Fortior Tech

DATASHEET

Future Is In Control

FS95A6AS

IPM for Three-phase Motor Control

Version No.: V1.0

Copyright Notice

Copyright by Fortior Technology (Shenzhen) Co., Ltd. All Rights Reserved.

Right to make changes — Fortior Technology (Shenzhen) Co., Ltd. reserves the right to make changes in the products - including circuits, standard cells, and/or software - described or contained herein in order to improve design and/or performance. The information contained in this manual is provided for the general use by our customers. Our customers shall ensure that they take appropriate action so that their use of our products does not infringe upon any patents. It is the policy of Fortior Technology (Shenzhen) Co., Ltd. to respect the valid patent rights of third parties and not to infringe upon or assist others to infringe upon such rights.

This manual is copyrighted by Fortior Technology (Shenzhen) Co., Ltd. You may not reproduce, transmit, transcribe, store in a retrieval system, or translate into any language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual, or otherwise, any part of this publication without the expressly written permission from Fortior Technology (Shenzhen) Co., Ltd. You may not alter or remove any copyright or other notice from copies of this content.

If there are any differences between the Chinese and the English contents, please take the Chinese version as the standard.

Disclaimer

This document is the property of Fortior Technology (Shenzhen) Co., Ltd. and its subsidiaries (hereinafter referred to as "Fortior Technology").

This document is provided on an "AS IS" basis, without warranty of any kind, express or implied, including but not limited to the warranties of merchantability, fitness for a particular purpose and non-infringement of intellectual property rights of any third-party. Any and all information stated herein is for reference purpose only, and Fortior Technology hereby disclaims any and all warranties and liabilities of any kind. The customer has the full responsibility to design, program and test their applications, evaluate functionality and safety of their final products and accept all loss or damage, legal claim or lawsuit whatsoever arising from the use of the information, data or opinions contained herein, and agrees to defend, hold harmless and indemnify Fortior Technology from and against any and all claims, damages and liabilities. No license, express or implied, to any intellectual property rights of Fortior Technology is granted by this document. Fortior Technology reserves the rights to amend, revise or make changes to this document at any time without prior notice.

In the event of any inconsistency, conflict or ambiguity under this disclaimer, the final interpretation is at the sole discretion of Fortior Technology.

Contents

Copyright Notice 1
Disclaimer 2
Contents 3
Explanation of Symbols 23
Abbreviations 24
1 System Introduction 26
1.1 Features 26
1.2 Applications 27
1.3 Overview 28
1.4 Functional Block Diagram 29
1.4.1 FS95A6AS 29
1.5 Memory Organization 30
1.5.1 Program Memory 30
1.5.2 Data Memory 30
1.5.3 SFR 31
1.5.4 XSFR 32
2 Pin Definitions 36
2.1 FS95A6AS SSOP A54-38 Pins 36
2.2 FS95A6AS SSOP A54-38 Pinout Diagram 39
3 Package Information 40
4 Ordering Information 41
5 Electrical Characteristics 42
5.1 Absolute Maximum Ratings 42
5.2 Recommended Operating Conditions 43
5.3 Global Electrical Characteristics 43
5.4 GPIO Electrical Characteristics 43

5.5 ADC Electrical Characteristics 44
5.6 Operational Amplifier Electrical Characteristics 45
5.7 BEMF Electrical Characteristics 45
5.8 OSC Electrical Characteristics 45
5.9 Reset Electrical Characteristics 46
5.10 LDO Electrical Characteristics 46
5.11 Switching Characteristics 46
5.12 Package Thermal Resistance 47
5.13 Mounting Torque 47
6 Reset Control 48
6.1 Reset Source(RST_SR) 48
6.2 Reset Enable 48
6.3 RSTEXT, RSTPOW 48
6.4 Low-voltage Reset 48
6.5 Watchdog Timeout Reset 49
6.6 RSTFED Reset 49
6.7 RSTDBG Reset 49
6.8 Soft Reset 49
6.9 Reset Register 50
6.9.1 RST_SR (0xC9) 50
7 Interrupt 51
7.1 Interrupt Introduction 51
7.2 Interrupt Source Enable 51
7.3 External Interrupt 52
7.4 Interrupt Introduction 52
7.5 Interrupt Register 54
7.5.1 IE (0xA8) 54
7.5.2 IP0 (0x8A) 54
7.5.3 IP1 (0x8B) 55
7.5.4 IP2 (0x8C) 55
7.5.5 IP3 (0x8D) 56
7.5.6 TCON (0x88) 56

7.5.7 EXT1L_IE (0xD1) 57
7.5.8 EXT1L_IF (0xD2) 58
7.5.9 EXT1H_IE (0xD3) 59
7.5.10 EXT1H_IF (0xD4) 60
8 Clock Gating 62
8.1 Clock Gating Register 63
8.1.1 CK_CR (0x91) 63
9 UART 64
9.1 UART Introduction 64
9.2 UART Operations 64
9.2.1 UART1 Operation Instructions 64
9.2.1.1 UART1 Mode0 64
9.2.1.2 UART1 Mode1 65
9.2.1.3 UART1 Mode2 65
9.2.1.4 UART1 Mode3 65
9.2.1.5 UART1 Interrupt Sources 66
9.2.2 UART2 Operation Instructions 66
9.2.2.1 UART2 Mode0 66
9.2.2.2 UART2 Mode1 66
9.2.2.3 UART2 Mode2 67
9.2.2.4 UART2 Mode3 67
9.2.2.5 UART2 Interrupt Sources 67
9.3 UART1 Registers 68
9.3.1 UT_CR (0x98) 68
9.3.2 UT_DR (0x99) 69
9.3.3 UT_BAUD (0x9A, 0x9B) 69
9.4 UART2 Registers 70
9.4.1 UT2_CR (0xD8) 70
9.4.2 UT2_DR (0x89) 71
9.4.3 UT2_BAUD (0x4042, 0x4043) 71
10 MDU 73
10.1 MDU Introduction 73

10.2 MDU Features 73
10.3 MDU Features 74
10.3.1 MDU Operations 74
10.3.2 16-bit Signed Multiplication with the Result Shifted Left by One-bit 74
10.3.3 16-bit Signed Multiplication 74
10.3.4 16-bit Unsigned Multiplication 75
10.3.5 32-bit/16-bit Unsigned Division 75
10.3.6 LPF 76
10.3.7 Coordinate Transformation (sin/cos Calculation) 76
10.3.8 Arctangent 77
10.3.9 PI/PID 78
10.3.9.1 PI/PID Introduction 78
10.3.9.2 PI/PID Features 79
10.3.9.3 PI/PID Operations 79
10.4 MDU Registers 81
10.4.1 MDU_CR (0xC1) 81
10.4.2 MUL0_MA (0x0FA0, 0x0FA1) 81
10.4.3 MUL0_MB (0x0FA2, 0x0FA3) 82
10.4.4 MUL0_MC (0x0FA4, 0x0FA5, 0x0FA6, 0x0FA7) 82
10.4.5 MUL1_MA (0x0F98, 0x0F99) 83
10.4.6 MUL1_MB (0x0F9A, 0x0F9B) 83
10.4.7 MUL1_MC (0x0F9C, 0x0F9D, 0x0F9E, 0x0F9F) 83
10.4.8 MUL2_MA (0x0F40, 0x0F41) 84
10.4.9 MUL2_MB (0x0F42, 0x0F43) 84
10.4.10 MUL2_MC (0x0F44, 0x0F45, 0x0F46, 0x0F47) 85
10.4.11 MUL3_MA (0x0F38, 0x0F39) 86
10.4.12 MUL3_MB (0x0F3A, 0x0F3B) 86
10.4.13 MUL3_MC (0x0F3C, 0x0F3D, 0x0F3E, 0x0F3F) 86
10.4.14 DIV0_DA (0x0F8C, 0x0F8D, 0x0F8E, 0x0F8F) 87
10.4.15 DIV0_DB (0x0F90, 0x0F91) 88
10.4.16 DIV0_DQ (0x0F92, 0x0F93, 0x0F94, 0x0F95) 88
10.4.17 DIV0_DR (0x0F96, 0x0F97) 89

10.4.18 DIV1_DA (0x0F80, 0x0F81, 0x0F82, 0x0F83) 89
10.4.19 DIV1_DB (0x0F84, 0x0F85) 90
10.4.20 DIV1_DQ (0x0F86, 0x0F87, 0x0F88, 0x0F89) 90
10.4.21 DIV1_DR (0x0F8A, 0x0F8B) 91
10.4.22 DIV2_DA (0x0F2C, 0x0F2D, 0x0F2E, 0x0F2F) 91
10.4.23 DIV2_DB (0x0F30, 0x0F31) 92
10.4.24 DIV2_DQ (0x0F32, 0x0F33, 0x0F34, 0x0F35) 92
10.4.25 DIV2_DR (0x0F36, 0x0F37) 93
10.4.26 DIV3_DA (0x0F20, 0x0F21, 0x0F22, 0x0F23) 93
10.4.27 DIV3_DB (0x0F24, 0x0F25) 94
10.4.28 DIV3_DQ (0x0F26, 0x0F27, 0x0F28, 0x0F29) 94
10.4.29 DIV3_DR (0x0F2A, 0x0F2B) 95
10.4.30 SCAT0_COS (0x0F16, 0x0F17) 95
10.4.31 SCAT0_SIN (0x0F18, 0x0F19) 96
10.4.32 SCAT0_THE (0x0F1A, 0x0F1B) 96
10.4.33 SCAT0_RES1 (0x0F1C, 0x0F1D) 96
10.4.34 SCAT0_RES2 (0x0F1E, 0x0F1F) 97
10.4.35 SCAT1_COS (0x0F0C, 0x0F0D) 97
10.4.36 SCAT1_SIN (0x0F0E, 0x0F0F) 98
10.4.37 SCAT1_THE (0x0F10, 0x0F11) 98
10.4.38 SCAT1_RES1 (0x0F12, 0x0F13) 98
10.4.39 SCAT1_RES2 (0x0F14, 0x0F15) 99
10.4.40 SCAT2_COS (0x0F02, 0x0F03) 99
10.4.41 SCAT2_SIN (0x0F04, 0x0F05) 99
10.4.42 SCAT2_THE (0x0F06, 0x0F07) 100
10.4.43 SCAT2_RES1 (0x0F08, 0x0F09) 100
10.4.44 SCAT2_RES2 (0x0F0A, 0x0F0B) 101
10.4.45 SCAT3_COS (0x0EF8, 0x0EF9) 101
10.4.46 SCAT3_SIN (0x0EFA, 0x0EFB) 101
10.4.47 SCAT3_THE (0x0EFC, 0x0EFD) 102
10.4.48 SCAT3_RES1 (0x0EFE, 0x0EFF) 102
10.4.49 SCAT3_RES2 (0x0F00, 0x0F01) 102

10.4.50 LPF0_K (0x0FD0, 0x0FD1) 103
10.4.51 LPF0_X (0x0FD2, 0x0FD3) 103
10.4.52 LPF0_Y (0x0FD4, 0x0FD5, 0x0FD6, 0x0FD7) 104
10.4.53 LPF1_K (0x0FC8, 0x0FC9) 104
10.4.54 LPF1_X (0x0FCA, 0x0FCB) 105
10.4.55 LPF1_Y (0x0FCC, 0x0FCD, 0x0FCE, 0x0FCF) 105
10.4.56 LPF2_K (0x0F78, 0x0F79) 106
10.4.57 LPF2_X (0x0F7A, 0x0F7B) 106
10.4.58 LPF2_Y (0x0F7C, 0x0F7D, 0x0F7E, 0x0F7F) 106
10.4.59 LPF3_K (0x0F70, 0x0F71) 107
10.4.60 LPF3_X (0x0F72, 0x0F73) 107
10.4.61 LPF3_Y (0x0F74, 0x0F75, 0x0F76, 0x0F77) 108
10.4.62 PI0_KP (0x0FB8, 0x0FB9) 108
10.4.63 PI0_EK1 (0x0FBA, 0x0FBB) 109
10.4.64 PI0_EK (0x0FBC, 0x0FBD) 109
10.4.65 PI0_KI (0x0FBE, 0x0FBF) 110
10.4.66 PI0_UKH (0x0FC0, 0x0FC1) 110
10.4.67 PI0_UKL (0x0FC2, 0x0FC3) 110
10.4.68 PI0_UKMAX (0x0FC4, 0x0FC5) 111
10.4.69 PI0_UKMIN (0x0FC6, 0x0FC7) 111
10.4.70 PI1_KP (0x0FA8, 0x0FA9) 111
10.4.71 PI1_EK1 (0x0FAA, 0x0FAB) 112
10.4.72 PI1_EK (0x0FAC, 0x0FAD) 112
10.4.73 PI1_KI (0x0FAE, 0x0FAF) 113
10.4.74 PI1_UKH (0x0FB0, 0x0FB1) 113
10.4.75 PI1_UKL (0x0FB2, 0x0FB3) 113
10.4.76 PI1_UKMAX (0x0FB4, 0x0FB5) 114
10.4.77 PI1_UKMIN (0x0FB6, 0x0FB7) 114
10.4.78 PI2_KP (0x0F5C, 0x0F5D) 114
10.4.79 PI2_EK1 (0x0F5E, 0x0F5F) 115
10.4.80 PI2_EK (0x0F60, 0x0F61) 115
10.4.81 PI2_KI (0x0F62, 0x0F63) 116

10.4.82 PI2_UKH (0x0F64, 0x0F65) 116
10.4.83 PI2_UKL (0x0F66, 0x0F67) 116
10.4.84 PI2_UKMAX (0x0F68, 0x0F69) 117
10.4.85 PI2_UKMIN (0x0F6A, 0x0F6B) 117
10.4.86 PI2_KD (0x0F6C, 0x0F6D) 117
10.4.87 PI2_EK2 (0x0F6E, 0x0F6F) 118
10.4.88 PI3_KP (0x0F48, 0x0F49) 118
10.4.89 PI3_EK1 (0x0F4A, 0x0F4B) 119
10.4.90 PI3_EK (0x0F4C, 0x0F4D) 119
10.4.91 PI3_KI (0x0F4E, 0x0F4F) 119
10.4.92 PI3_UKH (0x0F50, 0x0F51) 120
10.4.93 PI3_UKL (0x0F52, 0x0F53) 120
10.4.94 PI3_UKMAX (0x0F54, 0x0F55) 120
10.4.95 PI3_UKMIN (0x0F56, 0x0F57) 121
10.4.96 PI3_KD (0x0F58, 0x0F59) 121
10.4.97 PI3_EK2 (0x0F5A, 0x0F5B) 122
11 PFC 123
11.1 PFC Operating Instructions 123
11.1.1 PFC Introduction 123
11.1.2 Voltage Error Compensation Module 124
11.1.3 Voltage Feedforward Compensation Module 124
11.1.4 Calculation of Average Voltage PFC_UAVG 124
11.1.5 Current Error Compensation Module 125
11.1.6 PWM Output Module 126
11.1.7 Over-current Protection and Cycle-by-Cycle Current Limiting 126
11.1.8 PFC_UAC/PFC_IAC/UDC Sampling 127
11.1.8.1 UDC Sampling 127
11.1.8.2 PFC_IAC Sampling 127
11.1.8.3 PFC_UAC Sampling 127
11.2 PFC Registers 128
11.2.1 PFC_CR2 (0x4063) 128
11.2.2 PFC_CR0 (0x40E0) 129

11.2.3 PFC_CR1/UDC_UKMINH (0x40F2) 130
11.2.4 PFC_ADCCH (0x40E1) 131
11.2.5 PFC_CSO (0x40E2, 0x40E3) 132
11.2.6 PFC_ARR (0x40E4, 0x40E5) 132
11.2.7 PFC__UAVG (0x40E4, 0x40E5) 133
11.2.8 PFC__DR (0x40E6, 0x40E7) 133
11.2.9 UDC_REF (0x40E8, 0x40E9) 134
11.2.10 UDC__UK (0x40EA, 0x40EB) 134
11.2.11 UDC_KP (0x40EC, 0x40ED) 135
11.2.12 UDC_KI (0x40EE, 0x40EF) 135
11.2.13 UDC_UKMAX (0x40F0, 0x40F1) 135
11.2.14 UDC_UKMIN (0x40F2, 0x40F3) 136
11.2.15 PFC_KM (0x40F3) 136
11.2.16 IAC__REF (0x40F4, 0x40F5) 137
11.2.17 IAC__UK (0x40F6, 0x40F7) 137
11.2.18 IAC_KP (0x40F8, 0x40F9) 138
11.2.19 IAC_KI (0x40FA, 0x40FB) 138
11.2.20 IAC_UKMAX (0x40FC, 0x40FD) 138
11.2.21 IAC_UKMIN (0x40FE, 0x40FF) 139
11.2.22 PFC_TRGDLY/PFC_OUTARR (0x40FE, 0x40FF) 139
11.2.23 PFC__UAC (0x409A, 0x409B) 140
11.2.24 PFC__IAC (0x409C, 0x409D) 141
12 FOC 142
12.1 FOC Overview 142
12.1.1 FOC Introduction 142
12.1.2 Reference Input 143
12.1.3 PI Controller 143
12.1.4 Coordinate Transformation 143
12.1.4.1 Inverse Park Transformation 143
12.1.4.2 Clarke Transformation 144
12.1.4.3 Park Transformation 144
12.1.5 SVPWM 144

12.1.5.1 Continuous SVPWM 146
12.1.5.2 Discontinuous SVPWM 146
12.1.6 Overmodulation 147
12.1.7 Deadtime Compensation 147
12.1.8 Current and Voltage Sampling 147
12.1.8.1 Single-shunt Current Sampling Mode 147
12.1.8.2 Dual/Triple-shunt Current Sampling Mode 150
12.1.8.3 Current Sampling Offset 151
12.1.9 Angle Mode 151
12.1.9.1 Forced Ramping Angle 152
12.1.9.2 Forced Pulling Angle 152
12.1.9.3 Estimator Output Angle 153
12.1.9.3.1 Estimated Angle of Estimator 153
12.1.9.3.2 Forced Angle of Estimator 154
12.1.9.3.3 Angle Smooth Switching 154
12.1.9.3.4 Angle Compensation 155
12.1.10 Motor Real-time Parameters 155
12.1.10.1 Tailwind/headwind Detection 156
12.1.10.2 BEMF Detection 156
12.1.10.3 Motor Power 156
12.2 FOC Registers 157
12.2.1 FOC_CR0 (0x409F) 157
12.2.2 FOC_CR1 (0x40A0) 157
12.2.3 FOC_CR2 (0x40A1) 158
12.2.4 FOC_CR3 (0x409E) 159
12.2.5 FOC_TSMIN (0x40A2) 160
12.2.6 FOC_TGLI (0x40A3) 161
12.2.7 FOC_TBLO (0x40A4) 161
12.2.8 FOC_TRGDLY (0x40A5) 162
12.2.9 FOC_CSO (0x40A6, 0x40A7) 162
12.2.10 FOC_RTHESTEP (0x40A8, 0x40A9) 163
12.2.11 FOC_RTHEACC (0x40AA, 0x40AB) 163

12.2.12 FOC_EOMELPF (0x40AA, 0x40AB) 164
12.2.13 FOC_RTHECNT (0x40AC) 164
12.2.14 FOC_THECOR (0x40AD) 165
12.2.15 FOC_EMF (0x40AE, 0x40AF) 165
12.2.16 FOC_THECOMP (0x40AE, 0x40AF) 165
12.2.17 FOC_DMAX (0x4078) 166
12.2.18 FOC_DMIN (0x4079) 166
12.2.19 FOC_OMEEST (0x40B0, 0x40B1) 166
12.2.20 FOC_ATAN_THETA (0x40B2, 0x40B3) 167
12.2.21 FOC_QMAX (0x408A) 167
12.2.22 FOC_QMIN (0x408B) 167
12.2.23 FOC_UD (0x40B8, 0x40B9) 168
12.2.24 FOC_UQ (0x40BA, 0x40BB) 168
12.2.25 FOC_ID (0x40BC, 0x40BD) 168
12.2.26 FOC_IQ (0x40BE, 0x40BF) 169
12.2.27 FOC_IBET (0x40C0, 0x40C1) 169
12.2.28 FOC_IQ_LPFK (0x40C0) 170
12.2.29 FOC_ID_LPFK (0x40C1) 170
12.2.30 FOC_VBET (0x40C2, 0x40C3) 170
12.2.31 FOC_UDCPS (0x40C2, 0x40C3) 171
12.2.32 FOC_UQCPS (0x40C4, 0x40C5) 171
12.2.33 FOC_VALP (0x40C4, 0x40C5) 172
12.2.34 FOC_IC (0x40C6, 0x40C7) 172
12.2.35 FOC_LQ (0x40C8, 0x40C9) 172
12.2.36 FOC_IB (0x40C8, 0x40C9) 173
12.2.37 FOC_LD (0x40CA, 0x40CB) 173
12.2.38 FOC_IA (0x40CA, 0x40CB) 174
12.2.39 FOC_THETA (0x40CC, 0x40CD) 174
12.2.40 FOC_ETHETA (0x40CE, 0x40CF) 175
12.2.41 FOC_EALP (0x40D0, 0x40D1) 175
12.2.42 FOC_EBET (0x40D2, 0x40D3) 176
12.2.43 FOC_EOME (0x40D4, 0x40D5) 176

12.2.44 FOC_POW (0x40D8, 0x40D9) 176
12.2.45 FOC_EOMEKLPF (0x40D8) 177
12.2.46 FOC_IAMAX (0x40DA, 0x40DB) 177
12.2.47 FOC_IBMAX (0x40DC, 0x40DD) 178
12.2.48 FOC_ICMAX (0x40DE, 0x40DF) 178
12.2.49 FOC_DKP (0x4070, 0x4071) 179
12.2.50 FOC_EKP (0x4074, 0x4075) 179
12.2.51 FOC_EKI (0x4076, 0x4077) 179
12.2.52 FOC_EKLPFMIN (0x407A, 0x407B) 180
12.2.53 FOC_DKI (0x407C, 0x407D) 180
12.2.54 FOC_OMEKLPF (0x407E, 0x407F) 181
12.2.55 FOC_FBASE (0x4080, 0x4081) 181
12.2.56 FOC_EFREQACC (0x4082, 0x4083) 182
12.2.57 FOC_EFREQMIN (0x4084, 0x4085) 182
12.2.58 FOC_EFREQHOLD (0x4086, 0x4087) 183
12.2.59 FOC_EK3 (0x4088, 0x4089) 183
12.2.60 FOC_EK1 (0x408C, 0x408D) 184
12.2.61 FOC_EK2 (0x408E, 0x408F) 184
12.2.62 FOC_IDREF (0x4090, 0x4091) 184
12.2.63 FOC_IQREF (0x4092, 0x4093) 185
12.2.64 FOC_QKP (0x4094, 0x4095) 185
12.2.65 FOC_QKI (0x4096, 0x4097) 186
12.2.66 FOC_UDCFLT (0x4098, 0x4099) 186
13 Timer1 187
13.1 Timer1 Operations 187
13.1.1 Timer1 Counter Module 188
13.1.1.1 Prescaler 188
13.1.1.2 Basic Timer 189
13.1.1.3 Reload Timer 190
13.1.2 Position Detection 190
13.1.2.1 Position Detection Signal 190
13.1.2.2 CMP/GPIO Position Detection Event 191

13.1.2.3 ADC Position Detection Event 192
13.1.2.4 Sampling 193
13.1.2.5 Filtering 193
13.1.3 Write Timing Interrupt 194
13.1.4 Timer1 Interrupt 194
13.2 Square-wave Control for BLDC Motors 195
13.2.1 Six-step Phase Commutation of Square Wave Control 196
13.2.2 Square Wave Control Working Principle 196
13.2.2.1 60° Commutation Base Time 197
13.2.2.2 Forced Commutation at 60° 197
13.2.2.3 Diode Freewheeling Masking 198
13.2.2.4 Angle of Position Detection to Commutation 198
13.2.2.5 Cycle-by-cycle Current Limiting 198
13.3 Timer1 Registers 199
13.3.1 TIM1_CR0 (0x4068) 199
13.3.2 TIM1_CR1 (0x4069) 200
13.3.3 TIM1_CR2 (0x406A) 201
13.3.4 TIM1_CR3 (0x406B) 201
13.3.5 TIM1_CR4 (0x406C) 202
13.3.6 TIM1_IER (0x406D) 203
13.3.7 TIM1_SR (0x406E) 204
13.3.8 TIM1__BCOR (0x4070, 0x4071) 205
13.3.9 TIM1_CR5 (0x4072) 206
13.3.10 TIM1_DBR x (x = 1 ~ 7) (0x4072 + 2*x, 0x4073 + 2*x) 206
13.3.11 TIM1__BCNTR (0x4082, 0x4083) 208
13.3.12 TIM1__BCCR (0x4084, 0x4085) 208
13.3.13 TIM1__BARR (0x4086, 0x4087) 209
13.3.14 TIM1__RARR (0x4088, 0x4089) 209
13.3.15 TIM1__RCNTR (0x408A, 0x408B) 210
13.3.16 TIM1__UCOP (0x408C, 0x408D) 210
13.3.17 TIM1__UFLP (0x408E, 0x408F) 211
13.3.18 TIM1__URES (0x4090, 0x4091) 211

13.3.19 TIM1_KRMAX (0x4092) 211
13.3.20 TIM1_KFMIN (0x4093) 212
13.3.21 TIM1_KF (0x4094, 0x4095) 212
13.3.22 TIM1_KR (0x4096, 0x4097) 212
13.3.23 TIM1__ITRIP (0x4098, 0x4099) 213
14 Timer2 214
14.1 Timer2 Instructions 214
14.1.1 Prescaler 214
14.1.2 Reading, Writing and Counting of TIM2__CNTR 215
14.1.3 Output Mode 215
14.1.3.1 Reading and Writing of TIM2__ARR/TIM2__DR 215
14.1.3.2 High-/Low-level Output Mode 216
14.1.3.3 PWM Output 216
14.1.3.4 Interrupts 216
14.1.4 Input Signal Filtering and Edge Detection 217
14.1.5 Input Capture Mode 217
14.1.6 Input Counter Mode 219
14.2 Timer2 Registers 221
14.2.1 TIM2_CR0 (0xA1) 221
14.2.2 TIM2_CR1 (0xA9) 222
14.2.3 TIM2__CNTR (0xAA, 0xAB) 223
14.2.4 TIM2__DR (0xAC, 0xAD) 224
14.2.5 TIM2__ARR (0xAE, 0xAF) 224
15 Timer3/Timer4 225
15.1 Timer3/Timer4 Instructions 225
15.1.1 Prescaler 225
15.1.2 Reading, Writing and Counting of TIMx__CNTR 226
15.1.3 Output Mode 226
15.1.3.1 High-/Low-level Output Mode 226
15.1.3.2 PWM Output 227
15.1.3.3 Interrupts 227
15.1.4 Input Signal Filtering and Edge Detection 228

15.1.5 Input Capture Mode 228
15.1.6 Timer4 FG Generation Mode 230
15.2 Timer3/Timer4 Registers 231
15.2.1 TIMx_CR0 (0x9C/0x9E) (x = 3/4) 231
15.2.2 TIMx_CR1 (0x9D/0x9F) (x = 3/4) 232
15.2.3 TIMx__CNTR (0xA2, 0xA3/0x92, 0x93) (x = 3/4) 233
15.2.4 TIMx__DR (0xA4, 0xA5/0x94, 0x95) (x = 3/4) 234
15.2.5 TIMx__ARR (0xA6, 0xA7/0x96, 0x97) (x = 3/4) 234
16 Systick 235
16.1 Systick Instructions 235
16.2 Systick Registers 235
16.2.1 SYST_CR (0x4065) 235
17 Driver 236
17.1 Instructions 236
17.1.1 FS95A6AS Driver Introduction 236
17.1.2 Output Control Module 237
17.1.2.1 Count and Compare Module 237
17.1.2.2 Enable and Polarity of Output Signals 238
17.1.2.3 Deadtime Module 239
17.1.2.4 Main Output Enable (MOE) 240
17.1.2.5 Interrupts 241
17.1.2.5.1 Compare Match Interrupt 241
17.1.2.5.2 FG Interrupt 241
17.2 Driver Registers 242
17.2.1 PI_CR (0xF9) 242
17.2.2 DRV_CR (0x4062) 242
17.2.3 DRV_SR (0x4061) 243
17.2.4 DRV_OUT (0xF8) 244
17.2.5 DRV_CMR (0x405C, 0x405D) 245
17.2.6 DRV_ARR (0x405E, 0x405F) 246
17.2.7 DRV_COMR (0x405A, 0x405B) 247
17.2.8 DRV_DR (0x4058, 0x4059) 247

17.2.9 DRV_DTR (0x4060) 248
17.2.10 DRV__CNTR (0x4066, 0x4067) 248
18 WDT 250
18.1 WDT Notes 250
18.2 WDT Operations 250
18.3 WDT Registers 251
18.3.1 WDT_CR (0x4026) 251
18.3.2 WDT_ARR (0x4027) 251
18.3.3 CCFG1 (0x401E) 251
19 Clock 252
19.1 Introduction 252
19.2 Clock Calibration 252
19.3 Clock Calibration Registers 253
19.3.1 CAL_CR (0x4044, 0x4045) 253
20 RTC 254
20.1 RTC Functional Block Diagram 254
20.2 RTC Operations 254
20.3 RTC Registers 255
20.3.1 RTC_TM (0x402C, 0x402D) 255
20.3.2 RTC_STA (0x402E) 255
21 IO 257
21.1 IO Introduction 257
21.2 IO Operations 257
21.3 IO Registers 258
21.3.1 P0_OE (0xFC) 258
21.3.2 P1_OE (0xFD) 258
21.3.3 P2_OE (0xFE) 258
21.3.4 P3_OE (0xFF) 259
21.3.5 P1_AN (0x4050) 259
21.3.6 P2_AN (0x4051) 260
21.3.7 P3_AN (0x4052) 260

21.3.8 P0_PU (0x4053) 261
21.3.9 P1_PU (0x4054) 261
21.3.10 P2_PU (0x4055) 261
21.3.11 P3_PU (0x4056) 262
21.3.12 P4_PU (0x4057) 262
21.3.13 PX_PL (0x4048) 263
21.3.14 PH_SEL (0x404C) 263
21.3.15 PH_SEL1 (0x404D) 264
21.3.16 P0 (0x80) 264
21.3.17 P1 (0x90) 265
21.3.18 P2 (0xA0) 265
21.3.19 P3 (0xB0) 265
21.3.20 P4 (0xB8) 266
22 ADC 267
22.1 ADC Introduction 267
22.2 ADC Block Diagram 267
22.3 ADC Operations 268
22.3.1 Sequential Sampling Mode 268
22.3.2 Triggered Sampling Mode 269
22.3.3 Output Data Format 269
22.4 ADC Registers 270
22.4.1 ADC_CR (0x4039) 270
22.4.2 ADC_MASK (0x4036, 0x4037) 271
22.4.3 DAC_CR (0x4035) 271
22.4.4 ADC_SCYC (0x4038) 272
22.4.5 ADC0_DR (0x0FD8, 0x0FD9) 272
22.4.6 ADC1_DR (0x0FDA, 0x0FDB) 273
22.4.7 ADC2_DR (0x0FDC, 0x0FDD) 273
22.4.8 ADC3_DR (0x0FDE, 0x0FDF) 274
22.4.9 ADC4_DR (0x0FE0, 0x0FE1) 274
22.4.10 ADC5_DR (0x0FE2, 0x0FE3) 275
22.4.11 ADC6_DR (0x0FE4, 0x0FE5) 275

22.4.12 ADC7_DR (0x0FE6, 0x0FE7) 276
22.4.13 ADC8_DR (0x0FE8, 0x0FE9) 276
22.4.14 ADC9_DR (0x0FEA, 0x0FEB) 277
22.4.15 ADC10_DR (0x0FEC, 0x0FED) 277
23 DAC 278
23.1 DAC Introduction 278
23.2 DAC0 Functional Block Diagram 278
23.3 DAC1 Functional Block Diagram 279
23.4 DAC2 Functional Block Diagram 280
23.5 DAC Registers 281
23.5.1 DAC_CR (0x4035) 281
23.5.2 DAC0_DR (0x404B) 281
23.5.3 DAC1_DR (0x404A) 281
23.5.4 DAC2_DR (0x4049) 282
24 DMA 283
24.1 DMA Instructions 283
24.2 DMA Registers 284
24.2.1 DMA0_CR0 (0x403A) 284
24.2.2 DMA1_CR0 (0x403B) 285
24.2.3 DMA0_LEN (0x403C) 286
24.2.4 DMA0_BA (0x403E, 0x403F) 287
24.2.5 DMA1_LEN (0x403D) 287
24.2.6 DMA1_BA (0x4040, 0x4041) 288
25 VHALF 289
25.1 VHALF Instructions 289
25.2 VHALF Register 289
25.2.1 VHALF_CR (0x404F) 289
26 Operational Amplifier 290
26.1 Operational Amplifier Introduction 290
26.2 Operational Amplifier Instructions 290
26.2.1 Bus Current Sampling Operational Amplifier (AMP0) 290

26.2.1.1 AMP0 Normal Mode 290
26.2.1.2 AMP0 PGA Differential Input Mode 290
26.2.2 Phase Current Operational Amplifier (AMP1) 291
26.2.2.1 AMP1 PGA Differential Input Mode 291
26.2.3 Phase Current Operational Amplifier (AMP2) 292
26.2.3.1 AMP2 PGA Differential Input Mode 292
26.3 Operational Amplifier Registers 293
26.3.1 AMP_CR0 (0x0404E) 293
26.3.2 AMP_CR1 (0x4034) 293
27 Comparator 295
27.1 Comparator Operations 295
27.1.1 Comparator CMP3 295
27.1.1.1 Over-current Protection (OCP) 297
27.1.1.2 Cycle-by-cycle Current Limiting 297
27.1.2 Comparator CMP4 298
27.1.3 Comparator CMP5 299
27.1.4 Comparator Group (CMPG) 300
27.1.4.1 Built-in Three Comparators and Resistors Mode 300
27.1.4.2 Dual-comparator Mode 301
27.1.5 Comparator Sampling 301
27.2 Comparator Registers 303
27.2.1 CMP_CR0 (0xD5) 303
27.2.2 CMP_CR1 (0xD6) 303
27.2.3 CMP_CR2 (0xDA) 304
27.2.4 CMP_CR3 (0xDC) 306
27.2.5 CMP_CR4 (0xE1) 306
27.2.6 CMP_SAMR (0x40AD) 307
27.2.7 CMP_SR (0xD7) 308
27.2.8 HALL_CR (0xE2) 309
27.2.9 EVT_FILT (0xD9) 309
27.2.10 TSD_CR (0x402F) 310
28 Power Supply 311

28.1 LDO Instructions 311
28.2 Low Voltage Detection (LVD) 312
28.2.1 LVD Introduction 312
28.2.2 LVD Operations 312
28.2.3 LVD Registers 314
28.2.3.1 LVSR (0xDB) 314
29 Flash 315
29.1 Flash Introduction 315
29.2 Flash Operations 315
29.3 Flash Registers 316
29.3.1 FLA_CR (0x85) 316
29.3.2 FLA_KEY (0x84) 317
30 CRC 318
30.1 CRC Functional Block Diagram 318
30.2 CRC16 Polynomial 318
30.3 CRC16 Logic Diagram 319
30.4 CRC Operations 319
30.4.1 CRC Calculation of Single Byte 319
30.4.2 CRC Calculation of ROM Sector 319
30.5 CRC Registers 321
30.5.1 CRC_CR (0x4022) 321
30.5.2 CRC_DIN (0x4021) 321
30.5.3 CRC_DR (0x4023) 322
30.5.4 CRC_BEG (0x4024) 322
30.5.5 CRC_CNT (0x4025) 322
31 Sleep Mode 323
31.1 Introduction 323
31.2 Sleep Mode Register 324
31.2.1 PCON(0x87) 324
32 Code Protection 325
32.1 Introduction 325

32.2 Operating Instructions 325
33 Revision History 327

Explanation of Symbols

- The symbol “[ ]” following a register indicates a bit in the register. For example, ABCD[XY] indicates the XY bit in ABCD register.
- The symbol “x” in a register name indicates similar registers. For example, TIMx_CR0 indicates TIM3_CR0 and TIM4_CR0.
- [m:n] indicates a range of bits. For example, [3:0] means the bits from bit3 to bit0.
- Pm.n indicates the nth port of the Portm. P0.0 indicates the 0th port of Port0.
- Register read and write symbols:
  - R: Read only
  - W: Write only
  - R/W: Read/write
  - W0: Only 0 can be written
  - W1: Only 1 can be written
- The symbol “-” indicates an uncertainty value or invalid value.
- The RMW instruction cannot be used for registers with different read and written representations.
- Q (number) format is to store floating-point numbers using fixed-point numbers. MSB is the sign bit, followed by integer bits and fraction bits, where lower Q bits are assigned to the fractional part and the remaining bits are assigned to the integer part. For example, for Q12, bit15 is the sign bit, bit14 ~ bit12 represent the integer part and bit11 ~ bit0 represent the fraction part. The Q12 format has a decimal range -8 ~ 7.9998 (corresponding to 0x8000 ~ 0x7FFF).

Abbreviations

ADC	Analog to Digital Convertor
BEMF	Back Electromotive Force
BLDC	Brushless Direct Current
CRC	Cyclic Redundancy Check
DAC	Digital to Analog Convertor
DMA	Direct Memory Access
FG	Frequency Generator
FICE	Fortior Interactive Connectivity Establishment
FOC	Field Oriented Control
FOSC	Fast Oscillator
GPIO	General Purpose Input Output
IC	Integrated Circuit
IDE	Integrated Development Environment
IRAM	Internal RAM
LDO	Low Dropout Regulator
LPF	Low Pass Filter
LVD	Low Voltage Detection
MDU	Multiplication Division Unit
ME	Motor Engine
MSB	Most Significant Bit
MOSFET	Metal Oxide Semiconductor Field Effect Transistor
NC	Not Connected
PFC	Power Factor Correction
PGA	Programmable Gain Amplifier
PI/PID	Proportional Integral/Proportional Integral Derivative
PWM	Pulse Width Modulation
RAM	Random Access Memory
RMW	Read Modified Write
ROM	Read Only Memory

RTC	Real Time Clock
SAR	Successive Approximation Register
SFR	Special Function Register
SMO	Sliding Mode Observer
SOSC	Slow Oscillator
SVPWM	Space Vector PWM
TSD	Temperature Sensor Detect
UART	Universal Asynchronous Receiver/Transmitter
WDT	Watch Dog Timer
XRAM	External RAM
XSFR	External SFR

1 System Introduction

1.1 Features

- Power supply:
  - External power supply 13V~20V is connected to VCC pin, and internal LDO supplies VDD5 voltage.
- IGBT:
  - VCE Voltage = 650V
  - Single IGBT Drive Current (DC) = ± 6A (Max)
  - Single IGBT Drive Current (Pulse) = ± 12A (Max)
  - IGBT Saturation Voltage VCE (Sat.) = 1.73V (Typ.)
- Dual core: 8051 core and ME core
- An instruction cycle mostly takes 1 or 2 system clock cycle(s)
- 32kB Flash with CRC, self-program and code protection
- 256 bytes IRAM and 3.75k bytes XRAM
- ME: Core integrating PID module, FOC module, MDU auxiliary computing module and LPF module
- 16 interrupt sources with 4 configurable priority levels
- Number of GPIOs: 18
- Timers:
  - Timer1: Timer supporting square-wave drive timing control, automatic commutation, cycle-by-cycle current limiting and Hall/BEMF-based position sensing
  - Timer2: Timer supporting PWM output, measurement of duty cycle and period of input PWM wave, measurement of the time of set PWM wave numbers and QEP decoding.
  - Timer3/Timer4: Timers supporting PWM output, and measurement of duty cycle and period of input PWM wave. Timer4 supports FG generation and Timer3 supports up to 48MHz input
  - Systick Timer
  - RTC
- Communication interfaces:

- 2*UARTs (UART1 and UART2), supporting single-wire mode
- Dual-channel DMA: supporting data transmission via UART

- Analog peripherals:
  - 12-bit ADC, operating with 1μs conversion time
  - Number of ADC channels: 11. (Among them, AD0 and AD1 are internal channels)
  - Three standalone operational amplifier(s), where PGA is configurable
  - 4-channel analog comparator
  - DAC: Single-channel 9-bit, single-channel 8-bit, single-channel 6-bit
- Driver Type: PWM output
- FOC module supports single/dual/triple-shunt current sampling;
- FOC module supports overmodulation
- PFC
- System clock
  - Built-in 24MHz high-speed RC oscillator
  - Built-in 32.8kHz low-speed RC oscillator
- WDT
- LVD
- TSD
- Two-wire FICE protocol based in-circuit emulation

1.2 Applications

The chip can be used for the drive of sensorless or sensored BLDC/PMSM motors, single-phase/three-phase induction motors and servo motors.

- Application: Range hoods, air conditioner outdoor units, washing machines, high-speed hair dryers, and industrial fans, etc.

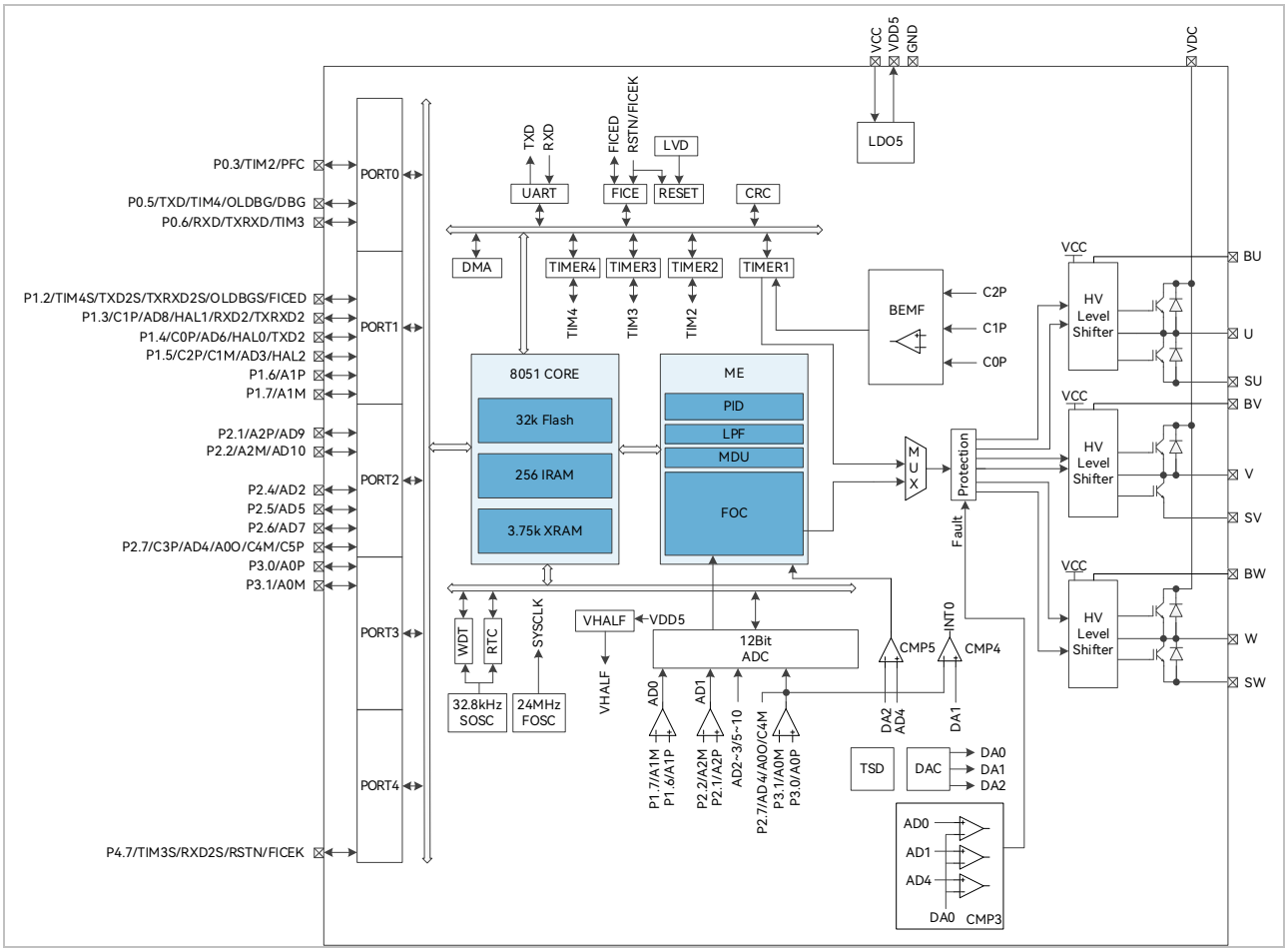
1.3 Overview

FS95A6AS is a fully-integrated IPM with motor controller, driver and power IC. Its high-performance motor drive chip incorporates ME core and 8051 core. ME core integrates FOC, MDU, LPF, PID and SVPWM modules that allow for automatic calculation of FOC or square-wave control by the hardware for sensored/sensorless BLDC/PMSM motors. 8051 core is used for parameter configuration and routine processing. Most of 8051 core instruction cycle takes 1T or 2T clock cycle(s). The dual cores work in parallel to achieve high-performance motor control. The chip integrates high-speed operational amplifiers, comparators, high-speed ADC, CRC, UART, Timers, PWM module and built-in high-voltage LDO, which are suitable for FOC or square-wave based BLDC/PMSM motors.

1.4 Functional Block Diagram

1.4.1 FS95A6AS

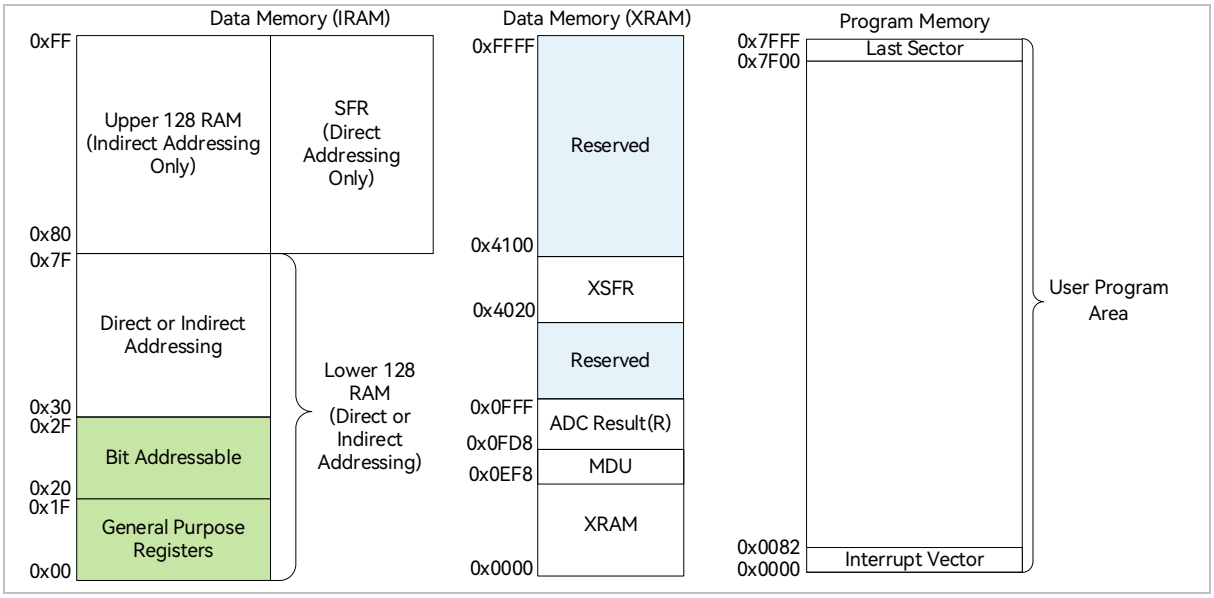
Figure 1-1 Functional Block Diagram of FS95A6AS



1.5 Memory Organization

The internal storage space is divided into Program Memory and Data Memory, which are independently addressed.

Figure 1-2 Memory Organization



1.5.1 Program Memory

The chip implements this program memory as Flash memory with a block from addresses 0x0000 to 0x7FFF to store control programs to store control programs.

The first sector (0x0000 ~ 0x0082) is the interrupt vector address area, which is used to store the start address of each interrupt subroutine. The last sector (0x7F00~0x7FFF) contains internal control bits of the chip.

1.5.2 Data Memory

The data memory is divided into External Data Memory and Internal Data Memory, as shown in Figure 1-2.

The External Data Memory is addressed from 0x0000 to 0xFFFF, which can be accessed only with MOVX instructions. It comprises XRAM (0x0000~0x0EF7), extended control register space (0x4020~0x40FF), MDU register space (0x0EF8~0x0FD7) and ADC result memory area (0x0FD8~0x0FFF).

The Internal Data Memory is addressed from 0x00 to 0xFF. Locations 0x00 ~ 0x1F are addressable as 4 banks

of general purpose registers, each bank consisting of 8 registers, adding up to 32 registers. Locations 0x20 ~ 0x7F are used for general purpose RAM memory, supporting direct and indirect addressing. Locations (0x20 ~ 0x2F) are 16-bit addressable. When locations 0x80 ~ 0xFF are accessed by indirect addressing, it points to RAM. When locations 0x80 ~ 0xFF are accessed by direct addressing, it points to SFR.

1.5.3 SFR

Table 1-1 SFR Address Mapping

Addr	0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)
0xF8	DRV_OUT	PI_CR			P0_OE	P1_OE	P2_OE	P3_OE
0xF0	B							
0xE8								
0xE0	ACC	CMP_CR4	HALL_CR					
0xD8	UT2_CR	EVT_FILT	CMP_CR2	LVSR	CMP_CR3			
0xD0	PSW	EXT1L_IE	EXT1L_IF	EXT1H_IE	EXT1H_IF	CMP_CR0	CMP_CR1	CMP_SR
0xC8		RST_SR						
0xC0		MDU_CR						
0xB8	P4							
0xB0	P3							
0xA8	IE	TIM2_CR1	TIM2__CNTRL	TIM2__CNTRH	TIM2__DRL	TIM2__DRH	TIM2__ARRL	TIM2__ARRH
0xA0	P2	TIM2_CR0	TIM3__CNTRL	TIM3__CNTRH	TIM3__DRL	TIM3__DRH	TIM3__ARRL	TIM3__ARRH
0x98	UT_CR	UT_DR	UT_BAUDL	UT_BAUDH	TIM3_CR0	TIM3_CR1	TIM4_CR0	TIM4_CR1
0x90	P1	CK_CR	TIM4__CNTRL	TIM4__CNTRH	TIM4__DRL	TIM4__DRH	TIM4__ARRL	TIM4__ARRH
0x88	TCON	UT2_DR	IP0	IP1	IP2	IP3		
0x80	P0	SP	DPL	DPH	FLA_KEY	FLA_CR		PCON

Note

- Registers with 4 low-order bits as 0 or 8 support addressing access
- Registers containing the symbol "__" are 16-bit snapshot registers. Snapshot registers are the dynamic registers which shall be read using variables. The value will be incorrect when the register is read directly
- 8-bit MCU shall read a 16-bit register twice to get the value, the 8 high-order bits and the 8 low-order bits respectively. The result will be incorrect when 8 low-order bits of the register change after MCU has read the 8 high-order bits. Therefore, when 8 high-order bits of the snapshot register are read by MCU, the corresponding 8 low-order bits are stored and read
- Snapshot register must be read as a whole, the 8 high-order bits first and then the 8 low-order bits

1.5.4 XSFR

Table 1-2 XSFR Address Mapping

Addr	0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)
0x40F8	IAC_KPH	IAC_KPL	IAC_KIH	IAC_KIL	IAC_UKMAXH	IAC_UKMAXL	IAC_UKMINH	IAC_UKMINL
							PFC_OUTARRH	PFC_OUTARRL
							PFC_TRGDLY	
0x40F0	UDC_UKMAXH	UDC_UKMAXL	UDC_UKMINH	UDC_UKMINL	IAC__REFH	IAC__REFL	IAC__UKH	IAC__UKL
			PFC_CR1	PFC_KM				
0x40E8	UDC_REFH	UDC_REFL	UDC__UKH	UDC__UKL	UDC_KPH	UDC_KPL	UDC_KIH	UDC_KIL
0x40E0	PFC_CR0	PFC_ADCCH	PFC_CSOH	PFC_CSOL	PFC_ARRH	PFC_ARRL	PFC__DRH	PFC__DRL
					PFC__UAVGH	PFC__UAVGL		
0x40D8	FOC__POWH	FOC__POWL	FOC__IAMAXH	FOC__IAMAXL	FOC__IBMAXH	FOC__IBMAXL	FOC__ICMAXH	FOC__ICMAXL
	FOC_EOMEKLPF							
0x40D0	FOC__EALPH	FOC__EALPL	FOC__EBETH	FOC__EBETL	FOC__EOMEH	FOC__EOMEL		
0x40C8	FOC__IBH	FOC__IBL	FOC__IAH	FOC__IAL	FOC__THETAH	FOC__THETAL	FOC__ETHETAH	FOC__ETHETAL
	FOC_LQH	FOC_LQL	FOC_LDH	FOC_LDL				
0x40C0	FOC__IBETH	FOC__IBETL	FOC__VBETH	FOC__VBETL	FOC__VALPH	FOC__VALPL	FOC__ICH	FOC__ICL
	FOC_IQ_LPFK	FOC_ID_LPFK	FOC_UDCPSH	FOC_UDCPSL	FOC_UQCPSH	FOC_UQCPSL		
0x40B8	FOC__UDH	FOC__UDL	FOC__UQH	FOC__UQL	FOC__IDH	FOC__IDL	FOC__IQH	FOC__IQL
0x40B0	FOC__OMEESTH	FOC__OMEESTL	FOC__ATAN_THETAH	FOC__ATAN_THETAL			FOC_QMINH	FOC_QMINL
0x40A8	FOC__RTHESTEPH	FOC__RTHESTEPL	FOC_RTHEACCH	FOC_RTHEACCL	FOC_RTHECNT	FOC_THECOR	FOC__THECOMPH	FOC__THECOMPL
			FOC__EOMELPFH	FOC__EOMELPFL		CMP_SAMR	FOC__EMFH	FOC__EMFL
0x40A0	FOC_CR1	FOC_CR2	FOC_TSMIN	FOC_TGLI	FOC_TBLO	FOC_TRGDLY	FOC_CSOH	FOC_CSOL
0x4098	FOC__UDCFLTH	FOC__UDCFLTL	PFC__UACH	PFC__UACL	PFC__IACH	PFC__IACL	FOC_CR3	FOC_CR0

Addr	0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)
	TIM1__ITRIPH	TIM1__ITRIPL						
0x4090	FOC_IDREFH	FOC_IDREFL	FOC_IQREFH	FOC_IQREFL	FOC_QKPH	FOC_QKPL	FOC_QKIH	FOC_QKIL
	TIM1__URESH	TIM1__URESL	TIM1_KRMAX	TIM1_KFMIN	TIM1_KFH	TIM1_KFL	TIM1_KRH	TIM1_KRL
0x4088	FOC_EK3H	FOC_EK3L	FOC_QMAX	FOC_QMIN	FOC_EK1H	FOC_EK1L	FOC_EK2H	FOC_EK2L
	TIM1__RARRH	TIM1__RARRL	TIM1__RCNTRH	TIM1__RCNTRL	TIM1__UCOPH	TIM1__UCOPL	TIM1__UFLPH	TIM1__UFLPL
0x4080	FOC_FBASEH	FOC_FBASEL	FOC_EFREQACCH	FOC_EFREQACCL	FOC_EFREQMINH	FOC_EFRQMINL	FOC_EFREQHOLDH	FOC_EFREQHOLDL
	TIM1_DBR7H	TIM1_DBR7L	TIM1__BCNTRH	TIM1__BCNTRL	TIM1__BCCRH	TIM1__BCCRL	TIM1__BARRH	TIM1__BARRL
0x4078	FOC_DMAX	FOC_DMIN	FOC_EKLPFMINH	FOC_EKLPFMINL	FOC_DKIH	FOC_DKIL	FOC_OMEKLPFH	FOC_OMEKLPFL
	TIM1_DBR3H	TIM1_DBR3L	TIM1_DBR4H	TIM1_DBR4L	TIM1_DBR5H	TIM1_DBR5L	TIM1_DBR6H	TIM1_DBR6L
0x4070	TIM1__BCORH	TIM1__BCORL	TIM1_CR5		FOC_EKPH	FOC_EKPL	FOC_EKIH	FOC_EKIL
	FOC_DKPH	FOC_DKPL			TIM1_DBR1H	TIM1_DBR1L	TIM1_DBR2H	TIM1_DBR2L
0x4068	TIM1_CR0	TIM1_CR1	TIM1_CR2	TIM1_CR3	TIM1_CR4	TIM1_IER	TIM1_SR	
0x4060	DRV_DTR	DRV_SR	DRV_CR	PFC_CR2	SYST_CR		DRV__CNTRH	DRV__CNTRL
0x4058	DRV_DRH	DRV_DRL	DRV_COMRH	DRV_COMRL	DRV_CMRH	DRV_CMRL	DRV_ARRH	DRV_ARRL
0x4050	P1_AN	P2_AN	P3_AN	P0_PU	P1_PU	P2_PU	P3_PU	P4_PU
0x4048	PX_PL	DAC2_DR	DAC1_DR	DAC0_DR	PH_SEL	PH_SEL1	AMP_CR0	VHALF_CR
0x4040	DMA1_BAH	DMA1_BAL	UT2_BAUDH	UT2_BAUDL	CAL_CRH	CAL_CRL		
0x4038	ADC_SCYC	ADC_CR	DMA0_CR0	DMA1_CR0	DMA0_LEN	DMA1_LEN	DMA0_BAH	DMA1_BAL
0x4030					AMP_CR1	DAC_CR	ADC_MASKH	ADC_MASKL
0x4028					RTC_TMH	RTC_TML	RTC_STA	TSD_CR
0x4020		CRC_DIN	CRC_CR	CRC_DR	CRC_BEG	CRC_CNT	WDT_CR	WDT_ARR
0x0FF0								
0x0FE8	AD8_DRH	AD8_DRL	AD9_DRH	AD9_DRL	AD10_DRH	AD10_DRL		
0x0FE0	AD4_DRH	AD4_DRL	AD5_DRH	AD5_DRL	AD6_DRH	AD6_DRL	AD7_DRH	AD7_DRL
0x0FD8	AD0_DRH	AD0_DRL	AD1_DRH	AD1_DRL	AD2_DRH	AD2_DRL	AD3_DRH	AD3_DRL
0x0FD0	LPF0_K		LPF0_X		LPF0_YH		LPF0_YL	
0x0FC8	LPF1_K		LPF1_X		LPF1_YH		LPF1_YL	
0x0FC0	PI0_UKH		PI0_UKL		PI0_UKMAX		PI0_UKMIN	
0x0FB8	PI0_KP		PI0_EK1		PI0_EK		PI0_KI	

Addr	0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)
0x0FB0	PI1_UKH		PI1_UKL		PI1_UKMAX		PI1_UKMIN	
0x0FA8	PI1_KP		PI1_EK1		PI1_EK		PI1_KI	
0x0FA0	MUL0_MA		MUL0_MB		MUL0_MCH		MUL0_MCL	
0x0F98	MUL1_MA		MUL1_MB		MUL1_MCH		MUL1_MCL	
0x0F90	DIV0_DB		DIV0_DQH		DIV0_DQL		DIV0_DR	
0x0F88	DIV1_DQL		DIV1_DR		DIV0_DAH		DIV0_DAL	
0x0F80	DIV1_DAH		DIV1_DAL		DIV1_DB		DIV1_DQH	
0x0F78	LPF2_K		LPF2_X		LPF2_YH		LPF2_YL	
0x0F70	LPF3_K		LPF3_X		LPF3_YH		LPF3_YL	
0x0F68	PI2_UKMAX		PI2_UKMIN		PI2_KD		PI2_EK2	
0x0F60	PI2_EK		PI2_KI		PI2_UKH		PI2_UKL	
0x0F58	PI3_KD		PI3_EK2		PI2_KP		PI2_EK1	
0x0F50	PI3_UKH		PI3_UKL		PI3_UKMAX		PI3_UKMIN	
0x0F48	PI3_KP		PI3_EK1		PI3_EK		PI3_KI	
0x0F40	MUL2_MA		MUL2_MB		MUL2_MCH		MUL2_MCL	
0x0F38	MUL3_MA		MUL3_MB		MUL3_MCH		MUL3_MCL	
0x0F30	DIV2_DB		DIV2_DQH		DIV2_DQL		DIV2_DR	
0x0F28	DIV3_DQL		DIV3_DR		DIV2_DAH		DIV2_DAL	
0x0F20	DIV3_DAH		DIV3_DAL		DIV3_DB		DIV3_DQH	
0x0F18	SCAT0_SIN		SCAT0_THE		SCAT0_RES1		SCAT0_RES2	
0x0F10	SCAT1_THE		SCAT1_RES1		SCAT1_RES2		SCAT0_COS	
0x0F08	SCAT2_RES1		SCAT2_RES2		SCAT1_COS		SCAT1_SIN	
0x0F00	SCAT3_RES2		SCAT2_COS		SCAT2_SIN		SCAT2_THE	
0x0EF8	SCAT3_COS		SCAT3_SIN		SCAT3_THE		SCAT3_RES1	

Note

- Registers with 4 low-order bits as 0 or 8 support addressing access
- Registers containing the symbol “__” are 16-bit snapshot registers. Snapshot registers are the dynamic registers which shall be read using variables. The value will be incorrect

when the register is read directly

- 8-bit MCU shall read a 16-bit register twice to get the value, the 8 high-order bits and the 8 low-order bits respectively. The result will be incorrect when 8 low-order bits of the register change after MCU has read the 8 high-order bits. Therefore, when 8 high-order bits of the snapshot register are read by MCU, the corresponding 8 low-order bits are stored and read
- Snapshot register must be read as a whole, the 8 high-order bits first and then the 8 low-order bits

2 Pin Definitions

The IO types are defined as follows:

- DI = Digital Input
- DO = Digital Output
- DB = Digital Bidirectional
- AI = Analog Input
- AO = Analog Output
- AB = Analog Bidirectional
- P = Power Supply

2.1 FS95A6AS SSOP A54-38 Pins

Table 2-1 FS95A6AS SSOP A54-38 Pins

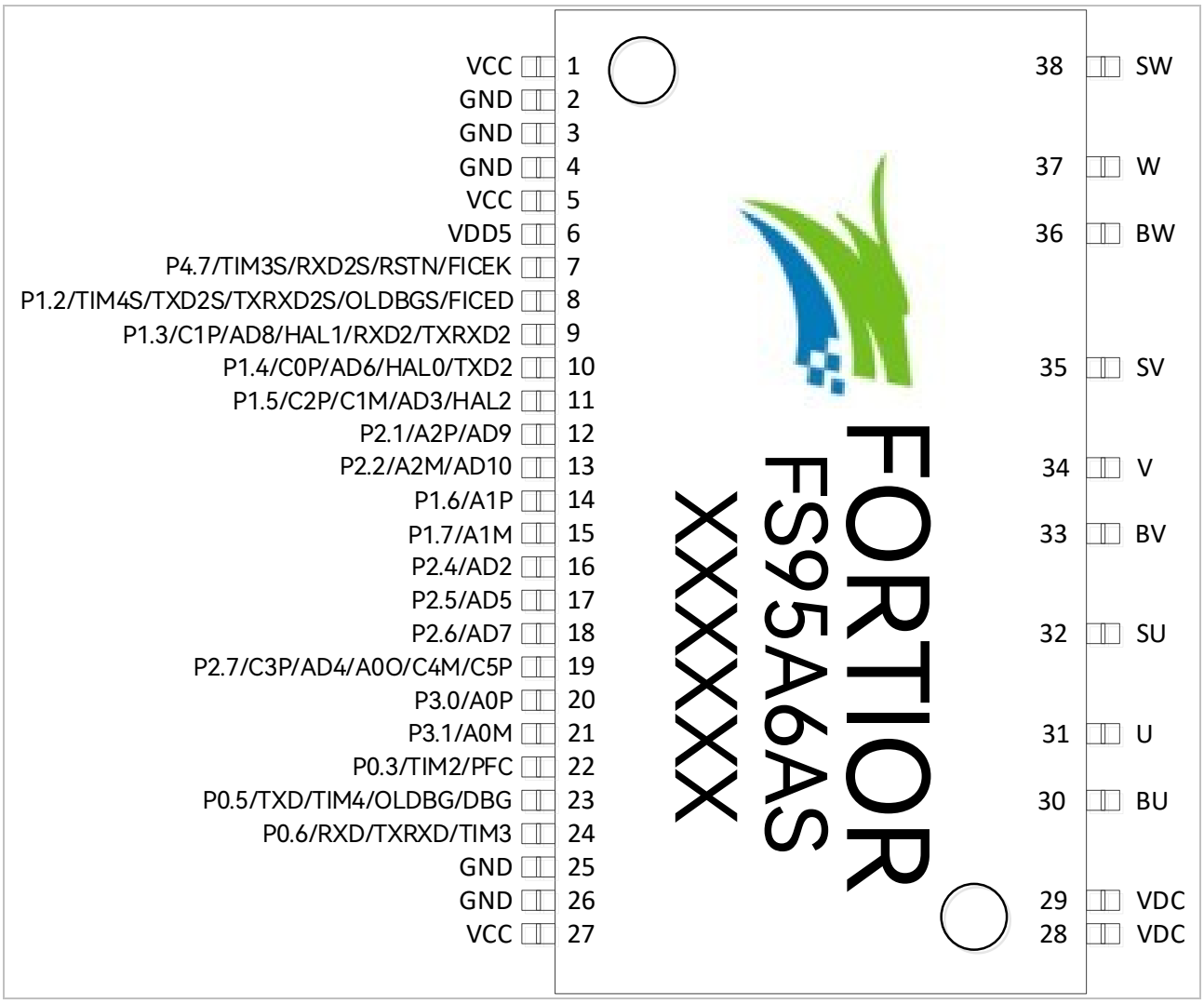
Pin	FS95A6AS SSOP A54-38	IO Type	Function Descriptions
VCC	1	P	Power input, with an external filter capacitor of 4.7μF or above. External power supply 13V~20V is connected to VCC pin, and internal LDO supplies VDD5 voltage.
GND	2	P	Ground
GND	3	P	Ground
GND	4	P	Ground
VCC	5	P	Power input, with an external filter capacitor of 4.7μF or above. External power supply 13V~20V is connected to VCC pin, and internal LDO supplies VDD5 voltage.
VDD5	6	P	5V power supply, with an external capacitor of 1μF or above
P4.7/	7	DI/	GPIO, used as an input only and configured with pull-up or pull-down resistor
TIM3S/		DI/	Timer3 input after function switching
RXD2S/		DI/	UART2 RXD input after function switching
RSTN/		DI/	Input of external reset, with built-in pull-up resistor
FICEK		DI	FICE clock line

Pin	FS95A6AS SSOP A54-38	IO Type	Function Descriptions
P1.2/ TIM4S/ TXD2S/ TXRXD2S/ OLDBGS/ FICED	8	DB/ DB/ DO/ DB/ DO/ DB	GPIO, configurable as INT1 input Timer4 input/output after function switching UART2 TXD output after function switching UART2 TXD output/RXD input in single-wire mode Debug output after function switching in single-wire mode FICE data line
P1.3/ C1P/ AD8/ HAL1/ RXD2/ TXRXD2	9	DB/ AI/ AI/ DI/ DI/ DB	GPIO, configurable as INT1 input CMP1 positive input for comparing V-phase BEMF voltage Input of ADC channel 8 Hall1 logic input UART2 RXD input UART2 TXD output/RXD input in single-wire mode
P1.4/ C0P/ AD6/ HAL0/ TXD2	10	DB/ AI/ AI/ DI/ DO	GPIO, configurable as INT1 input CMP0 positive input for comparing U-phase BEMF voltage Input of ADC channel 6 Hall0 logic level input UART2 TXD output
P1.5/ C2P/ C1M/ AD3/ HAL2	11	DB/ AI/ AI/ AI/ DI	GPIO, configurable as INT1 input CMP2 positive input for comparing W-phase BEMF voltage CMP0 negative input Input of ADC channel 3 Hall2 logic input
P2.1/ A2P/ AD9	12	DB/ AI/ AI	GPIO, configurable as INT1 input AMP2 positive input Input of ADC channel 9
P2.2/ A2M/ AD10	13	DB/ AI/ AI	GPIO, configurable as INT1 input AMP2 negative input Input of ADC channel 10
P1.6/ A1P	14	DB/ AI	GPIO, configurable as INT1 input AMP1 positive input
P1.7/ A1M	15	DB/ AI	GPIO, configurable as INT1 input AMP1 negative input
P2.4/ AD2	16	DB/ AI	GPIO, configurable as INT1 input Input of ADC channel 2 or bus voltage signal
P2.5/ AD5	17	DB/ AI	GPIO, configurable as INT1 input ADC3 CH5 input (for PFC_UAC current sampling)
P2.6/ AD7	18	DB/ AI	GPIO, configurable as INT1 input Input of ADC channel 7 for constant power function

Pin	FS95A6AS SSOP A54-38	IO Type	Function Descriptions
P2.7/ C3P/ AD4/ A0O/ C4M/ C5P	19	DB/ AI/ AI/ AO/ AI/ AI	GPIO, configurable as INT1 input CMP3 positive input Input of ADC channel 4 for bus current sampling AMP0 output CMP4 negative input CMP5 positive input
P3.0/ A0P	20	DB/ AI	GPIO, configurable as INT1 input AMP0 positive input
P3.1/ A0M	21	DB/ AI	GPIO, configurable as INT1 input AMP0 negative input
P0.3/ TIM2/ PFC	22	DB/ DB/ DO	GPIO, configurable as INT0 input Timer2 input capture mode or PWM output before function switching PFC PWM output
P0.5/ TXD/ TIM4/ OLDBG/ DBG	23	DB/ DO/ DB/ DO/ DO	GPIO, configurable as INT0/INT1 input UART1 TXD before function switching Timer4 input/output Single-wire Debug output DBGSEL output (square-wave control)
P0.6/ RXD/ TXRXD/ TIM3	24	DB/ DI/ DB/ DB	GPIO, configurable as INT0/INT1 input UART1 RXD UART1 TXD output/RXD input in single-wire mode Timer3 input/output
GND	25	P	Ground
GND	26	P	Ground
VCC	27	P	Power input, with an external filter capacitor of 4.7μF or above. External power supply 13V~20V is connected to VCC pin, and internal LDO supplies VDD5 voltage.
VDC	28	P	High-voltage power supply
VDC	29	P	High-voltage power supply
BU	30	P	U-phase high-side bootstrap supply
U	31	DO	U-phase output
SU	32	P	U-phase ground
BV	33	P	V-phase high-side bootstrap supply
V	34	DO	V-phase output
SV	35	P	V-phase ground
BW	36	P	W-phase high-side bootstrap supply
W	37	DO	W-phase output
SW	38	P	W-phase ground

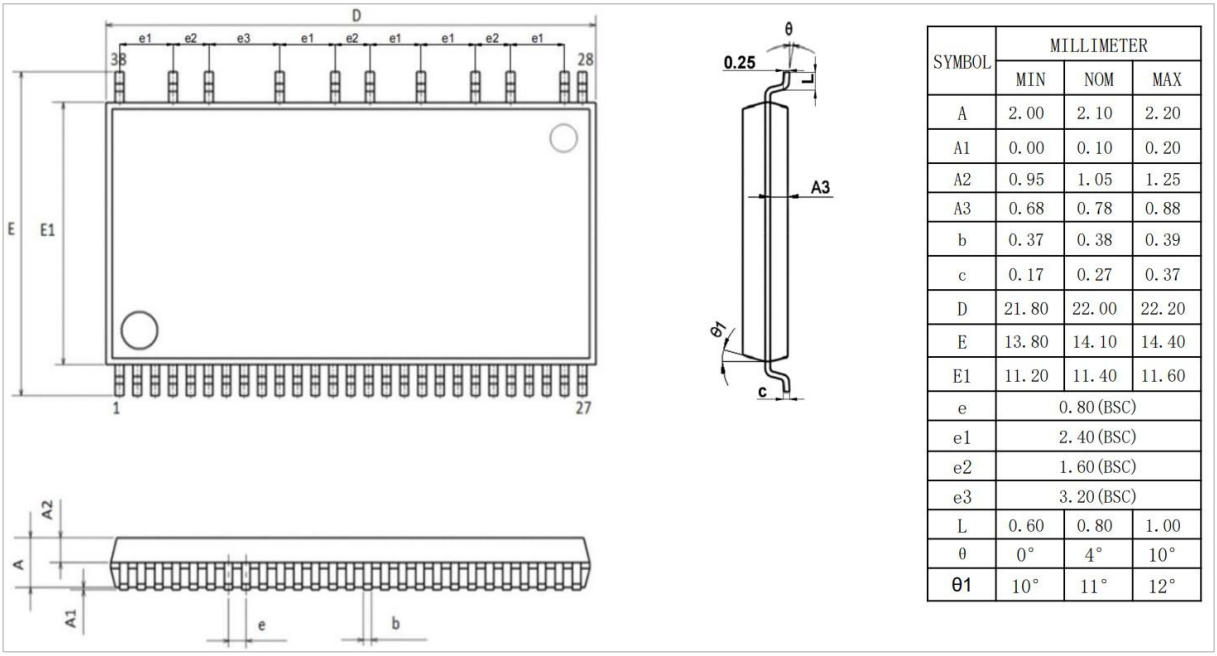
2.2 FS95A6AS SSOP A54-38 Pinout Diagram

Figure 2-1 FS95A6AS SSOP A54-38 Pinout Diagram



3 Package Information

Figure 3-1 FS95A6AS SSOP A54-38 Package Drawings and Dimensions



SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	2.00	2.10	2.20
A1	0.00	0.10	0.20
A2	0.95	1.05	1.25
A3	0.68	0.78	0.88
b	0.37	0.38	0.39
c	0.17	0.27	0.37
D	21.80	22.00	22.20
E	13.80	14.10	14.40
E1	11.20	11.40	11.60
e	0.80 (BSC)		
e1	2.40 (BSC)		
e2	1.60 (BSC)		
e3	3.20 (BSC)		
L	0.60	0.80	1.00
θ	0°	4°	10°
θ1	10°	11°	12°

4 Ordering Information

Table 4-1 Model Selections

Model	Saturation Voltage Reduction (H + L) (Ω)	Single IGBT Drive Current Average (A)	Clock Frequency (MHz)	Flash (kByte)	XRAM (kByte)	Clock Circuit		Drive Type		UART	DMA	GPIO	Timer	Analog Peripherals							Lead-free	Package
														ADC			DAC					
						Internal Fast Clock	Internal Slow Clock	Square-wave	FOC					Number	Channel	Bits	Number	Bits	Operational Amplifier	Comparator		
FS95A6AS	3.46	6	24	32	3.75	√	√	√	√	√	√	18	6	1	11	12	3	9\8\6	3	4	√	SSOP A54-38 (22x11.4mm)

5 Electrical Characteristics

5.1 Absolute Maximum Ratings

Stress values greater than "Absolute Maximum Ratings" listed above may cause irremediable damages to the device. These are stress ratings only, and it is NOT recommended to use your device in conditions that go beyond these stress ratings. Exposure to "Absolute Maximum Ratings" for extended periods may affect device reliability.

Table 5-1 Absolute Maximum Ratings

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
IGBT Collector-to-Emitter Voltage V_{CEMAX}		650	-	-	V
Continuous Collector Current $I_{CMAX(DC)}$ of Single IGBT	T_C = 100°C	-	-	6[1]	A
Continuous Collector Current $I_{CMAX(PLUS)}$ of Single IGBT	T_C = 25°C	-	-	12[1][2]	A
Power Dissipation P_d		-	-	3[3]	W
High-side VDC Supply Voltage V_{DC}		-0.3	-	650[1]	V
U/V/W-phase Output Voltage V_U,V_V,V_W		-0.3	-	650[1]	V
High-side Floating Absolute Voltage V_{BU},V_{BV},V_{BW}		-0.3	-	650[1]	V
High-side Floating Power Supply Voltage V_{BU} - V_U,V_{BV} - V_V,V_{BW} - V_W		-0.3	-	20	V
VCC to GND Voltage		-0.3	-	25	V
VDD5 to GND Voltage		-0.3	5	6.5	V
RSTN/GPIO to GND Voltage		-0.3	-	VDD5 + 0.3	V
Other IOs to GND Voltage (except VCC pins)		-0.3	-	VDD5 + 0.3	V
Junction Temperature T_J		-40	-	150	℃
Storage Temperature T_{STG}		-55	-	150	℃

Note

[1] Power dissipation shall not exceed Pd or ASO

[2] Pw ≤ 10μs and duty cycle ≤ 1%

[3] SSOP A54-38 package is applied to single-channel IGBT at ambient temperature of 25℃

5.2 Recommended Operating Conditions

Table 5-2 Recommended Operating Conditions

(T_A = 25℃ and VCC = 15V unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
High-side VDC Supply Voltage V_{DC}		-	310	400	V
VCC to GND Voltage		13	15	20	V

5.3 Global Electrical Characteristics

Table 5-3 Global Electrical Characteristics

(T_A = 25℃ and VCC = 15V unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Driver					
IGBT output					
IGBT Collector-to-Emitter Breakdown Voltage BV_{CES}	PWM = 0V	650	-	-	V
Drain-to-Source Leakage Current I_{CES}	PWM = 0V, Single IGBT	-	-	100	μA
Collector-to-Emitter Saturation Voltage $V_{CE(SAT)}$	I_D = 6A	-	1.73	2.2	Ω
Emitter-to-Collector Fast Recovery Diode Forward Voltage V_{SD}	I_D = 6A	-	1.9	2.4	V
Power Supply					
VB Quiescent Current I_{BBQ}	PWM = 0V, Single IGBT	25	55	100	μA
VB UVLO Release Voltage V_{BUVL}					
VB UVLO Release Voltage V_{BUVH}	$V_{BX} - V_X$	9.5	10.1	10.7	V
VB UVLO Lockout Voltage V_{BUVL}	$V_{BX} - V_X$	8.5	9.1	9.7	V

5.4 GPIO Electrical Characteristics

Table 5-4 GPIO Electrical Characteristics

(T_A = 25℃ and VCC = 13V ~ 20V unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Turn-on Rise Time	50pF load, from 10% to 90%, T_A = 25℃	-	15	-	ns
Turn-off Fall Time	50pF load, from 90% to 10%, T_A = 25℃	-	13	-	ns
V_{OH} High-level Output Voltage	I_{OH} = 4mA, T_A = -40℃ ~ 85℃	VDD5 - 0.7	-	-	V

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
V_{OL} Low-level Output Voltage	I_{OL} = 4mA, T_A = -40℃ ~ 85℃	-	-	GND + 0.7	V
V_{IH} High-level Input Voltage[1]		0.7*VDD5	-	-	V
V_{IL} Low-level Input Voltage		-	-	0.2*VDD5	V
Pull-up Resistor[2]	VDD5 = 5V	-	33	-	kΩ
Pull-up Resistor[3]	VDD5 = 5V	-	5	-	kΩ
Pull-down Resistor[4]	VDD5 = 5V	-	30	-	kΩ

Note

[1] When VDD5 = 5V, minimum value of VIH is 0.6*VDD5

[2] GPIOs except P1[5:3]

[3] P1[5:3]

[4] P4[7]

5.5 ADC Electrical Characteristics

Table 5-5 ADC Electrical Characteristics

(T_A = 25℃ and VCC = 13V ~ 20V unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
INL (Integral Nonlinearity)	12-bit	-	2	-	LSB
DNL (Differential Nonlinearity)	12-bit	-	1.5	-	LSB
OFFSET (Offset Error)	12-bit	-	6	-	LSB
SNR (Signal-to-noise Ratio)	f_{IN} = 350kHz	-	70.8	-	dB
ENOB (Effective Number of Bits)	f_{IN} = 350kHz	-	10.5	-	bit
SFDR (Spurious-free Dynamic Range)	f_{IN} = 350kHz	-	68.2	-	dB
THD (Total Harmonic Distortion)	f_{IN} = 350kHz	-	67	-	dB
R_{IN} Input Resistance		-	800	-	Ω
C_{IN} Input Capacitance		-	30	-	pF
Conversion Time		-	13	-	ADCLK[1]
Sampling Time		3	-	63	ADCLK2[2]

Note

[1] ADCLK = 24MHz

[2] ADCLK2 = 12MHz

5.6 Operational Amplifier Electrical Characteristics

Table 5-6 Operational Amplifier Electrical Characteristics

(T_A = 25℃ and VCC = 13V ~ 20V unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
V_{ICMR} Common-mode Input Voltage Range		0	-	VDD5 - 1.5	V
V_{OS} Operational Amplifier Offset Voltage	T_A = 25℃	-	5	10	mV
A_{OL} Open-loop Gain	R_L = 100kΩ	-	80	-	dB
Unity-gain Bandwidth (U_{GBW})	C_L = 40pF	6	10	-	MHz
Slew Rate (SR)	C_L = 40pF	10	15	-	V/μs
Operational Amplifier Gain[1][2]	AMP_CR1[AMP0_GAIN] = 001	3.76	4	4.24	-
	AMP_CR1[AMP0_GAIN] = 010	5.64	6	6.36	-
	AMP_CR1[AMP0_GAIN] = 011	7.5	8	8.5	-
	AMP_CR1[AMP0_GAIN] = 100	9.4	10	10.6	-

Note

[1] The operational amplifier gain is measured when both positive and negative inputs of the operational amplifier are connected in series with 2kΩ resistors. The operational amplifier gain varies with external resistors

[2] AMP0 is used as an example. For other operational amplifiers, see AMP_CR1 (0x4034) for the configurations.AMP_CR1 (0x4034)

5.7 BEMF Electrical Characteristics

Table 5-7 BEMF Electrical Characteristics

(T_A = 25℃ and VCC = 13V ~ 20V unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
BEMF Built-in Resistor		5.4	6.8	8.2	kΩ
Relative Accuracy between BEMF Built-in Resistors		-	1	-	%

5.8 OSC Electrical Characteristics

Table 5-8 OSC Electrical Characteristics

(T_A = -40℃ ~ 85℃, VCC = 13V ~ 20V unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
System Clock Rate		23.5	24	24.5	MHz
Low-speed Clock Rate		29	32.8	37	kHz

Note

SYSCLK refers to system clock rate, and T to system clock cycle. Unless otherwise specified, the system clock rate of chip is 24MHz and T = 1/SYSCLK

5.9 Reset Electrical Characteristics

Table 5-9 Reset Electrical Characteristics

(T_A = 25℃ and VCC = 13V ~ 20V unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Minimum Time for RSTN Released to Low		50	-	-	μs
VDD5 Reset Threshold	Reset Voltage LVR = 3.0V	2.8	3.0	3.2	V

5.10 LDO Electrical Characteristics

Table 5-10 LDO Electrical Characteristics

(T_A = 25℃ and VCC = 13V ~ 20V unless otherwise specified)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
VDD5 Voltage		4.8	5	5.2	V

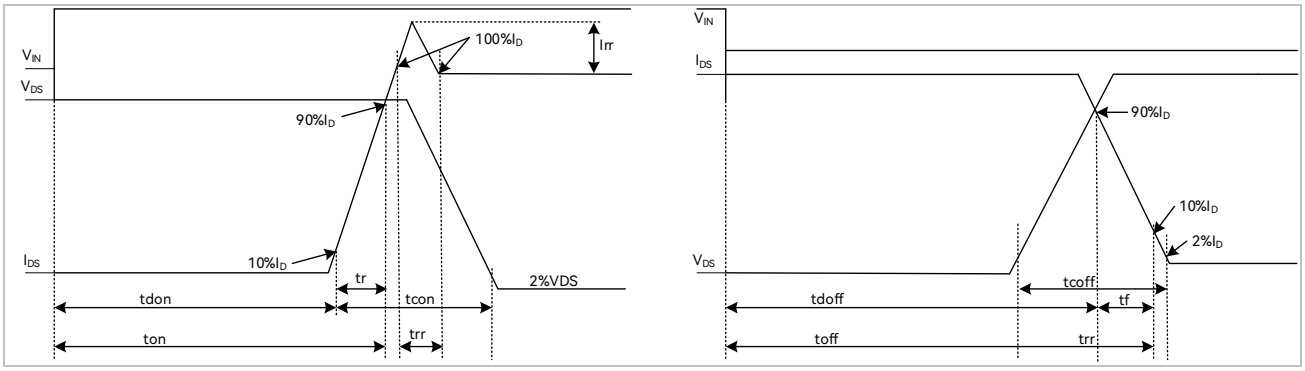
5.11 Switching Characteristics

Table 5-11 Switching Characteristics

(T_A = 25℃ and VCC = 15V unless otherwise specified)

Parameter	Test Conditions	Typ.	Unit
High-side Switching Time			
$t_{dH(on)}$	VDC = 300V, VCC = 15V, I_D = 2.5A	820	ns
t_{rH}		145	ns
t_{rrH}		180	ns
$t_{dH(off)}$		615	ns
t_{fH}		200	ns
Low-side Switching Time			
$t_{dL(on)}$	VDC = 300V, VCC = 15V, I_D = 2.5A	790	ns
t_{rL}		135	ns
t_{rrL}		180	ns
$t_{dL(off)}$		600	ns
t_{fL}		210	ns

Figure 5-1 Switching Time Waveform Definitions



5.12 Package Thermal Resistance

Table 5-12 SSOP A54-38 Package Thermal Resistance

Parameter	Test Conditions	Typ.	Unit
Junction-to-ambient Thermal Resistance θ_{JA}[1]	JEDEC standard, 1S0P PCB	42.5	℃/W
Junction-to-package-top Thermal Resistance Ψ_{JT}	JEDEC standard, 1S0P PCB	12.5	℃/W

Note

[1] The actual measurements may vary depending on the conditions

5.13 Mounting Torque

Table 5-13 Mounting Torque

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Mounting Torque	M3 screw[1]; Mounting torque 0.4N·m is suggested	0.3	0.4	0.5	N·m

Note

[1] When tightening the screws, it is recommended to use flat washers (ISO7089 ~ 7094)

6 Reset Control

6.1 Reset Source(RST_SR)

7 reset sources in the chip:

- Power-on Reset(RSTPOW)
- External Pin Reset(RSTEXT)
- Low-voltage Reset(RSTLVD)
- Watchdog Reset(RSTWDT)
- Flash Illegal Operation Reset(RSTFED)
- Debug Reset(RSTDBG)
- Soft Reset(SOFTR)

The reset flag is queryable and recorded in register RST_SR. Following the last reset, the affected reset flag is set to “1” and all other reset flags are cleared to “0”. In order to clear a reset flag, you can set RST_SR[RSTCLR] to “1” by software so that RST_SR[7:3]&RST_SR[0] are cleared. After reset, MCU starts the program from address 0.

6.2 Reset Enable

See the corresponding control registers.

6.3 RSTEXT, RSTPOW

The chip resets when RSTN pin remains low for 50μs.

The chip resets when the chip powers on and the voltage settles above the reset voltage threshold.

6.4 Low-voltage Reset

The chip’s internal circuitry monitors VCC voltage. When VCC voltage drops to a level below the reset voltage threshold, the internal monitor circuitry sends the reset signal to reset the chip.

Configuring corresponding register enables the low voltage monitor circuitry and sets the low voltage threshold.

6.5 Watchdog Timeout Reset

After enabling the watchdog timer, the software must repeatedly feed the dog in the program to clean the timer. If the program fails or fails to feed the watchdog, the watchdog timer will overflow or trigger a chip reset.

6.6 RSTFED Reset

The Flash memory can be programmed for read/write/erase operations with MOVX instructions. A Flash error detector reset (RSTFED) occurs if a Flash erase is attempted targeting the last sector (0x7F00 ~ 0x7FFF) or a Flash write is attempted targeting the last byte (0x7FFF). RSTFED is always enabled and cannot be disabled.

6.7 RSTDBG Reset

Click Reset button of IDE to send a debug reset signal when the chip enters the debug state.

6.8 Soft Reset

The chip resets immediately when RST_SR[SOFTR] is set to “1”. After reset, the flag RST_SR[SOFTR] is set to “1”.

6.9 Reset Register

6.9.1 RST_SR (0xC9)

Bit	7	6	5	4	3	2	1	0
Name	RSTPOW/ RSTCLR	RSTEXT	RSTLVD	RSV	RSTWDT	RSTFED	RSTDBG	SOFTR
Type	R/W1	R	R	-	R	R	R	R/W1
Reset	-	-	-	-	-	-	-	-

Bit	Name	Description
[7]	RSTPOW/ RSTCLR	Power-on Reset Read: 0: The last reset is not power-on reset 1: The last reset is power-on reset Write: 0: Meaningless 1: Clear the reset bit of RST_SR[7:3] & RST_SR[0]
[6]	RSTEXT	External Pin Reset 0: The last reset is not external pin reset 1: The last reset is external pin reset
[5]	RSTLVD	Low-voltage Reset 0: The last reset is not low-voltage reset 1: The last reset is low-voltage reset
[4]	RSV	Reserved
[3]	RSTWDT	Watchdog Reset 0: The last reset is not watchdog reset 1: The last reset is watchdog reset
[2]	RSTFED	Flash Illegal Operation Reset 0: The last reset is not Flash illegal operation reset 1: The last reset is Flash illegal operation reset
[1]	RSTDBG	Debug Reset 0: The last reset is not Debug reset 1: The last reset is Debug reset
[0]	SOFTR	Soft Reset Read: 0: The last reset is not soft reset 1: The last reset is soft reset Write: 0: Meaningless 1: Soft reset is triggered

7 Interrupt

7.1 Interrupt Introduction

The chip includes an interrupt system with a total of 16 interrupt sources. Each interrupt source can be individually programmed in IP0 ~ IP3 registers with one of four priority levels. Interrupt flags (IF) are located in an SFRs or XSFRs. The associated IF is set by the hardware to “1” when the internal circuitry or an external source meets the interrupt conditions. If IE[EA] = 1 and both the associated interrupt EA and IF bits are set to “1”, an interrupt request is generated and sent to CPU. If no other interrupt service routine (ISR) of greater priority is currently being serviced, the system enters interrupt state to service the requesting ISR.

Each interrupt source except the Reset Interrupt can be assigned a priority level. A low priority interrupt can be preempted by a high priority interrupt. The low priority interrupt will not be serviced until the ISR for the high priority interrupt completes. An interrupt will not be preempted by another of the same priority level. Each interrupt source can be individually configured to one of four priority levels in the Interrupt Priority (IP) register. Priority level assigned ascends from 0 to 3 and is defaulted to 0. If two interrupt requests are generated at the same time, the interrupt with the higher priority is serviced first. If two interrupt sources have the same priority level, a fixed priority order is used to arbitrate. See Table 7-1 for the interrupt sources and default priority orders, where the lower the mark the higher the priority level.

7.2 Interrupt Source Enable

IE[EA] is the global interrupt enable bit. MCU does not respond to any interrupt request when IE[EA] = 0.

Each interrupt source can be individually enabled or disabled by configuring the corresponding interrupt enable bit in an SFR or XSFR. When the enable bit of the global interrupt or an interrupt is cleared, the interrupt flag that is set to “1” is held in a pending state. Once the enable bit is set to “1”, MCU immediately enters the interrupt subroutine. Therefore, make sure to clear corresponding interrupt flag bit before enabling the interrupt.

7.3 External Interrupt

The external interrupt has 2 interrupt sources: INT0 and INT1. They both can be configured as interrupt on rising edge, interrupt on falling edge or interrupt on edge changes (rise or fall).

The digital input signals from P0.3, P0.5 ~ P0.6, P1.3 ~ P1.4, P2.5, P4.7 and output signals from CMP4 can be used to trigger an INT0. While INT0 is the interrupt trigger source of only one pin. The interrupt source is selected through LVSR[EXT0CFG] bit. These trigger sources share one interrupt entry point, one interrupt flag bit TCON[IF0] and one interrupt enable bit IE[EX0]. TCON[IT0] bit selects the interrupt edge. IP0[PX0] bit configures the priority level.

The digital input signals from P0.5 ~ P0.6, P1.2 ~ P1.7, P2.1, P2.2, P2.4 ~ P2.7 and P3.0 ~ P3.1 can be used to trigger an INT1. EXT1L_IF and EXT1H_IF are interrupt flag bits, and EXT1L_IE and EXT1H_IE are interrupt enable bits. Each trigger source has a corresponding interrupt flag bit and an interrupt enable bit. INT1 can select multiple trigger sources that are recognized by EXT1L_IF and EXT1H_IF in the interrupt subroutine. These 16 interrupt sources share one interrupt entry and one interrupt enable bit IE[EX1]. To enable INT1, first set IE[EX1] to “1” and then configure the corresponding enable bit. The interrupt edge is configured by TCON[IT1] bit, and the priority level by IP0[PX1] bit. See 7.5.7 EXT1L_IE (0xD1) ~ 7.5.10 EXT1H_IF (0xD4).

7.4 Interrupt Introduction

Table 7-1 Interrupt Introduction

Interrupt Source	Priority Order	Vector Address	Interrupt Flag	Cleared by Software?	Enable Bit	Priority Control
Reset	Highest	0x0000	None	N	Always enabled	Highest
LVW Interrupt TSD Interrupt	0	0x0003	LVSR[0] TCON[5]	Y	CCFG1[6] IE[1]	IP0[1:0]
INT0	1	0x000B	TCON[2]	Y	IE[0]	IP0[3:2]
INT1	2	0x0013	EXT1L_IF[7:0] EXT1H_IF[7:0]	Y	IE[2]	IP0[5:4]
FG Interrupt DRV CM Interrupt	3	0x001B	DRV_SR[5:4]	Y	DRV_SR[3] DRV_SR[2:0]	IP0[7:6]
Timer2 Interrupt	4	0x0023	TIM2_CR1[7:5]	Y	TIM2_CR1[4:3] TIM2_CR0[3]	IP1[1:0]
Timer1 Interrupt	5	0x002B	TIM1_SR[5:0]	Y	TIM1_IER[5:0]	IP1[3:2]

Interrupt Source	Priority Order	Vector Address	Interrupt Flag	Cleared by Software?	Enable Bit	Priority Control
ADC Interrupt	6	0x0033	ADC_CR[0]	Y	ADC_CR[1]	IP1[5:4]
CMP0/1/2 Interrupt Hall Interrupt	7	0x003B	CMP_SR[6:4] HALL_CR[7]	Y	CMP_CR0[5:0] HALL_CR[6]	IP1[7:6]
RTC Interrupt	8	0x0043	RTC_STA[6]	Y	IE[6]	IP2[1:0]
Timer3 Interrupt	9	0x004B	TIM3_CR1[7:5]	Y	TIM3_CR1[4:3] TIM3_CR0[3]	IP2[3:2]
Systick Interrupt	10	0x0053	DRV_SR[7]	Y	SYST_CR[1:0]	IP2[5:4]
Timer4 Interrupt	11	0x005B	TIM4_CR1[7:5]	Y	TIM4_CR1[4:3] TIM4_CR0[3]	IP2[7:6]
CMP3 Interrupt	12	0x0063	CMP_SR[7]	Y	CMP_CR0[7:6]	IP3[1:0]
UART1 Interrupt	13	0x006B	UT_CR[1:0]	Y	IE[4]	IP3[3:2]
UART2 Interrupt	14	0x0073	UT2_CR[1:0]	Y	UT2_BAUDH[5]	IP3[5:4]
DMA Interrupt	15	0x007B	DMA0_CR0[0] DMA1_CR0[0]	Y	DMA0_CR0[2]	IP3[7:6]

Note

- UT_CR[RI], UT_CR[TI], UT2_CR[UT2RI], UT2_CR[UT2TI], DMA0_CR0[DMAIF] and DMA1_CR0[DMAIF] flags can be cleared to “0” or set to “1” by software; when these flags are set to “1”, an interrupt request is generated. Other interrupt flags can only be cleared to “0” by software, and setting them to “1” has no meaning.
- For a register containing multiple interrupt flags, you can write a “1” to active interrupt flags in order to prevent clearing a interrupt flag to “0”. Take DRV_SR as an example. When DRV_SR[SYSTIF] is cleared to “0” by software, you can use the statement DRV_SR = (DRV_SR&0x7F) | 0x30 to prevent the software from clearing DRV_SR[FGIF] and DRV_SR[DCIF] to “0”.

7.5 Interrupt Register

7.5.1 IE (0xA8)

Bit	7	6	5	4	3	2	1	0
Name	EA	RTCIE	RSV	ES0	RSV	EX1	TSDIE	EX0
Type	R/W	R/W	-	R/W	-	R/W	R/W	R/W
Reset	0	0	-	0	-	0	0	0

Bit	Name	Description
[7]	EA	All Interrupts Enable 0: Disable 1: Enable
[6]	RTCIE	RTC Interrupt Enable 0: Disable 1: Enable
[5]	RSV	Reserved
[4]	ES0	UART1 Interrupt Enable 0: Disable 1: Enable
[3]	RSV	Reserved
[2]	EX1	INT1 Enable 0: Disable 1: Enable
[1]	TSDIE	TSD Interrupt Enable 0: Disable 1: Enable
[0]	EX0	INT0 Enable 0: Disable 1: Enable

7.5.2 IP0 (0x8A)

Bit	7	6	5	4	3	2	1	0
Name	PDRV		PX1		PX0		PLVW_TSD	
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:6]	PDRV	FG/DRV Compare Match Interrupt Priority Setting
[5:4]	PX1	External Interrupt 1 (INT1) Priority Setting

[3:2]	PX0	External Interrupt 0 (INT0) Priority Setting
[1:0]	PLVW_TSD	LVW/TSD Interrupt Priority Setting

Note

Priority level assigned ascends from 0 to 3, totaling 4 levels

7.5.3 IP1 (0x8B)

Bit	7	6	5	4	3	2	1	0
Name	PCMP		PADC		PTIM1		PTIM2	
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:6]	PCMP	CMP0/1/2 Interrupt Priority Setting
[5:4]	PADC	ADC Interrupt Priority Setting
[3:2]	PTIM1	Timer1 Interrupt Priority Setting
[1:0]	PTIM2	Timer2 Interrupt Priority Setting

Note

Priority level assigned ascends from 0 to 3, totaling 4 levels

7.5.4 IP2 (0x8C)

Bit	7	6	5	4	3	2	1	0
Name	PTIM4		PSYSTICK		PTIM3		PRTC	
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:6]	PTIM4	Timer4 Interrupt Priority Setting
[5:4]	PSYSTICK	Systick Interrupt Priority Setting
[3:2]	PTIM3	Timer3 Interrupt Priority Setting
[1:0]	PRTC	RTC Interrupt Priority Setting

Note

Priority level assigned ascends from 0 to 3, totaling 4 levels

7.5.5 IP3 (0x8D)

Bit	7	6	5	4	3	2	1	0
Name	PDMA		PUT2		PUT1		PCMP3	
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:6]	PDMA	DMA Interrupt Priority Setting
[5:4]	PUT2	UART2 Interrupt Priority Setting
[3:2]	PUT1	UART1 Interrupt Priority Setting
[1:0]	PCMP3	CMP3 Interrupt Priority Setting

Note

Priority level assigned ascends from 0 to 3, totaling 4 levels

7.5.6 TCON (0x88)

Bit	7	6	5	4	3	2	1	0
Name	RSV		TSDIF	IT1		IF0	IT0	
Type	-	-	R/W0	R/W	R/W	R/W0	R/W	R/W
Reset	-	-	0	0	0	0	0	0

Bit	Name	Description
[7:6]	RSV	Reserved
[5]	TSDIF	TSD Interrupt Flag This bit is set by hardware to “1” when an over-temperature event occurs. Read: 0: No interrupt pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect Note This flag is often used with the overtemperature status bit LVSR[TSDF]
[4:3]	IT1	INT1 Trigger Level Selection 00: Interrupt on rising edge 01: Interrupt on falling edge 1X: Interrupt on edge changes (rise or fall)
[2]	IF0	External Interrupt 0 (INT0) Interrupt Flag Read: 0: No Interrupt Pending

		1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[1:0]	IT0	INT0 Trigger Level Selection 00: Interrupt on rising edge 01: Interrupt on falling edge 1X: Interrupt on edge changes (rise or fall)

7.5.7 EXT1L_IE (0xD1)

Bit	7	6	5	4	3	2	1	0
Name	P17_IE	P16_IE	P15_IE	P14_IE	P13_IE	P12_IE	P06_IE	P05_IE
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	P17_IE	P1.7 INT1 Enable 0: Disable 1: Enable
[6]	P16_IE	P1.6 INT1 Enable 0: Disable 1: Enable
[5]	P15_IE	P1.5 INT1 Enable 0: Disable 1: Enable
[4]	P14_IE	P1.4 INT1 Enable 0: Disable 1: Enable
[3]	P13_IE	P1.3 INT1 Enable 0: Disable 1: Enable
[2]	P12_IE	P1.2 INT1 Enable 0: Disable 1: Enable
[1]	P06_IE	P0.6 INT1 Enable 0: Disable 1: Enable
[0]	P05_IE	P0.5 INT1 Enable 0: Disable 1: Enable

7.5.8 EXT1L_IF (0xD2)

Bit	7	6	5	4	3	2	1	0
Name	P17_IF	P16_IF	P15_IF	P14_IF	P13_IF	P12_IF	P06_IF	P05_IF
Type	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	P17_IF	P1.7 INT1 Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[6]	P16_IF	P1.6 INT1 Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[5]	P15_IF	P1.5 INT1 Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[4]	P14_IF	P1.4 INT1 Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[3]	P13_IF	P1.3 INT1 Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect

[2]	P12_IF	P1.2 INT1 Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[1]	P06_IF	P0.6 INT1 Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[0]	P05_IF	P0.5 INT1 Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect

7.5.9 EXT1H_IE (0xD3)

Bit	7	6	5	4	3	2	1	0
Name	P31_IE	P30_IE	P27_IE	P26_IE	P25_IE	P24_IE	P22_IE	P21_IE
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	P31_IE	P3.1 INT1 Enable 0: Disable 1: Enable
[6]	P30_IE	P3.0 INT1 Enable 0: Disable 1: Enable
[5]	P27_IE	P2.7 INT1 Enable 0: Disable 1: Enable
[4]	P26_IE	P2.6 INT1 Enable 0: Disable 1: Enable
[3]	P25_IE	P2.5 INT1 Enable

		0: Disable 1: Enable
[2]	P24_IE	P2.4 INT1 Enable 0: Disable 1: Enable
[1]	P22_IE	P2.2 INT1 Enable 0: Disable 1: Enable
[0]	P21_IE	P2.1 INT1 Enable 0: Disable 1: Enable

7.5.10 EXT1H_IF (0xD4)

Bit	7	6	5	4	3	2	1	0
Name	P31_IF	P30_IF	P27_IF	P26_IF	P25_IF	P24_IF	P22_IF	P21_IF
Type	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0
Reset	0	0	0	0	0	0	0	0

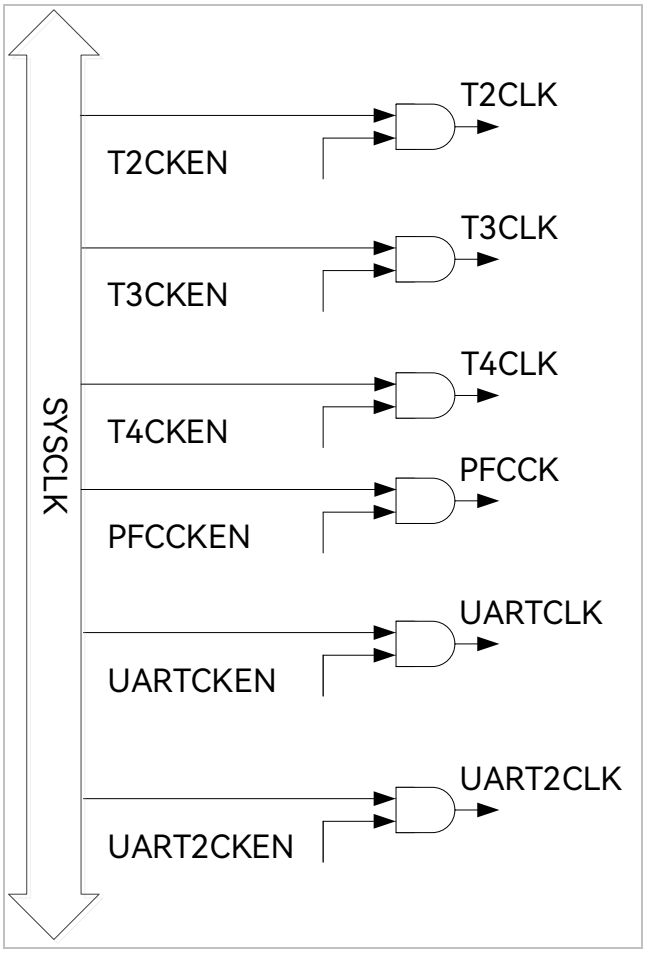
Bit	Name	Description
[7]	P31_IF	P3.1 INT1 Interrupt Flag Read: 0: No interrupt occurs 1: Interrupt occurs Write: 0: Cleared 1: Meaningless
[6]	P30_IF	P3.0 INT1 Interrupt Flag Read: 0: No interrupt occurs 1: Interrupt occurs Write: 0: Cleared 1: Meaningless
[5]	P27_IF	P2.7 INT1 Interrupt Flag Read: 0: No interrupt occurs 1: Interrupt occurs Write: 0: Cleared 1: Meaningless
[4]	P26_IF	P2.6 INT1 Interrupt Flag Read:

		0: No interrupt occurs 1: Interrupt occurs Write: 0: Cleared 1: Meaningless
[3]	P25_IF	P2.5 INT1 Interrupt Flag Read: 0: No interrupt occurs 1: Interrupt occurs Write: 0: Cleared 1: Meaningless
[2]	P24_IF	P2.4 INT1 Interrupt Flag Read: 0: No interrupt occurs 1: Interrupt occurs Write: 0: Cleared 1: Meaningless
[1]	P22_IF	P2.2 INT1 Interrupt Flag Read: 0: No interrupt occurs 1: Interrupt occurs Write: 0: Cleared 1: Meaningless
[0]	P21_IF	P2.1 INT1 Interrupt Flag Read: 0: No interrupt occurs 1: Interrupt occurs Write: 0: Cleared 1: Meaningless

8 Clock Gating

Timer2, Timer3, Timer4, PFC and UART modules are designed with a separate clock gating circuit. The module requires enabling the corresponding clock bit in CK_CR during operation.

Figure 8-1 Clock Tree



8.1 Clock Gating Register

8.1.1 CK_CR (0x91)

Bit	7	6	5	4	3	2	1	0
Name	UART2CKEN	UARTCKEN	PFCCKEN	RSV		T4CKEN	T3CKEN	T2CKEN
Type	R/W	R/W	R/W	-	-	R/W	R/W	R/W
Reset	0	0	0	-	-	0	0	0

Bit	Name	Description
[7]	UART2CKEN	UART2 Module Clock Enable 0: Disable 1: Enable
[6]	UARTCKEN	UART Clock Enable 0: Disable 1: Enable
[5]	PFCCKEN	PFC Module Clock Enable 0: Disable 1: Enable
[4:3]	RSV	Reserved
[2]	T4CKEN	Timer4 Module Clock Enable 0: Disable 1: Enable
[1]	T3CKEN	Timer3 Module Clock Enable 0: Disable 1: Enable
[0]	T2CKEN	Timer2 Module Clock Enable 0: Disable 1: Enable

9 UART

9.1 UART Introduction

UART is a full-duplex or half-duplex serial data exchange interface as shown in Figure 9-1.The baud rate is configurable and supports DMA transmission. Figure 9-2 depicts the UART timing.

Figure 9-1 Block Diagram of UART Module

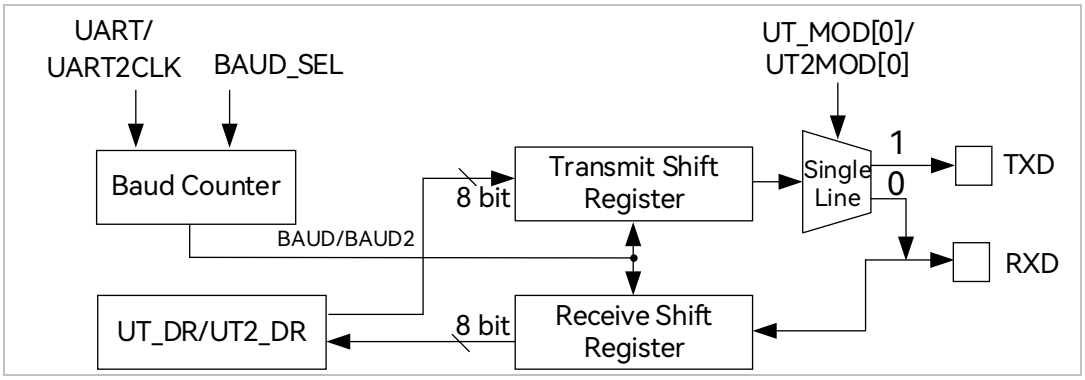
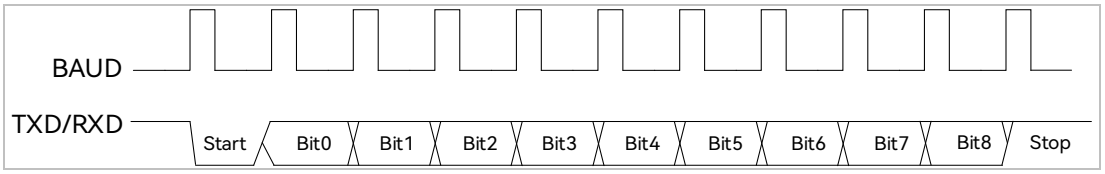


Figure 9-2 UART Timing Diagram



9.2 UART Operations

The corresponding registers shall be enabled before using UART feature. See 21.3.14 PH_SEL (0x404C) (bits [6] and [5]) ~ 21.3.15 PH_SEL1 (0x404D) (bit [7]) for more details.

9.2.1 UART1 Operation Instructions

9.2.1.1 UART1 Mode0

UART1 mode0 works in single-wire half-duplex mode. RXD pin is configured as both an output (Transmit Data Bus) and an input (Receive Data Bus). It uses a total of 10 bits (1 start bit, 8 data bits and 1 stop bit) to receive or transmit data. The baud rate is configured by UT_BAUD[BAUD].

Sending Data: Write the data to UT_DR and clear UT_CR[TI]. RXD outputs 10-bit data. UT_CR[TI] is set to "1" after the transmission is completed.

Receiving Data: Set UT_CR[REN] to "1" to receive the data and clear UT_CR[RI]. The data is received via RXD. After the data is received, UT_CR[RI] is set to “1” and UT_DR is read to obtain the data.

9.2.1.2 UART1 Mode1

UART1 mode1 works in full/half duplex mode. TXD pin is configured as an output (Transmit Data Bus), and RXD pin as an input (Receive Data Bus). It uses a total of 10 bits (1 start bit, 8 data bits and 1 stop bit) to receive or transmit data. The baud rate is configured by UT_BAUD[BAUD].

Sending Data: Write the data to UT_DR and clear UT_CR[TI]. TXD outputs 10-bit data. UT_CR[TI] is set to "1" after the transmission is completed.

Receiving Data: Set UT_CR[REN] to "1" to receive the data and clear UT_CR[RI]. The data is received via RXD. After the data is received, UT_CR[RI] is set to "1" and UT_DR is read to obtain the data.

9.2.1.3 UART1 Mode2

UART1 mode2 works in single-wire half-duplex mode. RXD pin is configured as both an output (Transmit Data Bus) and an input (Receive Data Bus). It uses a total of 11 bits (1 start bit, 9 data bits and 1 stop bit) to receive or transmit data. The baud rate is configured by UT_BAUD[BAUD].

Sending Data: Write the first 8 bits of the data to UT_DR and the 9^{th} bit to UT_CR[TB8], and clear UT_CR[TI]. TXD outputs 11-bit data. UT_CR[TI] is set to “1” after the transmission is completed.

Receiving Data: Set UT_CR[REN] to “1” to receive the data and clear UT_CR[RI]. The data is received via RXD. After the data is received, UT_CR[RI] is set to “1”. UT_CR[RB8] stores the 9^{th} bit of the data, and UT_DR stores the first 8 bits.

9.2.1.4 UART1 Mode3

UART1 mode3 works in full/half duplex mode. TXD pin is configured as an output (Transmit Data Bus), and RXD as an input (Receive Data Bus). It uses a total of 11 bits (1 start bit, 9 data bits and 1 stop bit) to receive or transmit data. The baud rate is configured by UT_BAUD[BAUD].

Sending Data: Write the first 8 bits of the data to UT_DR and the 9th bit to UT_CR[TB8], and clear UT_CR[TI]. TXD outputs 11-bit data. UT_CR[TI] is set to “1” after the transmission is completed.

Receiving Data: Set UT_CR[REN] to “1” to receive the data and clear UT_CR[RI]. The data is received via RXD. After the data is received, UT_CR[RI] is set to “1”. UT_CR[RB8] stores the 9th bit of the data, and UT_DR stores the first 8 bits.

9.2.1.5 UART1 Interrupt Sources

UART1 interrupt sources include:

- After the data is sent via UART1, UT_CR[TI] is set to “1” by hardware.
- After the data and STOP are received via UART1, UT_CR[RI] is set to “1” by hardware.

9.2.2 UART2 Operation Instructions

9.2.2.1 UART2 Mode0

UART2 mode0 works in single-wire half-duplex mode. RXD pin is configured as both an output (Transmit Data Bus) and an input (Receive Data Bus). It uses a total of 10 bits (1 start bit, 8 data bits and 1 stop bit) to receive or transmit data. The baud rate is configured by UT2_BAUD[BAUD2]

Sending Data: Write the data to UT2_DR and clear UT2_CR[UT2TI]. RXD outputs 10-bit data. UT2_CR[UT2TI] is set to "1" after the transmission is completed.

Receiving Data: Set UT2_CR[UT2REN] to "1" to receive the data and clear UT2_CR[UT2RI]. The data is received via RXD. After the data is received, UT2_CR[UT2RI] is set to “1” and UT2_DR is read to obtain the data.

9.2.2.2 UART2 Mode1

UART2 mode1 works in full/half duplex mode. TXD pin is configured as an output (Transmit Data Bus), and RXD pin as an input (Receive Data Bus). It uses a total of 10 bits (1 start bit, 8 data bits and 1 stop bit) to receive or transmit data. The baud rate is configured by UT2_BAUD[BAUD2].

Sending Data: Write the data to UT2_DR and clear UT2_CR[UT2TI]. TXD outputs 10-bit data. UT2_CR[UT2TI] is set to "1" after the transmission is completed.

Receiving Data: Set UT2_CR[UT2REN] to "1" to receive the data and clear UT2_CR[UT2RI]. The data is received via RXD. After the data is received, UT2_CR[UT2RI] is set to "1" and UT2_DR is read to obtain the data.

9.2.2.3 UART2 Mode2

UART2 mode2 works in single-wire half-duplex mode. RXD pin is configured as both an output (Transmit Data Bus) and an input (Receive Data Bus). It uses a total of 11 bits (1 start bit, 9 data bits and 1 stop bit) to receive or transmit data. The baud rate is configured by UT2_BAUD[BAUD2].

Sending Data: Write the first 8 bits of the data to UT2_DR and the 9th bit to UT2_CR[UT2TB8], and clear UT2_CR[UT2TI]. TXD outputs 11-bit data. UT2_CR[UT2TI] is set to “1” after the transmission is completed.

Receiving Data: Set UT2_CR[UT2REN] to “1” to receive the data and clear UT2_CR[UT2RI]. The data is received via RXD. After the data is received, UT2_CR[UT2RI] is set to “1”. UT2_CR[UT2RB8] stores the 9th bit of the data, and UT2_DR stores the first 8 bits.

9.2.2.4 UART2 Mode3

UART2 mode3 works in full/half duplex mode. TXD pin is configured as an output (Transmit Data Bus), and RXD as an input (Receive Data Bus). It uses a total of 11 bits (1 start bit, 9 data bits and 1 stop bit) to receive or transmit data. The baud rate is configured by UT2_BAUD[BAUD2].

Sending Data: Write the first 8 bits of the data to UT2_DR and the 9th bit to UT2_CR[UT2TB8], and clear UT2_CR[UT2TI]. TXD outputs 11-bit data. UT2_CR[UT2TI] is set to “1” after the transmission is completed.

Receiving Data: Set UT2_CR[UT2REN] to “1” to receive the data and clear UT2_CR[UT2RI]. The data is received via RXD. After the data is received, UT2_CR[UT2RI] is set to “1”. UT2_CR[UT2RB8] stores the 9th bit of the data, and UT2_DR stores the first 8 bits.

9.2.2.5 UART2 Interrupt Sources

UART2 interrupt sources include:

- After the data is sent via UART2, UT2_CR[UT2TI]is set to “1” by hardware.
- After the data and STOP are received via UART2, UT2_CR[UT2RI] is set to “1” by hardware.

9.3 UART1 Registers

9.3.1 UT_CR (0x98)

Bit	7	6	5	4	3	2	1	0
Name	UT_MOD		SM2	REN	TB8	RB8	TI	RI
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:6]	UT_MOD	Mode Selection 00: Mode0 01: Mode1 10: Mode2 11: Mode3
[5]	SM2	Communication Mode Selection 0: Single-device Communication 1: Multi-device Communication
[4]	REN	UART1 Receive Enable 0: Disable 1: Enable
[3]	TB8	Bit9 of the Transmitted Data in Mode2 and Mode3
[2]	RB8	Bit9 of the Received Data in Mode2 and Mode3
[1]	TI	Data Transmitting Completed Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0”. 1: The interrupt is generated.
[0]	RI	Data Receiving Completed Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0”. 1: The interrupt is generated.

9.3.2 UT_DR (0x99)

Bit	7	6	5	4	3	2	1	0
Name	UT_DR							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	UT_DR	Transmit / Receive Data Read: Data received Write: Data to be transmitted Note The UART1 data buffer consists of two independent buffers, i.e., a receive buffer and a transmit buffer, which can send and receive data at the same time. The transmit buffer can be written only, while the receive buffer can be read. Both buffers share a same address

9.3.3 UT_BAUD (0x9A, 0x9B)

UT_BAUDH (0x9B)								
Bit	15	14	13	12	11	10	9	8
Name	BAUD_SEL	UART_RX_INV	UART_TX_INV	RSV	BAUD[11:8]			
Type	R/W	R/W	R/W	-	R/W	R/W	R/W	R/W
Reset	0	0	0	-	0	0	0	0
UT_BAUDL(0x9A)								
Bit	7	6	5	4	3	2	1	0
Name	BAUD[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	0	0	1	1	0	1	1

Bit	Name	Description
[15]	BAUD_SEL	Frequency Multiplier Enable 0: Disable 1: Enable
[14]	UART_RX_INV	Receive Inverting Enable 0: Disable 1: Enable
[13]	UART_TX_INV	Transmit Inverting Enable 0: Disable 1: Enable

[12]	RSV	Reserved
[11:0]	BAUD	Baud Rate Setting Baud rate = UARTCLK/(16/(1 + UT_BAUD[BAUD_SEL]))/(UT_BAUD[BAUD] + 1) Example: If baud rate = 9600 and UT_BAUD[BAUD_SEL] = 0, then UT_BAUD[BAUD] = (24M/16/9600/(1 + 0)) - 1 = 155, i.e., 0x9B

9.4 UART2 Registers

9.4.1 UT2_CR (0xD8)

Bit	7	6	5	4	3	2	1	0
Name	UT2MOD		UT2SM2	UT2REN	UT2TB8	UT2RB8	UT2TI	UT2RI
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:6]	UT2MOD	Mode Selection 00: Mode0 01: Mode1 10: Mode2 11: Mode3
[5]	UT2SM2	Communication Mode Selection 0: Single-device Communication 1: Multi-device Communication
[4]	UT2REN	UART2 Receive Enable 0: Disable 1: Enable
[3]	UT2TB8	Bit9 of the Transmitted Data in Mode2 and Mode3
[2]	UT2RB8	Bit9 of the Received Data in Mode2 and Mode3
[1]	UT2TI	Data Transmitting Completed Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0”. 1: The interrupt is generated.
[0]	UT2RI	Data Receiving Completed Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0”.

		1: The interrupt is generated.

9.4.2 UT2_DR (0x89)

Bit	7	6	5	4	3	2	1	0
Name	UT2_DR							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	UT2_DR	Transmit / Receive Data Read: Data received Write: Data to be transmitted Note The UART2 data buffer consists of two independent buffers, i.e., a receive buffer and a transmit buffer, which can send and receive data at the same time. The transmit buffer can be written only, while the receive buffer can be read only. Both buffers share a same address

9.4.3 UT2_BAUD (0x4042, 0x4043)

UT2_BAUDH(0x4042)								
Bit	15	14	13	12	11	10	9	8
Name	BAUD2_SEL	UART2_RX_INV	UART2_TX_INV	UART2IEN	BAUD2[11:8]			
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
UT2_BAUDL(0x4043)								
Bit	7	6	5	4	3	2	1	0
Name	BAUD2[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	0	0	1	1	0	1	1

Bit	Name	Description
[15]	BAUD2_SEL	Frequency Multiplier Enable 0: Disable 1: Enable
[14]	UART2_RX_INV	Receive Inverting Enable 0: Disable 1: Enable
[13]	UART2_TX_INV	Transmit Inverting Enable 0: Disable

		1: Enable
[12]	UART2IEN	UART2 Interrupt Enable 0: Disable 1: Enable
[11:0]	BAUD2	Baud Rate Setting Baud rate = UART2CLK/(16/(1 + UT2_BAUD[BAUD2_SEL]))/(UT2_BAUD[BAUD2] + 1) Example: If baud rate = 9600 and UT2_BAUD[BAUD_SEL] = 0, then UT2_BAUD[BAUD2] = (24M/16/9600/(1 + 0)) - 1 = 155, i.e., 0x9B

10 MDU

10.1 MDU Introduction

MDU is a built-in computing co-processor that assists the CPU in processing complex operations efficiently. It supports multiplication, division, trigonometric operation, LPF operation and PID operation. MDU module can be invoked in different interrupt services and master programs, and the results are independent from each other.

10.2 MDU Features

The MDU module has the following features:

- Support invocation with nested interrupt
- Hardware acceleration to reduce CPU load
- Support the following modes:
  - 16-bit signed multiplication
  - 16-bit signed multiplication (result shifted left by one-bit)
  - 16-bit unsigned multiplication
  - 32-bit/16-bit unsigned division
  - Low-pass filter (LPF)
  - Coordinate transformation (SIN/COS)
  - Arctangent (ATAN)
  - PI/PID

10.3 MDU Features

10.3.1 MDU Operations

1. Configure MDU_CR[MDUMOD] register to select computing mode of the MDU module;
2. Write the data to the associated computing units, and configure MDU_CR[MDUSTA] to select computing unit of the MDU module, and start MDU computing;
3. Wait for MDU_CR[MDUBSY] to be cleared to “0” by hardware.

Note

When using MDU, ensure that computing mode and other data have been written before configuring MDU_CR[MDUSTA]

10.3.2 16-bit Signed Multiplication with the Result Shifted Left by One-bit

When MDU_CR[MDUMOD] = 000, MDU module works in the 16-bit signed multiplication mode with the result shifted left by one-bit. As shown in Table 10-1, after 16-bit signed data is written to MULx_MA and MULx_MB as multiplicand and multiplier respectively, 32-bit signed data is obtained by the product shifting left by one bit. The result is accessed by reading MULx_MC register.

Table 10-1 Register Definitions in 16-bit Signed Multiplication Mode with Result Shifted Left by One-bit

Data Register	Input	Output
MULx_MA	Multiplicand	-
MULx_MB	Multiplier	-
MULx_MC	-	Product

10.3.3 16-bit Signed Multiplication

When MDU_CR[MDUMOD] = 001, MDU module works in the 16-bit signed multiplication mode. As shown in Table 10-2, 31-bit signed data is obtained after 16-bit signed data is written to MULx_MA and MULx_MB as multiplicand and multiplier respectively. The result is accessed by reading MULx_MC register.

Table 10-2 Register Definitions in 16-bit Signed Multiplication Mode

Data Register	Input	Output
MULx_MA	Multiplicand	-
MULx_MB	Multiplier	-
MULx_MC	-	Product

10.3.4 16-bit Unsigned Multiplication

When MDU_CR[MDUMOD] = 010, MDU module works in the 16-bit unsigned multiplication mode. As shown in Table 10-3, 32-bit unsigned data is obtained after 16-bit unsigned data is written to MULx_MA and MULx_MB as multiplicand and multiplier respectively. The result is accessed by reading MULx_MC register.

Table 10-3 Register Definitions in 16-bit Unsigned Multiplication Mode

Data Register	Input	Output
MULx_MA	Multiplicand	-
MULx_MB	Multiplier	-
MULx_MC	-	Product

10.3.5 32-bit/16-bit Unsigned Division

When MDU_CR[MDUMOD] = 011, MDU module works in the 32-bit/16-bit unsigned division mode. As shown in Table 10-4, 32-bit unsigned quotient with 16-bit unsigned remainder is obtained after 32-bit dividend and a 16-bit divisor are written to DIVx_DA and DIVx_DB registers respectively. The quotient and remainder are accessed by reading DIVx_DQ and DIVx_DR registers respectively.

Table 10-4 Register Definitions in the Unsigned Division Mode

Data Register	Input	Output
DIVx_DA	Dividend	-
DIVx_DB	Divisor	-
DIVx_DQ	-	Quotient
DIVx_DR	-	Remainder

10.3.6 LPF

When MDU_CR[MDUMOD] = 110, MDU module works in LPF mode.

The calculation formula of LPF is:

$$Y_k = Y_{k-1} + K \times (X_k - Y_{k-1})$$

Where,

Y_k: Filtered data

Y_{k-1}: Previous filtered data

K: Filter coefficient

X_k: Data to be filtered

As shown in Table 10-5, Y_k and Y_{k-1} are 32-bit signed data, X_k and K are 16-bit signed data. Y_k is obtained after Y_{k-1} is written to LPFx_Y, K to LPFx_K and X_k to LPFx_X, and is accessed by reading LPFx_Y.

Table 10-5 Register Definitions in LPF Mode

Data Register	Input	Output
LPFx_X	X_k	-
LPFx_K	K	-
LPFx_Y	Y_{k-1}	Y_k

10.3.7 Coordinate Transformation (sin/cos Calculation)

When MDU_CR[MDUMOD] = 100, MDU module works in Coordinate Transformation mode. As shown in Figure 10-1, the coordinate transformation converts the components cos_i and sin_i of vector A under the x-y axis to the components cos_o and sin_o under the x'-y' axis, with the x'-y' axis lagging the x-y axis by θ.

The formula for coordinate transformation is:

$$cos_o = cos_i \times cos\,\theta - sin_i \times sin\,\theta$$

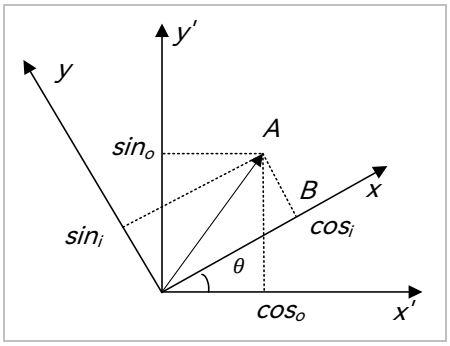
$$sin_o = cos_i \times sin\,\theta + sin_i \times cos\,\theta$$

In particular, when sin_i = 0, the coordinate transformation is a sine and cosine calculation with cos_i as the amplitude, calculated as:

$cos_o = cos_i \times cos\,\theta$

$sin_o = cos_i \times sin\,\theta$

Figure 10-1 Coordinate Transformation



As shown in Table 10-6, cos_i, sin_i, θ, cos_o and sin_o are all 16-bit signed data. cos_i is written to SCATx_COS, sin_i to SCATx_SIN and θ to SCATx_THE to calculate cos_o and sin_o. The results cos_o and sin_o are accessed by reading SCATx_RES1 and SCATx_RES2 respectively.

Table 10-6 Register Definitions in the Coordinate Transformation Mode

Data Register	Input	Output
SCATx_COS	cos_i	-
SCATx_SIN	sin_i	-
SCATx_THE	θ	-
SCATx_RES1	-	cos_o
SCATx_RES2	-	sin_o

10.3.8 Arctangent

When MDU_CR[MDUMOD] = 101, MDU module works in arctangent (ATAN) mode.

ATAN calculates the amplitude and angle of a vector based on sine and cosine inputs. The calculation formula is:

$U = \sqrt{(Usin\,\theta)^2 + (U\,cos\,\theta)^2}$

$\theta = tan^{-1}\left(\frac{U\,sin\,\theta}{Ucos\,\theta}\right)$

Where,

Usinθ: Sin component of the vector

Ucosθ: Cosine component of the vector

θ: Calculated vector angle

U: Calculated vector amplitude

As shown in Table 10-7, *Ucosθ* and *Usinθ*, *U* and *θ* are 16-bit signed data. $Ucos_\theta$ is written to SCATx_COS and $Usin_\theta$ to SCATx_SIN to calculate *U* and *θ*. *U* and *θ* are accessed by reading SCATx_RES1 and SCATx_RES2 respectively.

Table 10-7 Register Definitions in ATAN Mode

Data Register	Input	Output
SCATx_COS	*Ucosθ*	-
SCATx_SIN	*Usinθ*	-
SCATx_RES1	-	*U*
SCATx_RES2	-	*θ*

10.3.9 PI/PID

10.3.9.1 PI/PID Introduction

PI/PID regulator is a linear controller, where the output is generated by linear combination of error proportional, integral and differential actions, and then implemented by an actuator. In motor control system, it is used to for speed and position control.

PI algorithm:

$$U_k = U_{k-1} + Kp \times (E_k - E_{k-1}) + Ki \times E_k$$

PID algorithm:

$$U_k = U_{k-1} + Kp \times (E_k - E_{k-1}) + Ki \times E_k + Kd \times (E_k - 2 \times E_{k-1} + E_{k-2})$$

Where,

U_k: Output for round k of calculation

U_{k-1}: Output for round k-1 of calculation

E_k: Deviation for round k of input

E_{k-1}, E_{k-2}: Deviations for round k-1 and round k-2 of calculation

K_p, K_i and K_d: Proportional (P), integral (I) and differential (D) coefficients of regulator

The maximum U_k is represented as PIx_UKMAX (x = 0 ~ 3) and the minimum value as PIx_UKMIN.

10.3.9.2 PI/PID Features

- Parameter range is configurable.
- Support multiple invocations but not with nested interrupt.
- Produce a 32-bit result PIx_UK.
- Results are read after Busy Flag is reset to “0”.

10.3.9.3 PI/PID Operations

1. Initialize MDU before the operations, and configure K_p, K_i , K_d and the maximum and minimum values of U_k;
2. Set MDU_CR[MDUMOD] to “111”, and then select Comp_Unit0 and Comp_Unit1 as PI Mode, and Comp_Unit2 and Comp_Unit3 as PID Mode. Later, configure MDU_CR[MDUSTA] to select the desired computing unit and start PI/PID computing. At this time, busy flag MDU_CR[MDUBSY] is automatically set to “1”;
3. Read MDU_CR[MDUBSY] bit by software. When this bit is 0, it indicates that the calculation is completed, and calculation result PIx_UK is updated;
4. Read PIx_UK to obtain the output.

Note

- The data format of PI_KP is Q12 and that of other registers is Q15.
- PIx_UK and PIx_EK1 values default to the previous calculated U_k and E_k The related values change after PIx_EK1 and PIx_UK are written.
- When PI controller is invoked repeatedly, relevant parameters shall be saved after each PI operation, and initialized before the next PI operation. Initialization codes are shown as below:

```
PIx_KP = KP;          //Initialize Kp
PIx_KI = KI;          //Initialize Ki
PIx_KD = KD;          //Initialize Kd
```

```
Plx_UKMAX = UKMAX;          //Initialize maximum output
Plx_UKMIN = UKMIN;          //Initialize minimum output
Plx_EK1 = X;                // Initialize Ek-1
Plx_UKH = Y1;               //Initialize 16 high-order bits of Uk-1
Plx_UKL = Y2;               // Initialize16 low-order bits of Uk-1
```

10.4 MDU Registers

10.4.1 MDU_CR (0xC1)

Bit	7	6	5	4	3	2	1	0
Name	MDUBSY	MDUSTA				MDUMOD		
Type	R	R/W1	R/W1	R/W1	R/W1	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	MDUBSY	MDU Busy Flag A write of MDU_CR[6:3] to this bit starts MDU module. This bit is set to “1” after MDU completes operations.
[6:3]	MDUSTA	This bit is used to configure computing unit of the MDU. Four options are available. MDU module starts operation after initiating the computing. 0001: Comp_Unit0 activated 0010: Comp_Unit1 activated 0100: Comp_Unit2 activated 1000: Comp_Unit3 activated
[2:0]	MDUMOD	MDU Mode Selection 000: 16-bit Signed Multiplication (the result shifted left by one-bit) 001: 16-bit Signed Multiplication 010: 16-bit Unsigned Multiplication 011: 32-bit/16-bit Unsigned Division 100: Coordinate Transformation (SIN/COS) 101: ATAN 110: LPF 111: PI/PID mode, which decides the computing unit. Computing unit 0 and 1 are selected in PI mode, and computing unit 2 and 3 in PID mode.

10.4.2 MUL0_MA (0x0FA0, 0x0FA1)

MUL0_MAH(0x0FA0)								
Bit	15	14	13	12	11	10	9	8
Name	MUL0_MA[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
MUL0_MAL(0x0FA1)								
Bit	7	6	5	4	3	2	1	0
Name	MUL0_MA[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	MUL0_MA	Data register A of MUL0; Multiplicand of the multiplication

10.4.3 MUL0_MB (0x0FA2, 0x0FA3)

MUL0_MBH(0x0FA2)								
Bit	15	14	13	12	11	10	9	8
Name	MUL0_MB[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
MUL0_MBL(0x0FA3)								
Bit	7	6	5	4	3	2	1	0
Name	MUL0_MB[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	MUL0_MB	Data register B of MUL0; Multiplier of the multiplication

10.4.4 MUL0_MC (0x0FA4, 0x0FA5, 0x0FA6, 0x0FA7)

MUL0_MCHH(0x0FA4)								
Bit	31	30	29	28	27	26	25	24
Name	MUL0_MC[31:24]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
MUL0_MCHL(0x0FA5)								
Bit	23	22	21	20	19	18	17	16
Name	MUL0_MC[13:16]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
MUL0_MCLH(0x0FA6)								
Bit	15	14	13	12	11	10	9	8
Name					MUL0_MC[15:8]			
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
MUL0_MCLL(0x0FA7)								
Bit	7	6	5	4	3	2	1	0
Name	MUL0_MC[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[31:0]	MUL0_MC	Product of MUL0. The 16 high-order bits of the data is held by MUL0_MCH and the 16 low-order bits by MUL0_MCL.

10.4.5 MUL1_MA (0x0F98, 0x0F99)

MUL1_MAH(0x0F98)								
Bit	15	14	13	12	11	10	9	8
Name	MUL1_MA[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
MUL1_MAL(0x0F99)								
Bit	7	6	5	4	3	2	1	0
Name	MUL1_MA[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	MUL1_MA	Data register A of MUL1; Multiplicand of the multiplication

10.4.6 MUL1_MB (0x0F9A, 0x0F9B)

MUL1_MBH(0x0F9A)								
Bit	15	14	13	12	11	10	9	8
Name	MUL1_MB[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
MUL1_MBL(0x0F9B)								
Bit	7	6	5	4	3	2	1	0
Name	MUL1_MB[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	MUL1_MB	Data register B of MUL1; Multiplier of the multiplication

10.4.7 MUL1_MC (0x0F9C, 0x0F9D, 0x0F9E, 0x0F9F)

MUL1_MCHH(0x0F9C)								
Bit	31	30	29	28	27	26	25	24
Name	MUL1_MC[31:24]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

MUL1_MCHL(0x0F9D)								
Bit	23	22	21	20	19	18	17	16
Name	MUL1_MC[23:16]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
MUL1_MCLH(0x0F9E)								
Bit	15	14	13	12	11	10	9	8
Name	MUL1_MC[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
MUL1_MCLL(0x0F9F)								
Bit	7	6	5	4	3	2	1	0
Name	MUL1_MC[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[31:0]	MUL1_MC	Product of MUL1. The 16 high-order bits of the data is held by MUL1_MCH and the 16 low-order bits by MUL1_MCL.

10.4.8 MUL2_MA (0x0F40, 0x0F41)

MUL2_MAH(0x0F40)								
Bit	15	14	13	12	11	10	9	8
Name	MUL2_MA[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
MUL2_MAL(0x0F41)								
Bit	7	6	5	4	3	2	1	0
Name	MUL2_MA[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	MUL2_MA	Data register A of MUL2; Multiplicand of the multiplication

10.4.9 MUL2_MB (0x0F42, 0x0F43)

MUL2_MBH(0x0F42)								
Bit	15	14	13	12	11	10	9	8
Name	MUL2_MB[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Reset	0	0	0	0	0	0	0	0
MUL2_MBL(0x0F43)								
Bit	7	6	5	4	3	2	1	0
Name	MUL2_MB[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	MUL2_MB	Data register B of MUL2; Multiplier of the multiplication

10.4.10 MUL2_MC (0x0F44, 0x0F45, 0x0F46, 0x0F47)

MUL2_MCHH(0x0F44)								
Bit	31	30	29	28	27	26	25	24
Name	MUL2_MC[31:24]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
MUL2_MCHL(0x0F45)								
Bit	23	22	21	20	19	18	17	16
Name	MUL2_MC[23:16]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
MUL2_MCLH(0x0F46)								
Bit	15	14	13	12	11	10	9	8
Name	MUL2_MC[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
MUL2_MCLL(0x0F47)								
Bit	7	6	5	4	3	2	1	0
Name	MUL2_MC[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[31:0]	MUL2_MC	Product of MUL2. The 16 high-order bits of the data is held by MUL2_MCH and the 16 low-order bits by MUL2_MCL.

10.4.11 MUL3_MA (0x0F38, 0x0F39)

MUL3_MAH(0x0F38)								
Bit	15	14	13	12	11	10	9	8
Name	MUL3_MA[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
MUL3_MAL(0x0F39)								
Bit	7	6	5	4	3	2	1	0
Name	MUL3_MA[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	MUL3_MA	Data register A of MUL3; Multiplicand of the multiplication

10.4.12 MUL3_MB (0x0F3A, 0x0F3B)

MUL3_MBH(0x0F3A)								
Bit	15	14	13	12	11	10	9	8
Name	MUL3_MB[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
MUL3_MBL(0x0F3B)								
Bit	7	6	5	4	3	2	1	0
Name	MUL3_MB[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	MUL3_MB	Data register B of MUL3; Multiplier of the multiplication

10.4.13 MUL3_MC (0x0F3C, 0x0F3D, 0x0F3E, 0x0F3F)

MUL3_MCHH(0x0F3C)								
Bit	31	30	29	28	27	26	25	24
Name	MUL3_MC[31:24]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
MUL3_MCHL(0x0F3D)								
Bit	23	22	21	20	19	18	17	16
Name	MUL3_MC[23:16]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Reset	0	0	0	0	0	0	0	0
MUL3_MCLH(0x0F3E)								
Bit	15	14	13	12	11	10	9	8
Name	MUL3_MC[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
MUL3_MCLL(0x0F3F)								
Bit	7	6	5	4	3	2	1	0
Name	MUL3_MC[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[31:0]	MUL3_MC	Product of MUL3. The 16 high-order bits of the data is held by MUL3_MCH and the 16 low-order bits by MUL3_MCL.

10.4.14 DIV0_DA (0x0F8C, 0x0F8D, 0x0F8E, 0x0F8F)

DIV0_DAHH(0x0F8C)								
Bit	31	30	29	28	27	26	25	24
Name	DIV0_DA[31:24]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV0_DAHL(0x0F8D)								
Bit	23	22	21	20	19	18	17	16
Name	DIV0_DA[23:16]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV0_DALH(0x0F8E)								
Bit	15	14	13	12	11	10	9	8
Name	DIV0_DA[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV0_DALL(0x0F8F)								
Bit	7	6	5	4	3	2	1	0
Name	DIV0_DA[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[31:0]	DIV0_DA	Dividend of DIV0. The 16 high-order bits of the data is held by DIV0_DAH

		and the 16 low-order bits by DIV0_DAL.

10.4.15 DIV0_DB (0x0F90, 0x0F91)

DIV0_DBH(0x0F90)								
Bit	15	14	13	12	11	10	9	8
Name	DIV0_DB[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV0_DBL(0x0F91)								
Bit	7	6	5	4	3	2	1	0
Name	DIV0_DB[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	DIV0_DB	Data register B of DIV0; Divisor of the division

10.4.16 DIV0_DQ (0x0F92, 0x0F93, 0x0F94, 0x0F95)

DIV0_DQHH(0x0F92)								
Bit	31	30	29	28	27	26	25	24
Name	DIV0_DQ[31:24]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV0_DQHL(0x0F93)								
Bit	23	22	21	20	19	18	17	16
Name	DIV0_DQ[23:16]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV0_DQLH(0x0F94)								
Bit	15	14	13	12	11	10	9	8
Name	DIV0_DQ[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV0_DQLL(0x0F95)								
Bit	7	6	5	4	3	2	1	0
Name	DIV0_DQ[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description

[31:0]	DIV0_DQ[31:0]	Quotient of DIV0. The 16 high-order bits of the data is held by DIV0_DQH and the 16 low-order bits by DIV0_DQL.

10.4.17 DIV0_DR (0x0F96, 0x0F97)

DIV0_DRH(0x0F96)								
Bit	15	14	13	12	11	10	9	8
Name	DIV0_DR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV0_DRL(0x0F97)								
Bit	7	6	5	4	3	2	1	0
Name	DIV0_DR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	DIV0_DR	Remainder of DIV0

10.4.18 DIV1_DA (0x0F80, 0x0F81, 0x0F82, 0x0F83)

DIV1_DAHH(0x0F80)								
Bit	31	30	29	28	27	26	25	24
Name	DIV1_DA[31:24]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV1_DAHL(0x0F81)								
Bit	23	22	21	20	19	18	17	16
Name	DIV1_DA[23:16]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV1_DALH(0x0F82)								
Bit	15	14	13	12	11	10	9	8
Name	DIV1_DA[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV1_DALL(0x0F83)								
Bit	7	6	5	4	3	2	1	0
Name	DIV1_DA[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description

[31:0]	DIV1_DA	Dividend of DIV1. The 16 high-order bits of the data is held by DIV1_DA and the 16 low-order bits by DIV1_DA.

10.4.19 DIV1_DB (0x0F84, 0x0F85)

DIV1_DBH(0x0F84)								
Bit	15	14	13	12	11	10	9	8
Name	DIV1_DB[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV1_DBL(0x0F85)								
Bit	7	6	5	4	3	2	1	0
Name	DIV1_DB[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	DIV1_DB	Data register B of DIV1; Divisor of the division

10.4.20 DIV1_DQ (0x0F86, 0x0F87, 0x0F88, 0x0F89)

DIV1_DQHH(0x0F86)								
Bit	31	30	29	28	27	26	25	24
Name	DIV1_DQ[31:24]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV1_DQHL(0x0F87)								
Bit	23	22	21	20	19	18	17	16
Name	DIV1_DQ[23:16]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV1_DQLH(0x0F88)								
Bit	15	14	13	12	11	10	9	8
Name	DIV1_DQ[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV1_DQLL(0x0F89)								
Bit	7	6	5	4	3	2	1	0
Name	DIV1_DQ[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[31:0]	DIV1_DQ	Quotient of DIV1. The 16 high-order bits of the data is held by DIV1_DQH and the 16 low-order bits by DIV1_DQL.

10.4.21 DIV1_DR (0x0F8A, 0x0F8B)

DIV1_DRH(0x0F8A)								
Bit	15	14	13	12	11	10	9	8
Name	DIV1_DR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV1_DRL(0x0F8B)								
Bit	7	6	5	4	3	2	1	0
Name	DIV1_DR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	DIV1_DR	Remainder of DIV1

10.4.22 DIV2_DA (0x0F2C, 0x0F2D, 0x0F2E, 0x0F2F)

DIV2_DAHH(0x0F2C)								
Bit	31	30	29	28	27	26	25	24
Name	DIV2_DA[31:24]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV2_DAHL(0x0F2D)								
Bit	23	22	21	20	19	18	17	16
Name	DIV2_DA[23:16]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV2_DALH(0x0F2E)								
Bit	15	14	13	12	11	10	9	8
Name	DIV2_DA[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV2_DALL(0x0F2F)								
Bit	7	6	5	4	3	2	1	0
Name	DIV2_DA[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[31:0]	DIV2_DA	Dividend of DIV2. The 16 high-order bits of the data is held by DIV2_DA and the 16 low-order bits by DIV2_DA.

10.4.23 DIV2_DB (0x0F30, 0x0F31)

DIV2_DBH(0x0F30)								
Bit	15	14	13	12	11	10	9	8
Name	DIV2_DB[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV2_DBL(0x0F31)								
Bit	7	6	5	4	3	2	1	0
Name	DIV2_DB[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	DIV2_DB	Data register B of DIV2; Divisor of the division

10.4.24 DIV2_DQ (0x0F32, 0x0F33, 0x0F34, 0x0F35)

DIV2_DQHH(0x0F32)								
Bit	31	30	29	28	27	26	25	24
Name	DIV2_DQ[31:24]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV2_DQHL(0x0F33)								
Bit	23	22	21	20	19	18	17	16
Name	DIV2_DQ[23:16]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV2_DQLH(0x0F34)								
Bit	15	14	13	12	11	10	9	8
Name	DIV2_DQ[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV2_DQLL(0x0F35)								
Bit	7	6	5	4	3	2	1	0
Name	DIV2_DQ[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[31:0]	DIV2_DQ	Quotient of DIV2. The 16 high-order bits of the data is held by DIV2_DQH and the 16 low-order bits by DIV2_DQL.

10.4.25 DIV2_DR (0x0F36, 0x0F37)

DIV2_DRH(0x0F36)								
Bit	15	14	13	12	11	10	9	8
Name	DIV2_DR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV2_DRL(0x0F37)								
Bit	7	6	5	4	3	2	1	0
Name	DIV2_DR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	DIV2_DR	Remainder of DIV2

10.4.26 DIV3_DA (0x0F20, 0x0F21, 0x0F22, 0x0F23)

DIV3_DAHH(0x0F20)								
Bit	31	30	29	28	27	26	25	24
Name	DIV3_DA[31:24]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV3_DAHL(0x0F21)								
Bit	23	22	21	20	19	18	17	16
Name	DIV3_DA[23:16]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV3_DALH(0x0F22)								
Bit	15	14	13	12	11	10	9	8
Name	DIV3_DA[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV3_DALL(0x0F23)								
Bit	7	6	5	4	3	2	1	0
Name	DIV3_DA[7:0]							

Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[31:0]	DIV3_DA	Dividend of DIV3. The 16 high-order bits of the data is held by DIV3_DAH and the 16 low-order bits by DIV3_DAL.

10.4.27 DIV3_DB (0x0F24, 0x0F25)

DIV3_DBH(0x0F24)								
Bit	15	14	13	12	11	10	9	8
Name	DIV3_DB[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV3_DBL(0x0F25)								
Bit	7	6	5	4	3	2	1	0
Name	DIV3_DB[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	DIV3_DB	Data register B of DIV3; Divisor of the division

10.4.28 DIV3_DQ (0x0F26, 0x0F27, 0x0F28, 0x0F29)

DIV3_DQHH(0x0F26)								
Bit	31	30	29	28	27	26	25	24
Name	DIV3_DQ[31:24]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV3_DQHL(0x0F27)								
Bit	23	22	21	20	19	18	17	16
Name	DIV3_DQ[23:16]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV3_DQLH(0x0F28)								
Bit	15	14	13	12	11	10	9	8
Name	DIV3_DQ[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV3_DQLL(0x0F29)								
Bit	7	6	5	4	3	2	1	0

Name	DIV3_DQ[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[31:0]	DIV3_DQ	Quotient of DIV3. The 16 high-order bits of the data is held by DIV3_DQH and the 16 low-order bits by DIV3_DQL.

10.4.29 DIV3_DR (0x0F2A, 0x0F2B)

DIV3_DRH(0x0F2A)								
Bit	15	14	13	12	11	10	9	8
Name	DIV3_DR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
DIV3_DRL(0x0F2B)								
Bit	7	6	5	4	3	2	1	0
Name	DIV3_DR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	DIV3_DR	Remainder of DIV3

10.4.30 SCAT0_COS (0x0F16, 0x0F17)

SCAT0_COSH(0x0F16)								
Bit	15	14	13	12	11	10	9	8
Name	SCAT0_COS[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
SCAT0_COSL(0x0F17)								
Bit	7	6	5	4	3	2	1	0
Name	SCAT0_COS[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	SCAT0_COS	COS input in SIN/COS or ATAN mode of computing unit SCAT0

10.4.31 SCAT0_SIN (0x0F18, 0x0F19)

SCAT0_SINH(0x0F18)								
Bit	15	14	13	12	11	10	9	8
Name	SCAT0_SIN[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
SCAT0_SINL(0x0F19)								
Bit	7	6	5	4	3	2	1	0
Name	SCAT0_SIN[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	SCAT0_SIN	SIN input in SIN/COS or ATAN mode of computing unit SCAT0

10.4.32 SCAT0_THE (0x0F1A, 0x0F1B)

SCAT0_THEH(0x0F1A)								
Bit	15	14	13	12	11	10	9	8
Name	SCAT0_THE[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
SCAT0_THEL(0x0F1B)								
Bit	7	6	5	4	3	2	1	0
Name	SCAT0_THE[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	SCAT0_THE	THE input in SIN/COS mode of computing unit SCAT0

10.4.33 SCAT0_RES1 (0x0F1C, 0x0F1D)

SCAT0_RES1H(0x0F1C)								
Bit	15	14	13	12	11	10	9	8
Name	SCAT0_RES1[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
SCAT0_RES1L(0x0F1D)								
Bit	7	6	5	4	3	2	1	0
Name	SCAT0_RES1[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	SCAT0_RES1	COS output in SIN/COS mode of computing unit SCAT0; U output in ATAN mode

10.4.34 SCAT0_RES2 (0x0F1E, 0x0F1F)

SCAT0_RES2H(0x0F1E)								
Bit	15	14	13	12	11	10	9	8
Name	SCAT0_RES2[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
SCAT0_RES2L(0x0F1F)								
Bit	7	6	5	4	3	2	1	0
Name	SCAT0_RES2[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	SCAT0_RES2	SIN output in SIN/COS mode of computing unit SCAT0; θ output in ATAN mode

10.4.35 SCAT1_COS (0x0F0C, 0x0F0D)

SCAT1_COSH(0x0F0C)								
Bit	15	14	13	12	11	10	9	8
Name	SCAT1_COS[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
SCAT1_COSL(0x0F0D)								
Bit	7	6	5	4	3	2	1	0
Name	SCAT1_COS[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	SCAT1_COS	COS input in SIN/COS or ATAN mode of computing unit SCAT1

10.4.36 SCAT1_SIN (0x0F0E, 0x0F0F)

SCAT1_SINH(0x0F0E)								
Bit	15	14	13	12	11	10	9	8
Name	SCAT1_SIN[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
SCAT1_SINL(0x0F0F)								
Bit	7	6	5	4	3	2	1	0
Name	SCAT1_SIN[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	SCAT1_SIN	SIN input in SIN/COS or ATAN mode of computing unit SCAT1

10.4.37 SCAT1_THE (0x0F10, 0x0F11)

SCAT1_THEH(0x0F10)								
Bit	15	14	13	12	11	10	9	8
Name	SCAT1_THE[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
SCAT1_THEL(0x0F11)								
Bit	7	6	5	4	3	2	1	0
Name	SCAT1_THE[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	SCAT1_THE	THE input in SIN/COS mode of computing unit SCAT1

10.4.38 SCAT1_RES1 (0x0F12, 0x0F13)

SCAT1_RES1H(0x0F12)								
Bit	15	14	13	12	11	10	9	8
Name	SCAT1_RES1[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
SCAT1_RES1L(0x0F13)								
Bit	7	6	5	4	3	2	1	0
Name	SCAT1_RES1[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	SCAT1_RES1	COS output in SIN/COS mode of computing unit SCAT1; U output in ATAN mode

10.4.39 SCAT1_RES2 (0x0F14, 0x0F15)

SCAT1_RES2H(0x0F14)								
Bit	15	14	13	12	11	10	9	8
Name	SCAT1_RES2[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
SCAT1_RES2L(0x0F15)								
Bit	7	6	5	4	3	2	1	0
Name	SCAT1_RES2[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	SCAT1_RES2	SIN output in SIN/COS mode of computing unit SCAT1; U output in ATAN mode

10.4.40 SCAT2_COS (0x0F02, 0x0F03)

SCAT2_COSH(0x0F02)								
Bit	15	14	13	12	11	10	9	8
Name	SCAT2_COS[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
SCAT2_COSL(0x0F03)								
Bit	7	6	5	4	3	2	1	0
Name	SCAT2_COS[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	SCAT2_COS	COS input in SIN/COS or ATAN mode of computing unit SCAT2

10.4.41 SCAT2_SIN (0x0F04, 0x0F05)

SCAT2_SINH(0x0F04)								
Bit	15	14	13	12	11	10	9	8
Name	SCAT2_SIN[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SCAT2_SINL(0x0F05)								
Bit	7	6	5	4	3	2	1	0
Name	SCAT2_SIN[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	SCAT2_SIN	SIN input in SIN/COS or ATAN mode of computing unit SCAT2

10.4.42 SCAT2_THE (0x0F06, 0x0F07)

SCAT2_THEH(0x0F06)								
Bit	15	14	13	12	11	10	9	8
Name	SCAT2_THE[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
SCAT2_THEL(0x0F07)								
Bit	7	6	5	4	3	2	1	0
Name	SCAT2_THE[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	SCAT2_THE	THE input in SIN/COS mode of computing unit SCAT2

10.4.43 SCAT2_RES1 (0x0F08, 0x0F09)

SCAT2_RES1H(0x0F08)								
Bit	15	14	13	12	11	10	9	8
Name	SCAT2_RES1[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
SCAT2_RES1L(0x0F09)								
Bit	7	6	5	4	3	2	1	0
Name	SCAT2_RES1[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	SCAT2_RES1	COS output in SIN/COS mode of computing unit SCAT2; U output in ATAN mode

10.4.44 SCAT2_RES2 (0x0F0A, 0x0F0B)

SCAT2_RES2H(0x0F0A)								
Bit	15	14	13	12	11	10	9	8
Name	SCAT2_RES[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
SCAT2_RES2L(0x0F0B)								
Bit	7	6	5	4	3	2	1	0
Name	SCAT2_RES[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	SCAT2_RES2	SIN output in SIN/COS mode of computing unit SCAT2; θ output in ATAN mode

10.4.45 SCAT3_COS (0x0EF8, 0x0EF9)

SCAT3_COSH(0x0EF8)								
Bit	15	14	13	12	11	10	9	8
Name	SCAT3_COS[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
SCAT3_COSL(0x0EF9)								
Bit	7	6	5	4	3	2	1	0
Name	SCAT3_COS[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	SCAT3_COS	COS input in SIN/COS or ATAN mode of computing unit SCAT3

10.4.46 SCAT3_SIN (0x0EFA, 0x0EFB)

SCAT3_SINH(0x0EFA)								
Bit	15	14	13	12	11	10	9	8
Name	SCAT3_SIN[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
SCAT3_SINL(0x0EFB)								
Bit	7	6	5	4	3	2	1	0
Name	SCAT3_SIN[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	SCAT3_SIN	SIN input in SIN/COS or ATAN mode of computing unit SCAT3

10.4.47 SCAT3_THE (0x0EFC, 0x0EFD)

SCAT3_THEH(0x0EFC)								
Bit	15	14	13	12	11	10	9	8
Name	SCAT3_THE[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
SCAT3_THEL(0x0EFD)								
Bit	7	6	5	4	3	2	1	0
Name	SCAT3_THE[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	SCAT3_THE	THE input in SIN/COS mode of computing unit SCAT3

10.4.48 SCAT3_RES1 (0x0EFE, 0x0EFF)

SCAT3_RES1H(0x0EFE)								
Bit	15	14	13	12	11	10	9	8
Name	SCAT3_RES1[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
SCAT3_RES1L(0x0EFF)								
Bit	7	6	5	4	3	2	1	0
Name	SCAT3_RES1[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	SCAT3_RES1	COS output in SIN/COS mode of computing unit SCAT3; U output in ATAN mode

10.4.49 SCAT3_RES2 (0x0F00, 0x0F01)

SCAT3_RES2H(0x0F00)								
Bit	15	14	13	12	11	10	9	8
Name	SCAT3_RES[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
SCAT3_RES2L(0x0F01)								

Bit	7	6	5	4	3	2	1	0
Name	SCAT3_RES[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	SCAT3_RES2	SIN output in SIN/COS mode of computing unit SCAT3; θ output in ATAN mode

10.4.50 LPF0_K (0x0FD0, 0x0FD1)

LPF0_KH(0x0FD0)								
Bit	15	14	13	12	11	10	9	8
Name	LPF0_K[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
LPF0_KL(0x0FD1)								
Bit	7	6	5	4	3	2	1	0
Name	LPF0_K[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	LPF0_K[15:0]	K input of LPF0

10.4.51 LPF0_X (0x0FD2, 0x0FD3)

LPF0_XH(0x0FD2)								
Bit	15	14	13	12	11	10	9	8
Name	LPF0_X[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
LPF0_XL(0x0FD3)								
Bit	7	6	5	4	3	2	1	0
Name	LPF0_X[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	LPF0_X[15:0]	X input of LPF0

10.4.52 LPF0_Y (0x0FD4, 0x0FD5, 0x0FD6, 0x0FD7)

LPF0_YHH(0x0FD4)								
Bit	31	30	29	28	27	26	25	24
Name	LPF0_Y[31:24]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
LPF0_YHL(0x0FD5)								
Bit	23	22	21	20	19	18	17	16
Name	LPF0_Y[23:16]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
LPF0_YLH(0x0FD6)								
Bit	15	14	13	12	11	10	9	8
Name	LPF0_Y[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
LPF0_YLL(0x0FD7)								
Bit	7	6	5	4	3	2	1	0
Name	LPF0_Y[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[31:0]	LPF0_Y[31:0]	Input and output of the register in LPF0 Input: LPF0_Y_{k-1} Output: LPF0_Y_k

10.4.53 LPF1_K (0x0FC8, 0x0FC9)

LPF1_KH(0x0FC8)								
Bit	15	14	13	12	11	10	9	8
Name	LPF1_K[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
LPF1_KL(0x0FC9)								
Bit	7	6	5	4	3	2	1	0
Name	LPF1_K[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description

[15:0]	LPF1_K[15:0]	K input of LPF1

10.4.54 LPF1_X (0x0FCA, 0x0FCB)

LPF1_XH(0x0FCA)								
Bit	15	14	13	12	11	10	9	8
Name	LPF1_X[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
LPF1_XL(0x0FCB)								
Bit	7	6	5	4	3	2	1	0
Name	LPF1_X[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	LPF1_X[15:0]	X input of LPF1

10.4.55 LPF1_Y (0x0FCC, 0x0FCD, 0x0FCE, 0x0FCF)

LPF1_YHH(0x0FCC)								
Bit	31	30	29	28	27	26	25	24
Name	LPF1_Y[31:24]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
LPF1_YHL(0x0FCD)								
Bit	23	22	21	20	19	18	17	16
Name	LPF1_Y[23:16]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
LPF1_YLH(0x0FCE)								
Bit	15	14	13	12	11	10	9	8
Name	LPF1_Y[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
LPF1_YLL(0x0FCF)								
Bit	7	6	5	4	3	2	1	0
Name	LPF1_Y[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[31:0]	LPF1_Y[31:0]	Input and output of the register in LPF1

		Input: LPF1_Y_{k-1} Output: LPF1_Y_k

10.4.56 LPF2_K (0x0F78, 0x0F79)

LPF2_KH(0x0F78)								
Bit	15	14	13	12	11	10	9	8
Name	LPF2_K[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
LPF2_KL(0x0F79)								
Bit	7	6	5	4	3	2	1	0
Name	LPF2_K[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	LPF2_K[15:0]	K input of LPF2

10.4.57 LPF2_X (0x0F7A, 0x0F7B)

LPF2_XH(0x0F7A)								
Bit	15	14	13	12	11	10	9	8
Name	LPF2_X[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
LPF2_XL(0x0F7B)								
Bit	7	6	5	4	3	2	1	0
Name	LPF2_X[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	LPF2_X[15:0]	X input of LPF2

10.4.58 LPF2_Y (0x0F7C, 0x0F7D, 0x0F7E, 0x0F7F)

LPF2_YHH(0x0F7C)								
Bit	31	30	29	28	27	26	25	24
Name	LPF2_Y[31:24]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
LPF2_YHL(0x0F7D)								
Bit	23	22	21	20	19	18	17	16

Name	LPF2_Y[23:16]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
LPF2_YLH(0x0F7E)								
Bit	15	14	13	12	11	10	9	8
Name	LPF2_Y[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
LPF2_YLL(0x0F7F)								
Bit	7	6	5	4	3	2	1	0
Name	LPF2_Y[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[31:0]	LPF2_Y[31:0]	Input and output of the register in LPF2 Input: LPF2_Y_{k-1} Output: LPF2_Y_k

10.4.59 LPF3_K (0x0F70, 0x0F71)

LPF3_KH(0x0F70)								
Bit	15	14	13	12	11	10	9	8
Name	LPF3_K[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
LPF3_KL(0x0F71)								
Bit	7	6	5	4	3	2	1	0
Name	LPF3_K[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	LPF3_K[15:0]	K input of LPF3

10.4.60 LPF3_X (0x0F72, 0x0F73)

LPF3_XH(0x0F72)								
Bit	15	14	13	12	11	10	9	8
Name	LPF3_X[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
LPF3_XL(0x0F73)								

Bit	7	6	5	4	3	2	1	0
Name	LPF3_X[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	LPF3_K[15:0]	X input of LPF3

10.4.61 LPF3_Y (0x0F74, 0x0F75, 0x0F76, 0x0F77)

LPF3_YHH(0x0F74)								
Bit	31	30	29	28	27	26	25	24
Name	LPF3_Y[31:24]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
LPF3_YHL(0x0F75)								
Bit	23	22	21	20	19	18	17	16
Name	LPF3_Y[23:16]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
LPF3_YLH(0x0F76)								
Bit	15	14	13	12	11	10	9	8
Name	LPF3_Y[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
LPF3_YLL(0x0F77)								
Bit	7	6	5	4	3	2	1	0
Name	LPF3_Y[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[31:0]	LPF3_Y[31:0]	Input and output of the register in LPF3 Input: LPF3_Y_{k-1} Output: LPF3_Y_k

10.4.62 PI0_KP (0x0FB8, 0x0FB9)

PI0_KPH(0x0FB8)								
Bit	15	14	13	12	11	10	9	8
Name	PI0_KP[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

PI0_KPL(0x0FB9)								
Bit	7	6	5	4	3	2	1	0
Name	PI0_KP[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI0_KP	Proportional coefficient of PI0

10.4.63 PI0_EK1 (0x0FBA, 0x0FBB)

PI0_EK1H(0x0FBA)								
Bit	15	14	13	12	11	10	9	8
Name	PI0_EK1[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI0_EK1L(0x0FBB)								
Bit	7	6	5	4	3	2	1	0
Name	PI0_EK1[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI0_EK1	Previous input deviation of PI0

10.4.64 PI0_EK (0x0FBC, 0x0FBD)

PI0_EKH(0x0FBC)								
Bit	15	14	13	12	11	10	9	8
Name	PI0_EK[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI0_EKL(0x0FBD)								
Bit	7	6	5	4	3	2	1	0
Name	PI0_EK[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI0_EK	Present input deviation of PI0

10.4.65 PI0_KI (0x0FBE, 0x0FBF)

PI0_KIH(0x0FBE)								
Bit	15	14	13	12	11	10	9	8
Name	PI0_KI[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI0_KIL(0x0FBF)								
Bit	7	6	5	4	3	2	1	0
Name	PI0_KI[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI0_KI	Integral coefficient of PI0

10.4.66 PI0_UKH (0x0FC0, 0x0FC1)

PI0_UKHH(0x0FC0)								
Bit	15	14	13	12	11	10	9	8
Name	PI0_UKH[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI0_UKHL(0x0FC1)								
Bit	7	6	5	4	3	2	1	0
Name	PI0_UKH[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI0_UKH	16 high-order bits of PI0 output

10.4.67 PI0_UKL (0x0FC2, 0x0FC3)

PI0_UKLH(0x0FC2)								
Bit	15	14	13	12	11	10	9	8
Name	PI0_UKL[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI0_UKLL(0x0FC3)								
Bit	7	6	5	4	3	2	1	0
Name	PI0_UKL[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI0_UKL	16 low-order bits of PI0 output

10.4.68 PI0_UKMAX (0x0FC4, 0x0FC5)

PI0_UKMAXH(0x0FC4)								
Bit	15	14	13	12	11	10	9	8
Name	PI0_UKMAX[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI0_UKMAXL(0x0FC5)								
Bit	7	6	5	4	3	2	1	0
Name	PI0_UKMAX[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI0_UKMAX	Maximum output of PI0

10.4.69 PI0_UKMIN (0x0FC6, 0x0FC7)

PI0_UKMINH(0x0FC6)								
Bit	15	14	13	12	11	10	9	8
Name	PI0_UKMIN[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI0_UKMINL(0x0FC7)								
Bit	7	6	5	4	3	2	1	0
Name	PI0_UKMIN[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI0_UKMIN	Minimum output of PI0

10.4.70 PI1_KP (0x0FA8, 0x0FA9)

PI1_KPH(0x0FA8)								
Bit	15	14	13	12	11	10	9	8
Name	PI1_KP[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

PI1_KPL(0x0FA9)								
Bit	7	6	5	4	3	2	1	0
Name	PI1_KP[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI1_KP	Proportional coefficient of PI1

10.4.71 PI1_EK1 (0x0FAA, 0x0FAB)

PI1_EK1H(0x0FAA)								
Bit	15	14	13	12	11	10	9	8
Name	PI1_EK1[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI1_EK1L(0x0FAB)								
Bit	7	6	5	4	3	2	1	0
Name	PI1_EK1[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI1_EK1	Previous input deviation of PI1

10.4.72 PI1_EK (0x0FAC, 0x0FAD)

PI1_EKH(0x0FAC)								
Bit	15	14	13	12	11	10	9	8
Name	PI1_EK[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI1_EKL(0x0FAD)								
Bit	7	6	5	4	3	2	1	0
Name	PI1_EK[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI1_EK	Present input deviation of PI1

10.4.73 PI1_KI (0x0FAE, 0x0FAF)

PI1_KIH(0x0FAE)								
Bit	15	14	13	12	11	10	9	8
Name	PI1_KI[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI1_KIL(0x0FAF)								
Bit	7	6	5	4	3	2	1	0
Name	PI1_KI[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI1_KI	Integral coefficient of PI1

10.4.74 PI1_UKH (0x0FB0, 0x0FB1)

PI1_UKHH(0x0FB0)								
Bit	15	14	13	12	11	10	9	8
Name	PI1_UKH[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI1_UKHL(0x0FB1)								
Bit	7	6	5	4	3	2	1	0
Name	PI1_UKH[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI1_UKH	16 high-order bits of PI1 output

10.4.75 PI1_UKL (0x0FB2, 0x0FB3)

PI1_UKLH(0x0FB2)								
Bit	15	14	13	12	11	10	9	8
Name	PI1_UKL[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI1_UKLL(0x0FB3)								
Bit	7	6	5	4	3	2	1	0
Name	PI1_UKL[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI1_UKL	16 low-order bits of PI1 output

10.4.76 PI1_UKMAX (0x0FB4, 0x0FB5)

PI1_UKMAXH(0x0FB4)								
Bit	15	14	13	12	11	10	9	8
Name	PI1_UKMAX[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI1_UKMAXL(0x0FB5)								
Bit	7	6	5	4	3	2	1	0
Name	PI1_UKMAX[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI1_UKMAX	Maximum output of PI1

10.4.77 PI1_UKMIN (0x0FB6, 0x0FB7)

PI1_UKMINH(0x0FB6)								
Bit	15	14	13	12	11	10	9	8
Name	PI1_UKMIN[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI1_UKMINL(0x0FB7)								
Bit	7	6	5	4	3	2	1	0
Name	PI1_UKMIN[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI1_UKMIN	Minimum output of PI1

10.4.78 PI2_KP (0x0F5C, 0x0F5D)

PI2_KPH(0x0F5C)								
Bit	15	14	13	12	11	10	9	8
Name	PI2_KP[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

PI2_KPL(0x0F5D)								
Bit	7	6	5	4	3	2	1	0
Name	PI2_KP[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI2_KP	Proportional coefficient of PI2

10.4.79 PI2_EK1 (0x0F5E, 0x0F5F)

PI2_EK1H(0x0F5E)								
Bit	15	14	13	12	11	10	9	8
Name	PI2_EK1[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI2_EK1L(0x0F5F)								
Bit	7	6	5	4	3	2	1	0
Name	PI2_EK1[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI2_EK1	Previous input deviation of PI2

10.4.80 PI2_EK (0x0F60, 0x0F61)

PI2_EKH(0x0F60)								
Bit	15	14	13	12	11	10	9	8
Name	PI2_EK[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI2_EKL(0x0F61)								
Bit	7	6	5	4	3	2	1	0
Name	PI2_EK[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI2_EK	Present input deviation of PI2

10.4.81 PI2_KI (0x0F62, 0x0F63)

PI2_KIH(0x0F62)								
Bit	15	14	13	12	11	10	9	8
Name	PI2_KI[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI2_KIL(0x0F63)								
Bit	7	6	5	4	3	2	1	0
Name	PI2_KI[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI2_KI	Integral coefficient of PI2

10.4.82 PI2_UKH (0x0F64, 0x0F65)

PI2_UKHH(0x0F64)								
Bit	15	14	13	12	11	10	9	8
Name	PI2_UKH[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI2_UKHL(0x0F65)								
Bit	7	6	5	4	3	2	1	0
Name	PI2_UKH[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI2_UKH	16 high-order bits of PI2 output

10.4.83 PI2_UKL (0x0F66, 0x0F67)

PI2_UKLH(0x0F66)								
Bit	15	14	13	12	11	10	9	8
Name	PI2_UKL[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI2_UKLL(0x0F67)								
Bit	7	6	5	4	3	2	1	0
Name	PI2_UKL[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI2_UKL	16 low-order bits of PI2 output

10.4.84 PI2_UKMAX (0x0F68, 0x0F69)

PI2_UKMAXH(0x0F68)								
Bit	15	14	13	12	11	10	9	8
Name	PI2_UKMAX[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI2_UKMAXL(0x0F69)								
Bit	7	6	5	4	3	2	1	0
Name	PI2_UKMAX[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI2_UKMAX	Maximum output of PI2

10.4.85 PI2_UKMIN (0x0F6A, 0x0F6B)

PI2_UKMINH(0x0F6A)								
Bit	15	14	13	12	11	10	9	8
Name	PI2_UKMIN[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI2_UKMINL(0x0F6B)								
Bit	7	6	5	4	3	2	1	0
Name	PI2_UKMIN[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI2_UKMIN	Minimum output of PI2

10.4.86 PI2_KD (0x0F6C, 0x0F6D)

PI2_KDH(0x0F6C)								
Bit	15	14	13	12	11	10	9	8
Name	PI2_KD[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

PI2_KDL(0x0F6D)								
Bit	7	6	5	4	3	2	1	0
Name	PI2_KD[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI2_KD	Differential coefficient of PI2

10.4.87 PI2_EK2 (0x0F6E, 0x0F6F)

PI2_EK2H(0x0F6E)								
Bit	15	14	13	12	11	10	9	8
Name	PI2_EK2[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI2_EK2L(0x0F6F)								
Bit	7	6	5	4	3	2	1	0
Name	PI2_EK2[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI2_EK2	Deviation before previous input of PI2

10.4.88 PI3_KP (0x0F48, 0x0F49)

PI3_KPH(0x0F48)								
Bit	15	14	13	12	11	10	9	8
Name	PI3_KP[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI3_KPL(0x0F49)								
Bit	7	6	5	4	3	2	1	0
Name	PI3_KP[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI3_KP	Proportional coefficient of PI3

10.4.89 PI3_EK1 (0x0F4A, 0x0F4B)

PI3_EK1H(0x0F4A)								
Bit	15	14	13	12	11	10	9	8
Name	PI3_EK1[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI3_EK1L(0x0F4B)								
Bit	7	6	5	4	3	2	1	0
Name	PI3_EK1[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI3_EK1	Previous input deviation of PI3

10.4.90 PI3_EK (0x0F4C, 0x0F4D)

PI3_EKH(0x0F4C)								
Bit	15	14	13	12	11	10	9	8
Name	PI3_EK[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI3_EKL(0x0F4D)								
Bit	7	6	5	4	3	2	1	0
Name	PI3_EK[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI3_EK	Present input deviation of PI3

10.4.91 PI3_KI (0x0F4E, 0x0F4F)

PI3_KIH(0x0F4E)								
Bit	15	14	13	12	11	10	9	8
Name	PI3_KI[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI3_KIL(0x0F4F)								
Bit	7	6	5	4	3	2	1	0
Name	PI3_KI[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI3_KI	Integral coefficient of PI3

10.4.92 PI3_UKH (0x0F50, 0x0F51)

PI3_UKHH(0x0F50)								
Bit	15	14	13	12	11	10	9	8
Name	PI3_UKH[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI3_UKHL(0x0F51)								
Bit	7	6	5	4	3	2	1	0
Name	PI3_UKH[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI3_UKH	16 high-order bits of PI3 output

10.4.93 PI3_UKL (0x0F52, 0x0F53)

PI3_UKLH(0x0F52)								
Bit	15	14	13	12	11	10	9	8
Name	PI3_UKL[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI3_UKLL(0x0F53)								
Bit	7	6	5	4	3	2	1	0
Name	PI3_UKL[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI3_UKL	16 low-order bits of PI3 output

10.4.94 PI3_UKMAX (0x0F54, 0x0F55)

PI3_UKMAXH(0x0F54)								
Bit	15	14	13	12	11	10	9	8
Name	PI3_UKMAX[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

PI3_UKMAXL(0x0F55)								
Bit	7	6	5	4	3	2	1	0
Name	PI3_UKMAX[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI3_UKMAX	Maximum output of PI3

10.4.95 PI3_UKMIN (0x0F56, 0x0F57)

PI3_UKMINH(0x0F56)								
Bit	15	14	13	12	11	10	9	8
Name	PI3_UKMIN[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI3_UKMINL(0x0F57)								
Bit	7	6	5	4	3	2	1	0
Name	PI3_UKMIN[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI3_UKMIN	Minimum output of PI3

10.4.96 PI3_KD (0x0F58, 0x0F59)

PI3_KDH(0x0F58)								
Bit	15	14	13	12	11	10	9	8
Name	PI3_KD[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI3_KDL(0x0F59)								
Bit	7	6	5	4	3	2	1	0
Name	PI3_KD[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI3_KD	Differential coefficient of PI3

10.4.97 PI3_EK2 (0x0F5A, 0x0F5B)

PI3_EK2H(0x0F5A)								
Bit	15	14	13	12	11	10	9	8
Name	PI3_EK2[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PI3_EK2L(0x0F5B)								
Bit	7	6	5	4	3	2	1	0
Name	PI3_EK2[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PI3_EK2	Deviation before previous input of PI3

11 PFC

11.1 PFC Operating Instructions

11.1.1 PFC Introduction

Power Factor Correction (PFC) improves power efficiency and power density, optimizes voltage regulation stability and reduces electromagnetic compatibility and electromagnetic interference.

PFC module has the following features:

- Full-automatic hardware
- ADC automatic sampling
- Over-current protection and cycle-by-cycle current limiting

Note

PFC module is available in single-shunt current sampling mode only

Figure 11-1 Structure Diagram of PFC Module

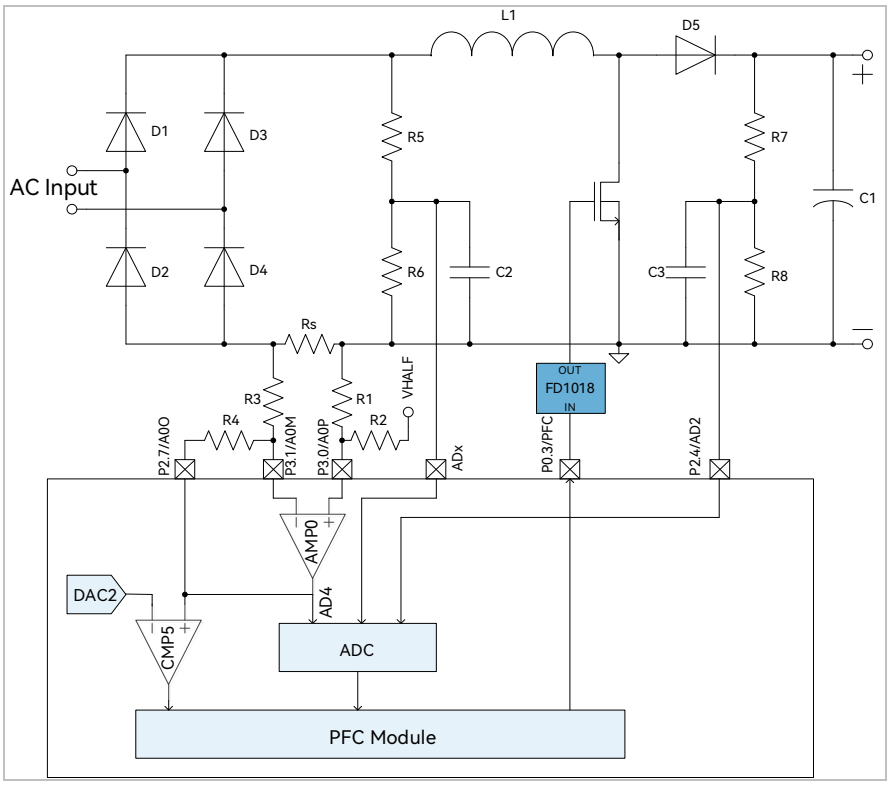
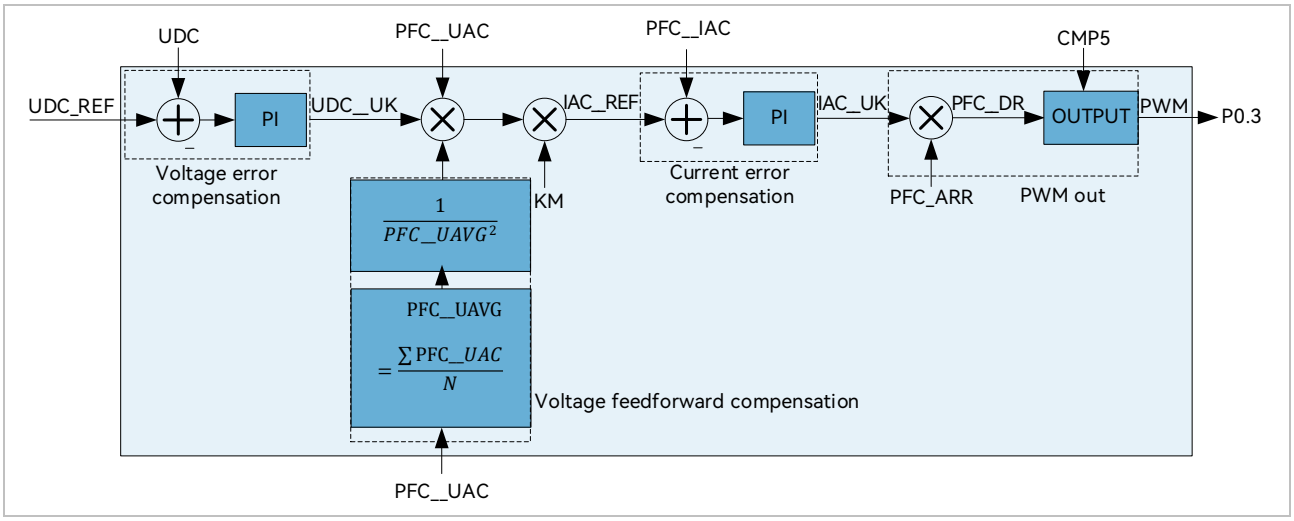


Figure 11-2 Block Diagram of PFC Module



PFC module includes voltage error compensation module, voltage feedforward compensation module, current error compensation module and PWM output module.

11.1.2 Voltage Error Compensation Module

Voltage error compensation module is outer loop of the PFC module. Its input is the difference between UDC_REF (user defined DC voltage reference) and UDC (ADC-sampled DC voltage), which is transmitted to PI controller to generate control output UDC__UK. Outer loop frequency = Inner loop frequency / PFC_OUTARR = 24M / PFC_ARR / PFC_OUTARR.

11.1.3 Voltage Feedforward Compensation Module

The voltage feed-forward compensation module is mainly used to maintain a constant output power under unstable input AC voltage.

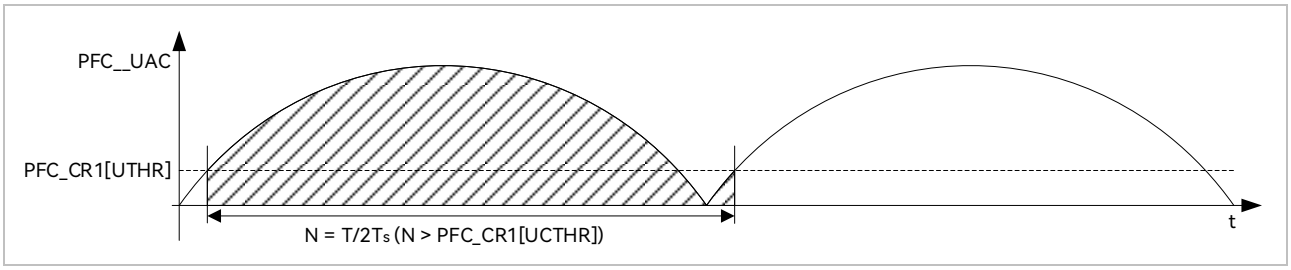
11.1.4 Calculation of Average Voltage PFC__UAVG

PFC__UAVG is the rectified average voltage of AC voltage (PFC__UAC). The PFC module calculates PFC__UAVG by hardware automatically. This function shall be disabled for some special applications and UAVG will be calculated by software. PFC__UAVG is calculated as

$$\text{PFC__UAVG} = \frac{\sum PFC__UAC}{N}$$

where, PFC__UAC is the input AC voltage, N is the sampling number in time period T_s.

Figure 11-3 Calculation of PFC_UAVG



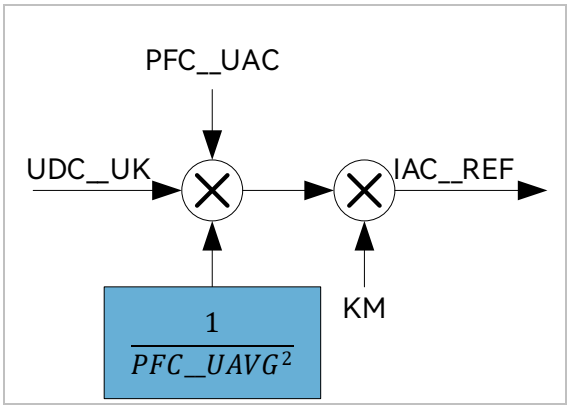
PFC_CR0[UAVGCDIS] is set to "0" to start the calculation of PFC_UAVG by hardware. The calculation period of PFC_UAVG is a half period of power frequency T. PFC_1[UTHR] is the first and the last value of sampling voltage. Ts is the sampling period of PFC_UAC. PFC_UCTHR is the minimum value of PFC_UAC sampling number. When PFC_UAC in this sampling period is bigger than PFC_1[UTHR] and the former one is smaller, this sampling period is the start or the end of a calculation period. To decrease the influence of sampling distortion, N should be adequate, larger than PFC_1[UCTHR].

PFC_CR0[UAVGCDIS] is set to "1" to start the calculation of PFC_UAVG by software. When PFC_CR0[UAVGSW] = 1, the calculation period is the time between the end of last calculation period and this sampling period (PFC_CR0[UAVGSW]). Systick or other Timer can be used to generate a frequency for the accurate calculation of PFC_UAVG.

11.1.5 Current Error Compensation Module

Current error compensation module is inner loop of the PFC module. Its input is the difference between IAC_REF (current reference calculated by outer loop) and PFC_IAC (sampled current by ADC), which is transmitted to PI controller to generate control output IAC_UK. Inner loop frequency = 24MHz/PFC_ARR.

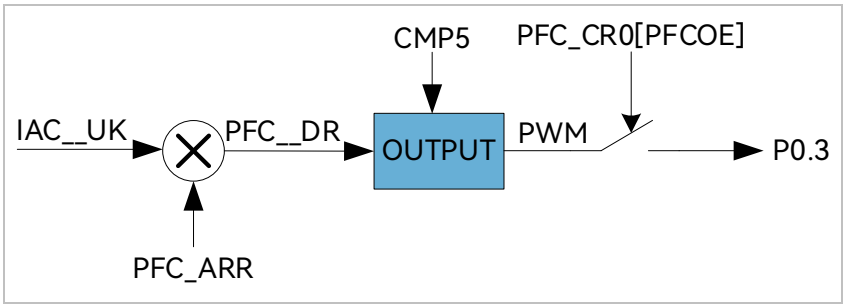
Figure 11-4 Block Diagram of the Calculation of IACREG



As shown above, multiply UDC_UK, PFC__UAC, the output voltage of feedforward compensation module and the constant KM, the result is IAC__REF.

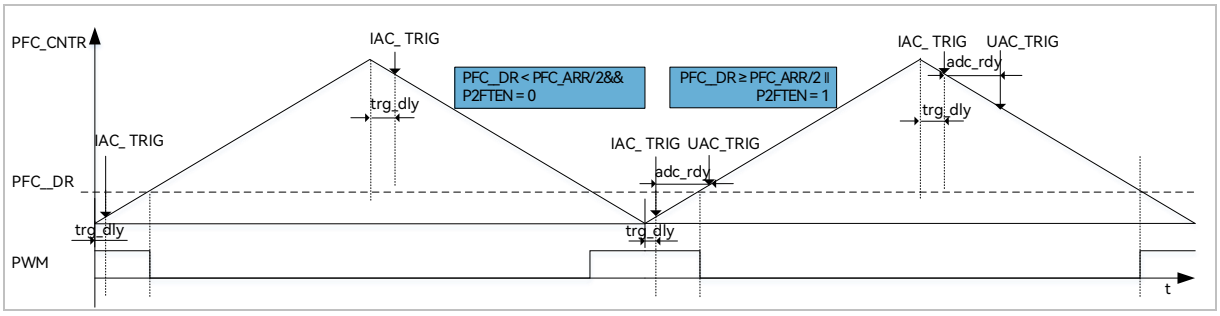
11.1.6 PWM Output Module

Figure 11-5 Block Diagram of PWM Output



IAC__UK is the duty cycle of PWM module, duty cycle = 100%*IAC__UK/32768, and PWM signal is generated by comparator and output to P0.3. As shown above, PFC__DR, the value IAC__UK*PFC_ARR/32768, is compared with PFC timer to generate PWM signal. When PFC__DR > PFC_CNTR, PWM module outputs logical 1, and when PFC__DR < PFC_CNTR, PWM module outputs logical 0. If PFC_CR0[PFCOE] is set to “1”, P0.3 serves as output of the PWM module.

Figure 11-6 Diagram of PWM Output and PFC__IAC / PFC__UAC Sampling



11.1.7 Over-current Protection and Cycle-by-Cycle Current Limiting

Configuring CMP_CR4[CMP5EN] to “1” enables CMP5 and over-current protection feature. The filter factor of CMP5 is determined by PFC_CR0[CPM5DIV]. When the input of CMP5 is logical 1, over-current protection forces PWM module to output low voltage level. Configuring PFC_CR0[PFCOE] to “0” disables the over-current protection feature.

11.1.8 PFC_UAC/PFC_IAC/UDC Sampling

11.1.8.1 UDC Sampling

- UDC is sampled by FOC module every carrier period.
- ADC channel 2 is used for the sampling

11.1.8.2 PFC_IAC Sampling

- Sample once on overflow point or underflow point of every inner loop period
- The sampling point is determined according to PFC_CR2[P2FTEN] and PFC_DR (as shown in Figure 11-6). If PFC_CR2[P2FTEN] = 0 and PFC_DR < PFC_ARR/2, sampling is triggered when PFC_CNTR reaches the underflow point. If PFC_CR2[P2FTEN] = 1 or PFC_DR ≥PFC_ARR/2, sampling is triggered when PFC_CNTR reaches the overflow point. The time delay for sampling is configurable by PFC_TRGDLY. The overflow point can be set at PFC_TRGDLY*8, and the underflow point at (PFC_ARR - PFC_TRGDLY*8)
- ADC channel 4 is used for the sampling
- When PFC_CR0[CCHSEL] = 0, a data can be written to PFC_CSO to set PFC_IAC offset. Providing the voltage range of ADC is 0V ~ 5V and the reference is 2.5V, then PFC_CSO = 32768 × 2.5 / 5V = 16384 (0x4000)

11.1.8.3 PFC_UAC Sampling

- Set PFC_CR1[UACSAMSEL] to configure PFC_UAC sampling period, once every 1/2/4/8 inner loop periods. PFC_UAC is sampled after sampling of PFC_IAC.
- ADC channel 5 is used for the sampling by default. Configuring PFC_ADCCH[UAC_TRIG_CH] selects other ADC channels.
- When PFC_CR0[CCHSEL] = 1, a data can be written to PFC_CSO to set PFC_UAC offset. Providing the voltage range of ADC is 0V ~ 5V and the reference is 2.5V, then PFC_CSO = 32768 × 2.5 / 5V = 16384 (0x4000).

11.2 PFC Registers

11.2.1 PFC_CR2 (0x4063)

Bit	7	6	5	4	3	2	1	0
Name	PFC_BLK_MD		P2FTEN	DCLREN	PIAUTOEN	ADTRIGEN	DRALEN	PFCCEN
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:6]	PFC_BLK_MD	When PFC module is disabled (PFC_CR0[PFCEN] = 0), UDC_PI/IAC_PI works as a general PI controller. PFC_CR2[7] serves as the enable bit of UDC_PI, and PFC_CR2[6] as the enable bit of IAC_PI. This bit is set by software to logical 1, and will be cleared by hardware at next clock. It is invalid to write "0" to this bit. When PFC module is disabled, this bit reflects busy status of the PI controller. 0: Disable 1: Enable When PFC module is enabled (PFC_CR0[PFCEN] = 1), this bit is used to select the mask time for PFC_UAC/PFC_IAC sampling. 00: The mask time equals the deadtime 01: The mask time is 1/2 of the deadtime 10: The mask time is twice the deadtime
[5]	P2FTEN	When PFC module is enabled (PFC_CR0[PFCEN] = 1), this bit and PFC_DR determine triggered sampling point for PFC_IAC sampling. See PFC__IAC Sampling for more details.
[4]	DCLREN	PFC Timer Synchronization Enable With this bit is enabled, DRV timer is synchronized with PFC timer. When DRV timer generates an underflow event, PFC timer is cleared to "0", so that both DRV timer and PFC timer count from 0. It is mainly used to start ADC automatic sampling when PFC/DRV timer reaches a certain value (by setting PFC_DR) if PFC module is disabled. 0: Disable 1: Enable
[3]	PIAUTOEN	UDC_PI/IAC_PI Automatic Enable When PFC module is disabled (PFC_CR0[PFCEN] = 0), UDC_PI/IAC_PI works as a general PI controller. With this bit enabled, the two PI controllers automatically operate once per cycle of the PFC timer. When PFC_CR0[PFCEN] = 1, this bit is automatically set to "1". 0: Disable 1: Enable
[2]	ADTRIGEN	ADC Automatic Enable

		0: Disable 1: Enable
[1]	DRALEN	PFC__DR Automatic Loading Enable With this bit enabled, PFC__DR is calculated at the underflow point of each PFC period based on the duty cycle. The formula is IAC__UK/32768*PFC_ARR. The result is updated to PFC__DR. When PFC_CR0[PFCEN] = 1, this bit is automatically set to “1”. 0: Disable 1: Enable
[0]	PFCCEN	PFC Timer Enable When PFC_CR0[PFCEN] = 1, this bit is automatically set to “1”. 0: Disable 1: Enable

11.2.2 PFC_CR0 (0x40E0)

Bit	7	6	5	4	3	2	1	0
Name	UAVGCSW	CMP5DIV		UAVGDIS	PFCOA	CCHSEL	PFCOE	PFCEN
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	UAVGCSW	Start PFC_UAVG Calculation Start PFC_UAVG calculation and update the result to the value of PFC_UAVG. This bit is set to “1” by software, and will be cleared by hardware at next clock. It is invalid to write “0” to this bit. 0: PFC_UAVG calculation doesn’t start. 1: PFC_UAVG calculation starts.
[6:5]	CMP5DIV	Filter Period Selection of CMP5 When input pulse width of CMP5 is less than the set value, it will be considered as noise and automatically filtered out by the hardware. 00: Disable 01: 4 system clocks 10: 8 system clocks 11: 16 system clocks
[4]	UAVGCDIS	PFC_UAVG Calculation by Hardware Enable When this bit is disabled, the hardware does not calculate PFC_UAVG at each power frequency period. You need to configure PFC_CR0[UAVGSW] to start PFC_UAVG calculation. PFC_UAVG calculation also starts automatically when internal timer overflows. 0: Enable 1: Disable

[3]	PFCOA	Cycle-by-cycle Current Limiting Feature Enable After CMP5 is enabled, over-current protection feature is enabled by default. When over-current event occurs, PFC module turns off the outputs. After PFCOA is enabled, the hardware automatically restores the output after the over-current protection state is released, that is, cycle-by-cycle current limiting feature. 0: Disable 1: Enable
[2]	CCHSEL	ADC Offset Selection Configuring this bit writes data to PFC_CSO to set ADC offset for PFC_IAC or PFC_UAC sampling. 0: PFC_IAC sampling 1: PFC_UAC sampling
[1]	PFCOE	PFC Output Enable With this bit enabled, PWM signal generated by PFC module is output to P0.3. 0: Disable 1: Enable
[0]	PFCEN	PFC Enable 0: Disable 1: Enable

11.2.3 PFC_CR1/UDC_UKMINH (0x40F2)

Bit	7	6	5	4	3	2	1	0
Name	UACSAMSEL		UTHR	UCTHR				
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:6]	UACSAMSEL	PFC_UAC Sampling Period Sample PFC_UAC once every x PFC periods. 00: 1 PFC period 01: 2 PFC periods 10: 4 PFC periods 11: 8 PFC periods
[5]	UTHR	PFC_UAVG Calculation Period Threshold The sampling value, which is higher or lower than this threshold, is set as the first and the last value of a calculation period. The threshold is based on UAC_BASE. 0: 1/16*UAC_BASE 1: 1/8*UAC_BASE
[4:0]	UCTHR	Minimum Number of PFC_UAC Sampling Times The calculation value of PFC_[UAVG] is reasonable under sampling times no less than this value. The minimum sampling times = PFC_CR1[UCTHR]*32

Note

- PFC_CR1 is valid only when PFC module is enabled (PFC_CR0[PFCEN] = 1)
- When PFC module is enabled (PFC_CR0[PFCEN] = 1), this register is used for PFC_CR1 configuration. When PFC module is disabled, it is the minimum value of UDC_PI, or UDC_UKMINH

11.2.4 PFC_ADCCH (0x40E1)

Bit	7	6	5	4	3	2	1	0
Name	IAC_TRIG_CH				UAC_TRIG_CH			
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	1	0	1	0	1	0	1

Bit	Name	Description
[7:4]	IAC_TRIG_CH	ADC Channel Selection for PFC_IAC Sampling Choose ADC channel 4 when PFC is enabled, otherwise the operational amplifier is unavailable. Table 11-1 ADC Channel Selection for PFC_IAC Sampling
[3:0]	UAC_TRIG_CH	ADC channel selection for PFC_UAC sampling Table 11-2 ADC Channel Selection for PFC_UAC Sampling

IAC_TRIG_CH	ADC Channel	IAC_TRIG_CH	ADC Channel
0000	Channel 0	0001	Channel 1
0010	Channel 2	0011	Channel 3
0100	Channel 4	0101	Channel 5
0110	Channel 6	0111	Channel 7
1000	Channel 8	1001	Channel 9
1010	Channel 10	1011	RSV
1100	RSV	1101	RSV
1110	RSV	1111	RSV

UAC_TRIG_CH	ADC Channel	UAC_TRIG_CH	ADC Channel
0000	Channel 0	0001	Channel 1
0010	Channel 2	0011	Channel 3
0100	Channel 4	0101	Channel 5
0110	Channel 6	0111	Channel 7
1000	Channel 8	1001	Channel 9
1010	Channel 10	1011	RSV
1100	RSV	1101	RSV
1110	RSV	1111	RSV

11.2.5 PFC_CSO (0x40E2, 0x40E3)

PFC_CSOH(0x40E2)								
Bit	15	14	13	12	11	10	9	8
Name	PFC_CSO[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PFC_CSOL(0x40E3)								
Bit	7	6	5	4	3	2	1	0
Name	PFC_CSO[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PFC_CSO	PFC_IAC/PFC_UAC Sampling Reference Configure PFC_CR0[CCHSEL] to write data to PFC_CSO set PFC_IAC/PFC_UAC ADC offset. Range [0,32767]. MSB is always 0. Note Providing ADC voltage range is 0V ~ 5V, the offset is 2.5V, then PFC_CSO = 2.5/5V*32768 = 16384 (0x4000)

11.2.6 PFC_ARR (0x40E4, 0x40E5)

PFC_ARRH(0x40E4)								
Bit	15	14	13	12	11	10	9	8
Name	RSV				PFC_ARR[11:8]			
Type	-	-	-	-	W	W	W	W
Reset	-	-	-	-	0	0	0	0
PFC_ARRL(0x40E5)								
Bit	7	6	5	4	3	2	1	0
Name	PFC_ARR[7:0]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:12]	RSV	Reserved
[11:0]	PFC_ARR	Reload Value of PFC Timer. It configures the carrier period and operation mode (center-alignment mode). An overflow event occurs when PFC timer counts from 0 and reaches PFC_ARR, and then it counts down to 0. This register is write-only. Range [0,4095]

11.2.7 PFC_UAVG (0x40E4, 0x40E5)

PFC_UAVGH(0x40E4)								
Bit	15	14	13	12	11	10	9	8
Name	PFC_UAVG[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
PFC_UAVGL(0x40E5)								
Bit	7	6	5	4	3	2	1	0
Name	PFC_UAVG[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PFC_UAVG	Average PFC_UAC in One Power Frequency Range [-32768,32767]

11.2.8 PFC_DR (0x40E6, 0x40E7)

PFC_DRH(0x40E6)								
Bit	15	14	13	12	11	10	9	8
Name	RSV				PFC_DR[11:8]			
Type	-	-	-	-	R/W	R/W	R/W	R/W
Reset	-	-	-	-	0	0	0	0
PFC_DRL(0x40E7)								
Bit	7	6	5	4	3	2	1	0
Name	PFC_DR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:12]	RSV	Reserved
[11:0]	PFC_DR	Comparison Value by PFC to Generate PWM Signal When PFC_DR > PFC_CNTR, PWM module outputs logical 1, and when PFC_DR < PFC_CNTR, PWM module outputs logical 0. When PFC module is enabled, PFC_DR is automatically updated by hardware. Range [0,4095]

11.2.9 UDC_REF (0x40E8, 0x40E9)

UDC_REFH(0x40E8)								
Bit	15	14	13	12	11	10	9	8
Name	UDC_REF[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
UDC_REFL(0x40E9)								
Bit	7	6	5	4	3	2	1	0
Name	UDC_REF[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	UDC_REF	This register is the UDC reference when PFC is enabled (PFC_CR0[PFCEN] = 1). This register is EK of UDC_PI when PFC is disabled (PFC_CR0[PFCEN] = 0). Range [-32768,32767]

11.2.10 UDC__UK (0x40EA, 0x40EB)

UDC__UKH(0x40EA)								
Bit	15	14	13	12	11	10	9	8
Name	UDC__UK[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
UDC__UKL(0x40EB)								
Bit	7	6	5	4	3	2	1	0
Name	UDC__UK[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	UDC__UK	Output value UK of UDC PI controller Range [-32768,32767]

11.2.11 UDC_KP (0x40EC, 0x40ED)

UDC_KPH(0x40EC)								
Bit	15	14	13	12	11	10	9	8
Name	UDC_KP[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
UDC_KPL(0x40ED)								
Bit	7	6	5	4	3	2	1	0
Name	UDC_KP[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	UDC_KP	KP coefficient of UDC PI controller Range [0,32767]. MSB is always 0. The data format is Q10.

11.2.12 UDC_KI (0x40EE, 0x40EF)

UDC_KIH(0x40EE)								
Bit	15	14	13	12	11	10	9	8
Name	UDC_KI[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
UDC_KIL(0x40EE)								
Bit	7	6	5	4	3	2	1	0
Name	UDC_KI[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	UDC_KI	KI coefficient of UDC PI controller Range [0,32767]. MSB is always 0. The data format is Q15.

11.2.13 UDC_UKMAX (0x40F0, 0x40F1)

UDC_UKMAXH(0x40F0)								
Bit	15	14	13	12	11	10	9	8
Name	UDC_UKMAX[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
UDC_UKMAXL(0x40F1)								
Bit	7	6	5	4	3	2	1	0
Name	UDC_UKMAX[7:0]							

Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	UDC_UKMAX	Maximum output of UDC PI controller Range [-32768,32767]

11.2.14 UDC_UKMIN (0x40F2, 0x40F3)

UDC_UKMINH(0x40F2)								
Bit	15	14	13	12	11	10	9	8
Name	UDC_UKMIN[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
UDC_UKMINL(0x40F3)								
Bit	7	6	5	4	3	2	1	0
Name	UDC_UKMIN[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	UDC_UKMIN	This bit is the minimum output of UDC PI controller when PFC module is disabled (PFC_CR0[PFCEN] = 0). Range [-32768,32767] This bit is used for PFC_CR1 and PFC_KM registers when PFC module is enabled (PFC_CR0[PFCEN] = 1). In this case, UDC_UKMIN is 0 by default.

11.2.15 PFC_KM (0x40F3)

Bit	7	6	5	4	3	2	1	0
Name	PFC_KM							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	PFC_KM	KM coefficient of PFC Range [0,255] Note This bit is valid only when PFC module is enabled (PFC_CR0[PFCEN] = 1)

11.2.16 IAC_REF (0x40F4, 0x40F5)

IAC_REFH(0x40F4)								
Bit	15	14	13	12	11	10	9	8
Name	IAC_REF[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
IAC_REFL(0x40F5)								
Bit	7	6	5	4	3	2	1	0
Name	IAC_REF[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	IAC_REF	This register sets EK of IAC_PI when PFC is disabled (PFC_CR0[PFCEN] = 0). This register sets PFC_IAC reference when PFC is enabled (PFC_CR0[PFCEN] = 1). Range [-32768,32767]

11.2.17 IAC_UK (0x40F6, 0x40F7)

IAC_UKH(0x40F6)								
Bit	15	14	13	12	11	10	9	8
Name	IAC_UK[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
IAC_UKL(0x40F7)								
Bit	7	6	5	4	3	2	1	0
Name	IAC_UK[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	IAC_UK	Output value UK of PFC_IAC PI controller Range [-32768,32767]

11.2.18 IAC_KP (0x40F8, 0x40F9)

IAC_KPH(0x40F8)								
Bit	15	14	13	12	11	10	9	8
Name	IAC_KP[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
IAC_KPL(0x40F9)								
Bit	7	6	5	4	3	2	1	0
Name	IAC_KP[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	IAC_KP	KP coefficient of PFC_IAC PI controller Range [0,32767]. MSB is always 0. The data format is Q10.

11.2.19 IAC_KI (0x40FA, 0x40FB)

IAC_KIH(0x40FA)								
Bit	15	14	13	12	11	10	9	8
Name	IAC_KI[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
IAC_KIL(0x40FB)								
Bit	7	6	5	4	3	2	1	0
Name	IAC_KI[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	IAC_KI	KI coefficient of PFC_IAC PI controller Range [0,32767]. MSB is always 0. The data format is Q15.

11.2.20 IAC_UKMAX (0x40FC, 0x40FD)

IAC_UKMAXH(0x40FC)								
Bit	15	14	13	12	11	10	9	8
Name	IAC_UKMAX[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
IAC_UKMAXL(0x40FD)								
Bit	7	6	5	4	3	2	1	0

Name	IAC_UKMAX[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	IAC_UKMAX	Maximum output of PFC_IAC PI controller Range [-32768,32767]

11.2.21 IAC_UKMIN (0x40FE, 0x40FF)

IAC_UKMINH(0x40FE)								
Bit	15	14	13	12	11	10	9	8
Name	IAC_UKMIN[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
IAC_UKMINL(0x40FF)								
Bit	7	6	5	4	3	2	1	0
Name	IAC_UKMIN[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	IAC_UKMIN	This bit is the minimum output of PFC_IAC PI controller when PFC module is disabled (PFC_CR0[PFCEN] = 0). Range [-32768,3767] This bit is used for PFC_TRGDLY and PFC_OUTARR registers when PFC module is enabled (PFC_CR0[PFCEN] = 1). In this case, IAC_UKMIN is 0 by default.

11.2.22 PFC_TRGDLY/PFC_OUTARR (0x40FE, 0x40FF)

PFC_OUTARRH(0x40FE)								
Bit	15	14	13	12	11	10	9	8
Name	PFC_TRGDLY				PFC_OUTARR[11:8]			
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
PFC_OUTARRL(0x40FF)								
Bit	7	6	5	4	3	2	1	0
Name	PFC_OUTARR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description

[15:12]	PFC_TRGDLY	ADC Sampling Delay of PFC_IAC Channel If the MCU clock is 24MHz (41.67ns) and PFC_TRGDLY = 5, the sampling time is delayed by 41.67*8*5 = 1664ns. Range [0,15] Note This bit is valid only when PFC module is enabled (PFC_CR0[PFCEN] = 1)
[11:0]	PFC_OUTARR	Period of PFC Outer Loop This bit configures the period of PFC outer loop, i.e., UDC PI controller. Period of outer loop = Period of inner loop / PFC_OUTARR = 24M / PFC_ARR / PFC_OUTARR If MCU clock is 24MHz (41.67ns), PFC_ARR = 150 and PFC_OUTARR = 200, the period of inner loop = 24000000 / 2 /PFC_ARR = 80000Hz; the period of outer loop = 80000 /PFC_OUTARR = 400Hz. Range [0,2047] Note This bit is valid only when PFC module is enabled (PFC_CR0[PFCEN] = 1)

11.2.23 PFC_UAC (0x409A, 0x409B)

PFC_UACH(0x409A)								
Bit	15	14	13	12	11	10	9	8
Name	PFC_UAC[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
PFC_UACL(0x409B)								
Bit	7	6	5	4	3	2	1	0
Name	PFC_UAC[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PFC_UAC	Sampling threshold triggered by ADC voltage

11.2.24 PFC_IAC (0x409C, 0x409D)

PFC_IACH(0x409C)								
Bit	15	14	13	12	11	10	9	8
Name	PFC_IAC[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
PFC_IACL(0x409D)								
Bit	7	6	5	4	3	2	1	0
Name	PFC_IAC[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	PFC_IAC	Sampling threshold triggered by ADC current

12 FOC

12.1 FOC Overview

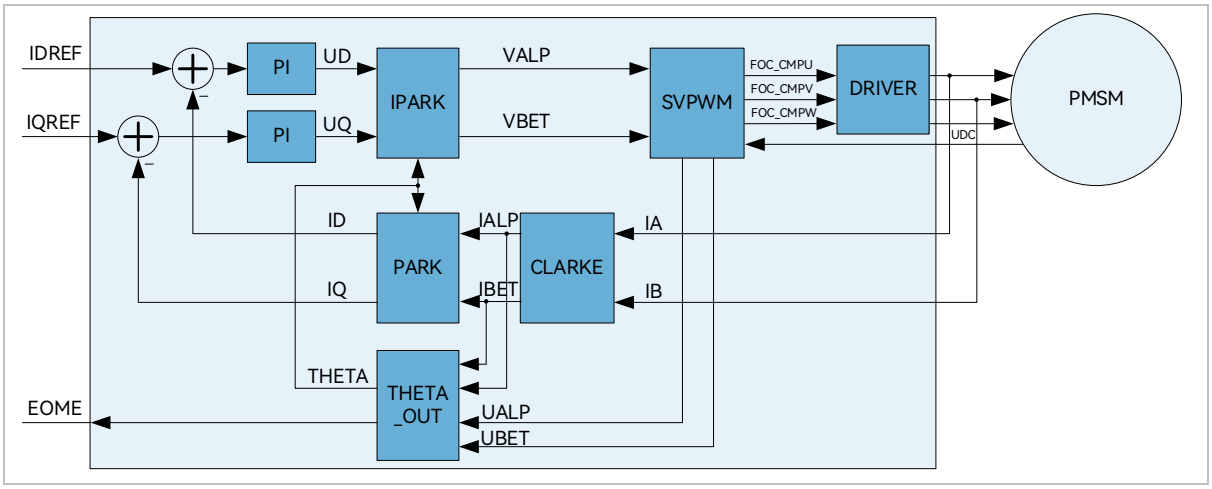
12.1.1 FOC Introduction

FOC module is used in sensorless and sensored FOC motor drive applications and SVPWM-based motor control applications. When DRV_CR[FOCEN] = 0, FOC module is inactivated and FOC clock stops. The relevant FOC registers are forced into the reset state and cannot be written.

FOC module consists of angle estimator, PI controller, coordinate transform module, current sampling module and PWM output module. The angle estimator uses the sampling motor current to estimate the rotor position and implement sensorless FOC-based motor control. MCU can also process the signals from the position sensor to implement sensored FOC-based control.

- Sensorless FOC: Angle for coordinate transformation is obtained by angle estimator, and the motor speed is estimated for speed closed-loop control.
- Sensor-based FOC: FOC module provides the angle input interface. MCU samples position sensor signals and calculates electrical angle of the motor, and sends the result to FOC module for coordinate transformation.

Figure 12-1 FOC Block Diagram



12.1.2 Reference Input

The current loop of FOC module uses the d-axis current reference value FOC_IDREF and the q-axis current reference value FOC_IQREF as the reference, and uses the d-axis current sampling value and the q-axis current sampling value as the feedback. FOC module outputs real-time estimated motor speed FOC_EOME. MCU can use FOC_EOME as the feedback to build speed loop and send the output of speed loop to FOC_IQREF to implement the speed-current dual closed-loop control.

12.1.3 PI Controller

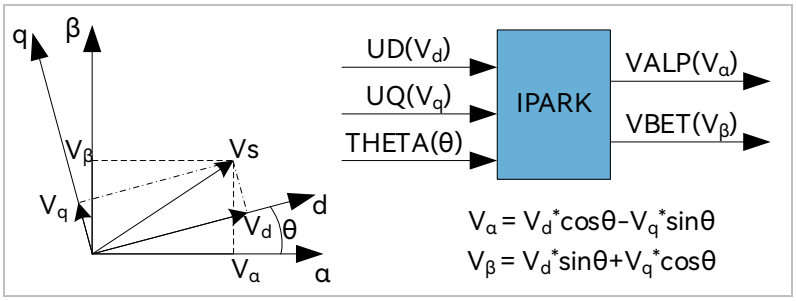
FOC module integrates two PI controllers:

1. Flux control: PI controller of d-axis current, with reference current FOC_IDREF minus feedback current as deviation input, proportional coefficient FOC_DKP and integral coefficient FOC_DKI for the adjustment of PI controller performance, FOC_DMAX and FOC_DMIN for limiting of the output amplitude. The output is voltage reference of d-axis FOC_UD.
2. Torque control: PI controller of q-axis current, with current reference FOC_IQREF minus feedback current as the error input, proportional coefficient FOC_QKP and the integral coefficient FOC_QKI for adjustment of PI performance, and FOC_QMAX and FOC_QMIN for limiting of the output amplitude. The output is voltage reference of q-axis FOC_UQ.

12.1.4 Coordinate Transformation

12.1.4.1 Inverse Park Transformation

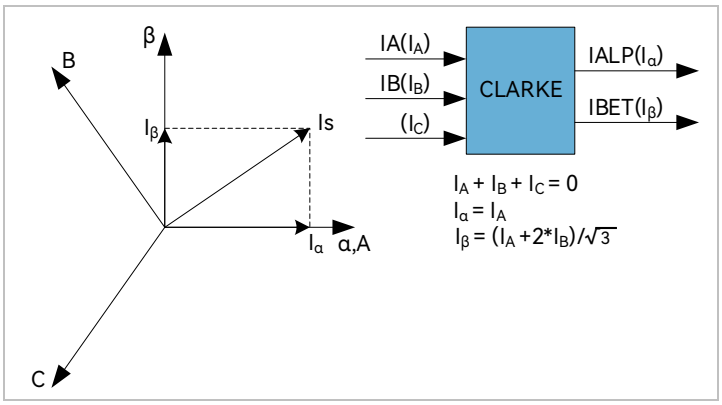
Figure 12-2 Inverse Park Transformation



Inverse Park transformation is used to transform two voltage vectors obtained by PI controller, FOC_UD and FOC_UQ, from d/q-axis coordinate to α/β-axis coordinate.

12.1.4.2 Clarke Transformation

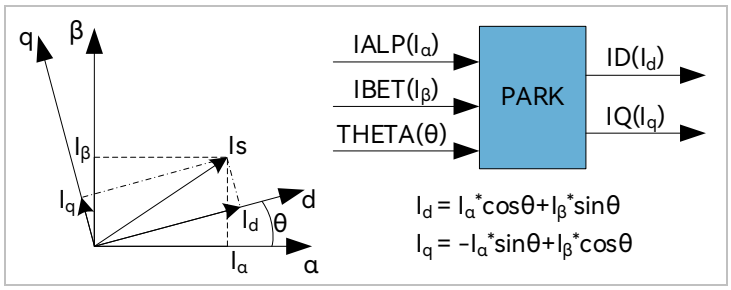
Figure 12-3 Clarke Transformation



Clarke transformation is used to transform the sampled current from 3-phase stationary coordinate to α/β -axis coordinate.

12.1.4.3 Park Transformation

Figure 12-4 Park Transformation



Park transformation is used to transform the current vectors from α/β-axis coordinate to d/q-axis coordinate to get the sampled d/q-axis current.

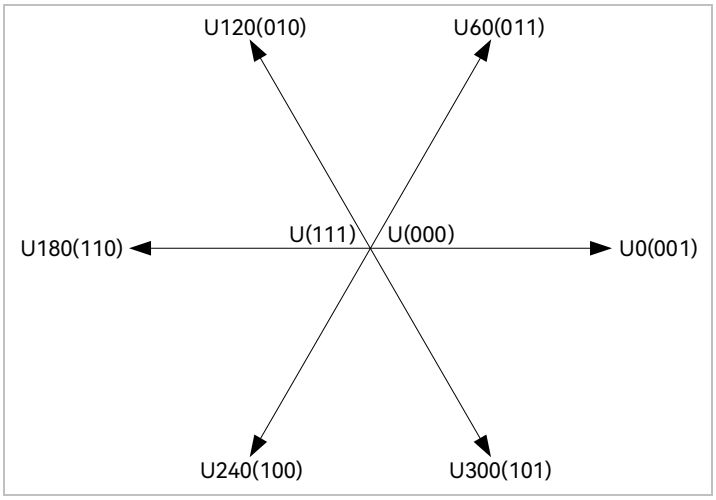
12.1.5 SVPWM

SVPWM algorithm is an important part of FOC. The main idea is to obtain quasi-circular rotating magnetic field by switching the inverter space voltage vectors. This method decreases harmonic components of the phase current, harmonic losses of the motor and torque ripple, and achieves high voltage utilization.

SVPWM generates pulse-width modulation signals for the 3-phase motor voltage control, whose process can be reduced to a few simple equations. Since high side and low side of the inverter cannot be turned on simultaneously, there are two states for a phase, i.e., phase connected to bus voltage (represented by 1) or

phase connected to ground (represented by 0). Therefore, voltage vector output of THE inverter has a total of 2^3 = 8 possible states. $X_C X_B X_A$ represents the voltage vectors, where X_C represents the state of C-phase, X_B represents the state of B-phase and X_A represents the state of A-phase. For example, “100” represents the state that C-phase voltage is connected to bus voltage and A, B-phases are connected to ground. When the states of 3-phase are all 1 or 0, there is no voltage drop between two phases and the state is called inactive state or zero voltage vector. The other 6 states which have voltage output are active voltage vectors with an adjacent state rotation offset of 60 degrees.

Figure 12-5 SVPWM Voltage Vector



SVPWM uses the sum of two adjacent vectors to generate any voltage vector located in the voltage vector space. As shown in Figure 12 6, UOUT is the desired vector and it is in the sector between U60 and U0. Based on the principle of equal impulse, the effect, U0 applied 2*T1 time and U60 applied 2*T2 time, is equivalent to the UOUT. The rest of time (T0) is applied by zero voltage vector.

Figure 12-6 SVPWM Voltage Vector Synthesis

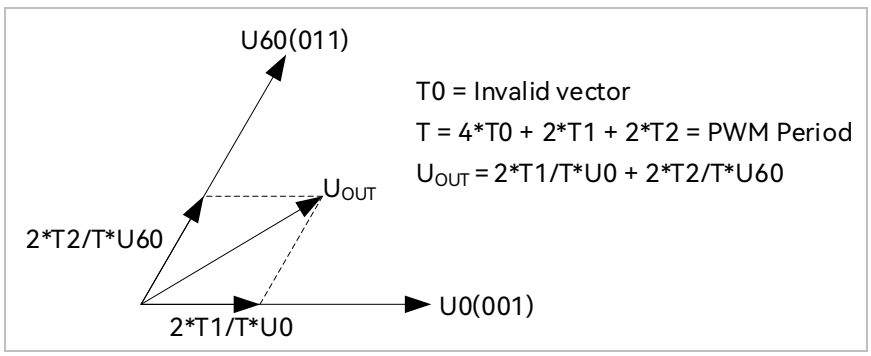


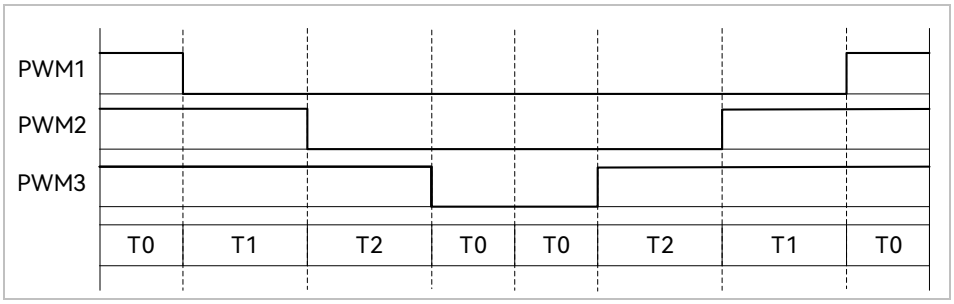
Table 12-1 States of SVPWM Inverter

Phase C	Phase B	Phase A	U_{ALP}	U_{BET}	Vector
0	0	0	0	0	000
0	0	1	$2/3*U_{DC}$	0	001
0	1	1	$1/3*U_{DC}$	$1/3*U_{DC}$	011
0	1	0	$-1/3*U_{DC}$	$1/3*U_{DC}$	010
1	1	0	$-2/3*U_{DC}$	0	110
1	0	0	$-1/3*U_{DC}$	$-1/3*U_{DC}$	100
1	0	1	$1/3*U_{DC}$	$-1/3*U_{DC}$	101
1	1	1	0	0	111

12.1.5.1 Continuous SVPWM

In single-shunt current sampling mode, continuous SVPWM is always used. In dual/triple-shunt current sampling mode, FOC_CR2[F5SEG] is set to “0” to select continuous SVPWM as the output mode.

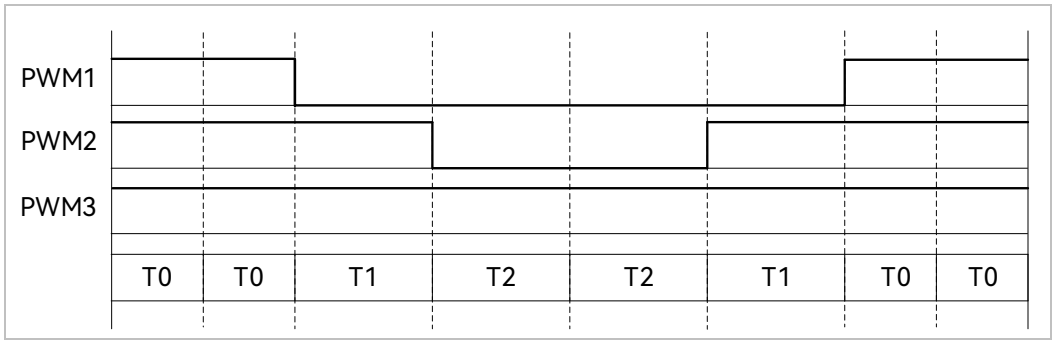
Figure 12-7 Output Level of Continuous SVPWM



12.1.5.2 Discontinuous SVPWM

Discontinuous SVPWM is available in dual/triple-shunt current sampling mode. FOC_CR2[F5SEG] is set to “1” to activate this mode.

Figure 12-8 Output Level of Discontinuous SVPWM



12.1.6 Overmodulation

Overmodulation is available in single/dual/triple-shunt current sampling mode. Configuring FOC_CR1[OVMDL] = 1 enables overmodulation feature. The FOC_UD, FOC_UQ, related limit amplitudes and voltage output are multiplied by 1.15 in this mode.

12.1.7 Deadtime Compensation

Deadtime compensation is available in dual/triple-shunt current sampling mode. The compensation value of deadtime is configured by FOC_TSMIN. This mode improves the quality of phase current at low speed.

12.1.8 Current and Voltage Sampling

In FOC mode, bus voltage and phase current are sampled by hardware automatically. Before FOC module operates, ADC and operational amplifier shall be enabled and the corresponding control registers be configured. No configuration is required for ADC channel and scan mode. Single/dual/triple-shunt current sampling mode is selected by setting FOC_CR1[CSM]. In single-shunt current sampling mode, channel 4 is the default sampling channel of the bus current (itrip). In dual-shunt current sampling mode, channel 0 and channel 1 are the default sampling channels of A-phase current (ia) and B-phase current (ib) respectively. In triple-shunt current sampling mode, channel 0, channel 1 and channel 4 are the default sampling channels of ia, ib and C-phase current (ic) respectively. Channel 2 is used for bus voltage sampling.

12.1.8.1 Single-shunt Current Sampling Mode

FOC_CR1[CSM] is set to “0” to select the single-shunt current sampling mode. In this mode, FOC module samples itrip (channel 4) twice during the timer counting-up operation, and samples bus voltage during the timer counting-down operation and after FOC module completes the calculation.

Since deadtime affects the accuracy of current sampling, FOC module samples within T1’ and T2’, which is the applied time of active voltage vector with deadtime removed. FOC_TRGDLY is the register which advances or delays the current sampling time, and this register shall be configured reasonably to ensure sampling is completed within T1’ and T2’. For example, if FOC_TRGDLY = 5, the sampling time is delayed by 5*T = 208ns; and if FOC_TRGDLY = 0xFB(−5), the sampling time is advanced by 5*T = 208ns.

Figure 12-9 Single-shunt Current Sampling Timing

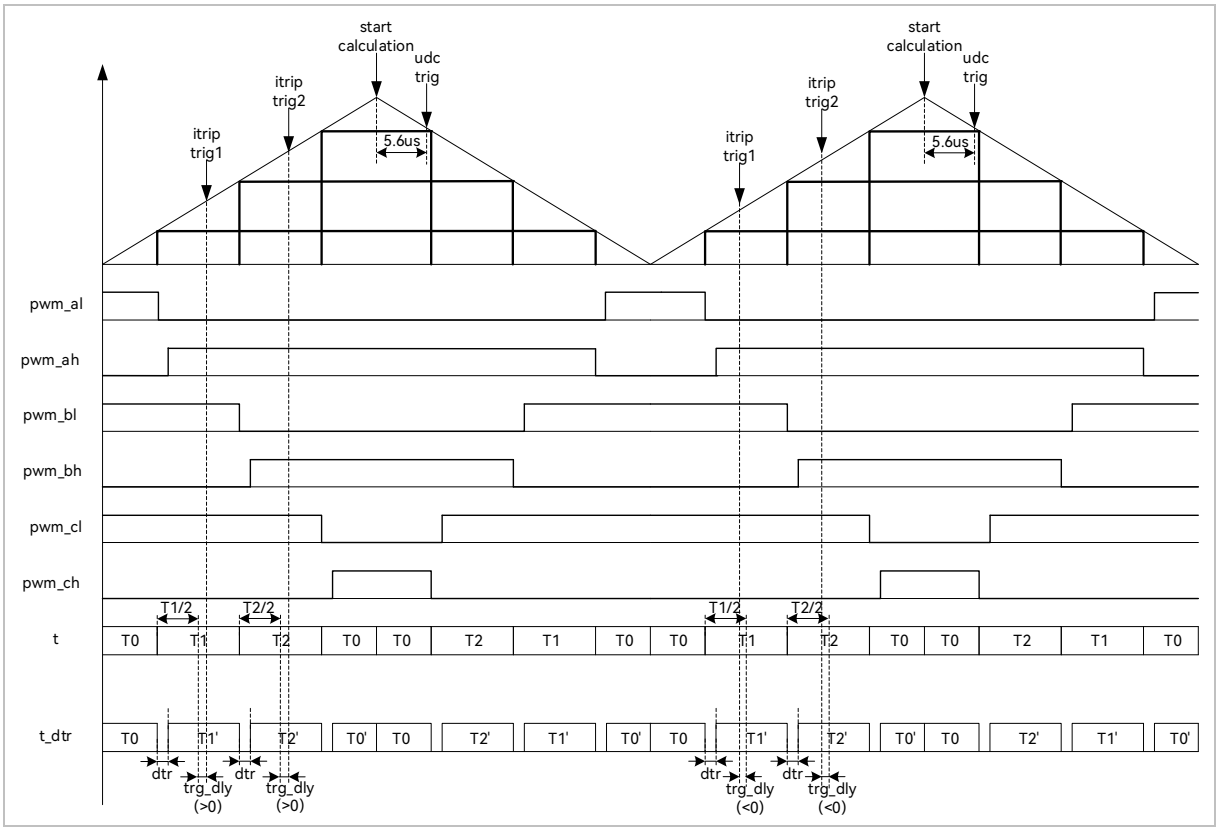
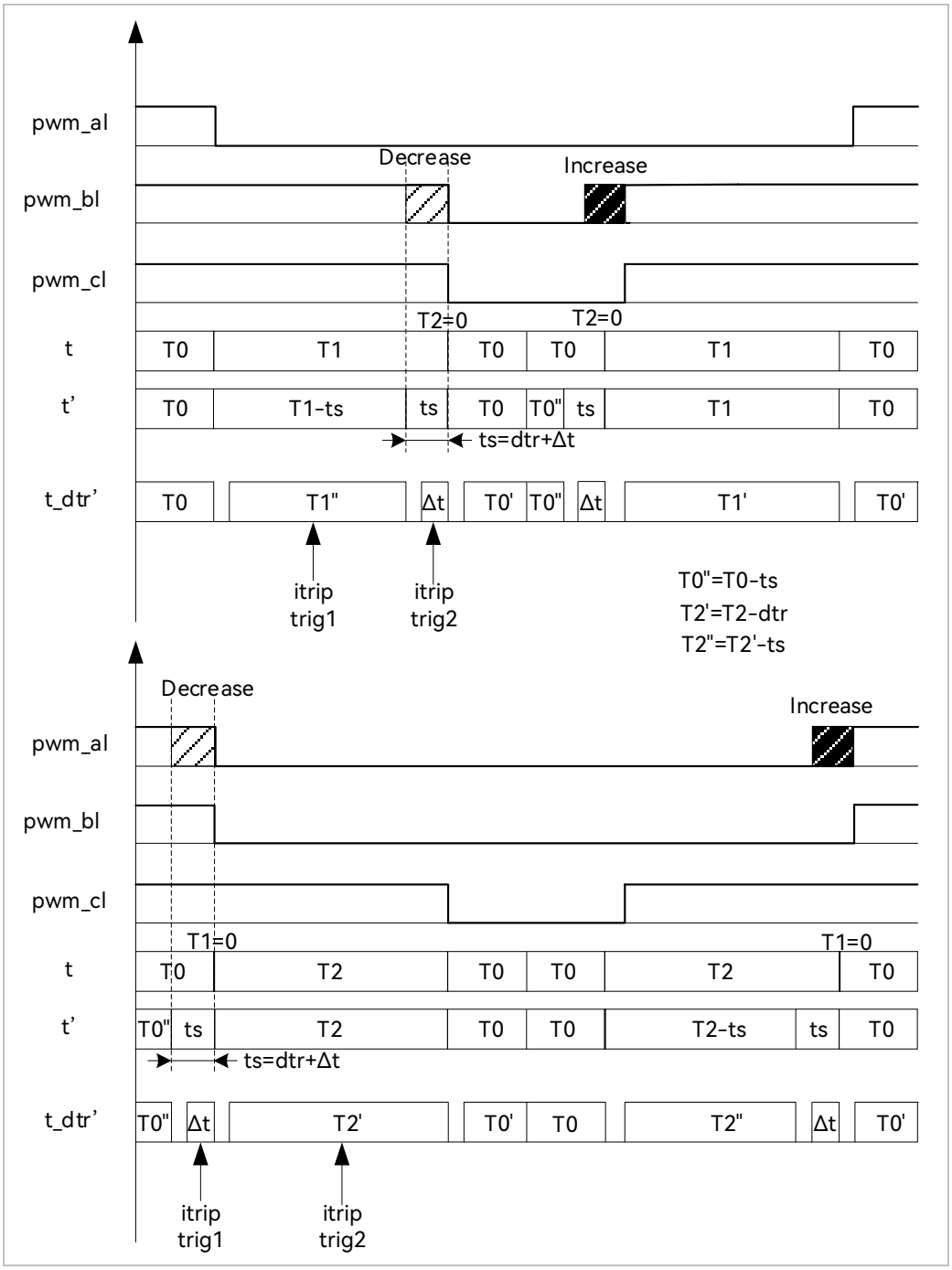


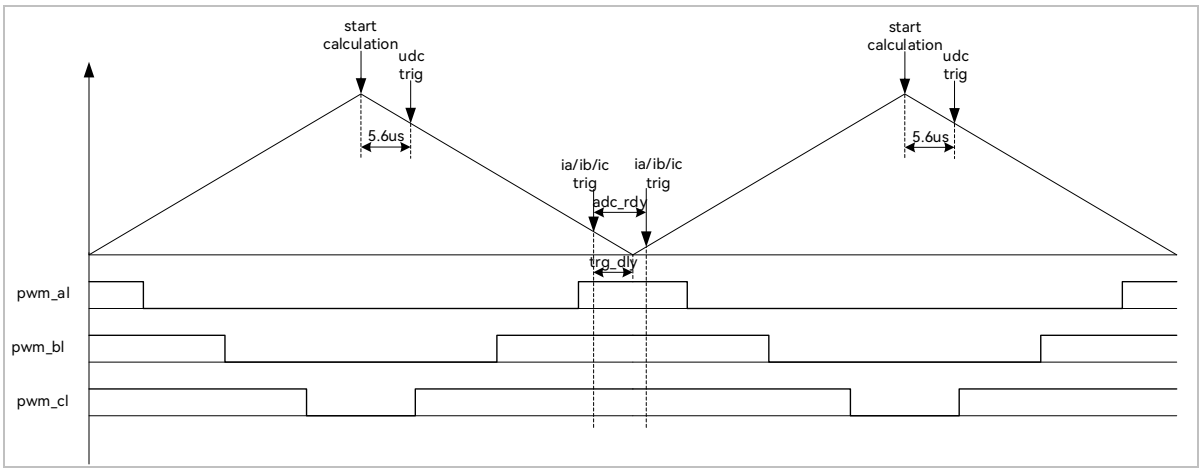
Figure 12-10 Single-shunt Current Sampling Time Compensation



The time of single-shunt current sampling window may be not enough to sample the current in low modulation index and sector switching area. PWM waveform shall be adjusted to ensure the minimum sampling window required in the case. FOC_TSMIN (FOC_TSMIN = minimum sampling window + deadtime) is used to configure the compensation value of deadtime, and FOC module adjusts the PWM waveform automatically.

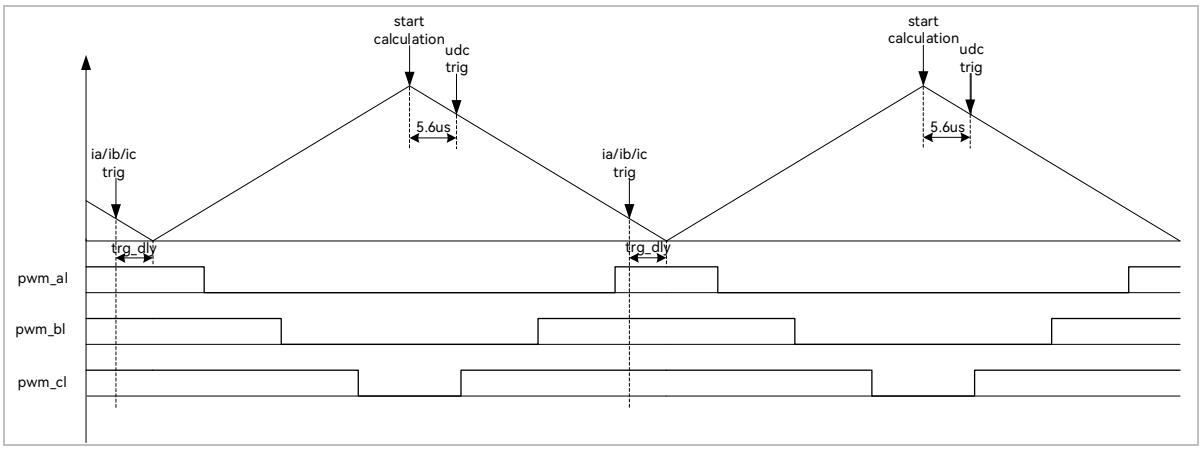
12.1.8.2 Dual/Triple-shunt Current Sampling Mode

Figure 12-11 Dual/Triple-shunt Sequential Current Sampling Mode



FOC_CR1[CSM] is set to “10/11” and FOC_CR2[DSS] to “0” to select dual/triple-shunt sequential current sampling mode. In triple-shunt sequential current sampling mode, FOC_TRGDLY is used to configure the sampling time of a phase current (ia/ib/ic is determined according to the sector), and other phases are sampled at the end of previous sampling. In dual-shunt sequential current sampling mode, FOC_TRGDLY is used to configure the sampling time of ia, and ib is sampled at the end of ia sampling. FOC_TRGDLY shall be configured reasonably to ensure current sampling time is within zero voltage vector (000). For example, when FOC_TRGDLY = 0xB2 and FOC timer counts down, ia/ib/ic is sampled at 50*T = 2.08μs before an underflow event, and then the other phases of ia/ib/ic are sampled.

Figure 12-12 Dual/Triple-shunt Alternate Current Sampling Mode



FOC_CR1[CSM] is set to “10/11” and FOC_CR2[DSS] to “1” to select dual/triple-shunt alternating current sampling mode. In this mode, FOC module performs calculation in every PWM cycle. However, only one phase current is sampled at each PWM cycle (ia/ib/ic is determined according to the sector). The first carrier cycle samples one phase of the ia/ib/ic, and the second carrier cycle samples the current of the other phase, so as to alternately sample the current of two phases in three phases. FOC_TRGDLY is used to configure the sampling time of ia (channel 0), ib (channel 1) and ic (channel 4). FOC_TRGDLY shall be configured reasonably to ensure sampling time for the ia/ib/ic current is within zero voltage vector (000). For example, when FOC_TRGDLY = 0xB2 and FOC timer counts-down, phase current is sampled at 50*T = 2.08μs before an underflow event.

In dual/triple-shunt current sampling mode, bus voltage is sampled when driver timer is down-counting and FOC module completes the calculation.

12.1.8.3 Current Sampling Offset

The current sampling offset voltage shall be added to sample full range of current due to the existence of the positive and negative phase current. When phase current is 0, ADC result is the offset value. ADC result minus this value, 0x4000 by default, is the sampling current. Since ADC reference voltage and hardware are nonideal, there is a deviation between the default value and the real value. Therefore, it is necessary to calibrate the offset. The calibration procedure is as follows. When FOC module does not work and there is no current in three phases, MCU starts to sample the corresponding channel several times, averages all the sampled value, and writes the averaged value to FOC_CSO. Providing ADC sampling range is 0V ~ 5V and the reference is 2.5V, FOC_CSO = 2.5V/5V*32768= 16384 (0x4000).

- When FOC_CR2[CSOC] = 00/11, FOC_CSO is written to modify the offset of itrip and ic.
- When FOC_CR2[CSOC] = 01, FOC_CSO is written to modify the offset of ia.
- When FOC_CR2[CSOC] = 10, FOC_CSO is written to modify the offset of ib.

12.1.9 Angle Mode

Angle module includes angle estimation module, ramping module and estimated angle smooth switching module. The sources of angle are as follows:

- Forced ramping angle

- Forced pulling angle
- Estimated angle of estimator
- Forced angle of estimator

Table 12-2 Sources of Angle

FOC_CR1[RFAE]	FOC_CR1[ANGM]	FOC_CR1[EFAE]	Source
1	X	X	Forced ramping angle
0	0	X	Forced pulling angle
0	1	0	Estimated angle of estimator
0	1	1	- omega > FOC_EFREQMIN: Estimated angle of estimator - omega < FOC_EFREQMIN: Forced angle of estimator

12.1.9.1 Forced Ramping Angle

Forced ramping angle is controlled by angle register FOC_THETA, speed register FOC_RTHESTEP, acceleration register FOC_RTHEACC and ramping counter FOC_RTHECNT. The formula is:

FOC_RTHESTEP(32 bits) = FOC_RTHESTEP(32 bits) + FOC_RTHEACC(16 low-order bits)

THETA_OL(16 bits) = THETA_OL(16 bits) + FOC_RTHESTEP(16 high-order bits)

Where, THETA_OL is an internal variable of the chip. In forced ramping angle mode, THETA_OL is written to FOC_THETA as the used angle. If the software writes a value to FOC_THETA, this value is written to THETA_OL as well.

Forced ramping angle has the highest priority. Configuring FOC_CR1[RFAE] to “1” enables the ramping feature. Ramping module makes a ramping operation in every PWM cycle and the counter is added by 1. When the value of the counter reaches the set value by FOC_RTHECNT, FOC_CR1[RFAE] is cleared by hardware, and then the ramping is completed. Thereafter, according to the value of FOC_CR1[ANGM], the angle comes from estimator (FOC_CR1[ANGM] = 1) or forced pulling angle (FOC_CR1[ANGM] = 0).

12.1.9.2 Forced Pulling Angle

Forced pulling angle is controlled by angle register FOC_THETA and speed register FOC_RTHESTEP.

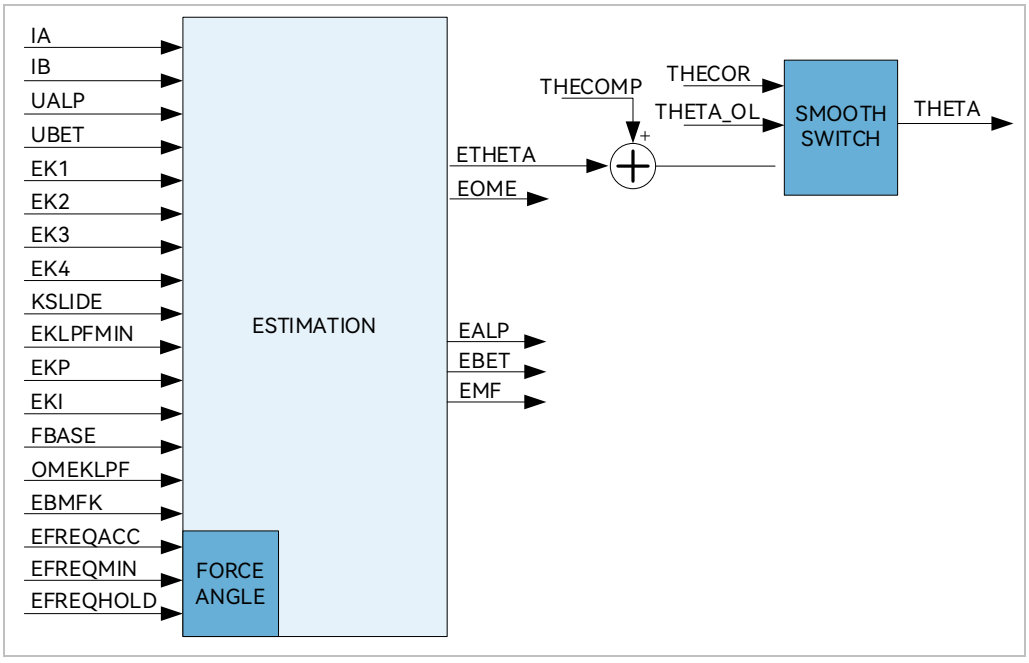
The formula is: THETA_OL(16 bits) = THETA_OL(16 bits) + FOC_RTHESTEP(16 high-order bits)

Where, THETA_OL is an internal variable of the chip. In forced pulling angle mode, THETA_OL is written to FOC_THETA as the used angle. If the software writes a value to FOC_THETA, this value is written to THETA_OL as well.

- When FOC_CR1[RFAE] is set to “1” and FOC_CR1[ANGM] to “0”, MCU switches to forced pulling angle mode after forced ramping angle mode. The speed is the cumulative result after the ramping force angle mode. This mode implements a forced uniform speed control.
- When FOC_CR1[RFAE] is set to “0” and FOC_CR1[ANGM] to “0”, the angle is the forced pulling angle and FOC_RTHESTEP is the initial speed written by software. Configuring FOC_RTHESTEP to “0” enables the pre-position feature. The sensor-based FOC is implemented after the motor speed is set with FOC_RTHESTEP. (Principle of sensor-based FOC: The angle and speed are written to FOC_THETA and FOC_RTHESTEP by software, and FOC module generates an angle in each PWM cycle based on the written values.)

12.1.9.3 Estimator Output Angle

Figure 12-13 Schematic Block Diagram of Estimator

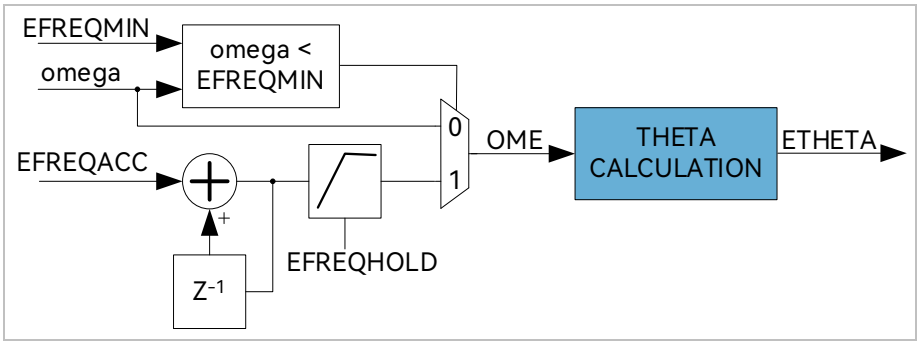


12.1.9.3.1 Estimated Angle of Estimator

The estimator builds the motor model based on the motor parameters and control parameters, and outputs the estimated angle based on the sampled current and voltage.

12.1.9.3.2 Forced Angle of Estimator

Figure 12-14 Schematic Diagram of Forced Angle of Estimator



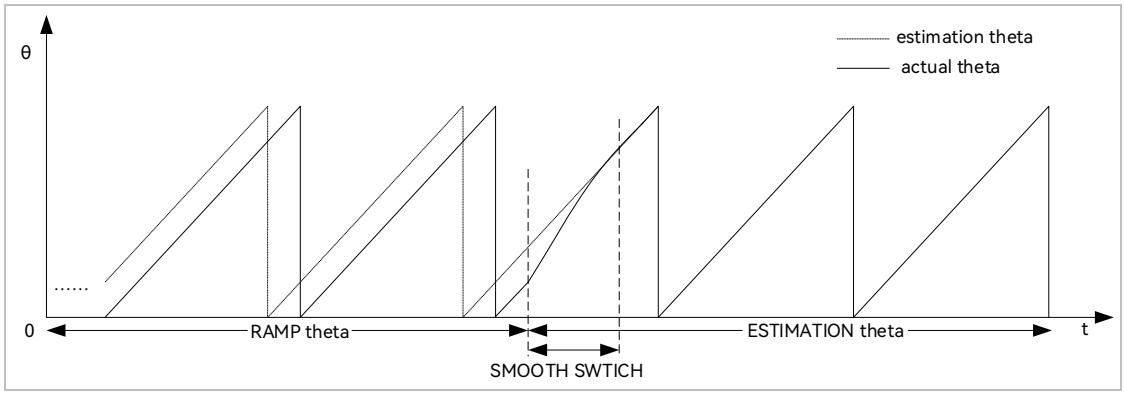
This feature is similar to the ramping feature. Due to the low speed at motor starting process, there may be a deviation in angle and speed estimation with the small effective signal, resulting in startup failure. In this case, the estimator outputs the forced angle to ensure the motor starts normally.

The forced angle feature of the estimator is enabled when FOC_CR1[RFAE] is set to “0”, FOC_CR1[ANGM] to “1” and FOC_CR1[EFAE] to “1”. As shown in Figure 12-14, the estimator compares the value of real-time estimated speed (omega) and FOC_EFREQMIN to determine omega or forced speed (FOC_ETHETA) as the used speed (OME). When omega < FOC_EFREQMIN, the forced speed is selected as OME. The forced speed starts with 0 and increases by FOC_EFREQACC in each PWM cycle, with the maximum value FOC_EFREQHOLD. When omega ≥ FOC_EFREQMIN, omega is selected as OME.

Estimated speed of the estimator FOC_EOME is the low-pass filtering result of OME with the coefficient set by FOC_OMEKLPF.

12.1.9.3.3 Angle Smooth Switching

Figure 12-15 Angle Smooth Switching Curve



When FOC_CR1[RFAE] is set to “1” and FOC_CR1[ANGM] to “1”, the motor starts with ramping feature, and it switches to estimator angle mode after the ramping. However, there is usually a deviation between the estimated angle (FOC_ETHETA) and the forced ramping angle (THETA_OL). If the angle is switched from forced ramping angle to estimated angle directly, motor jitter may occur due to such a sudden change. To deal with this problem, a smooth switching is preferred.

After ramping, if the deviation between FOC_ETHETA and THETA_OL is less than or equal to FOC_THECOR, FOC_ETHETA is selected as the output angle. But if the deviation is larger than FOC_THECOR, THETA_OL is modified smoothly with the step of FOC_THECOR at every PWM cycle until it is close to FOC_ETHETA. After the deviation is less than FOC_THECOR, FOC_ETHETA is selected as the output angle.

12.1.9.3.4 Angle Compensation

Angle compensation value FOC_THECOMP is used to compensate for the estimated angle FOC_ETHETA. If FOC_THECOMP is negative, the lag angle is compensated; if it is positive, the lead angle is compensated.

12.1.10 Motor Real-time Parameters

MCU monitors the state of motor using the following real time variables provided by FOC module:

- Used angle FOC_THETA
- Estimated angle FOC_ETHETA and estimated speed FOC_EOME
- d-axis voltage FOC_UD and q-axis voltage FOC_UQ
- α-axis voltage FOC_VALP and β-axis voltage FOC_VBET
- Bus voltage FOC_UDCFLT
- Phase current FOC_IA, FOC_IB, FOC_IC and maximum phase current FOC_IAMAX, FOC_IBMAX, FOC_ICMAX
- α-axis current (equal to FOC_IA) and β-axis current FOC_IBET
- α-axis BEMF FOC_EALP and β-axis BEMF FOC_EBET
- Magnitude of BEMF FOC_EMF
- Motor power FOC_POW

12.1.10.1 Tailwind/headwind Detection

FOC module provides tailwind/headwind detection feature. FOC module starts to operate when FOC_CR0[ESCMS] is set to “1”, FOC_IDREF to “0” and FOC_IQREF to “0”. Motor’s rotor state is detected by FOC_ETHETA and FOC_EOME. If FOC_ETHETA decreases or FOC_EOME is a negative value, the motor rotates in the headwind state and it is necessary to brake first and then start the motor with ramping forced angle mode. If FOC_ETHETA increases or FOC_EOME is a positive value, the motor rotates in the tailwind state and can be started using estimated angle directly.

12.1.10.2 BEMF Detection

Estimator estimates α-axis BEMF FOC_EALP and β-axis BEMF FOC_EBET with the motor parameters, and calculates the magnitude of FOC_EMF, which implements protection features, such as motor lock protection, phase loss protection, etc.

12.1.10.3 Motor Power

FOC module calculates motor power based on the sampling current, modulation index of SVPWM and filtered bus voltage.

12.2 FOC Registers

12.2.1 FOC_CR0 (0x409F)

Bit	7	6	5	4	3	2	1	0
Name	OMIF	RSV	MERRS		KSLIDE_SEL		ESCMS	EDIS
Type	R	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	-	0	0	0	0	0	0

Bit	Name	Description
[7]	OMIF	omega < FOC_EFREQMIN flag. This bit is valid even if FOC_CR1[EFAE] is 0. 0: omega ≥ FOC_EFREQMIN 1: omega < FOC_EFREQMIN
[6]	RSV	Reserved
[5:4]	MERRS	Maximum Error of SMO Algorithm 00: 0.5 01: 0.25 10: 0.125 11: 1.0
[3:2]	KSLIDE_SEL	KSLIDE Selection 00: 0.85 01: 0.25 10: 0.5 11: 0.75
[1]	ESCMS	Angle Mode Selection 0: Internal Test Mode 1: Recommended Mode
[0]	EDIS	FOC_EALP/FOC_EBET Auto-computation Disable 0: Disable 1: Enable

12.2.2 FOC_CR1 (0x40A0)

Bit	7	6	5	4	3	2	1	0
Name	OVMDL	EFAE	RFAE	ANGM	CSM		RSV	SVPWMEN
Type	R/W	R/W	R/W	R/W	R/W	R/W	-	R/W
Reset	0	0	0	0	0	0	-	0

Bit	Name	Description
[7]	OVMDL	Overmodulation Enable 0: Disable 1: Enable

[6]	EFAE	Forced Angle of Estimator Enable When this feature is enabled, angle mode is determined by the estimator, and it switches to estimated angle mode automatically. 0: Disable 1: Enable
[5]	RFAE	Forced Ramping Angle Enable When this feature is enabled, angle mode is determined by the ramping module. After ramping, it switches to estimated mode or forced pulling mode according to FOC_CR1[ANGM]. FOC_CR1[RFAE] is cleared to “0” by hardware as well. 0: Disable 1: Enable
[4]	ANGM	Angle Mode When FOC_CR1[RFAE] = 0, angle mode is determined by this bit. When FOC_CR1[RFAE] = 1, angle mode is determined by this bit after ramping. 0: Forced Pulling Angle Mode 1: Estimated Angle of Estimator Mode
[3:2]	CSM	Current Sampling Mode 00: Single-shunt Current Sampling 01: Dual-shunt Current Sampling 10: Reserved 11: Triple-shunt Current Sampling
[1]	RSV	Reserved
[0]	SVPWMEN	SVPWM Module Enable 0: Disable 1: Enable

12.2.3 FOC_CR2 (0x40A1)

Bit	7	6	5	4	3	2	1	0
Name	RSV	ICLR	F5SEG	DSS	CSOC		UQD	UDD
Type	-	R/W1	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	0	0	0	0	0	0	0

Bit	Name	Description
[7]	RSV	Reserved
[6]	ICLR	Clear FOC_IAMAX/FOC_IBMAX/FOC_ICMAX to “0” 0: No effect 1: This bit is automatically set to “0” after FOC_IAMAX/FOC_IBMAX/FOC_ICMAX is cleared to “0”
[5]	F5SEG	SVPWM Mode Selection 0: Continuous SVPWM 1: Discontinuous SVPWM (it cannot be selected in single-shunt current sampling mode)

[4]	DSS	Dual/Triple-shunt Current Sampling Mode 0: Sequential Sampling Mode, where two-phase currents are sampled in each carrier period 1: Alternate Sampling Mode. FOC module completes the calculation in every carrier cycle. Two-phase currents are sampled alternately in two adjacent carrier cycles.
[3:2]	CSOC	Current Sampling Offset Calibration This bit is written to select the offset of FOC_CSO. In single-shunt sampling, “00” or “11” is written to calibrate itrip offset. In dual-shunt sampling, “01” is written to calibrate ia offset and “10” to calibrate ib offset. In triple-shunt sampling, “01” is written to calibrate ia offset, “10” to calibrate ib offset and “00” or “11” to calibrate ic offset. 00: itrip and ic 01: ia 10: ib 11: itrip and ic
[1]	UQD	q-axis PI Controller Disable When it is enabled, FOC_UQ value is no longer updated by the PI controller. 0: Disable 1: Enable
[0]	UDD	d-axis PI Controller Disable When it is enabled, FOC_UD value is no longer updated by the PI controller. 0: Disable 1: Enable

12.2.4 FOC_CR3 (0x409E)

Bit	7	6	5	4	3	2	1	0
Name	EFAM	TAMD	MFP_EN	FOC_THECOMP_DIS	FOCFEN	MFP_MD	TSMINH9	TSMINH8
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	EFAM	OMEGA Startup Force Enable When FOC_CR1[EFAE] is set to “0” and FOC_CR3[EFAM] to “1”, FOC_OMEGA register is forced to stay as FOC_EFREQHOLD. 0: Disable 1: Enable
[6]	TAMD	Angle Calculation Method The angle derived from atan (ealpha/ebeta) is used as FOC_THETA. 0: Disable 1: Enable

[5]	MFP_EN	Adaptive Observer Enable 0: Disable 1: Enable
[4]	FOC_THECOMP_DIS	Algorithm without Compensation Angle Enable With this feature enabled, angle compensation of 26.5° is not executed even if the SMO or AO algorithm is selected. 0: Disable 1: Enable
[3]	FOCFEN	FOC Force Enable When DRV_CR[MESEL] is set to "1", FOC module performs calculation even if DRV_CR[OCS] = 0. 0: Disable 1: Enable
[2]	MFP_MD	Adaptive Observer Anti-Windup Enable 0: Disable 1: Enable. Internal calculation precision is reduced during high-speed operation (12 PWM cycles per electrical cycle) and high-current conditions
[1:0]	TSMINH	Scale up by two bits of FOC_TSMIN, forming 10-bit data with 0x40A2 register

12.2.5 FOC_TSMIN (0x40A2)

Bit	7	6	5	4	3	2	1	0
Name	FOC_TSMIN							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	FOC_TSMIN	Single-shunt Current Sampling Mode: minimum window for sampling Dual/triple-shunt Current Sampling Mode: deadtime compensation Range [0,255] TSMIN = sampling window T_{window} + deadtime T_{DT} Example: Assuming that T_{window} = 1μs, T_{DT} = 1μs, TSMIN = 2μs and carrier period = 62.5μs, then FOC_TSMIN = (1 + 1)/62.5*4096 = 131

12.2.6 FOC_TGLI (0x40A3)

Bit	7	6	5	4	3	2	1	0
Name	GLI_H_DIS	GLI_L_DIS	GLI_TIME					
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	GLI_H_DIS	High-side Positive Narrow Pulse Elimination Disable (Low-duty-cycle Pulses) 0: Disable 1: Enable
[6]	GLI_L_DIS	High-side Negative Narrow Pulse Elimination Disable (High-duty-cycle Pulses) 0: Disable 1: Enable
[5:0]	GLI_TIME	GLI_TIME Narrow Pulse Elimination Time, 41.67ns*n*2 This feature is designed for high-voltage applications. The minimum pulse required by high side of the HV level shifter must be longer than a certain time. The level shifter is not turned on if the pulse is less than the value set by this bit. Range [0,63] Example: If GLI_TIME is set to 10, the deadtime to remove the narrow pulse is 41.67ns*10*2

12.2.7 FOC_TBLO (0x40A4)

Bit	7	6	5	4	3	2	1	0
Name	FOC_TBLO							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	FOC_TBLO	Sampling Masking Time in Triple-shunt Current Sampling Mode If the low side of the level shifter is turned on for less than FOC_TBLO, the phase current is not sampled and obtained through special process. Range [0,255] Assuming that the phase current is not sampled if the low side of the driver is turned on for less than 1μs, then FOC_TBLO = 1000/(2*41.67)ns = 12.

12.2.8 FOC_TRGDLY (0x40A5)

Bit	7	6	5	4	3	2	1	0
Name	FOC_TRGDLY							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	FOC_TRGDLY	Time Configuration for Current Sampling When FOC_TRGDLY is set to “0”, FOC module samples the current as follows. Single-shunt Current Sampling Mode: Midpoint between deadtime and applied time of active voltage vector Dual/Triple-shunt Current Sampling Mode: Midpoint of vector 000 (Driver count value = 0). Range [-128,127] Single-shunt Current Sampling Mode: If FOC_TRGDLY = 5, it delays by 5*T = 208ns to sample the current, and if FOC_TRGDLY = 0xFB (complement) or FOC_TRGDLY = -5, it advances by 5*T=208ns. Dual/Triple-shunt Current Sampling Mode: If FOC_TRGDLY = 0x85 (the highest bit, and the remaining 7 bits are absolute values) and Driver timer counts down, it samples the current at 5*T = 208ns before an overflow event occurs. If FOC_TRGDLY = 5 and Driver timer counts up, it samples the current at 5*T = 208ns after an overflow event occurs.

12.2.9 FOC_CSO (0x40A6, 0x40A7)

FOC_CSOH(0x40A6)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_CSO[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	1	0	0	0	0	0	0
FOC_CSOL(0x40A7)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_CSO[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_CSO	Current Sampling Offset FOC_CR2[CSOC] is configured to select the current, and FOC_CSO is written to calibrate current sampling offset of itrip in single-shunt current sampling mode, ia, ib in dual-shunt current sampling mode and

		ia, ib and ic in triple-shunt current sampling mode. Range [0,32767]. The MSB is always 0. Example: Assuming that the ADC voltage falls within 0V ~ 5V with a reference value of 2.5V, then FOC_CSO = 2.5V/5V*32768 = 16384(0x4000)

12.2.10 FOC_RTHESTEP (0x40A8, 0x40A9)

FOC_RTHESTEPH(0x40A8)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_RTHESTEP[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_RTHESTEPL(0x40A9)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_RTHESTEP[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_RTHESTEP	Speed of Ramping Module FOC_RTHESTEP is an internal 32-bit variable. MSB is sign bit. 16 high-order 16 bits are written by software. Range [-32768,32767] FOC_RTHESTEP (32 bits) = FOC_RTHESTEP (32 bits) + FOC_RTHEACC (16 low-order bits) THETA_OL (16 bits) = THETA_OL(16 bits) + FOC_RTHESTEP (16 high-order bits)

12.2.11 FOC_RTHEACC (0x40AA, 0x40AB)

FOC_RTHEACCH(0x40AA)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_RTHEACC[15:8]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0
FOC_RTHEACCL(0x40AB)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_RTHEACC[7:0]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_RTHEACC	Ramping Acceleration FOC_RTHEACC is an internal 32-bit variable. MSB is sign bit. 16 low-order bits are written by software, and 16 high-order bits are always 0. Range [-32768,32767] FOC__RTHESTEP (32 bits) = FOC__RTHESTEP (32 bits) + FOC_RTHEACC (16 low-order bits) THETA_OL (16 bits) = THETA_OL (16 bits) + FOC__RTHESTEP (16 high-order bits)

12.2.12 FOC__EOMELPF (0x40AA, 0x40AB)

FOC__EOMELPFH(0x40AA)								
Bit	15	14	13	12	11	10	9	8
Name	FOC__EOMELPF[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__EOMELPFL(0x40AB)								
Bit	7	6	5	4	3	2	1	0
Name	FOC__EOMELPF[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC__EOMELPF	Filtered Estimated Speed of Estimator The filter coefficient is FOC_EOMEKLPF, and the LPF frequency is the PWM cycle. Range [-32768,32767]

12.2.13 FOC_RTHECNT (0x40AC)

Bit	7	6	5	4	3	2	1	0
Name	FOC_RTHECNT							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	FOC_RTHECNT	Max. ramping counts = FOC_RTHECNT*256 When ramping feature is enabled, the ramping angle increases in each PWM cycle. After FOC_RTHECNT*256 times, ramping feature is disabled.

12.2.14 FOC_THECOR (0x40AD)

Bit	7	6	5	4	3	2	1	0
Name	FOC_THECOR							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R
Reset	0	0	0	0	0	0	0	1

Bit	Name	Description
[7:0]	FOC_THECOR	Angle smooth switching correction: The step value of angle smooth switching after ramping. The format is the same as FOC_THETA. Range [0,255]

12.2.15 FOC_EMF (0x40AE, 0x40AF)

FOC_EMFH(0x40AE)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_EMF[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC_EMFL(0x40AF)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_EMF[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_EMF	Estimated BEMF of Estimator This value is the root of sum of square of FOC_EALP and square of FOC_EBET. Range [0,32767]

12.2.16 FOC_THECOMP (0x40AE, 0x40AF)

FOC_THECOMPH(0x40AE)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_THECOMP[15:8]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0
FOC_THECOMPL(0x40AF)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_THECOMP[7:0]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_THECOMP	Angle Compensation Value The output angle FOC_THETA is derived from estimated angle of the estimator (FOC_ETHETA) + compensation value; the format is same as that of FOC_THETA. Range [-32768,32767]

12.2.17 FOC_DMAX (0x4078)

Bit	7	6	5	4	3	2	1	0
Name	FOC_DMAX							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	FOC_DMAX	Max. output of d-axis PI controller Range [-128,127]

12.2.18 FOC_DMIN (0x4079)

Bit	7	6	5	4	3	2	1	0
Name	FOC_DMIN							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	FOC_DMIN	Min. output of d-axis PI controller Range [-128,127]

12.2.19 FOC_OMEEST (0x40B0, 0x40B1)

FOC_OMEESTH(0x40B0)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_OMEEST[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC_OMEESTL(0x40B1)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_OMEEST[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_OMEEST	FOC Calculated Speed of Estimator Range (0,32767)

12.2.20 FOC_ATAN_THETA (0x40B2, 0x40B3)

FOC_ATAN_THETAH (0x40B2)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_ATAN_THETA[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC_ATAN_THETAL(0x40B3)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_ATAN_THETA[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_ATAN_THETA	Angle in ATAN mode, directly calculated by FOC_EALP/FOC_EBET Range [-32768,32767]

12.2.21 FOC_QMAX (0x408A)

Bit	7	6	5	4	3	2	1	0
Name	FOC_QMAX							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	FOC_QMAX	Max. output of q-axis PI controller Range [-128,127]

12.2.22 FOC_QMIN (0x408B)

Bit	7	6	5	4	3	2	1	0
Name	FOC_QMIN							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	FOC_QMIN	Min. output of q-axis PI controller Range [-128,127]

12.2.23 FOC_UD (0x40B8, 0x40B9)

FOC_UDH(0x40B8)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_UD[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_UDL(0x40B9)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_UD[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_UD	d-axis voltage calculated by d-axis PI controller Range [-32768,32767]

12.2.24 FOC_UQ (0x40BA, 0x40BB)

FOC_UQH(0x40BA)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_UQ[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_UQL(0x40BB)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_UQ[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_UQ	q-axis voltage calculated by q-axis PI controller Range [-32768,32767]

12.2.25 FOC_ID (0x40BC, 0x40BD)

FOC_IDH(0x40BC)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_ID[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC_IDL(0x40BD)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_ID[7:0]							

Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_ID	d-axis current from coordinate transformation Range [-32768,32767]

12.2.26 FOC_IQ (0x40BE, 0x40BF)

FOC_IQH(0x40BE)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_IQ[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC_IQL(0x40BF)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_IQ[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_IQ	q-axis current from coordinate transformation Range [-32768,32767]

12.2.27 FOC_IBET (0x40C0, 0x40C1)

FOC_IBETH(0x40C0)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_IBET[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC_IBETL(0x40C1)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_IBET[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_IBET	β-axis current from coordinate transformation Range [-32768,32767]

12.2.28 FOC_IQ_LPFK (0x40C0)

Bit	7	6	5	4	3	2	1	0
Name	FOC_IQ_LPFK							
Type	W	W	W	W	W	W	W	W
Reset	1	1	1	1	1	1	1	1

Bit	Name	Description
[7:0]	FOC_IQ_LPFK	LPF coefficient of FOC_IQ, set to 0xFF by default Range [0,255]

12.2.29 FOC_ID_LPFK (0x40C1)

Bit	7	6	5	4	3	2	1	0
Name	FOC_IQ_LPFK							
Type	W	W	W	W	W	W	W	W
Reset	1	1	1	1	1	1	1	1

Bit	Name	Description
[7:0]	FOC_ID_LPFK	LPF coefficient of FOC_ID, set to 0xFF by default Range [0, 255]

12.2.30 FOC__VBET (0x40C2, 0x40C3)

FOC__VBETH(0x40C2)								
Bit	15	14	13	12	11	10	9	8
Name	FOC__VBET[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__VBETL(0x40C3)								
Bit	7	6	5	4	3	2	1	0
Name	FOC__VBET[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC__VBET	β-axis Output Voltage of FOC Module Range [-32768,32767]

12.2.31 FOC_UDCPS (0x40C2, 0x40C3)

FOC_UDCPSH(0x40C2)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_UDCPS[15:8]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0
FOC_UDCPSL(0x40C3)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_UDCPS[7:0]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_UDCPS	d-axis Voltage Compensation Value The result of d-axis PI controller (FOC_UD) added to FOC_UDCPS is transferred to the next module. Range [-32768,32767]

12.2.32 FOC_UQCPS (0x40C4, 0x40C5)

FOC_UQCPSH(0x40C4)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_UQCPS[15:8]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0
FOC_UQCPSL(0x40C5)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_UQCPS[7:0]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_UQCPS	q-axis Voltage Compensation Value The result of q-axis PI controller (FOC_UQ) added to FOC_UQCPS is transferred to the next module. Range [-32768,32767]

12.2.33 FOC_VALP (0x40C4, 0x40C5)

FOC_VALPH(0x40C4)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_VALP[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC_VALPL(0x40C5)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_VALP[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_VALP	α-axis Output Voltage of FOC Module Range [-32768,32767]

12.2.34 FOC_IC (0x40C6, 0x40C7)

FOC_ICH(0x40C6)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_IC[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC_ICL(0x40C7)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_IC[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_IC	Sampled Phase-C Current Range [-32768,32767]

12.2.35 FOC_LQ (0x40C8, 0x40C9)

FOC_LQH(0x40C8)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_LQ[15:8]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0
FOC_LQ(0x40C9)								
Bit	7	6	5	4	3	2	1	0

Name	FOC_LQ[7:0]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_LQ	Q-axis Inductance Range [0,32767]

12.2.36 FOC__IB (0x40C8, 0x40C9)

FOC__IBH(0x40C8)								
Bit	15	14	13	12	11	10	9	8
Name	FOC__IB[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC__IBL(0x40C9)								
Bit	7	6	5	4	3	2	1	0
Name	FOC__IB[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC__IB	Sampled Phase-B Current Range [-32768,32767]

12.2.37 FOC_LD (0x40CA, 0x40CB)

FOC_LDH(0x40CA)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_LD[15:8]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0
FOC_LD(0x40CB)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_LD[7:0]							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_LD	D-axis Inductance Range [0,32767]

12.2.38 FOC_IA (0x40CA, 0x40CB)

FOC_IAH(0x40CA)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_IA[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC_IAL(0x40CB)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_IA[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_IA	Sampled Phase-A Current Range [-32768,32767]

12.2.39 FOC_THETA (0x40CC, 0x40CD)

FOC_THETAH(0x40CC)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_THETA[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_THETAL(0x40CD)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_THETA[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_THETA	Output Angle of FOC Module Range [-32768,32767] The bit value -32768 ~ 32767 corresponds to angle range -180°~ 180°. Example: Assuming that FOC_THETA = 8192, the output angle is 8192/32768*180°= 45°.

12.2.40 FOC_ETHETA (0x40CE, 0x40CF)

FOC_ETHETAH(0x40CE)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_ETHETA[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_ETHETAL(0x40CF)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_ETHETA[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_ETHETA	Read: Output Angle of Estimator (angle before FOC_THECOMP is applied); the format is same as that of FOC_THETA. Write: Initial Angle of Estimator Range [-32768,32767]

12.2.41 FOC_EALP (0x40D0, 0x40D1)

FOC_EALPH(0x40D0)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_EALP[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC_EALPL(0x40D1)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_EALP[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_EALP	α-axis estimated BEMF Range [-32768,32767]

12.2.42 FOC_EBET (0x40D2, 0x40D3)

FOC_EBETH(0x40D2)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_EBET[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC_EBETL(0x40D3)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_EBET[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_EBET	β-axis estimated BEMF Range [-32768,32767]

12.2.43 FOC_EOME (0x40D4, 0x40D5)

FOC_EOMEH(0x40D4)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_EOME[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_EOMEL(0x40D5)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_EOME[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_EOME	Output Speed of Estimator Range [-32768,32767]

12.2.44 FOC_POW (0x40D8, 0x40D9)

FOC_POWH(0x40D8)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_POW[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC_POWL(0x40D9)								
Bit	7	6	5	4	3	2	1	0

Name	FOC_POW[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_POW	Motor Power Range [-32768,32767]

12.2.45 FOC_EOMEKLPF (0x40D8)

Bit	7	6	5	4	3	2	1	0
Name	FOC_EOMEKLPF							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	FOC_EOMEKLPF	LPF coefficient of estimated speed FOC_EOMELPF of the estimator LPF is calculated in every PWM cycle Range [1,255] mapping[1/32768,255/32768]

12.2.46 FOC_IAMAX (0x40DA, 0x40DB)

FOC_IAMAXH(0x40DA)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_IAMAX[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC_IAMAXL(0x40DB)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_IAMAX[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_IAMAX	Recorded maximum value of phase-A current; This value may be unreliable unless the motor rotates in a full electrical period. This value will not be cleared to “0” automatically unless FOC_CR2[ICLR] is set to “1” Range [-32768,32767]

12.2.47 FOC_IBMAX (0x40DC, 0x40DD)

FOC_IBMAXH(0x40DC)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_IBMAX[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC_IBMAXL(0x40DD)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_IBMAX[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_IBMAX	Max. Phase-B Current Recorded maximum value of phase-B current. This value may be unreliable unless the motor rotates in a full electrical period. This value will not be cleared to “0” automatically unless FOC_CR2[ICLR] is set to “1”. Range [-32768,32767]

12.2.48 FOC_ICMAX (0x40DE, 0x40DF)

FOC_ICMAXH(0x40DE)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_ICMAX[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC_ICMAXL(0x40DF)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_ICMAX[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_ICMAX	Max. Phase-C Current Recorded maximum value of phase-C current. This value may be unreliable unless the motor rotates in a full electrical period. This value will not be cleared to “0” automatically unless FOC_CR2[ICLR] is set to “1”. Range [-32768,32767]

12.2.49 FOC_DKP (0x4070, 0x4071)

FOC_DKPH(0x4070)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_DKP[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_DKPL(0x4071)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_DKP[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_DKP	KP of D-axis PI Controller Range [0,32767]. MSB is always 0. Q12 format

12.2.50 FOC_EKP (0x4074, 0x4075)

FOC_EKPH(0x4074)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_EKP[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_EKPL(0x4075)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_EKP[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_EKP	KP of PI controller used for estimated angle of the estimator. MSB is always 0. Q12 format. Range [0,32767]

12.2.51 FOC_EKI (0x4076, 0x4077)

FOC_EKIH(0x4076)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_EKI[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_EKIL(0x4077)								

Bit	7	6	5	4	3	2	1	0
Name	FOC_EKI[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_EKI	KI of PI controller used for estimated angle of the estimator; MSB is always 0; Q15 format Range [0,32767]

12.2.52 FOC_EKLPFMIN (0x407A, 0x407B)

FOC_EKLPFMINH(0x407A)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_EKLPFMIN[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_EKLPFMINH(0x407B)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_EKLPFMIN[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_EKLPFMIN	Minimum value of SMO BEMF LPF Coefficient. The estimator-calculated LPF coefficient is forced to be this value when it is lower than this value. Q15 format

12.2.53 FOC_DKI (0x407C, 0x407D)

FOC_DKIH(0x407C)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_DKI[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_DKIL(0x407D)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_DKI[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_DKI	KI of D-axis PI Controller Range [0,32767]. MSB is always 0. Q15 format

12.2.54 FOC_OMEKLPF (0x407E, 0x407F)

FOC_OMEKLPFH(0x407E)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_OMEKLPF[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_OMEKLPFL(0x407F)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_OMEKLPF[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_OMEKLPF	LPF coefficient of estimated speed of the estimator. MSB is always 0. Q15 format Range [0,32767]

12.2.55 FOC_FBASE (0x4080, 0x4081)

FOC_FBASEH(0x4080)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_FBASE[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_FBASEL(0x4081)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_FBASE[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_FBASE	Frequency Base of Estimator Range [0,32767] FOC_FBASE = fbase*Ts*32768 Example: Assuming that fbase = 200Hz, Ts = 62.5μs, then FOC_FBASE = 200*0.0000625*32768 = 409(0x199)

12.2.56 FOC_EFREQACC (0x4082, 0x4083)

FOC_EFREQACCH(0x4082)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_EFREQACC[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_EFREQACCL(0x4083)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_EFREQACC[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_EFREQACC	Speed Increment of Forced Angle Mode. FOC_EFREQACC is an internal 24-bit variable and MSB is sign bit. 16 low-order bits are written by software. Range [0,65535] Example: Assuming that fbase = 200Hz and pp (Pole_Pairs) = 4, then speed_base = 60*fbase/pp = 3000rpm. If speed increment = 3rpm, then FOC_EFREQACC = 3rpm/speed_base*32768*256 = 8388(0x20C4).

12.2.57 FOC_EFREQMIN (0x4084, 0x4085)

FOC_EFREQMINH(0x4084)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_EFREQMIN[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_EFREQMINL(0x4085)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_EFREQMIN[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_EFREQMIN	Switch Threshold of the Estimated Angle FOC_EFREQMIN is an internal 24-bit variable, and MSB is sign bit. 16 high-order bits are written by software. With Forced Angle of Estimator Mode enabled, FOC module outputs forced angle when the estimated angle is smaller than this bit value. Range [-32768,32767] Example: Assuming that fbase = 200Hz and pp (Pole_Pairs) = 4, then

		speed_base = 60*fbase/pp = 3000rpm. Assuming that the min. switching speed = 30rpm, then FOC_EFREQMIN = 30rpm/speed_base*32768 = 327(0x147).

12.2.58 FOC_EFREQHOLD (0x4086, 0x4087)

FOC_EFREQHOLDH(0x4086)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_EFREQHOLD[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_EFREQHOLDL(0x4087)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_EFREQHOLD[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_EFREQHOLD	Maximum Value of Forced Speed of the Estimator FOC_EFREQHOLD is an internal 24-bit variable, and MSB is sign bit. 16 high-order bits are written by the software. If the estimator outputs a speed below FOC_EFREQMIN, the speed is forced to increase to this value and is not increased further. Range [-32768,32767] Example: Assuming that fbase = 200Hz and pp (Pole_Pairs) = 4, then speed_base = 60*fbase/pp = 3000rpm. If max. forced speed = 60rpm, then FOC_EFREQHOLD = 60rpm/speed_base*32768 = 655(0x028F).

12.2.59 FOC_EK3 (0x4088, 0x4089)

FOC_EK3H(0x4088)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_EK3[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_EK3L(0x4089)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_EK3[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_EK3	The 3rd coefficient of the current model in estimator, and MSB is always 0. Q15 format; Range [0,32767]

12.2.60 FOC_EK1 (0x408C, 0x408D)

FOC_EK1H(0x408C)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_EK1[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_EK1L(0x408D)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_EK1[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_EK1	The 1st coefficient of the current model in estimator, and MSB is always 0. Q15 format; Range [0,32767]

12.2.61 FOC_EK2 (0x408E, 0x408F)

FOC_EK2H(0x408E)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_EK2[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_EK2L(0x408F)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_EK2[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_EK2	The 2nd coefficient of the current model in estimator, and MSB is always 0. Q15 format; Range [0,32767]

12.2.62 FOC_IDREF (0x4090, 0x4091)

FOC_IDREFH(0x4090)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_IDREF[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_IDREFL(0x4091)								
Bit	7	6	5	4	3	2	1	0

Name	FOC_IDREF[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_IDREF	User-defined d-axis Current Range [-32768,32767]

12.2.63 FOC_IQREF (0x4092, 0x4093)

FOC_IQREFH(0x4092)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_IQREF[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_IQREFL(0x4093)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_IQREF[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_IQREF	User-defined q-axis Current Range [-32768,32767]

12.2.64 FOC_QKP (0x4094, 0x4095)

FOC_QKPH(0x4094)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_QKP[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_QKPL(0x4095)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_QKP[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_QKP	KP coefficient of q-axis PI controller. MSB is always 0. Q12 format Range [0,32767] corresponds to range of Q12 [0,8]

12.2.65 FOC_QKI (0x4096, 0x4097)

FOC_QKIH(0x4096)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_QKI[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
FOC_QKIL(0x4097)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_QKI[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_QKI	KI coefficient of q-axis PI controller. MSB is always 0. Q15 format. Range [0,32767] corresponds to range of Q15 [0,1]

12.2.66 FOC_UDCFLT (0x4098, 0x4099)

FOC_UDCFLTH(0x4098)								
Bit	15	14	13	12	11	10	9	8
Name	FOC_UDCFLT[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
FOC_UDCFLTL(0x4099)								
Bit	7	6	5	4	3	2	1	0
Name	FOC_UDCFLT[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	FOC_UDCFLT	Filtered Bus Voltage FOC module samples the bus voltage and filters it to obtain FOC_UDCFLT. ADC channel 2 (external voltage divider) can be selected. Range [0,32767] Example: The bus voltage is scaled down by 1/6 before feeding into the ADC module, ADC VREF = 5V (namely, the sampling range is [0V~30V]) and FOC_UDCFLT = 19661(0x4CCD), then bus voltage = 19661/32768*5V*6 = 18V.

13 Timer1

13.1 Timer1 Operations

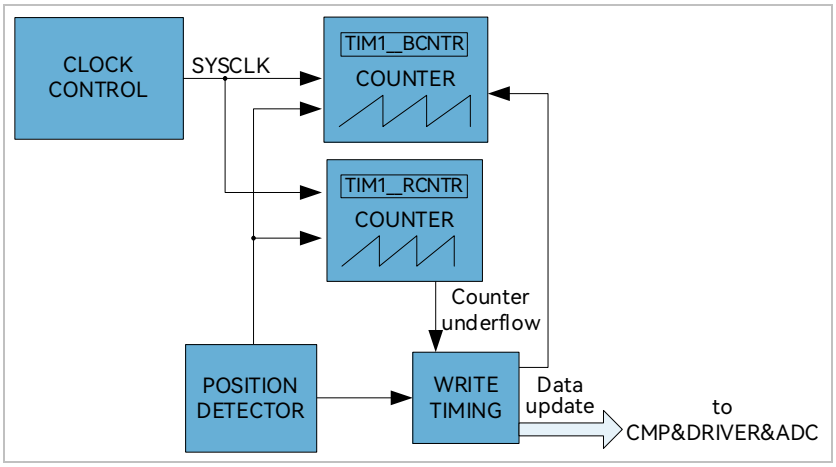
Timer1 consists of a 16-bit up-counting Base Timer and a 16-bit up-counting Reload Timer. Timer1 can be used in the applications of square-wave controlled BLDC motor drive.

Timer1 features as follows.

- 16-bit up-counting Base Timer is used to record the time between two position detection events or two phase commutation events (60 degree time) and also can be used for forced phase commutation control when position detection fails
- 16-bit up-counting Reload Timer is used to control the time from position detection to phase commutation, as well as masking time for diode freewheeling after phase commutation (prohibit position detection time)
- 3-bit programmable frequency prescaler divides the system clock. The divided clock is used as the clock source of the two timers
- Configurable filtering signals and sampling delay for position detection
- Position detection module generates the position signal required for phase commutation according to the input signal
- 7 groups state register control comparator and drive output
- 6 interrupt sources

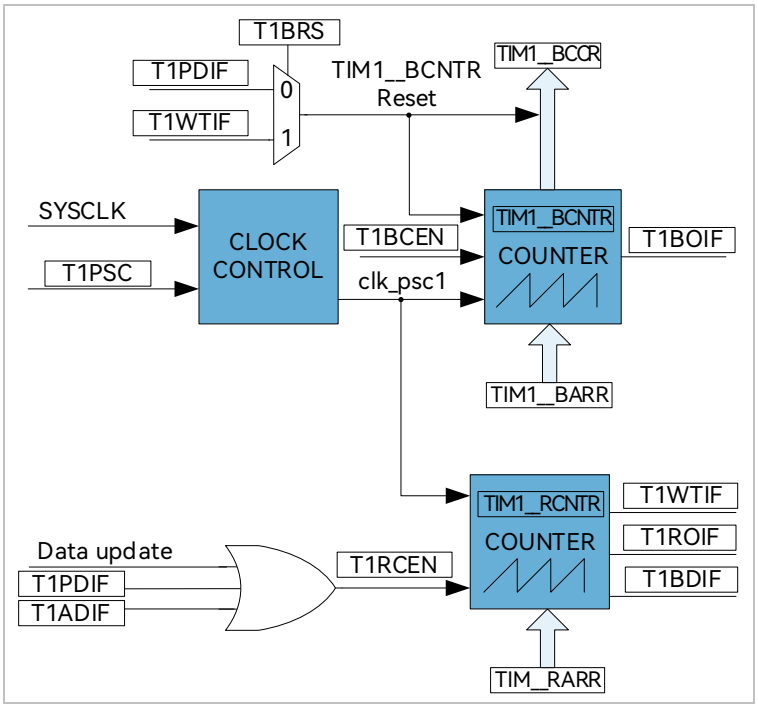
The internal structure of Timer1 is shown in Figure 13-1.

Figure 13-1 Timer1 Internal Structure



13.1.1 Timer1 Counter Module

Figure 13-2 Timebase Unit



Timer1 consists of a frequency prescaler, a 16-bit up-counting Base Timer and a 16-bit up-counting Reload Timer.

13.1.1.1 Prescaler

Prescaler divides the system clock frequency and generates the counter clock source for Base Timer and Reload Timer. It offers 8 division coefficients and can be selected through TIM1_CR3[T1PSC]. Since this register has no buffer, the clock rate is immediately updated after the division coefficient is written. Therefore, the division coefficient shall be configured when both the Basic Timer and Reload Timer are not working. The clock rate clk_psc1 = SYSCLK/(2^TIM1_CR3[T1PSC]). The clock rate corresponding to TIM1_CR3[T1PSC] is shown in Table 13-1.

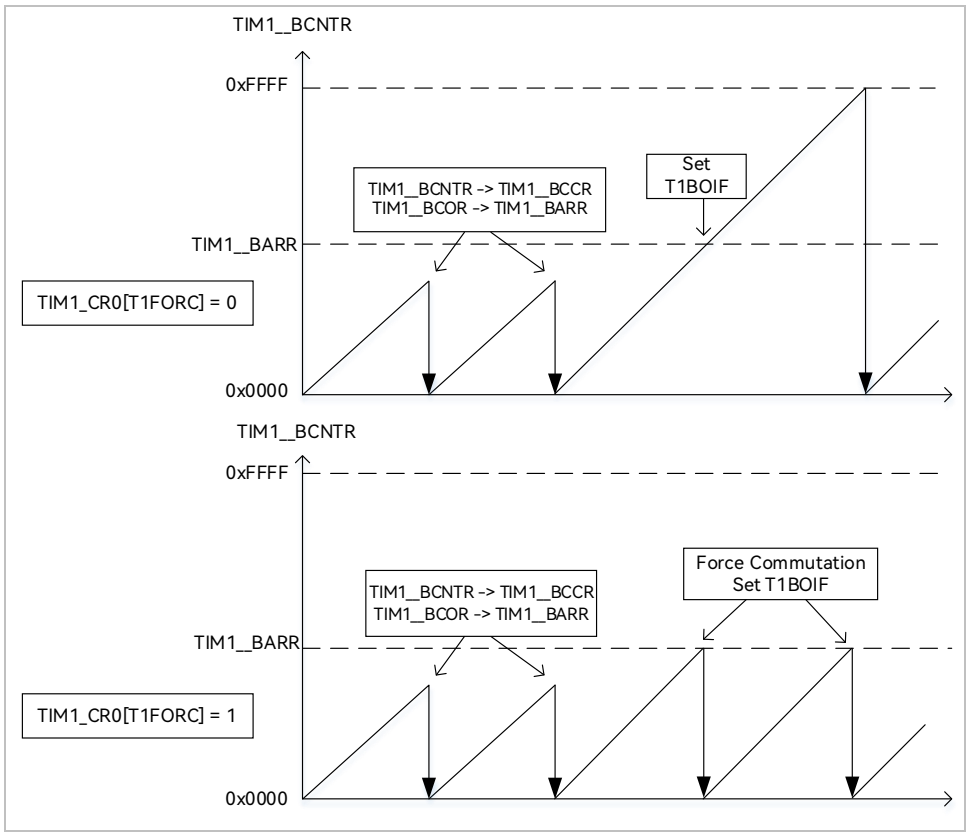
Table 13-1 Mapping between Clock Rate and TIM1_CR3[T1PSC]

TIM1_CR3[T1PSC]	Division Factor	clk_psc1(Hz)	TIM1_CR3[T1PSC]	Division Factor	clk_psc1(Hz)
000	1	24M	100	16	1.5M
001	2	12M	101	32	750k
010	4	6M	110	64	375k
011	8	3M	111	128	187.5k

13.1.1.2 Basic Timer

The Basic Timer is a 16-bit up-counting timer with its count value held in the TIM1_BCNTR register. TIM1_BCNTR value is loaded into Capture Register TIM1_BCCR upon a Position Detection Interrupt TIM1_SR[T1PDIF] or a Write Timing Interrupt TIM1_SR[T1WTIF] (selected by TIM1_CR2[T1BRS]). Meanwhile, TIM1_BCNTR is cleared to "0" and restarts the counter cycle. TIM1_BCCR captures the time between two Position Detection Interrupts or two Write Timing Interrupts (i.e. 60° commutation time). These time inputs are averaged multiple times (programmed by TIM1_CR0[T1CFLT]) before loading the average as a 60° commutation base into the TIM1_BCOR register. When Auto-load Register TIM1_BARR is enabled (TIM1_CR1[BAPE] is set to "1"), TIM1_BARR loads the value of TIM1_BCOR by hardware. When count value of TIM1_BCNTR increases to TIM1_BARR, overflow interrupt flag TIM1_SR[T1BOIF] of the Basic Timer is set to "1". If forced commutation feature is enabled, phase commutation occurs and the Basic Timer is cleared to "0". Otherwise, the Basic Timer Register will not be cleared until it counts up to 0xFFFF and becomes overflowed.

Figure 13-3 Waveform of Basic Timer

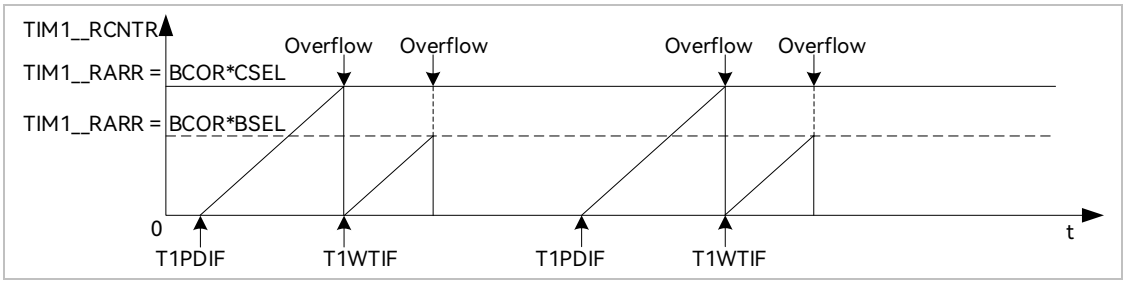


In Manual mode (TIM1_IER[T1MAME] = 1), TIM1__BCNTR is cleared by Basic Timer Overflow event instead of TIM1_CR2[T1BRS].

13.1.1.3 Reload Timer

The Reload Timer is a 16-bit up-counting timer with its count value held in TIM1__RCNTR. The timer overflows when TIM1__RCNTR increases to TIM1__RARR. It stops counting when TIM1_SR[T1ROIF] (overflow interrupt flag of the reload counter) is set to “1”, and TIM1__RCNTR and TIM1_CR0[T1RCEN] are cleared to “0”. TIM1_CR0[T1RCEN] is set to “1” to restart Reload Timer when position detection interrupt or write timing interrupt is generated.

Figure 13-4 Waveform of Reload Timer

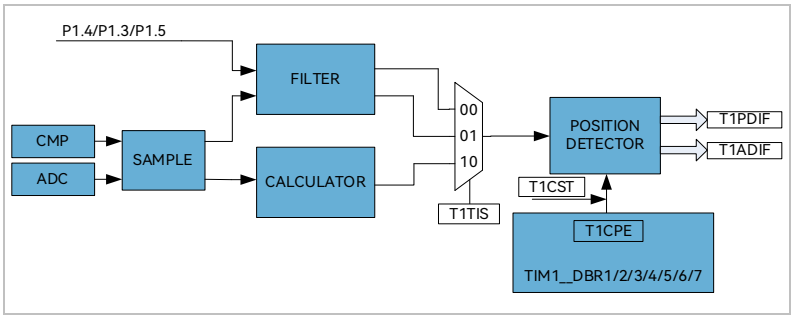


13.1.2 Position Detection

13.1.2.1 Position Detection Signal

TIM1_CR3[T1TIS] selects the sources of Position Detection signal, including CMP0/1/2 (CMP Position Detection), GPIO (Hall Sensor Position Detection) or ADC (ACD Position Detection). GPIO signals are sourced by P1.4/P1.3/P1.5. TIM1_CR3[T1INM] decides whether CMP/GPIO signal is filtered. A Position Detection Interrupt is generated upon the completion of position detection. Position Detection Interrupts are divided into CMP/GPIO Position Detection Interrupt and ADC Position Detection Interrupt.

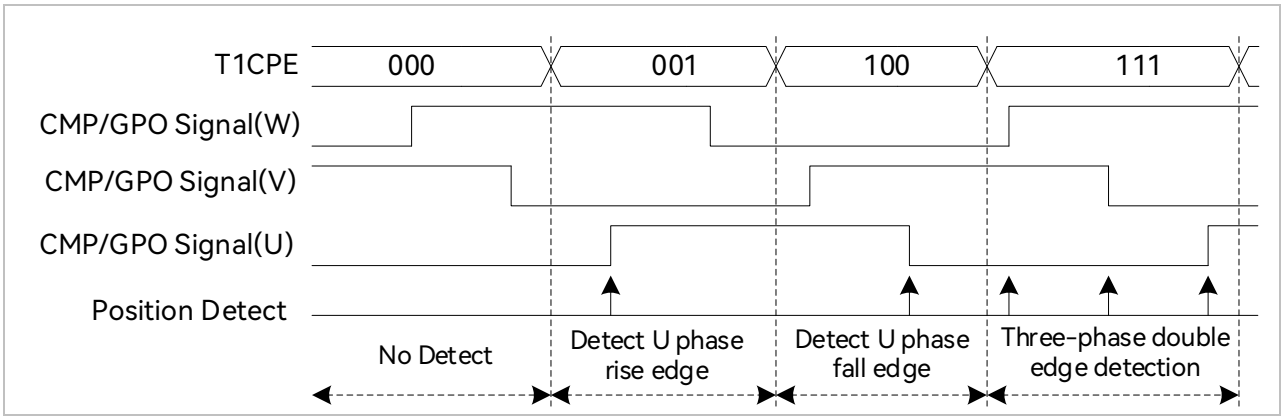
Figure 13-5 Functional Block Diagram of Position Detection



13.1.2.2 CMP/GPIO Position Detection Event

The register bank TIM1_DBR1/2/3/4/5/6/7[T1CPE] is configured to select the active edge of position detection signal. When an active edge of CMP/GPIO Position Detection signal is detected, it indicates the position detection is successfully done, allowing the CMP/GPIO Position Detection Interrupt Flag TIM1_SR[T1PDIF] to become "1". TIM1_CR4[T1CST] selects TIM1_DBR1/2/3/4/5/6/7[T1CPE] timing.

Figure 13-6 Timing Diagram of CMP/GPIO Position Detection



The relation between active edge and TIM1_DBR1/2/3/4/5/6/7[T1CPE] is shown in Table 13-2.

Table 13-2 Mapping between Active Edge and TIM1_DBR1/2/3/4/5/6/7[T1CPE]

T1CPE	Description	T1CPE	Description
000	0	100	Phase-U corresponding comparator is enabled when falling edge of phase-U is detected.
001	Phase-U corresponding comparator is enabled when rising edge of phase-U is detected.	101	phase-W corresponding comparator is enabled when rising edge of phase-W is detected.
010	phase-W corresponding comparator is enabled when falling edge of phase-W is detected.	110	Phase-V corresponding comparator is enabled when falling edge of phase-V is detected.
011	Phase-V corresponding comparator is enabled when rising edge of phase-V is detected.	111	Phase-U+W+V corresponding comparator is enabled when rising or falling edge of phase-U+W+V is detected.

13.1.2.3 ADC Position Detection Event

TIM1_CR3[T1TIS] is configured to select the position detection signal from ADC. Timer1 controls ADC to sample the voltage of active phase and floating phase, which are calculated in the following equation:

$$TIM1_URES = K \times TIM1_UCOP - TIM1_UFLP$$

Where,

K: ADC Position Detection Coefficient

TIM1_UCOP: ADC sampled value of active phase

TIM1_UFLP: ADC sampled value of floating phase

K, *TIM1_UCOP* and *TIM1_UFLP* definitions are determined by TIM1_DBR1/2/3/4/5/6/7[T1CPE], as detailed in Table 13-3.

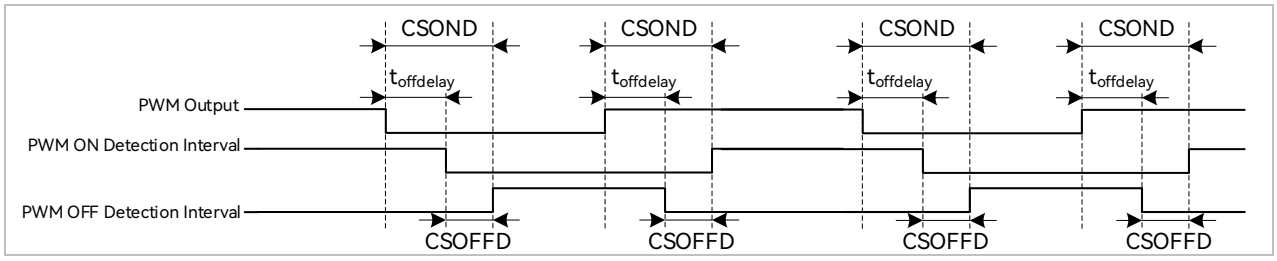
Table 13-3 Relation between TIM1_DBR1/2/3/4/5/6/7[T1CPE] and *K*, *TIM1_UCOP* and *TIM1_UFLP*

T1CPE	Description
000	Reserved
001	TIM1_KR for *K*, W-phase voltage for *TIM1_UCOP*, and U-phase voltage for *TIM1_UFLP*
010	TIM1_KF for *K*, U-phase voltage for *TIM1_UCOP*, and W-phase voltage for *TIM1_UFLP*
011	TIM1_KR for *K*, U-phase voltage for *TIM1_UCOP*, and V-phase voltage for *TIM1_UFLP*
100	TIM1_KF for *K*, V-phase voltage for *TIM1_UCOP*, and U-phase voltage for *TIM1_UFLP*
101	TIM1_KR for *K*, V-phase voltage for *TIM1_UCOP*, and W-phase voltage for *TIM1_UFLP*
110	TIM1_KF for *K*, W-phase voltage for *TIM1_UCOP*, and V-phase voltage for *TIM1_UFLP*
111	Reserved

When TIM1_URES has a negative step or a positive step, an ADC Position Detection Interrupt is generated and TIM1_SR[T1ADIF] (Position Detection Interrupt Flag) is set to "1". The position at which ADC Position Detection Interrupt is generated is controlled by setting the coefficient *K*. In this case, the phase commutation degree can be controlled flexibly.

13.1.2.4 Sampling

Figure 13-7 Timing Diagram of Sampling

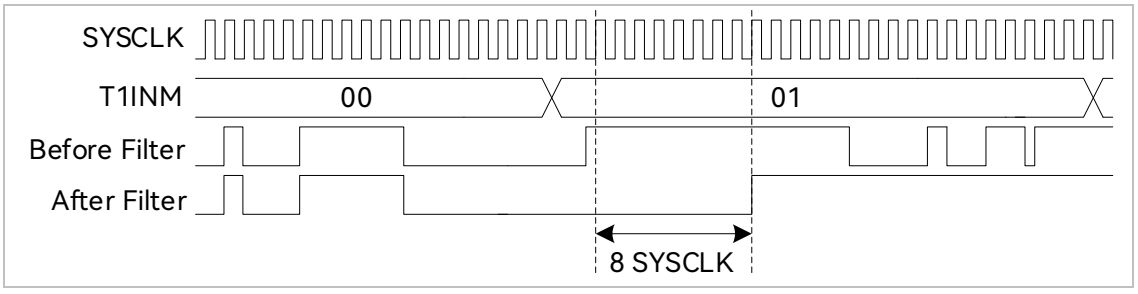


Affected by switching rate of the power device, BEMF signal lags behind PWM output. CMP_SAMR[CSOFFD], CMP_SAMR[CSOND] and CMP_CR4[FAEN] shall be set reasonably to adjust the sampling interval and obtain the valid position detection signal. When TIM1_CR3[T1TIS] = 01 or 10, Timer1 enables CMP0/1/2 to output the compare results between phase BEMF and neutral point, or starts ADC module to sample floating voltage.

See section 27.1.5 for details.

13.1.2.5 Filtering

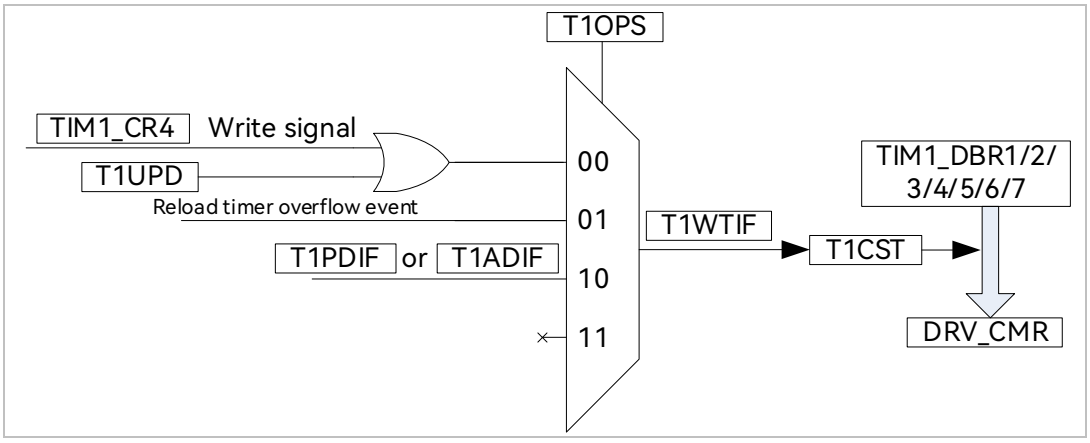
Figure 13-8 Timing Diagram of Filtering Module



According to TIM1_CR3[T1INM] and CMP_CR4[FAEN], the filtered pulse width of input noise can be selected as 8/16/24/32/64/96 system clocks. After this feature is enabled, the signal is delayed by about 8/16/24/32/64/96 system clocks.

13.1.3 Write Timing Interrupt

Figure 13-9 Write Timing Block Diagram



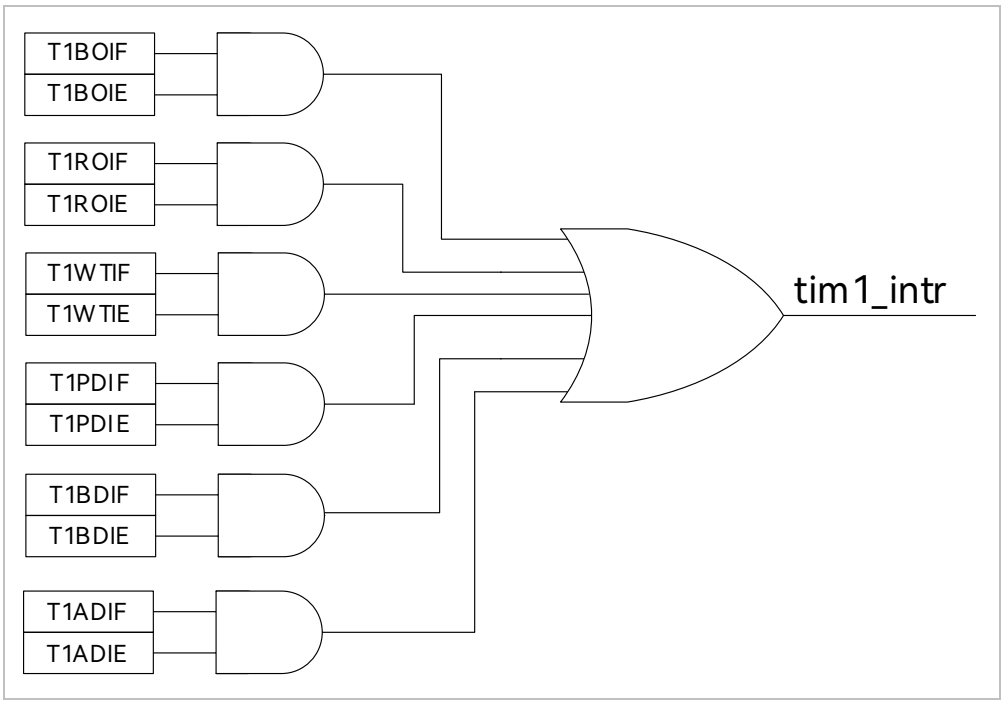
When the control logic, predefined in TIM1_DBR1/2/3/4/5/6/7, is sent to driver register DRV_CMR, a write timing interrupt is generated. The triggered source is selected by the configuration of TIM1_CR0[T1OPS], that is, software, Reload Timer overflow event or position detection event can be selected. When a write timing interrupt is generated, write timing interrupt flag TIM1_SR[T1WTIF] is set to “1”. If TIM1_CR4[T1CST] ranges in 001~110, TIM1_CR4[T1CST] adds 1 automatically.

13.1.4 Timer1 Interrupt

Timer1 supports 6 interrupt sources:

- Basic timer overflow interrupt
- Reload timer overflow interrupt
- Write timing interrupt
- Diode freewheeling end interrupt
- CMP/GPIO position detection interrupt
- ADC position detection interrupt

Figure 13-10 Timer1 Interrupt Sources



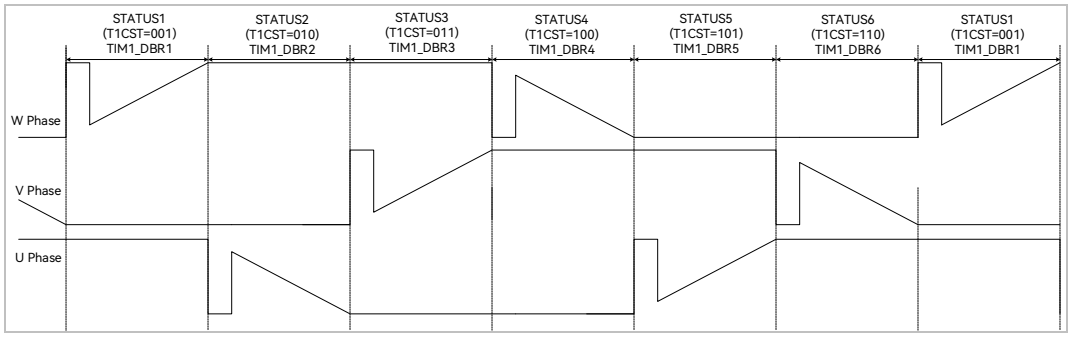
13.2 Square-wave Control for BLDC Motors

For BLDC motor square-wave control application, Timer1 works with CMP0/1/2 and Driver module to achieve the following features:

- Automatic record of 60 degree time, filtered as 60 degree reference time
- Automatic forced phase commutation when no position signal is detected
- Automatic diode freewheeling masking, i.e., stopping comparator sampling during diode freewheeling
- Automatic control of the time from position detection to phase commutation to achieve automatic commutation
- Take over CMP_CR2[CMP0SEL] to control CMP0/1/2 automatically
- Comparator signal can be configured to avoid ringing at the switch node of the power IC, and the signal can be filtered after sampling
- Take over DRV_CMR register to control six PWM outputs automatically

13.2.1 Six-step Phase Commutation of Square Wave Control

Figure 13-11 Diagram of Six-step Phase Commutation of Square Wave Control

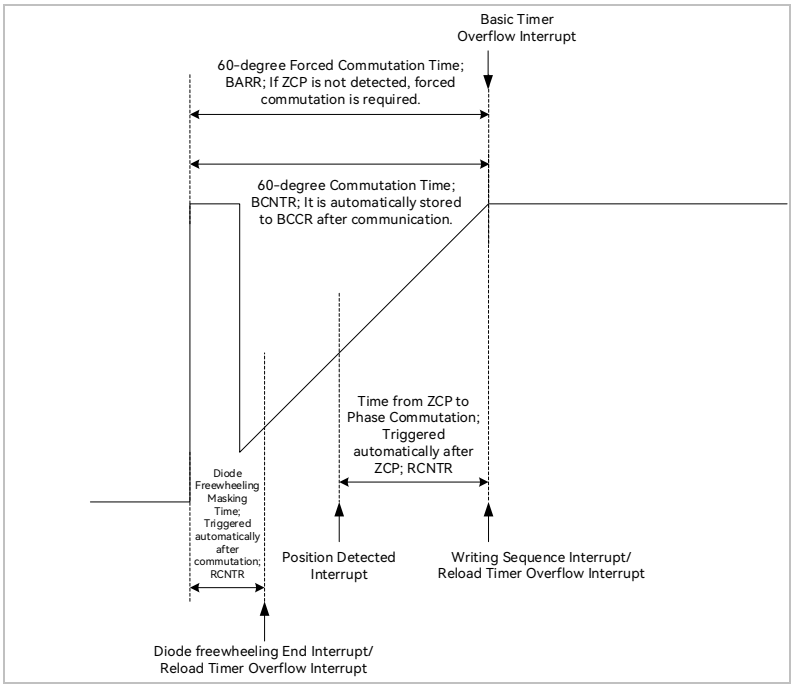


TIM1_CR4[T1CST] is the commutation state machine. Among them, state 0 is used to output off state, and state 7 is customizable for braking, pre-charging, pre-positioning, startup, etc. States 1 ~ 6 are used for six-step automatic commutation, and the state machine TIM1_CR4[T1CST] automatically adds 1 after phase commutation.

The states 1~7 maps to the TIM1_DBR1~7. When write timing interrupt occurs, TIM1_DBRx corresponding to the current state is automatically transferred to DRV_CMR and CMP_CR2[CMP0SEL] for phase commutation and position detection.

13.2.2 Square Wave Control Working Principle

Figure 13-12 Square Wave Control Working Principle



13.2.2.1 60° Commutation Base Time

TIM1__BCCR captures the time of previous 60 degree. TIM1_CR2[T1BRS] is set to “0” to capture the time between two write timing interrupts and TIM1_CR2[T1BRS] to “1” to capture the time between two position detection interrupts.

TIM1_BCOR is the filtered 60 degree time, i.e., 60 degree base time. TIM1_CR0[T1CFLT] can select the previous 1/2/4/8 TIM1__BCCR averaged to obtain TIM1_BCOR.

In square-wave control mode, the diode freewheeling masking time, the time from position detection to commutation, and the time to forced commutation are determined by the 60 degree base time TIM1_BCOR.

When Basic Timer is auto-load enabled (TIM1_CR1[T1BAPE] = 1), and is reset due to a position detection interrupt or a write timing interrupt, TIM1_BCOR is transferred to TIM1__BARR to control the forced phase commutation.

13.2.2.2 Forced Commutation at 60°

When the motor rotates smoothly, a ZCP is generally detected after 30 degrees of rotation after a phase commutation and a position detection interrupt is generated. If no ZCP is detected in 60 degree after the phase commutation, position detection fails and a forced phase commutation is required.

In this case, TIM1_CR0[T1FORC] is set to “1” to enable the forced commutation feature. During previous commutation, the timer TIM1__BCNTR is cleared to “0” by write timing interrupt and restarts counting, while TIM1__BCCR captures the count value held in TIM1__BCNTR, which is filtered and stored in TIM1_BCOR as the 60 degree base time. When auto-load feature is enabled (TIM1_CR1[T1BAPE] = 1), the value held in TIM1_BCOR is loaded into TIM1__BARR after the Basic Timer is cleared. If no ZCP is detected in 60 degree after commutation (TIM1__BCNTR matches TIM1__BARR), TIM1_SR[T1BOIF] (overflow interrupt flag of the Basic Timer) is set to “1” for forced phase commutation, and the timer TIM1__BCNTR is cleared to “0”.

Note

But if an ZCP is detected within 60 degrees after phase commutation, even when TIM1__BCNTR > TIM1__BARR, the forced commutation will not be triggered and TIM1_SR[T1BOIF] will not be set to “1”

When forced commutation feature is disabled (TIM1_CR0[T1FORC] = 0) and TIM1__BCNTR > TIM1__BARR, the interrupt flag TIM1_SR[T1BOIF] is set to “1” and no forced phase commutation is automatically performed. Phase commutation can be performed manually by Basic Timer overflow interrupt flag and the position detection interrupt flag.

13.2.2.3 Diode Freewheeling Masking

After the commutation, inductance energy of the phase is released to the power supply or ground through the diode since the original active phase becomes a floating phase. During diode freewheeling, the floating phase BEMF signal cannot be measured. By masking comparator signal or ADC sampling value during diode freewheeling time, wrong commutation caused by wrong signal generated by the freewheeling is avoided. After freewheeling masking, the freewheeling masking end interrupt flag TIM1_SR[T1BDIF] is generated.

Freewheeling masking time is set by TIM1_CR1[BSEL]: Masking angle = TIM1_CR1[BSEL]/128*60°.

13.2.2.4 Angle of Position Detection to Commutation

After commutation, a ZCP is detected (generating a position detection interrupt) and the hardware starts counting according to the software-set time between ZCP and the commutation. After the counting ends, the hardware automatically implements phase commutation and generates the write timing interrupt flag TIM1_SR[T1WTIF].

The time between ZCP and commutation is set by TIM1_CR2[CSEL] with the formula: Commutation angle = TIM1_CR2[CSEL]/128*60°.

13.2.2.5 Cycle-by-cycle Current Limiting

See section 27.1.1.2.

13.3 Timer1 Registers

13.3.1 TIM1_CR0 (0x4068)

Bit	7	6	5	4	3	2	1	0
Name	T1RWEN	T1CFLT		T1FORC	T1OPS		T1BCEN	T1RCEN
Type	W1	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	T1RWEN	Write to TIM1_CR0[T1RCEN] Enable 0: No effect. 1: When TIM1_CR0 is updated, TIM1_CR0[T1RWEN] and TIM1_CR0[T1RCEN] shall be configured simultaneously to enable or disable TIM1_CR0[T1RCEN]. A write of “0x81” to TIM1_CR0 enables TIM1_CR0[T1RCEN], and “0x80” to disables TIM1_CR0[T1RCEN].
[6:5]	T1CFLT	60 Degree Base Time Filtering Selection The average of previous x times 60 degree (TIM1__BCCR) is used as the base time (TIM1__BCOR). 00: 1 times 60 degree 01: 2 times 60 degree 10: 4 times 60 degree 11: 8 times 60 degree
[4]	T1FORC	Forced Phase Commutation at 60° Enable 0: Disable 1: Enable Note If a ZCP is detected, forced phase commutation will not be implemented even if this bit is enabled
[3:2]	T1OPS	Commutation Trigger Signal Selection This bit selects the trigger signal for TIM1_DBRx to transfer data to DRV_CMR. 00: The transfer is triggered upon a write of “1” to TIM1_IER[T1UPD] in software or a write to TIM1_CR4[T1CST]. 01: The transfer is triggered upon an overflow interrupt of reload timer commutation counter. 10: The transfer is triggered upon a Position Detection Interrupt. 11: Reserved
[1]	T1BCEN	Basic Timer Enable 0: Disable 1: Enable

[0]	T1RCEN	Reload Timer Enable When TIM1_CR0 is updated, TIM1_CR0[T1RWEN] and TIM1_CR0[T1RCEN] must be configured simultaneously to enable or disable TIM1_CR0[T1RCEN]. A write of "0x81" to TIM1_CR0 enables TIM1_CR0[T1RCEN] and "0x80" disables TIM1_CR0[T1RCEN]. TIM1_CR0[T1RCEN] is automatically enabled upon a Position Detection Interrupt and a Write Timing Interrupt. TIM1_CR0[T1RCEN] is cleared to "0" by hardware upon a Reload Timer Overflow Interrupt. TIM1_CR0[T1RCEN] cannot be automatically enabled or disabled by hardware in manual mode. 0: Disable 1: Enable

13.3.2 TIM1_CR1 (0x4069)

Bit	7	6	5	4	3	2	1	0
Name	T1BAPE	BSEL						
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	T1BAPE	TIM1__BARR Register Auto-load Enable With this bit enabled, TIM1__BCOR is written to TIM1__BARR when Basic Timer is reset due to a Position Detection Interrupt or a Write Timing Interrupt. It is used for forced phase commutation at 60°when no ZCP is detected. It has no effect on TIM1__BARR Register auto-load feature in manual mode. 0: Disable 1: Enable
[6:0]	BSEL	Diode Freewheeling Masking Angle Selection This bit is used to configure the angle of diode freewheeling masking after phase commutation. Position is not detected during diode freewheeling masking. Equation: Diode freewheeling masking angle = TIM1_CR1[BSEL]/128*60°  Note This bit is invalid in manual mode

13.3.3 TIM1_CR2 (0x406A)

Bit	7	6	5	4	3	2	1	0
Name	T1BRS	CSEL						
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	T1BRS	Basic Timer Reset Source Selection This bit is invalid in manual mode (TIM1_IER[T1MAME] = 1). TIM1__BCNTR can only be cleared by a BCNTR Overflow Interrupt. 0: Write Timing Reset 1:Position Detection Interrupt Reset
[6:0]	CSEL	Phase Commutation Angle Selection After a position detection event, phase commutation is implemented after the degree configured by TIM1_CR2[CSEL]. Equation: Commutation angle = TIM1_CR2[CSEL]/128*60°

13.3.4 TIM1_CR3 (0x406B)

Bit	7	6	5	4	3	2	1	0
Name	T1COM_MD	T1PSC			T1TIS		T1INM	
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	1	0	0

Bit	Name	Description
[7]	T1COM_MD	CMP0/1/2 output as position detection signal: triggering conditions 0: Valid level detected 1: Valid edge detected ADC output as position detection signal: triggering conditions 0: When DRV_CR[DDIR] = 0, valid edge is detected. When DRV_CR[DDIR] = 1, valid level is detected. 1: When DRV_CR[DDIR] = 0, valid level is detected. When DRV_CR[DDIR] = 1, valid edge is detected. GPIO output as position detection signal: No effect
[6:4]	T1PSC	Timer Clock Source Frequency Selection These bits are configured to divide the system clock as the clock source for Basic Timer and Reload Timer. The clock source frequency of the two timers: 000: 24MHz 001: 12MHz 010: 6MHz 011: 3MHz 100: 1.5MHz 101: 750kHz 110: 375kHz 111: 187.5kHz

[3:2]	T1TIS	Position Detection Signal Selection 00: GPIO (P1.4, P1.3, P1.5) 01: CMP0/1/2 output 10: ADC output 11: Reserved
[1:0]	T1INM	Filter Pulse Width for Position Detection Signal Selection When pulse width of the input signal is less than the set value, it is filtered as noise. The filtering time changes according to CMP_CR4[FAEN]. When CMP_CR4[FAEN] = 0: 00: 4 system clock cycles 01: 8 system clock cycles 10: 16 system clock cycles 11: 24 system lock cycles When CMP_CR4[FAEN] = 1: 00: 32 system clock cycles 01: 64 system clock cycles 10: 96 system clock cycles 11: 128 system lock cycles

13.3.5 TIM1_CR4 (0x406C)

Bit	7	6	5	4	3	2	1	0
Name	RSV					T1CST		
Type	-	-	-	-	-	R/W	R/W	R/W
Reset	-	-	-	-	-	0	0	0

Bit	Name	Description
[7:3]	RSV	Reserved
[2:0]	T1CST	Commutation State Machine The state machine corresponds to different TIM1_DBRx at different states. When TIM1_CR4[T1CST] reads 001 ~ 111, Timer1 automatically enables or disables CMP0/1/2 according to TIM1_DBRx[T1CPE]. When TIM1_CR4[T1CST] reads 001 ~ 110, Timer1 automatically adds by “1” each cycle upon a Write Timing Interrupt.

Table 13-4 Mapping between TIM1_CR4[T1CST] and TIM1_DBRx

TIM1_CR4[T1CST]	TIM1_DBRx	TIM1_CR4[T1CST]	TIM1_DBRx
000	0	100	TIM1_DBR4
001	TIM1_DBR1	101	TIM1_DBR5
010	TIM1_DBR2	110	TIM1_DBR6
011	TIM1_DBR3	111	TIM1_DBR7

13.3.6 TIM1_IER (0x406D)

Bit	7	6	5	4	3	2	1	0
Name	T1UPD	T1MAME	T1ADIE	T1BOIE	T1ROIE	T1WTIE	T1PDIE	T1BDIE
Type	W1	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	T1UPD	When TIM1_CR0[T1OPS] = 00, a write of “1” to this bit enables data transfer. This bit is cleared to “0” by hardware after “1” is written.
[6]	T1MAME	Manual Mode Enable With this bit enabled, Basic Timer and Reload Timer acts as separate counters. Details: TIM1_BCNTR of the Basic Timer is cleared by a Basic Timer Overflow Interrupt, instead of TIM1_CR2[T1BRS] TIM1_CR0[T1RCEN] of the Reload Timer cannot be cleared to “0” or set to “1” automatically, and is operated by software only. TIM1_RCNTR of the Reload Timer can be cleared to “0” upon a Reload Timer Overflow Interrupt only. TIM1_RARR of the Reload Timer cannot be updated automatically, and is operated by software only. 0: Disable 1: Enable
[5]	T1ADIE	ADC Position Detection Interrupt Enable 0: Disable 1: Enable
[4]	T1BOIE	Basic Timer Overflow Interrupt Enable 0: Disable 1: Enable
[3]	T1ROIE	Reload Timer Overflow Interrupt Enable 0: Disable 1: Enable
[2]	T1WTIE	Write Timing Interrupt Enable 0: Disable 1: Enable
[1]	T1PDIE	CMP/GPIO Position Detection Interrupt Enable 0: Disable 1: Enable
[0]	T1BDIE	Diode Freewheeling Masking Interrupt Enable 0: Disable 1: Enable

13.3.7 TIM1_SR (0x406E)

Bit	7	6	5	4	3	2	1	0
Name	RSV		T1ADIF	T1BOIF	T1ROIF	T1WTIF	T1PDIF	T1BDIF
Type	-	-	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0
Reset	-	-	0	0	0	0	0	0

Bit	Name	Description
[7:6]	RSV	Reserved
[5]	T1ADIF	ADC Position Detection Interrupt Flag A position detection interrupt is generated when TIM1_DBRx[T1CPE] matches ACD position detection signal. Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0”. 1: No effect
[4]	T1BOIF	Basic Timer Overflow Interrupt Flag An overflow event occurs when Basic Timer counts up and TIM1__BCNTR matches with TIM1__BARR. Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[3]	T1ROIF	Reload Timer Overflow Interrupt Flag An overflow event occurs and TIM1__RCNTR is cleared to “0” when TIM1__RCNTR matches TIM1__RARR. Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect

[2]	T1WTIF	Write Timing Interrupt Flag A write timing interrupt is generated when TIM1_DBRx is transferred to DRV_CMR. Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[1]	T1PDIF	CMP/GPIO Position Detection Interrupt Flag A position detection interrupt is generated when CMP/GPIO position detection signal matches TIM1_DBRx[T1CPE]. Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[0]	T1BDIF	Diode Freewheeling Masking End Interrupt Flag Diode freewheeling masking starts after phase commutation and an interrupt is generated at end. Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect

13.3.8 TIM1_BCOR (0x4070, 0x4071)

TIM1_BCORH(0x4070)								
Bit	15	14	13	12	11	10	9	8
Name	TIM1_BCOR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM1_BCORL(0x4071)								
Bit	7	6	5	4	3	2	1	0
Name	TIM1_BCOR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	TIM1_BCOR	This bit is configured to capture filtered count values held in the Basic Timer.

		TIM1_BCCR hods the filtered count value, i.e., 60 Degree Base Time.

13.3.9 TIM1_CR5 (0x4072)

Bit	7	6	5	4	3	2	1	0
Name	T1POP	T1WTS	RSV		ITRIP_DIS	UCOP_DIS	T1AFL	
Type	R/W	R/W	-	-	R/W	R/W	R/W	R/W
Reset	0	0	-	-	0	0	0	0

Bit	Name	Description
[7]	T1POP	Data Transfer Triggered by Driver Counter Overflow This bit is valid only when TIM_CR0[T1OPS] = 00. With it enabled, data transfer is triggered by Driver Counter Overflow, namely, commuting the phase once every PWM cycle. 0: Disable 1: Enable
[6]	T1WTS	PWM Synchronization Enable 0: Disable 1: Enable
[5:4]	RSV	Reserved
[3]	ITRIP_DIS	Bus Current Sampling Disable 0: Disable 1: Enable
[2]	UCOP_DIS	Active Phase Voltage Sampling Disable 0: Disable 1: Enable
[1:0]	T1AFL	ADC Sampled Voltage Calculation Filtering Counts 00: 1 01: 2 10: 4 11: 8

13.3.10 TIM1_DBR x (x = 1 ~ 7) (0x4072 + 2*x, 0x4073 + 2*x)

TIM1_DBRx (x = 1~7) corresponds to the data when T1CST = 1/2/3/4/5/6, respectively. The TIM1_DBRx register is described below using TIM1_DBR1 as an example.

TIM1_DBR1H(0x4074)								
Bit	15	14	13	12	11	10	9	8
Name	RSV	T1CPE			T1WHP	T1WLP	T1VHP	T1VLP
Type	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	0	0	0	0	0	0	0
TIM1_DBR1L(0x4075)								
Bit	7	6	5	4	3	2	1	0

Name	T1UHP	T1ULP	T1WHE	T1WLE	T1VHE	T1VLE	T1UHE	T1ULE
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15]	RSV	Reserved
[14:12]	T1CPE	Position Detection Input Edge Polarity and Comparator Enable Selection This bit is used to define the edge of Position Detection Input and enable or disable the associated comparators. The detected edge in input signal, corresponding to the configuration, generates a position detection interrupt. See CMP/GPIO Position Detection Event and Table 13-2
[11]	T1WHP	High-side Output Polarity of Phase-W 0: Active High 1: Active Low
[10]	T1WLP	Low-side Output Polarity of Phase-W 0: Active High 1: Active Low
[9]	T1VHP	High-side Output Polarity of Phase-V 0: Active High 1: Active Low
[8]	T1VLP	Low-side Output Polarity of Phase-V 0: Active High 1: Active Low
[7]	T1UHP	High-side Output Polarity of Phase-U 0: Active High 1: Active Low
[6]	T1ULP	Low-side Output Polarity of Phase-U 0: Active High 1: Active Low
[5]	T1WHE	High-side Output Enable of Phase-W 0: Disable 1: Enable
[4]	T1WLE	Low-side Output Enable of Phase-W 0: Disable 1: Enable
[3]	T1VHE	High-side Output Enable of Phase-V 0: Disable 1: Enable
[2]	T1VLE	Low-side Output Enable of Phase-V 0: Disable 1: Enable
[1]	T1UHE	High-side Output Enable of Phase-U

		0: Disable 1: Enable
[0]	T1ULE	Low-side Output Enable of Phase-U 0: Disable 1: Enable

Note

The high-side and low-side outputs of W, V and U-phases are complementary and deadtime is automatically added (same for TIM1_DBR2 ~ TIM1_DBR7) when TIM1_DBR1[T1WLE] and TIM1_DBR1[T1WHE], TIM1_DBR1[T1VLE] and TIM1_DBR1[T1VHE] or TIM1_DBR1[T1ULE] and TIM1_DBR1[T1UHE] are set to "1"

13.3.11 TIM1_BCNTR (0x4082, 0x4083)

TIM1_BCNTRH(0x4082)								
Bit	15	14	13	12	11	10	9	8
Name	TIM1_BCNTR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	0	0	0	0	0	0	0
TIM1_BCNTRL(0x4083)								
Bit	7	6	5	4	3	2	1	0
Name	TIM1_BCNTR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	TIM1_BCNTR	This bit holds count values of the Basic Timer and is used for clocking commutation at 60°. Auto mode: TIM1_BCNTR register selects the reset source according to TIM1_CR2[T1BRS], and TIM1_BCNTR does not restart when TIM1_BCNTR overflow interrupt is generated. Manual mode: TIM1_BCNTR restarts when TIM1_BCNTR overflow interrupt is generated.

13.3.12 TIM1_BCCR (0x4084, 0x4085)

TIM1_BCCRH(0x4084)								
Bit	15	14	13	12	11	10	9	8
Name	TIM1_BCCR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM1_BCCRL(0x4085)								
Bit	7	6	5	4	3	2	1	0
Name	TIM1_BCCR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	TIM1_BCCR	This bit is configured to capture count values held in Basic Timer. Auto mode: When the Basic Timer is reset on a Position Detection Interrupt or a Write Timing Interrupt, the count values before the reset are stored into TIM1_BCCR. Manual mode: When the Basic Timer is reset on an Overflow Interrupt, the count values before the reset are stored into TIM1_BCCR.

13.3.13 TIM1_BARR (0x4086, 0x4087)

TIM1_BARRH(0x4086)								
Bit	15	14	13	12	11	10	9	8
Name	TIM1_BARR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM1_BARRL(0x4087)								
Bit	7	6	5	4	3	2	1	0
Name	TIM1_BARR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	TIM1_BARR	Reload Value of Basic Timer When the count value of the Basic Timer equals to TIM1_BARR, an overflow interrupt is generated and the timer is cleared to “0”.

13.3.14 TIM1_RARR (0x4088, 0x4089)

TIM1_RARRH(0x4088)								
Bit	15	14	13	12	11	10	9	8
Name	TIM1_RARR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM1_RARRL(0x4089)								
Bit	7	6	5	4	3	2	1	0
Name	TIM1_RARR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	TIM1_RARR	Auto-reload Value of Reload Timer When the count of Reload Timer is equal to TIM1_RARR, an overflow

		interrupt is generated and the value of the timer is cleared to “0”. Auto mode: The value of diode freewheeling masking angle held in TIM1_CR1[BSEL] is updated to TIM1_RARR when a write timing interrupt is generated. The commutation angle held in TIM1_CR2[CSEL] is updated to TIM1_RARR when a position detection interrupt occurs. Manual mode: TIM1_RARR is not updated automatically.

13.3.15 TIM1_RCNTR (0x408A, 0x408B)

TIM1_RCNTRH(0x408A)								
Bit	15	14	13	12	11	10	9	8
Name	TIM1_RCNTR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1
TIM1_RCNTRL(0x408B)								
Bit	7	6	5	4	3	2	1	0
Name	TIM1_RCNTR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Bit	Name	Description
[15:0]	TIM1_RCNTR	Count value of the Reload Timer for counting numbers of diode freewheeling masking and ZCP to phase commutation. Note In manual mode, TIM1_RCNTR is cleared to “0” only by a Reload Timer overflow interrupt

13.3.16 TIM1_UCOP (0x408C, 0x408D)

TIM1_UCOPH(0x408C)								
Bit	15	14	13	12	11	10	9	8
Name	TIM1_UCOP[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM1_UCOPL(0x408D)								
Bit	7	6	5	4	3	2	1	0
Name	TIM1_UCOP[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	TIM1_UCOP	ADC sampled value of active phase voltage (second-highest bit

		alignment)

13.3.17 TIM1_UFLP (0x408E, 0x408F)

TIM1_UFLPH(0x408E)								
Bit	15	14	13	12	11	10	9	8
Name	TIM1_UCOP[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM1_UFLPL(0x408F)								
Bit	7	6	5	4	3	2	1	0
Name	TIM1_UCOP[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	TIM1_UFLP	ADC sampled value of floating phase voltage (second-highest bit alignment)

13.3.18 TIM1_URES (0x4090, 0x4091)

TIM1_URESH(0x4090)								
Bit	15	14	13	12	11	10	9	8
Name	TIM1_URES[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM1_URESL(0x4091)								
Bit	7	6	5	4	3	2	1	0
Name	TIM1_URES[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	TIM1_URES	Result of ADC position detection formula; Q15 format

13.3.19 TIM1_KRMAX (0x4092)

Bit	7	6	5	4	3	2	1	0
Name	TIM1_KRMAX							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	TIM1_KRMAX	Max. Coefficient of Raising Edge

		Range [0,255]

13.3.20 TIM1_KFMIN (0x4093)

Bit	7	6	5	4	3	2	1	0
Name	TIM1_KFMIN							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	TIM1_KFMIN	Min. Coefficient of Falling Edge Range [0,255]

13.3.21 TIM1_KF (0x4094, 0x4095)

TIM1_KFH(0x4094)								
Bit	15	14	13	12	11	10	9	8
Name	TIM1_KF[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM1_KFL(0x4095)								
Bit	7	6	5	4	3	2	1	0
Name	TIM1_KF[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	TIM1_KF	ADC position detection coefficient when floating phase voltage drops Range [0,32767]

13.3.22 TIM1_KR (0x4096, 0x4097)

TIM1_KRH(0x4096)								
Bit	15	14	13	12	11	10	9	8
Name	TIM1_KR[15:8]							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM1_KRL(0x4097)								
Bit	7	6	5	4	3	2	1	0
Name	TIM1_KR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description

[15:0]	TIM1_KR	ADC position detection coefficient when floating phase voltage rises Range [0,32767]

13.3.23 TIM1__ITRIP (0x4098, 0x4099)

TIM1__ITRIPH(0x4098)								
Bit	15	14	13	12	11	10	9	8
Name	TIM1__ITRIP[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
TIM1__ITRIPL(0x4099)								
Bit	7	6	5	4	3	2	1	0
Name	TIM1__ITRIP[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	TIM1__ITRIP	Filtered Bus Current When DRV_CNTR = 0, the hardware automatically samples the bus current and filters it for software application. The default channel is ADC channel 4. Range [0,32767] Note The value is obtained by averaging the instantaneous current values of 8 samples

14 Timer2

14.1 Timer2 Instructions

Timer2 has the following three working modes:

- Output mode: PWM output
- Input capture mode: Detect the duration of high and low level of input PWM
- Input counter mode: Detect input time of the set PWM wave numbers

Timer2 features:

- 3-bit programmable prescaler divides the system clock
- 16-bit up-counting Basic Timer; Counting clock source serves as the output of prescaler
- 16-bit up/down-counting special timer for input counter mode, with external input signal as clock source
- Input filter module
- Edge detection module
- PWM output module
- Interrupt event

14.1.1 Prescaler

Prescaler divides the system clock frequency and generates clock source for Basic Timer. 8 frequency division coefficients of prescaler are available and can be selected by TIM2_CR0[T2PSC]. Since this register has no buffer, the clock source frequency is updated immediately after TIM2_CR0[T2PSC] is written. Therefore, the frequency division coefficients shall be configured when Basic Timer is not working. The clock source frequency formula is: clk_psc2 = T2CLK/(2^TIM2_CR0[T2PSC]). The clock rate corresponding to different TIM2_CR0[T2PSC] value as shown in Table 14-1.

Table 14-1 Mapping between Clock Rate and TIM2_CR0[T2PSC]

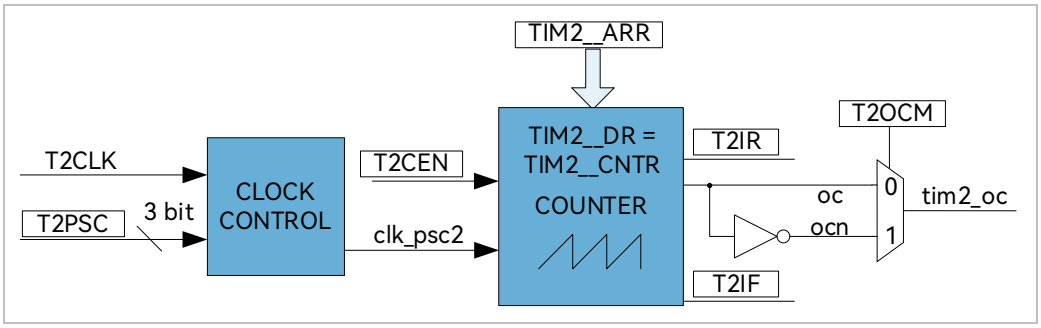
TIM2_CR0[T2PSC]	Division Factor	clk_psc2(Hz)	TIM2_CR0[T2PSC]	Division Factor	clk_psc2(Hz)
000	1	24M	100	16	1.5M
001	2	12M	101	32	750k
010	4	6M	110	64	375k
011	8	3M	111	128	187.5k

14.1.2 Reading, Writing and Counting of TIM2__CNTR

When TIM2_CR1[T2CEN] = 1, TIM2__CNTR starts to count. The write operation to TIM2__CNTR directly changes the value of the register, so Basic Timer shall be disabled before the write operation. When reading TIM2__CNTR, the software reads the high-order bytes first, and the hardware synchronously caches the low-order bytes. When reading the low-order bytes, the software reads the cached data.

14.1.3 Output Mode

Figure 14-1 Output Mode Block Diagram



The output mode generates output signals according to TIM2_CR0[T2OCM], and the comparison results between TIM2__CNTR and registers TIM2__DR, TIM2__ARR. Meanwhile, corresponding interrupts events are generated.

14.1.3.1 Reading and Writing of TIM2__ARR/TIM2__DR

In output mode, TIM2__ARR/TIM2__DR contains preload registers and shadow registers. When the software writes TIM2__ARR/TIM2__DR register, the data is saved in the preload register. When the overflow event TIM2_CR1[T2IF] is generated or the Basic Timer stops working (TIM2_CR1[T2CEN] = 0), the set value is transferred to the shadow register.

TIM2__ARR/TIM2__DR is a 16-bit register, which requires to write the high-order bytes first and then the

low-order bytes. The hardware ensures that the data in the preload register is not transferred to the shadow register after the high-order bytes are written and before the low-order bytes are written.

For example, TIM2__DR is a preload register and DR_SH is a shadow register. PWM is generated by comparing TIM2__CNTR with DR_SH. When software writes TIM2__DR, TIM2__DR is not updated to DR_SH immediately, and is updated to TIM2__DR at the end of a PWM (TIM2__CNTR overflow event).

14.1.3.2 High-/Low-level Output Mode

When TIM2_CR0[T2OCM] = 0, if TIM2__DR > TIM2__ARR, the output signal is always low. When TIM2_CR0[T2OCM] = 1, if TIM2__DR > TIM2__ARR, the output signal is always high.

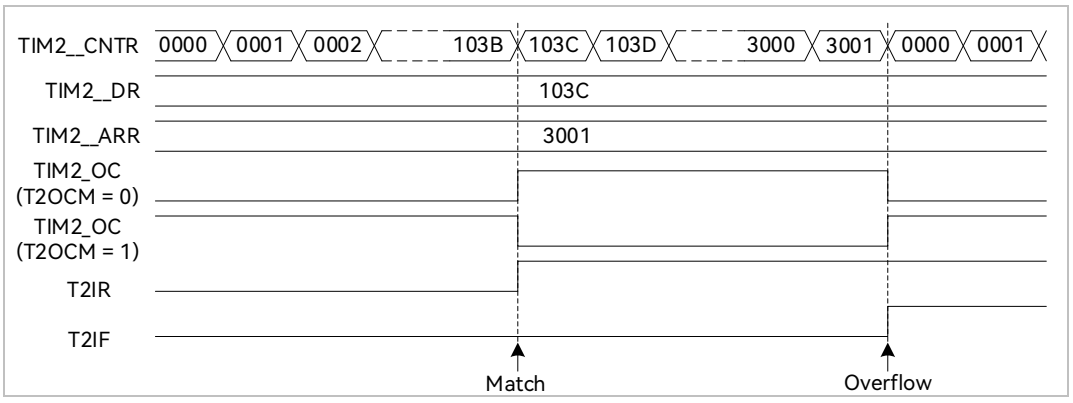
14.1.3.3 PWM Output

In PWM output mode, TIM2__ARR determines PWM cycle, TIM2__DR determines duty cycle, and duty cycle = TIM2__DR/TIM2__ARR*100%. If TIM2_CR0[T2OCM] = 0, the low level is output when TIM2__CNTR < TIM2__DR, and the high level is output when TIM2__CNTR ≥ TIM2__DR. If TIM2_CR0[T2OCM] = 1, the high level is output when TIM2__CNTR < TIM2__DR, and the low level is output when TIM2__CNTR ≥ TIM2__DR. When TIM2__CNTR is increased to TIM2__ARR, the output signal is reversed.

14.1.3.4 Interrupts

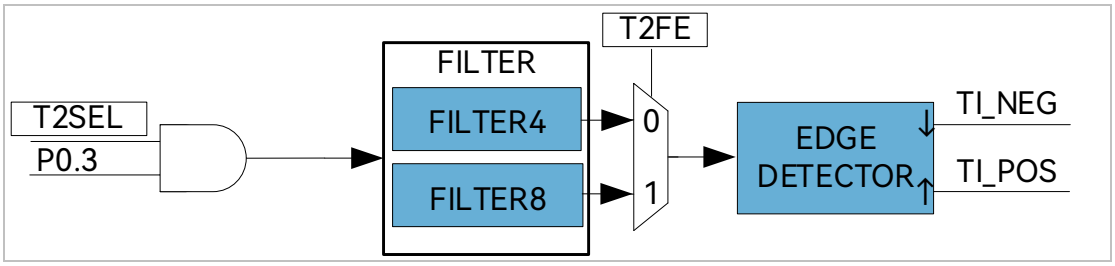
- When TIM2__CNTR = TIM2__DR, a compare match event is generated and the interrupt flag bit TIM2_CR1[T2IR] is set to “1”. The timer continues.
- When TIM2__CNTR = TIM2__ARR, an overflow event is generated, and the interrupt flag bit TIM2_CR1[T2IF] is set to “1”. The timer is cleared to “0” and then restarts

Figure 14-2 Output Mode Waveform



14.1.4 Input Signal Filtering and Edge Detection

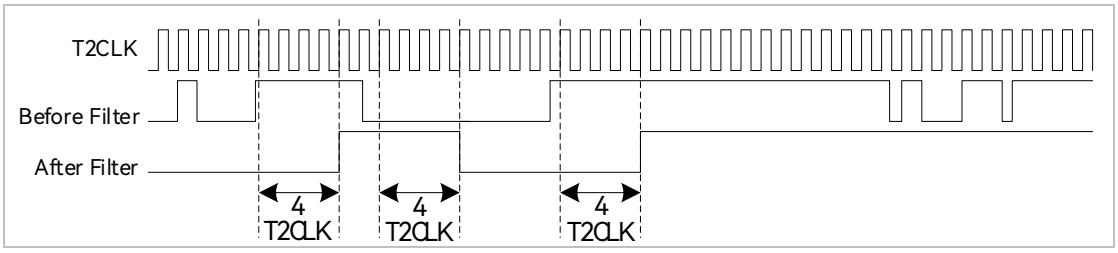
Figure 14-3 Block Diagram of Input Signal Filtering and Edge Detection



The input signal of Timer2 comes from P0.3. The filter of input signal is optional.

The filtering circuit filters out the input noise below 4/8 SYSCLK cycles. The filtering period is selected by setting TIM2_CR1[T2FE]. When TIM2_CR1[T2FE] is set to “0”, filtering circuit filters signals every 4 system cycles; and when TIM2_CR1[T2FE] is set to “1”, filtering circuit filters signals every 8 system cycles. The filtered signal is 4/8 clock cycles later than the signal before filtering. TIM2_CR0[T2CES] determines the active edge to count.

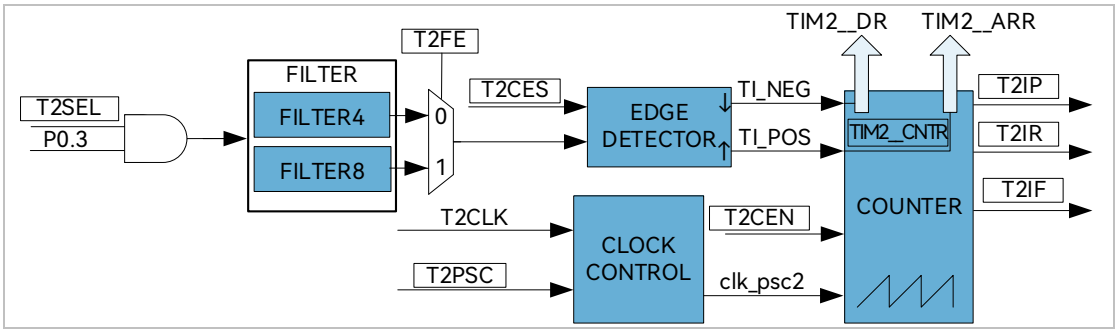
Figure 14-4 Timing Diagram of Filter Module



The edge detection module detects filtered input signals and records rising edge and falling edge for input capture mode or input counter modes.

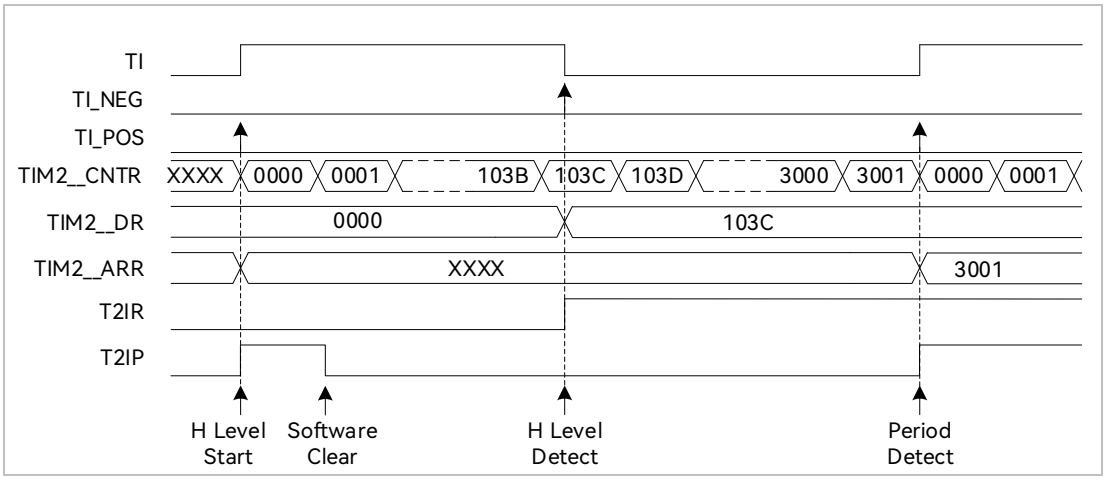
14.1.5 Input Capture Mode

Figure 14-5 Schematic Diagram of Input Capture Mode



The input capture mode detects duty cycle and period of the PWM signal. When TIM_CR0[T2CES] = 0, the time between two adjacent rising edges forms one cycle, and the time from rising edge to falling edge forms the pulse width (HIGH). When TIM_CR0[T2CES] = 1, the time between two adjacent falling edges forms one cycle, and the time from falling edge to rising edge forms the pulse width (LOW). When the predefined edge arrives, the count value TIM2__CNTR is stored in TIM2__DR and TIM2__ARR respectively to calculate the period and duty cycle of PWM waveform.

Figure 14-6 Timing Diagram of Input Capture Mode (TIM2_CR0[T2CES] = 0)

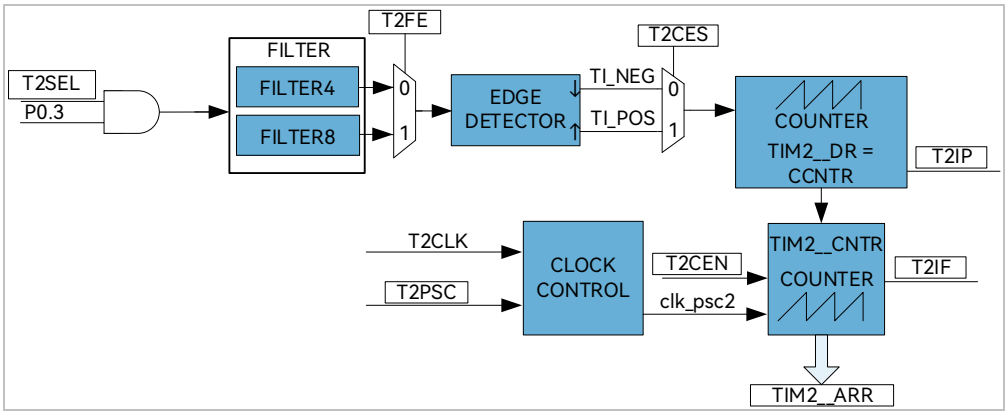


For example, when TIM2_CR0[T2CES] = 0, TIM2_CR1[T2CEN] is set to “1” to enable the Basic Timer. When the first rising edge of the input (falling edge is invalid) is detected, TIM2__CNTR is cleared and restarts. When falling edge of the input is detected, the value of TIM2__CNTR is stored in TIM2__DR, and the interrupt flag TIM2_CR1[T2IR] is set to “1”, and TIM2__CNTR continues to count. When the second rising edge of input is detected, the value of TIM2__CNTR is stored in TIM2__ARR. Meanwhile, the interrupt flag TIM2_CR1[T2IP] is set to “1”, and TIM2__CNTR is cleared to “0” and restarts.

An overflow event occurs if Timer2 does not detect the second rising edge of the input and TIM2__CNTR reaches 0xFFFF. In this case, the interrupt flag TIM2_CR1[T2IF] is set to “1”, and TIM2__CNTR is cleared to “0” and restarts. At this point, TIM2__ARR value is 0xFFFF, and the TIM2__DR value is determined by the input level and TIM2_CR0[T2OCM] XOR.

14.1.6 Input Counter Mode

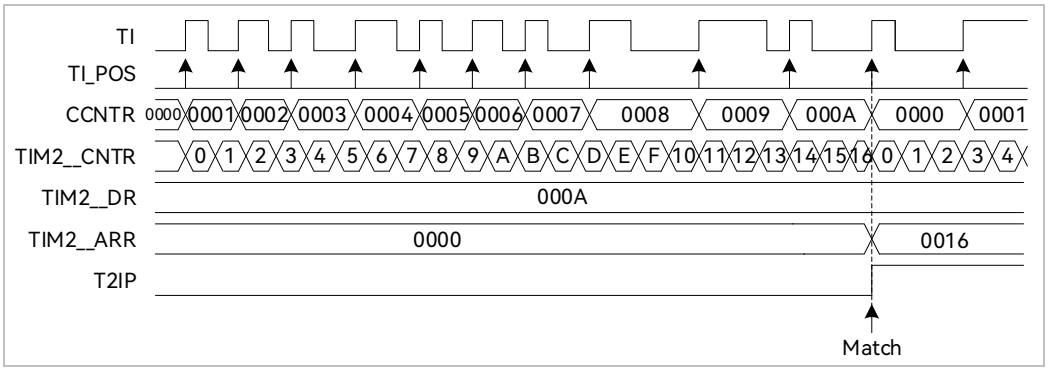
Figure 14-7 Schematic Diagram of Input Counter Mode



In input counter mode, TIM2__DR includes preload register and shadow register. When the software writes TIM2__DR register, the data is saved in the preload register first, and then sent to the shadow register in case of compare match event (TIM2_CR1[T2IP] = 1), overflow event (TIM2_CR1[T2IF] = 1) or special timer disable (TIM2_CR1[T2CEN] = 0). TIM2__DR is a 16-bit register, which requires the software writes the high-order bytes first and then the low-order bytes. The hardware ensures that the data in the preload register is not updated to the shadow register after the high-order bytes are written and before the low-order bytes are written.

The input counter mode is used to detect the time to input the set PWM wave. When the number of input PWM counted by the special timer CCNTR reaches the set value (TIM2__DR), TIM2__CNTR of the Basic Timer is stored in TIM2__ARR. When TIM2_CR0[T2CES] is set to “1”, the rising edge of the input PWM signal serves as the active counting edge of the special timer; when TIM2_CR0[T2CES] is set to “0”, the falling edge of the input signal as the active edge.

Figure 14-8 Timing Diagram of Input Counter Mode



The Basic Timer is enabled when TIM2_CR1[T2CEN] is set to “1”. If the first active edge of the input signal is detected, TIM2__CNTR is cleared to “0” and restarts. Whenever active edge of the input signal arrives, one is added to the count value of the special timer CCNTR. When the count value reaches TIM2__DR, TIM2__CNTR is stored in TIM2__ARR. When TIM2_CR1[T2IP] is set to “1”, TIM2__CNTR and CCNTR are cleared to “0” and restart.

When the number of input PWM does not reach the set value and TIM2__CNTR reaches 0xFFFF, an overflow event occurs, and the interrupt flag TIM2_CR1[T2IF] is set to “1”. TIM2__CNTR is cleared to “0” with CCNTR uncleared. TIM2__CNTR starts counting from 0, and CCNTR continues counting with the previous value.

14.2 Timer2 Registers

14.2.1 TIM2_CR0 (0xA1)

Bit	7	6	5	4	3	2	1	0
Name	T2PSC			T2OCM	T2IRE	T2CES	T2MOD	
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:5]	T2PSC	Basic Timer Clock Prescaler Selection It is configured to divide the system clock frequency and generate the clock source for Basic Timer. The prescaled clock rates are configured as follows: 000: 24MHz 001: 12MHz 010: 6MHz 011: 3MHz 100: 1.5MHz 101: 750kHz 110: 375kHz 111: 187.5kHz
[4]	T2OCM	Output Mode: Output Mode Selection 0: “0” is output when TIM2_CNTR < TIM2_DR; “1” is output when TIM2_CNTR ≥ TIM2_DR. 1: “1” is output when TIM2_CNTR < TIM2_DR; “0” is output when TIM2_CNTR ≥ TIM2_DR. Input Counter Mode: No effect Input Capture Mode: TIM2_DR indicates the input level to be selected when timer TIM2_CNTR becomes overflowed. 0: TIM2_DR is reset to “0” by hardware for low-level input upon an overflow interrupt and is set to “0xFFFF” for high-level input upon an overflow interrupt. 1: TIM2_DR is reset to “0” by hardware for high-level input upon an overflow interrupt and is set to “0xFFFF” for low-level output upon an overflow interrupt.
[3]	T2IRE	Output Mode: Compare Match Interrupt Enable Input Capture Mode: Pulse Width Detection Interrupt Enable Input Counter Mode: No effect
[2]	T2CES	Output Mode: No effect Input Capture Mode: Counting Edge Selection 0: The time between two adjacent raising edges forms one cycle, and the time from rising edge to falling edge forms the pulse width (HIGH). 1: The time between two adjacent falling edges forms one cycle, and the time from falling edge to raising edge forms the pulse width (LOW). Input Counter Mode: Active Edge Selection 0: Falling Edge Count 1: Raising Edge Count
[1:0]	T2MOD	Mode Selection 00: Input Capture Mode

		01: Output Mode 10: Input Counter Mode 11: Reserved

14.2.2 TIM2_CR1 (0xA9)

Bit	7	6	5	4	3	2	1	0
Name	T2IR	T2IP	T2IF	T2IPE	T2IFE	T2FE	T2DIR	T2CEN
Type	R/W0	R/W0	R/W0	R/W	R/W	R/W	R	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	T2IR	Output Mode: Compare Match Interrupt Flag Input Capture Mode: Pulse Width Detection Interrupt Flag Input Counter Mode: No effect Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[6]	T2IP	Output Mode: No effect Input Capture Mode: PWM Cycle Detection Interrupt Flag Input Counter Mode: PWM Input Count Match Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[5]	T2IF	Output Mode: Basic Timer Overflow Interrupt Flag, which is set to “1” when TIM2_CNTR matches TIM2_ARR. Input Capture Mode: Basic Timer Overflow Interrupt Flag, which is set to “1” when the Timer has not detected an input PWM cycle but the timer TIM2_CNTR value reaches 0xFFFF. Input Counter Mode: Special timer overflow Interrupt Flag, which is set to “1” when the input PWM cycle has not reached the preset TIM2_DR value but the Basic Timer TIM2_CNTR value reaches 0xFFFF. Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect

[4]	T2IPE	Output Mode: No effect Input Capture Mode: PWM Cycle Detection Interrupt Enable Input Counter Mode: PWM Input Count Match Interrupt Enable 0: Disable 1: Enable
[3]	T2IFE	Output Mode: Basic Timer Overflow Interrupt Enable Input Capture Mode: Basic Timer Overflow Interrupt Enable Input Count Mode: Basic Timer Overflow Interrupt Enable 0: Disable 1: Enable
[2]	T2FE	Input Signal Filter Selection Input signals are filtered out as noise if the pulse width is less than 4/8 clock cycle. Assuming that the system clock runs at 24MHz (41.67ns) , then the pulse width for filtering is 166.67/333.34ns. 0: Signals filtered on every 4 clock cycles 1: Signals filtered on every 8 clock cycles
[1]	T2DIR	Rotation direction of the motor is determined according to the direction signal P0.3. 0: Forward 1: Backward
[0]	T2CEN	Basic Timer Enable 0: Disable 1: Enable

14.2.3 TIM2_CNTR (0xAA, 0xAB)

TIM2_CNTRH(0xAB)								
Bit	15	14	13	12	11	10	9	8
Name	TIM2_CNTR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM2_CNTRL(0xAA)								
Bit	7	6	5	4	3	2	1	0
Name	TIM2_CNTR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	TIM2_CNTR	Output Mode/Input Capture Mode/Input Counter Mode: Count values held in the Basic Timer

14.2.4 TIM2_DR (0xAC, 0xAD)

TIM2_DRH(0xAD)								
Bit	15	14	13	12	11	10	9	8
Name	TIM2_DR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM2_DRL(0xAC)								
Bit	7	6	5	4	3	2	1	0
Name	TIM2_DR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	TIM2_DR	Output Mode: Compare match value (written by software) Input Capture Mode: Count value of the detected input pulse width (written by hardware) Input Counter Mode: PWM cycles to be counted (written by software)

14.2.5 TIM2_ARR (0xAE, 0xAF)

TIM2_ARRH(0xAF)								
Bit	15	14	13	12	11	10	9	8
Name	TIM2_ARR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIM2_ARRL(0xAE)								
Bit	7	6	5	4	3	2	1	0
Name	TIM2_ARR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	TIM2_ARR	Output Mode: PWM cycle (written by software) Input Capture Mode: Count value held in Basic Timer of a PWM cycle (written by hardware) Input Counter Mode: Count value held in Basic Timer when the input PWM count matches (written by hardware)

15 Timer3/Timer4

15.1 Timer3/Timer4 Instructions

Timer3/Timer4 support output and input modes:

- Output mode: Generate PWM
- Input capture mode: Detect the duration of high and low level of input PWM, which can be used to calculate PWM duty cycle

Timer3/Timer4 Features:

- 3-bit programmable prescaler divides system clock as the clock source for Basic Timer (clock source of Timer3 can be doubled to 48MHz in input capture mode)
- 16-bit up-counting Basic Timer; The output of the prescaler serves as the counting clock source
- Input signal filtering
- Input signal edge detection
- Output PWM signal, single compare output
- FG generation (Timer4)
- Interrupt event

15.1.1 Prescaler

Prescaler divides the system clock frequency and generates counter clock source for Basic Timer. 8 frequency division coefficients of prescaler are available and can be selected by TIMx_CR0[TxPSC]. Since this register has no buffer, the clock source frequency is updated immediately after TIMx_CR0[TxPSC] is written. Therefore, the frequency division coefficients shall be configured when Basic Timer is not working. The clock source frequency formula is: clk_pscx = TxCLK/(2^TxPSC). The clock rate corresponding to different TIMx_CR0[TxPSC] value as shown in Table 15-1.

Table 15-1 Mapping between Clock Rate and TIMx_CR0[TxPSC]

TIMx_CR0[TxPSC]	Division Factor	clk_pscx(Hz)	TIMx_CR0[TxPSC]	Division Factor	clk_pscx(Hz)
000	0x1	24M	100	0x10	1.5M
001	0x2	12M	101	0x20	750k
010	0x4	6M	110	0x40	375k
011	0x8	3M	111	0x80	187.5k

Note

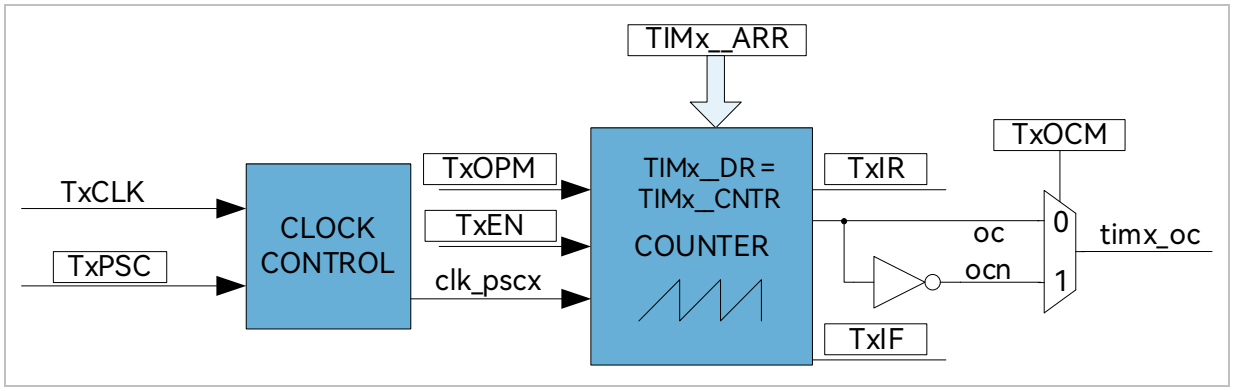
In Input Capture Mode of Timer3, the clock rate is 48MHz when TIM3_CR0[T3PSC] = 111

15.1.2 Reading, Writing and Counting of TIMx_CNTR

TIMx_CNTR starts when TIMx_CR1[TxEN] = 1. The write operation to TIMx_CNTR directly changes the value of the register, so it is required to disable the timer before the write operation. When reading TIMx_CNTR, the software reads high-order bytes first and then low-order bytes, and the hardware caches the low-order bytes simultaneously. When reading the low-order bytes, the software reads the cached data.

15.1.3 Output Mode

Figure 15-1 Output Mode Block Diagram



The output mode generates output signals according to TIMx_CR0[TxOCM], and the comparison results between TIMx_CNTR and registers TIMx_DR, TIMx_ARR. Meanwhile, corresponding interrupts is generated.

15.1.3.1 High-/Low-level Output Mode

When TIMx_CR0[TxOCM] = 0 and TIMx_DR > TIMx_ARR, the output signals are always low. When TIMx_CR0[TxOCM] = 1 and TIMx_DR > TIMx_ARR, the output signals are always high.

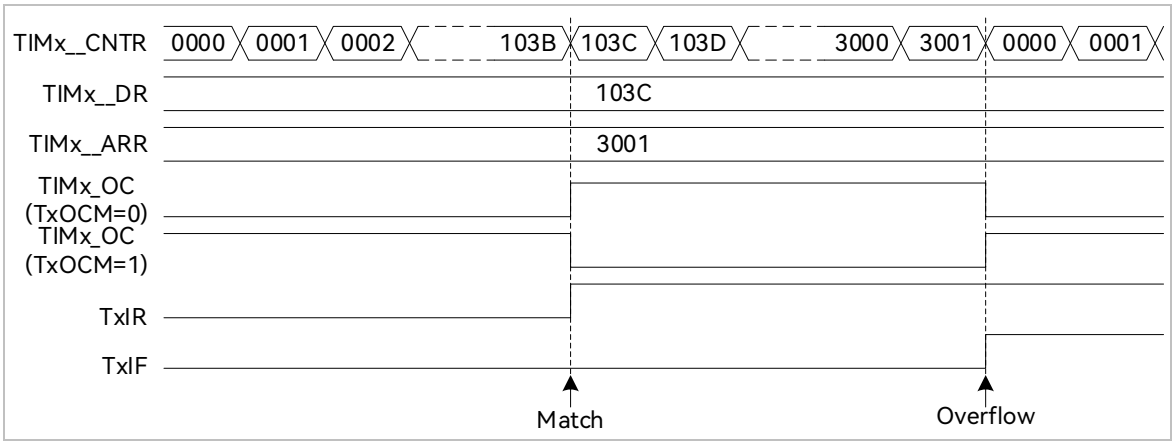
15.1.3.2 PWM Output

In PWM output mode, TIMx_ARR determines PWM cycle, and TIMx_DR determines the duty cycle, and duty cycle = TIMx_DR/TIMx_ARR*100%. If TIMx_CR0[TxOCM] = 0, the low level is output when TIMx_CNTR < TIMx_DR, and the high level is output when TIMx_CNTR ≥ TIMx_DR. If TIMx_CR0[TxOCM] = 1, the high level is output when TIMx_CNTR < TIMx_DR, and low level is output when TIMx_CNTR ≥ TIMx_DR. When TIMx_CNTR > TIMx_ARR, the output signal is reversed.

15.1.3.3 Interrupts

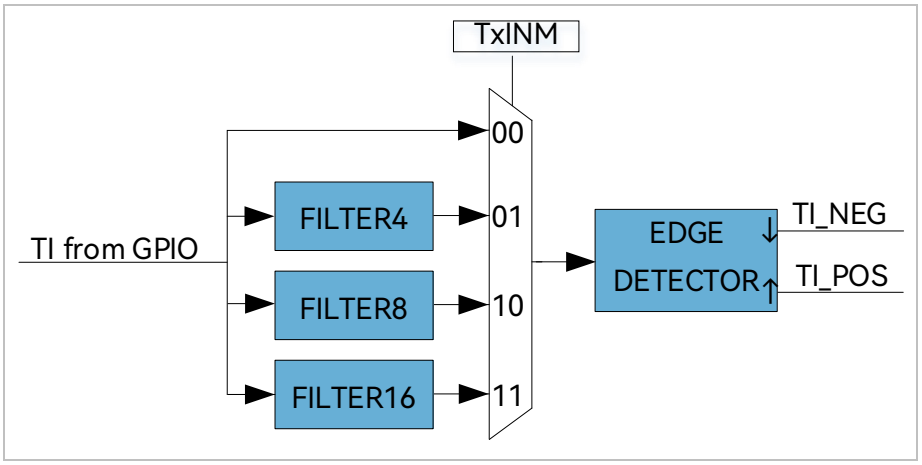
- When TIMx_CNTR = TIMx_DR, a compare match interrupt is generated. The interrupt flag TIMx_CR1[TxIR] is set to “1”, and the timer continues.
- When TIMx_CNTR = TIMx_ARR, an overflow event is generated. The interrupt flag TIMx_CR1[TxIF] is set to “1”, and the timer is cleared to “0”. TIMx_CR0[TxOPM] determines whether the timer recounts. The timer stops when TIMx_CR0[TxOPM]= 1, and restarts when TIMx_CR0[TxOPM]= 0.

Figure 15-2 Output Waveform of Output Mode



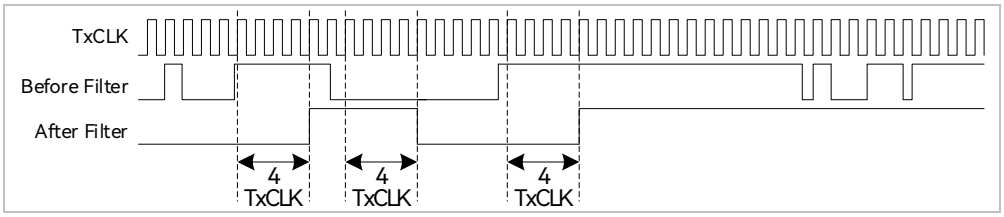
15.1.4 Input Signal Filtering and Edge Detection

Figure 15-3 Block Diagram of Input Signal Filtering and Edge Detection



The input signals of Timer3/Timer4 come from GPIO pin. TIMx_CR1[TxINM] is configured to disable the filtering circuit or filter out the input noise below 4/8/16 system clock cycles. The filtered signal is 4/8/16 system clock cycles delayed than the signal before filtering.

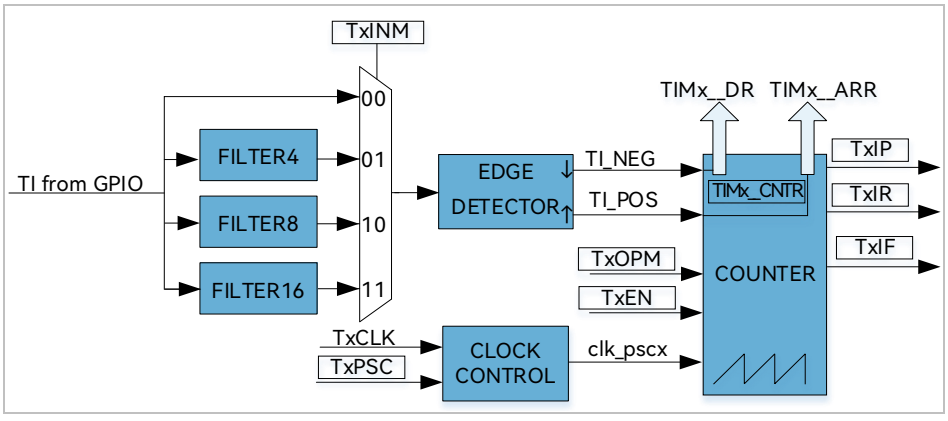
Figure 15-4 Timing Diagram of Filter Module



The edge detection module detects the filtered input signal from filtering module, and records the rising edge and falling edge for input capture mode.

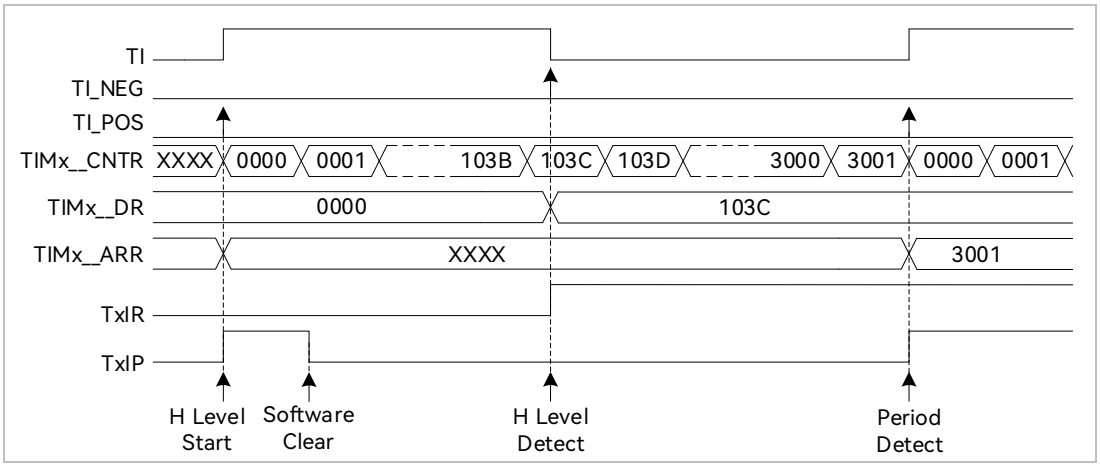
15.1.5 Input Capture Mode

Figure 15-5 Schematic Diagram of Input Capture Mode



The Input Capture Mode detects pulse width and waveform period of the input PWM signals. When TIMx_CR0[TxOCM] = 0, the time between two adjacent rising edges forms one cycle, and the time from rising edge to falling edge forms the pulse width (HIGH). When TIMx_CR0[TxOCM] = 1, the time between two adjacent falling edges forms one cycle, and the time from falling edge to rising edge forms the pulse width (LOW). The pulse width and the period obtained by TIMx_CNTR are stored in TIMx_DR and TIMx_ARR respectively.

Figure 15-6 Timing Diagram of Input Capture Mode (TIMx_CR0[TxOCM] = 0)



For example, when TIMx_CR0[TxOCM]= 0, TIMx_CR1[TxEN] is set to “1” to enable the timer. The Basic Timer is cleared to “0” and restarts when the first raising edge is detected. When the falling edge is detected, the value of TIMx_CNTR is stored into TIMx_DR. Meanwhile, the interrupt flag TIMx_CR1[TxIR] is set to “1”, and TIMx_CNTR continues to count. When the second rising edge is detected, the value of TIMx_CNTR is saved into TIMx_ARR. The interrupt flag TIMx_CR1[TxIP] is set to “1” and TIMx_CNTR is cleared to “0”. TIMx_CR0[TxOPM] determines whether the timer restarts. If TIMx_CR0[TxOPM] = 1, the timer stops; and if TIMx_CR0[TxOPM] = 0, it restarts.

An overflow event occurs if Timer3/Timer4 does not detect the second rising edge of the input and TIMx_CNTR reaches 0xFFFF. In this case, the interrupt flag bit TIMx_CR1[TxIF] is set to “1”, and TIMx_CNTR is cleared to “0”. TIMx_CR0[TxOPM] determines whether the timer restarts. If TIMx_CR0[TxOCM]= 1, the timer stops counting, and if TIMx_CR0[TxOPM] = 0, it restarts. At this point, TIMx_ARR is 0xFFFF, and TIMx_DR is determined by the input level and TIMx_CR0[TxOCM] XOR.

15.1.6 Timer4 FG Generation Mode

FG signal is generated by FOC module and Timer4. The FOC module calculates FGBASE using the formula: FGBASE = FBASE*32768*4/187500*X (where FBASE represents the frequency reference, X denotes the FG frequency scaling coefficient [which may be a floating-point value], and 187500 is Timer 4's prescaler value). The software computes the estimator speed OMEGA*FGBASE, right-shifts the 32-bit product by 8 bits, and extracts the 24 low-order bits as DELTA_THETA. The 16 high-order bits of DELTA_THETA are written to TIM4__ARR, while the 8 low-order bits are written to TIM4__DR[15:8]. When TIM4_CR0[T4FGM] is set to “1”, Timer 4 enters FG generation mode, where it outputs the FG signal upon counting overflow and sets the interrupt flag TIMx_CR1[T4IF] to “1”.

15.2 Timer3/Timer4 Registers

15.2.1 TIMx_CR0 (0x9C/0x9E) (x = 3/4)

Bit	7	6	5	4	3	2	1	0
Name	TxPSC			TxOCM	TxIRE	T4FGM	TxOPM	TxMOD
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:5]	TxPSC	Basic Timer Clock Prescaler Selection It is configured to divide the system clock frequency and generate the clock source for Basic Timer. The prescaled clock rates are configured as follows: 000: 24MHz 001: 12MHz 010: 6MHz 011: 3MHz 100: 1.5MHz 101: 750kHz 110: 375kHz 111: 187.5kHz Note In Input Capture Mode of Timer3, the clock rate is 48MHz when this bit is set to “111”.
[4]	TxOCM	Output Mode: Output Mode Selection 0: “0” is output when TIMx_CNTR < TIMx_DR; “1” is output when TIMx_CNTR ≥ TIMx_DR. 1: “1” is output when TIMx_CNTR < TIMx_DR; “0” is output when TIMx_CNTR ≥ TIMx_DR. Input Capture Mode: TIMx_DR indicates the input level to be selected when the active edge is detected or the timer becomes overflowed. Active Edge Selection 0: The time between two adjacent raising edges forms one cycle, and the time from rising edge to falling edge forms the pulse width (HIGH). 1: The time between two adjacent falling edges forms one cycle, and the time from falling edge to raising edge forms the pulse width (LOW). TIMx_DR indicates the input level to be selected when the timer becomes overflowed. 0: TIMx_DR is reset to “0” by hardware for low-level input upon an overflow interrupt and is set to “0xFFFF” by hardware for high-level input upon an overflow interrupt. 1: TIMx_DR is reset to “0” by hardware for high-level input upon an overflow interrupt and is set to “0xFFFF” for low-level input upon an overflow interrupt.

[3]	TxIRE	Output Mode: Compare Match Interrupt Enable Input Capture Mode: Pulse Width Detection Interrupt Enable 0: Disable 1: Enable
[2]	T4FGM	Timer4 FG Generation Mode Enable 0: Disable 1: Enable
[1]	TxOPM	Single Mode Basic Timer stops in any of the following events: Output Mode: Basic Timer overflow event Input Capture Mode: PWM Cycle Detection or Basic Timer overflow event 0: Basic Timer does not stop 1: Basic Timer stops (TIMx_CR1[TxEN] is reset to “0”)
[0]	TxMOD	Working Mode Selection 0: Input Capture Mode 1: Output Mode

15.2.2 TIMx_CR1 (0x9D/0x9F) (x = 3/4)

Bit	7	6	5	4	3	2	1	0
Name	TxIR	TxIP	TxIF	TxIPE	TxIFE	TxINM		TxEN
Type	R/W0	R/W0	R/W0	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	TxIR	Output Mode: Compare Match Interrupt Flag Input Capture Mode: Pulse Width Detection Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[6]	TxIP	Output Mode: No effect Input Capture Mode: PWM Cycle Detection Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[5]	TxIF	Output Mode: Basic Timer Overflow Interrupt Flag, which is set to”1” when TIMx_CNTR matches TIMx_ARR.

		Input Capture Mode: Basic Timer Overflow Interrupt Flag, which is set to “1” when the Timer does not detect an input PWM cycle but TIMx_CNTR reaches 0xFFFF. Timer4 FG Generation Mode: It is set to “1” when FG CNTR overflows. Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[4]	TxIPE	Output Mode: No effect Input Capture Mode: PWM Cycle Detection Interrupt Enable 0: Disable 1: Enable
[3]	TxIFE	Output Mode: Basic Timer Overflow Interrupt Input Capture Mode: Basic Timer Overflow Interrupt Enable Timer4 FG Generation Mode: FG CNTR Overflow Interrupt Enable 0: Disable 1: Enable
[2:1]	TxINM	Input Signal Filtering Pulse Width Selection Input signals are filtered as noise if pulse width is less than the defined value. 00: Disable 01: Filtered on every 4 system clock cycles 10: Filtered on every 8 system clock cycles 11: Filtered on every 16 system clock cycles
[0]	TxEN	Basic Timer Enable 0: Disable 1: Enable

15.2.3 TIMx_CNTR (0xA2, 0xA3/0x92, 0x93) (x = 3/4)

TIMx_CNTRH(0xA3/0x93)								
Bit	15	14	13	12	11	10	9	8
Name	TIMx_CNTR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIMx_CNTRL(0xA2/0x92)								
Bit	7	6	5	4	3	2	1	0
Name	TIMx_CNTR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	TIMx_CNTR	Count values held in Basic Timer

15.2.4 TIMx_DR (0xA4, 0xA5/0x94, 0x95) (x = 3/4)

TIMx_DRH(0xA5/0x95)								
Bit	15	14	13	12	11	10	9	8
Name	TIMx_DR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIMx_DRL(0xA4/0x94)								
Bit	7	6	5	4	3	2	1	0
Name	TIMx_DR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	TIMx_DR	Output Mode: Compare match values (written by software) Timer4 FG Generation Mode: 8 low-order bits of DELTA_THETA are written to 8 high-order bits of this register Input Capture Mode: Count value of the detected input pulse width (written by hardware)

15.2.5 TIMx_ARR (0xA6, 0xA7/0x96, 0x97) (x = 3/4)

TIMx_ARRH(0xA7/0x97)								
Bit	15	14	13	12	11	10	9	8
Name	TIMx_ARR[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
TIMx_ARRL(0xA6/0x96)								
Bit	7	6	5	4	3	2	1	0
Name	TIMx_ARR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	TIMx_ARR	Output Mode: Reload value (written by software) Timer4 FG Generation Mode: 16 high-order bits of DELTA_THETA are written to this register Input Capture Mode: Count value of a detected PWM cycle (written by hardware)

16 Systick

16.1 Systick Instructions

The chip generates Systick interrupts at a fixed interval, with the interrupt cycle set by SYST_CR[SYST_SEL]. Systick interrupt is enabled when SYST_CR[SYST_SEL] is not “00”, and the interrupt entry is accessed by P10.

16.2 Systick Registers

16.2.1 SYST_CR (0x4065)

Bit	7	6	5	4	3	2	1	0
Name	RSV						SYST_SEL	
Type	-	-	-	-	-	-	R/W	R/W
Reset	-	-	-	-	-	-	0	0

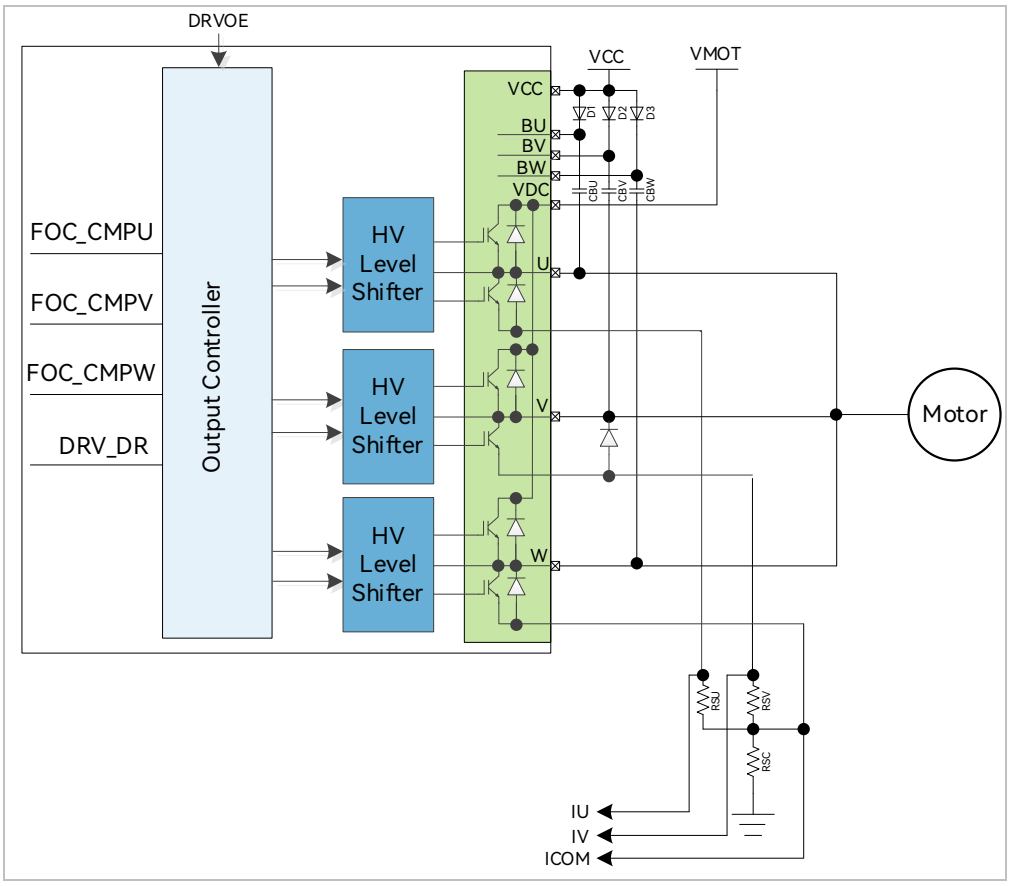
Bit	Name	Description
[7:2]	RSV	Reserved
[1:0]	SYST_SEL	Systick Interrupt Cycle 00: Disable 01: 0.25ms 10: 0.5ms 11: 1ms Note See DRV_SR (0x4061) for Systick interrupt flag

17 Driver

17.1 Instructions

17.1.1 FS95A6AS Driver Introduction

Figure 17-1 Block Diagram of FS95A6AS Driver Module



FOC_CMPU/V/W is the three-way comparison value output by FOC module, and DRV_DR is the comparison value set by the software. The above comparison value outputs three sets of level signals U/V/W to HV Level Shifter after passing through the output control module. The U/V/W three-way output is applied to DC brushless motor control

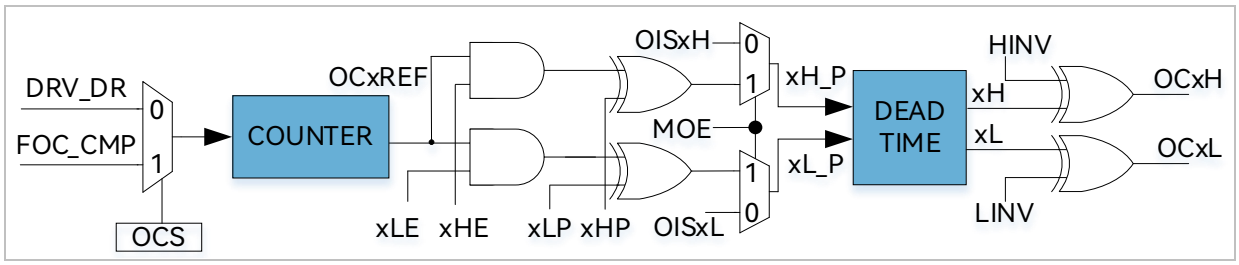
DRVOE is the enable signal for the HV Level Shifter, where the outputs are wired to 6 NMOS respectively to drive U/V/W-phases.

Table 17-1 Output Truth Values of Built-in HV Level Shifter

Input		Output	
UH/VH/WH	UL/VL/WL	HU/HV/HW	LU/LV/LW
L	L	L	L
L	H	L	H
H	L	H	L
H	H	L	L

17.1.2 Output Control Module

Figure 17-2 Block Diagram of Output Control Module



Before Driver module works, DRV_CR[MESEL] is set to “1” to select FOC mode or to “0” to select square-wave control mode.

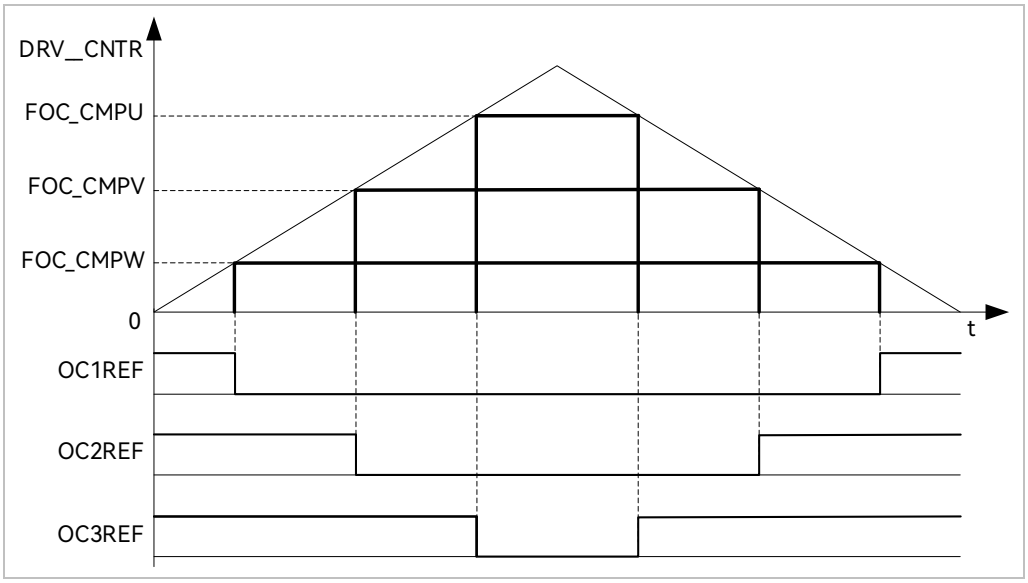
If DRV_CR[OCS] = 0, comparison value of PWM comes from DRV_DR. Otherwise, it comes from FOC_CMP and U/V/W/X output signals (OCxREF) are generated. DRV_CMR[xHE], DRV_CMR[xLE], polarity control bits DRV_CMR[xHP] and DRV_CMR[xLP] are configured for logic processing of OCxREF signal. Enabling DRV_OUT[MOE] outputs PWM waveform, otherwise, the idle level. xH_P and xL_P output signals are transferred to the deadtime module to generate xH and xL signals. PI_CR[HINV] and PI_CR[LINV] are configured to output PWM drive signals required by the high and low levels.

17.1.2.1 Count and Compare Module

DRV_CR[OCS] is configured to select the comparison value of PWM from FOC_CMPU/V/W of FOC module or DRV_DR set by software. The comparison value is sent to the counter for comparison to obtain the 3-phase original PWM signal OCxREF, and DRV_ DR is used for motor pre-charging, braking and square-wave control. If DRV_CNTR is smaller than the comparison value, OCxREF outputs high-level signal, and if DRV_CNTR is larger than DRV_DR, OCxREF outputs low-level signal.

When DRV_CR[OCS] = 1, FOC_CMPU/V/W is compared with the count value to generate the duty cycle OC1REF/OC2REF/OC3REF.

Figure 17-3 PMW Generation



When DRV_CR[OCS] = 0, DRV_DR set by software is compared with the count value to generate OC1REF/OC2REF/OC3REF with the same duty cycle. Duty cycle = DRV_DR/DRV_ARR*100%.

17.1.2.2 Enable and Polarity of Output Signals

The driver's high-side and low-side outputs are controlled by DRV_CMR[xHE] and DRV_CMR[xLE], and the output polarity by DRV_CMR[xHP] and DRV_CMR[xLP]. DRV_DR, DRV_ARR and DRV_CMR can be configured to implement functions, including pre-charging, brake, etc.

Figure 17-4 Pre-charge Waveform

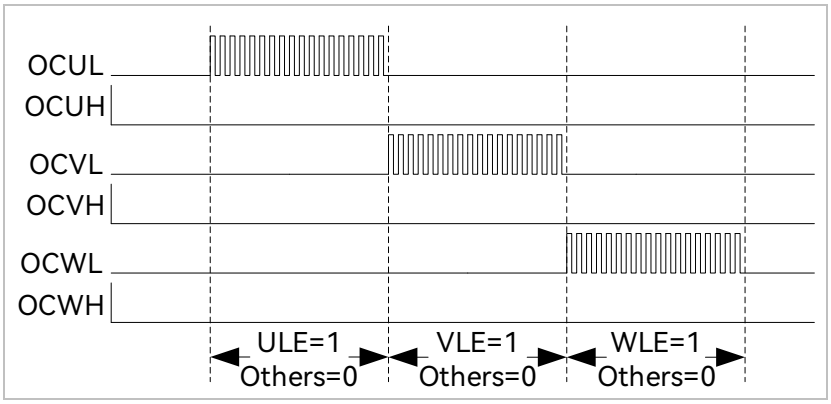
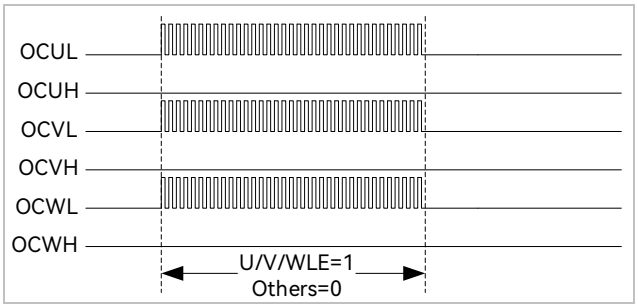
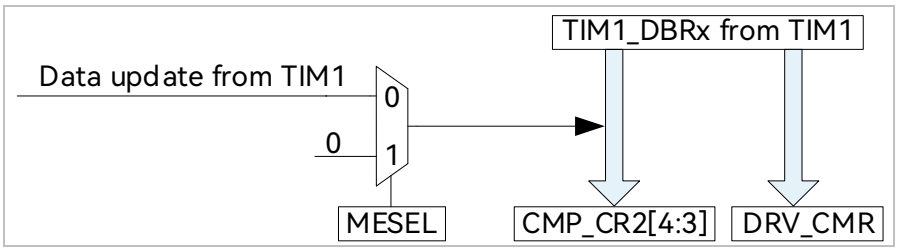


Figure 17-5 Brake Waveform



For square-wave control, Timer1 automatically controls DRV_CMR to implement phase commutation. Configuring DRV_CR[MESEL] = 0 enables the Square Wave Drive Mode. After Timer1 generates a write timing interrupt, the data stored in the corresponding TIM1_DBRx are transferred to DRV_CMR and CMP_CR2[4:3].

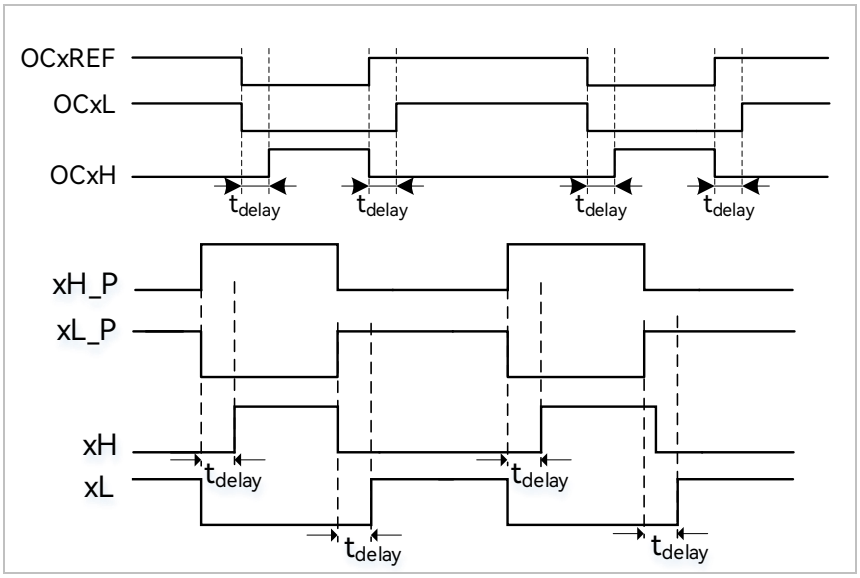
Figure 17-6 Timer1 Automatic Control of DRV_CMR and CMP_CR2[4:3]



17.1.2.3 Deadtime Module

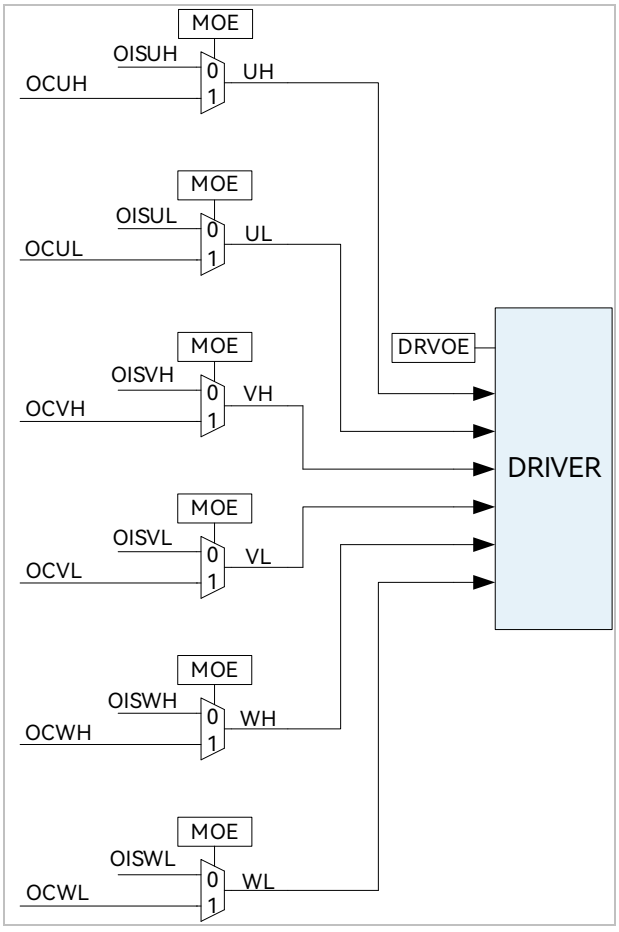
xH_P and xL_P signals are available for deadtime insertion. For complementary outputs, the deadtime insertion is enabled when DRV_DTR is not “0”. Each channel has an 8-bit deadtime generator, and three channels have the same deadtime, which is set by DRV_DTR. When rising edge signals are detected, output high level of xH and xL is delayed for a period of time set in DRV_DTR; if the delayed time is greater than the output pulse width, the associated channel pulse width is not delayed.

Figure 17-7 Complementary Outputs with Deadtime Insertion



17.1.2.4 Main Output Enable (MOE)

Figure 17-8 Block Diagram of Output Control Module



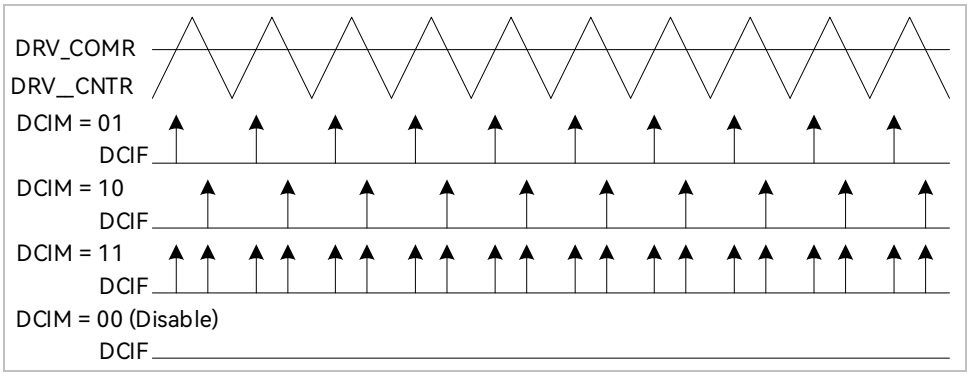
When DRV_OUT[MOE] is enabled, MOE module uses comparison value of the counter for motor control. When DRV_OUT[MOE] is disabled, the module outputs idle level set by the software to keep the motor at shutdown state.

17.1.2.5 Interrupts

17.1.2.5.1 Compare Match Interrupt

The generation conditions and time for compare match interrupt are configured by DRV_SR[DCIM] and DRV_COMR respectively. When the timer reaches the value set in DRV_COMR and the conditions set by DRV_SR[DCIM] are met, a compare match interrupt is generated and the interrupt flag DRV_SR[DCIF] is set to “1” by hardware.

Figure 17-9 Driver Compare Match Interrupt



17.1.2.5.2 FG Interrupt

FG interrupt is enabled when DRV_SR[FGIE] is set to “1”. The motor generates an interrupt for every electrical cycle.

17.2 Driver Registers

17.2.1 PI_CR (0xF9)

Bit	7	6	5	4	3	2	1	0
Name	RSV					DRV_MD	HINV	LINV
Type	-	-	-	-	-	R/W	R/W	R/W
Reset	-	-	-	-	-	0	0	0

Bit	Name	Description
[7:3]	RSV	Reserved
[2]	DRV_MD	Count Mode 0: Triangular Wave Mode 1: Sawtooth Wave Mode (FOC disabled)
[1]	HINV	High Side Reverse Enable 0: Disable 1: Enable
[0]	LINV	Low Side Reverse Enable 0: Disable 1: Enable

17.2.2 DRV_CR (0x4062)

Bit	7	6	5	4	3	2	1	0
Name	DRVEN	DDIR	FOCEN	DRPE	OCS	MESEL	RSV	DRVOE
Type	R/W	R/W	R/W	R/W	R/W	R/W	-	R/W
Reset	0	0	0	0	0	0	-	0

Bit	Name	Description
[7]	DRVEN	Counter Enable 0: Disable 1: Enable
[6]	DDIR	Output Direction (Forward/Reverse) This bit decides motor rotation directions, and is valid in both square-wave drive and FOC drive modes. In sensorless FOC mode, setting this bit changes motor rotation. In sensored FOC mode, it is also required to modify the angle by the software. In square-wave control mode, parameters related to Timer1 shall be configured. 0: Forward 1: Reverse
[5]	FOCEN	FOC Module Enable 0: Disable 1: Enable
[4]	DRPE	DRV_DR Pre-load Enable

		When preload is enabled, the data written to DRV_DR is updated after a timer underflow event occurs. When preload is disabled, the data written to DRV_DR is updated immediately. 0: Disable 1: Enable
[3]	OCS	Comparison Source Selection 0: DRV_DR 1: FOC_CMP
[2]	MESEL	ME Operating Mode Selection 0: Square Wave Drive 1: FOC Drive
[1]	RSV	Reserved
[0]	DRVOE	Driver Enable 0: Disable 1: Enable

17.2.3 DRV_SR (0x4061)

Bit	7	6	5	4	3	2	1	0
Name	SYSTIF	RSV	FGIF	DCIF	FGIE	DCIP	DCIM	
Type	R/W0	-	R/W0	R/W0	R/W	R/W	R/W	R/W
Reset	0	-	0	0	0	0	0	0

Bit	Name	Description
[7]	SYSTIF	Systick Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[6]	RSV	Reserved
[5]	FGIF	FG Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[4]	DCIF	Driver Compare Match Interrupt Flag When the Driver count value is equal to DRV_COMR, the system decides whether to generate an interrupt according to DRV_SR[DCIM]. Read:

		0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[3]	FGIE	FG Interrupt Enable After the interrupt feature is enabled, an FG Interrupt is generated in each electric cycle under FOC/square-wave control mode. 0: Disable 1: Enable
[2]	DCIP	Number of PWM cycles to generate a Compare Match Interrupt 0: 1 PWM cycle 1: 2 PWM cycles
[1:0]	DCIM	Compare Match Interrupt Mode Selection When the Driver count value is equal to DRV_COMR, the system decides whether to generate an interrupt according to DRV_SR[DCIM]. 00: No interrupt is generated. 01: An interrupt is generated when the timer counts up. 10: An interrupt is generated when the timer counts down. 11: An interrupt is generated when the timer counts up/down.

17.2.4 DRV_OUT (0xF8)

Bit	7	6	5	4	3	2	1	0
Name	MOE	RSV	OISWL	OISWH	OISVL	OISVH	OISUL	OISUH
Type	R/W	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	-	0	0	0	0	0	0

Bit	Name	Description
[7]	MOE	Main Output Enable This bit selects the sources for high and low sides of the driver of 3-phase output signals. It can be set to “1” and “0” by software. When bus current protection occurs, it is automatically cleared to “0” to turn off the output (see section 27.1.1.1). 0: Disable. The output sourced from the idle levels set by DRV_OUT[OISUH]/ DRV_OUT[OISVH]/DRV_OUT[OISWH] and DRV_OUT[OISUL]/DRV_OUT[OISVL]/ DRV_OUT[OISWL]. 1: Enable. The output sourced from the comparison value of the timer.
[6]	RSV	Reserved
[5]	OISWL	Output idle level of WL See descriptions on OISUH.
[4]	OISWH	Output idle level of WH See descriptions on OISUH.
[3]	OISVL	Output idle level of VL

		See descriptions on OISUH.
[2]	OISVH	Output idle level of VH See descriptions on OISUH.
[1]	OISUL	Output idle level of UL See descriptions on OISUH.
[0]	OISUH	Output idle level of UH This bit sets the UH output in IDLE state. When DRV_OUT[MOE] = 0, it outputs idle level to disable MOS. 0: Low level 1: High level

17.2.5 DRV_CMR (0x405C, 0x405D)

DRV_CMRH(0x405C)								
Bit	15	14	13	12	11	10	9	8
Name	RSV				WHP	WLP	VHP	VLP
Type	-	-	-	-	R/W	R/W	R/W	R/W
Reset	-	-	-	-	0	0	0	0
DRV_CMRL(0x405D)								
Bit	7	6	5	4	3	2	1	0
Name	UHP	ULP	WHE	WLE	VHE	VLE	UHE	ULE
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:12]	RSV	Reserved
[11]	WHP	High-side Polarity Control of Phase-W 0: Active High 1: Active Low
[10]	WLP	Low-side Polarity Control of Phase-W 0: Active High 1: Active Low
[9]	VHP	High-side Polarity Control of Phase-V 0: Active High 1: Active Low
[8]	VLP	Low-side Polarity Control of Phase-V 0: Active High 1: Active Low
[7]	UHP	High-side Polarity Control of Phase-U 0: Active High 1: Active Low
[6]	ULP	Low-side Polarity Control of Phase-U 0: Active High

		1: Active Low
[5]	WHE	High-side Output Enable of Phase-W 0: Disable 1: Enable
[4]	WLE	Low-side Output Enable of Phase-W 0: Disable 1: Enable
[3]	VHE	High-side Output Enable of Phase-V 0: Disable 1: Enable
[2]	VLE	Low-side Output Enable of Phase-V 0: Disable 1: Enable
[1]	UHE	High-side Output Enable of Phase-U 0: Disable 1: Enable
[0]	ULE	Low-side Output Enable of Phase-U 0: Disable 1: Enable

Note

- When DRV_CMR[W/V/ULE] and DRV_CMR[W/V/UHE] are set to "1", high-side and low-side outputs of W/V/U-phases are complementary to generate PWM signals with deadtime insertion
- For square-wave control, Timer1 automatically controls DRV_CMR register

17.2.6 DRV_ARR (0x405E, 0x405F)

DRV_ARRH(0x405E)								
Bit	15	14	13	12	11	10	9	8
Name	RSV		DRV_ARR[13:8]					
Type	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	-	0	0	0	0	0	0
DRV_ARRL(0x405F)								
Bit	7	6	5	4	3	2	1	0
Name	DRV_ARR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:14]	RSV	Reserved
[13:0]	DRV_ARR	Timer reload value, which determines PWM frequency (center-aligned) Driver timer up-counts from 0 to DRV_ARR/2 - 1 and an overflow event occurs. Then it down-counts to 0. Calculation formula: $f_{carrier}$ = SYSCLK * 2 / DRV_ARR

		DRV_ARR value is calculated using SYSCLK * 2, and falls within the range [0,16383] Note The LSB is always 0, and a write of “1” has no effect

17.2.7 DRV_COMR (0x405A, 0x405B)

DRV_COMRH(0x405A)								
Bit	15	14	13	12	11	10	9	8
Name	RSV				DRV_COMR[11:8]			
Type	-	-	-	-	R/W	R/W	R/W	R/W
Reset	-	-	-	-	0	0	0	0
DRV_COMRL(0x405B)								
Bit	7	6	5	4	3	2	1	0
Name	DRV_COMR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:12]	RSV	Reserved
[11:0]	DRV_COMR	Timer Compare Match Value The compare match interrupt is generated when the count value is equal to DRV_COMR and the conditions set in DRV_SR[DCIM] are met. The clock rate for the calculation is 12MHz. Duty cycle at the match point = DRV_COMR*4/DRV_ARR*100% DRV_COMR value is calculated using 12MHz clock rate, and falls within the range [0,4095]

17.2.8 DRV_DR (0x4058, 0x4059)

DRV_DRH(0x4058)								
Bit	15	14	13	12	11	10	9	8
Name	RSV		DRV_DR[13:8]					
Type	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	-	0	0	0	0	0	0
DRV_DRL(0x4059)								
Bit	7	6	5	4	3	2	1	0
Name	DRV_DR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description

[15:14]	RSV	Reserved
[13:0]	DRV_DR	PWM Duty Cycle Setting in Software Duty cycle = DRV_DR/DRV_ARR*100% DRV_DR value is calculated using SYSCLK * 2, and falls within the range [0,16383]. Note When this register is used as a comparison source, PWM is referenced to the high side of the driver and a deadtime is inserted in the complementary output of the low side of driver

17.2.9 DRV_DTR (0x4060)

Bit	7	6	5	4	3	2	1	0
Name	DRV_DTR							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	DRV_DTR	Deadtime Setting Deadtime = (DRV_DTR + 1)*T Example: If DRV_DTR is configured to “11”, then deadtime = 12*41.67ns = 500ns. Note If DRV_DTR is configured to “0”, deadtime insertion is disabled

17.2.10 DRV__CNTR (0x4066, 0x4067)

DRV__CNTRH(0x4066)								
Bit	15	14	13	12	11	10	9	8
Name	RSV				DRV__CNTR[11:8]			
Type	-	-	-	-	R/W	R/W	R/W	R/W
Reset	-	-	-	-	0	0	0	0
DRV__CNTRL(0x4067)								
Bit	7	6	5	4	3	2	1	0
Name	DRV__CNTR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:12]	RSV	Reserved
[11:0]	DRV__CNTR	Count Value of Timer The clock rate for the calculation is 12MHz. Driver duty cycle = DRV__CNTR*4/DRV_ARR*100%

		Range [0,4095]  Note The DRV_CNTR register is writable only when DRV_CR[DRVEN] = 1

18 WDT

The watchdog timer (WDT) is a timer that works on the internal slow clock to monitor the master program operation and prevent the MCU running out. Watchdog works as follows: After watchdog operates, WDT starts counting. When WDT overflows, watchdog sends a signal to reset the MCU and the program restarts running from address 0. During the operation of master program, WDT has to be initialized at regular intervals to prevent WDT overflowing.

After being enabled, WDT starts counting from 0. When it reaches 0xFFFC, watchdog outputs a signal that is 4 internal slow clock cycles wide to reset MCU, and the program starts running from address 0. WDT has to be initialized at regular intervals during operation, and WDT rolls over to WDT_ARR and restarts counting.

18.1 WDT Notes

- When MCU enters standby or sleep mode, WDT stops counting, but the count values are retained.
- WDT is automatically disabled during emulation.
- RST_SR[RSTWDT] is set to “1” when MCU is reset by WDT timer overflow.

18.2 WDT Operations

1. Set CCFG1[WDT_EN] = 1 to start WDT, which then starts counting from 0;
2. Set WDT_ARR (this operation can also be performed before starting WDT);
3. Set WDT_CR[WDTRF] = 1 in the running of program, and WDT rolls over to WDT_ARR setting.

18.3 WDT Registers

18.3.1 WDT_CR (0x4026)

Bit	7	6	5	4	3	2	1	0
Name	RSV							WDTRF
Type	-	-	-	-	-	-	-	R/W
Reset	-	-	-	-	-	-	-	0

Bit	Name	Description
[7:1]	RSV	Reserved
[0]	WDTRF	WDT Initialization 0: No effect 1: WDT rolls over to WDT_ARR setting and restarts counting.

18.3.2 WDT_ARR (0x4027)

Bit	7	6	5	4	3	2	1	0
Name	WDT_ARR							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	WDT_ARR	WDT Reload Timer This bit sets 8 high-order bits of the initialized value of WDT.

18.3.3 CCFG1 (0x401E)

Bit	7	6	5	4	3	2	1	0
Name	LVW_EN_B	LVW_IE	WDT_EN	RSV				
Type	R/W	R/W	R/W	-	-	-	-	-
Reset	0	0	0	-	-	-	-	-

Bit	Name	Description
[7]	LVW_EN_B	VCC Low-voltage Warning Enable 0: Disable 1: Enable
[6]	LVW_IE	LVW Detection Interrupt Enable 0: Disable 1: Enable
[5]	WDT_EN	WDT Enable 0: Disable 1: Enable
[4:0]	RSV	Reserved

19 Clock

19.1 Introduction

The clock consists of two modules: Internal Fast Clock and Internal Slow Clock. The system clock operates as an Internal Fast Clock. Internal Slow Clock is used as WDT for configuring overflow time of the watchdog.

19.2 Clock Calibration

Clock calibration is a feature that uses internal slow clock to calibrate the internal fast clock. Working principles: A 13-bit timer is used to count the length of 8 slow clock cycles with the fast clock as the clock source.

Calibration operations:

1. Set CAL_CR[CAL_STA] = 1 in software to start the calibration;
2. Read CAL_CR[CAL_BUSY] flag bit to check if the calibration process is completed;
3. When the calibration is completed (CAL_CR[CAL_BUSY] = 0), the readout of CAL_CR[CAL_ARR] is the value of the length of counting 8 slow clock cycles

19.3 Clock Calibration Registers

19.3.1 CAL_CR (0x4044, 0x4045)

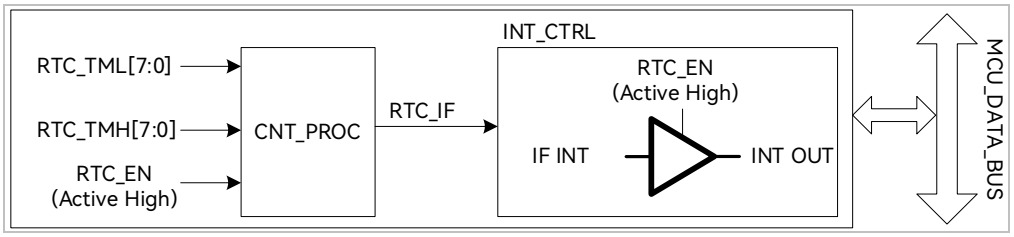
CAL_CRH(0x4044)								
Bit	15	14	13	12	11	10	9	8
Name	CAL_STA/ CAL_BUSY	RSV		CAL_ARR[12:8]				
Type	R/W1	-	-	R/W	R/W	R/W	R/W	R/W
Reset	1	-	-	0	0	0	0	0
CAL_CRL(0x4045)								
Bit	7	6	5	4	3	2	1	0
Name	CAL_ARR[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15]	CAL_STA/ CAL_BUSY	Clock Calibration Enable Read: 0: Calibration is completed. 1: Calibration is in progress. Write: 0: No effect 1: Clock Calibration starts.
[14:13]	RSV	Reserved
[12:0]	CAL_ARR	Calibration Counts The count values of the fast clock to continuously count eight slow clock cycles  Note When this value is 0, it indicates that no corresponding slow clock input exists, and when this value is 0x1FFF, it indicates that the count overflows (slow clock is too slow or fast clock is too fast)

20 RTC

20.1 RTC Functional Block Diagram

Figure 20-1 RTC Functional Block Diagram



20.2 RTC Operations

A write to RTC_TM sets RTC reload value. RTC is enabled when RTC_STA[RTC_EN] is set to “1”.

20.3 RTC Registers

20.3.1 RTC_TM (0x402C, 0x402D)

RTC_TMH(0x402C)								
Bit	15	14	13	12	11	10	9	8
Name	RTC_TM[15:8]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1
RTC_TML(0x402D)								
Bit	7	6	5	4	3	2	1	0
Name	RTC_TM[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Bit	Name	Description
[15:0]	RTC_TM	RTC Count Register Read: Instantaneous value of the timer Write: RTC timer up-counts at a rate of 32768Hz from 0 to the written value and becomes overflowed. Meanwhile, an interrupt request is generated, and the timer is cleared and restarts counting.

20.3.2 RTC_STA (0x402E)

Bit	7	6	5	4	3	2	1	0
Name	RTC_EN	RTC_IF	SCKSEL	ISOSCEN	RSV			
Type	R/W	R/W0	R/W	R/W	-	-	-	-
Reset	0	0	0	0	-	-	-	-

Bit	Name	Description
[7]	RTC_EN	RTC Enable 0: Disable 1: Enable
[6]	RTC_IF	RTC Interrupt Flag This bit is set to “1” when the timer value matches RTC_TM setting. Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[5]	SCKSEL	Slow Clock Source Selection 0: Internal Slow Clock

		1: External Slow Clock
[4]	ISOSCEN	Internal Slow Clock Enable 0: Disable 1: Enable
[3:0]	RSV	Reserved

21 IO

21.1 IO Introduction

FS95A6AS has up to 18 GPIO pins, including P0.3, P0.5 ~ P0.6, P1.2 ~ P1.7, P2.1 ~ P2.2, P2.4 ~ P2.7, P3.0 ~ P3.1 and P4.7.

21.2 IO Operations

Each GPIO port pin has relevant registers to meet different application requirements. For example, P0.0 is mapped to register P0, and P1.0 to register P1. P0_OE and P1_OE registers are configured for digital input and output.

- P4.7 works in FICEK mode by default.
- P4.7 does not support external reset in IO mode.
- The signal frequency on P4.7 shall be below 100kHz.
- The enable bits of pull-up resistors and pull-down resistors are configured to “1”. See 21.3.8 P0_PU (0x4053) ~ 21.3.13 PX_PL (0x4048) for port pins and registers.
- See 5.4 GPIO Electrical Characteristics for the values of pull-up resistors and pull-down resistors.
- The relevant bits of P1_AN, P2_AN and P3_AN registers are configured to “1” to activate analog signal mode. See 21.3.5 P1_AN (0x4050) ~ 21.3.7 P3_AN (0x4052) for port pins and registers. After the port pins are configured to analog mode, all their digital features are disabled and the port state is 0 by reading relevant bits in P1, P2 and P3 registers.
- Pull-up resistors of P1.2 ~ P1.7, P2.1 ~ P2.2, P2.4 ~ P2.7 and P3.0 ~ P3.1 are automatically disabled when the port pins are configured as analog mode.

21.3 IO Registers

21.3.1 P0_OE (0xFC)

Bit	7	6	5	4	3	2	1	0
Name	RSV	P06_OE	P05_OE	RSV	P03_OE	RSV		
Type	-	R/W	R/W	-	R/W	-	-	-
Reset	-	0	0	-	0	-	-	-

Bit	Name	Description
[7]	RSV	Reserved
[6]	P06_OE	P0.6 Digital I/O Selection 0: Input 1: Output
[5]	P05_OE	P0.5 Digital I/O Selection 0: Input 1: Output
[4]	RSV	Reserved
[3]	P03_OE	P0.3 Digital I/O Selection 0: Input 1: Output
[2:0]	RSV	Reserved

21.3.2 P1_OE (0xFD)

Bit	7	6	5	4	3	2	1	0
Name	P1_OE						RSV	
Type	R/W	R/W	R/W	R/W	R/W	R/W	-	-
Reset	0	0	0	0	0	0	-	-

Bit	Name	Description
[7:2]	P1_OE	P1.7 ~ P1.2 Digital I/O Selection 0: Input 1: Output
[1:0]	RSV	Reserved

21.3.3 P2_OE (0xFE)

Bit	7	6	5	4	3	2	1	0
Name	P27_OE	P26_OE	P25_OE	P24_OE	RSV	P22_OE	P21_OE	RSV
Type	R/W	R/W	R/W	R/W	-	R/W	R/W	-
Reset	0	0	0	0	-	0	0	-

Bit	Name	Description

[7]	P27_OE	P2.7 Digital I/O Selection 0: Input 1: Output
[6]	P26_OE	P2.6 Digital I/O Selection 0: Input 1: Output
[5]	P25_OE	P2.5 Digital I/O Selection 0: Input 1: Output
[4]	P24_OE	P2.4 Digital I/O Selection 0: Input 1: Output
[3]	RSV	Reserved
[2]	P22_OE	P2.2 Digital I/O Selection 0: Input 1: Output
[1]	P21_OE	P2.1 Digital I/O Selection 0: Input 1: Output
[0]	RSV	Reserved

21.3.4 P3_OE (0xFF)

Bit	7	6	5	4	3	2	1	0
Name	RSV					P3_OE		
Type	-	-	-	-	-	-	R/W	R/W
Reset	-	-	-	-	-	-	0	0

Bit	Name	Description
[7:2]	RSV	Reserved
[1:0]	P3_OE	P3.1 ~ P3.0 Digital I/O Selection 0: Input 1: Output

21.3.5 P1_AN (0x4050)

Bit	7	6	5	4	3	2	1	0
Name	P1_AN					RSV		HDIO
Type	R/W	R/W	R/W	R/W	R/W	-	-	R/W
Reset	0	0	0	0	0	-	-	0

Bit	Name	Description
[7:3]	P1_AN	P1.7 ~ P1.3 Analog Mode Enable 0: Disable 1: Enable

[2:1]	RSV	Reserved
[0]	HDIO	IO Driver Capability for PWM Output 0: Normal drive capability 1: High drive capability

21.3.6 P2_AN (0x4051)

Bit	7	6	5	4	3	2	1	0
Name	P27_AN	P26_AN	P25_AN	P24_AN	RSV	P22_AN	P21_AN	RSV
Type	R/W	R/W	R/W	R/W	-	R/W	R/W	-
Reset	0	0	0	0	-	0	0	-

Bit	Name	Description
[7]	P27_AN	P2.7 Analog Mode Enable 0: Disable 1: Enable
[6]	P26_AN	P2.6 Analog Mode Enable 0: Disable 1: Enable
[5]	P25_AN	P2.5 Analog Mode Enable 0: Disable 1: Enable
[4]	P24_AN	P2.4 Analog Mode Enable 0: Disable 1: Enable
[3]	RSV	Reserved
[2]	P22_AN	P2.2 Analog Mode Enable 0: Disable 1: Enable
[1]	P21_AN	P2.1 Analog Mode Enable 0: Disable 1: Enable
[0]	RSV	Reserved

21.3.7 P3_AN (0x4052)

Bit	7	6	5	4	3	2	1	0
Name	RSV						P3_AN	
Type	-	-	-	-	-	-	R/W	R/W
Reset	-	-	-	-	-	-	0	0

Bit	Name	Description
[7:2]	RSV	Reserved

[1:0]	P3_AN	P3.1 ~ P3.0 Analog Mode Enable 0: Disable 1: Enable

21.3.8 P0_PU (0x4053)

Bit	7	6	5	4	3	2	1	0
Name	RSV	P06_PU	P05_PU	RSV	P03_PU	RSV		
Type	-	R/W	R/W	-	R/W	-	-	-
Reset	-	0	0	-	0	-	-	-

Bit	Name	Description
[7]	RSV	Reserved
[6]	P06_PU	P0.6 Pull-up Resistor Enable 0: Disable 1: Enable
[5]	P05_PU	P0.5 Pull-up Resistor Enable 0: Disable 1: Enable
[4]	RSV	Reserved
[3]	P03_PU	P0.3 Pull-up Resistor Enable 0: Disable 1: Enable
[2:0]	RSV	Reserved

21.3.9 P1_PU (0x4054)

Bit	7	6	5	4	3	2	1	0
Name	P1_PU						RSV	
Type	R/W	R/W	R/W	R/W	R/W	R/W	-	-
Reset	0	0	0	0	0	0	-	-

Bit	Name	Description
[7:2]	P1_PU	P1.7 ~ P1.2 Pull-up Resistor Enable 0: Disable 1: Enable
[1:0]	RSV	Reserved

21.3.10 P2_PU (0x4055)

Bit	7	6	5	4	3	2	1	0
Name	P27_PU	P26_PU	P25_PU	P24_PU	RSV	P22_PU	P21_PU	RSV
Type	R/W	R/W	R/W	R/W	-	R/W	R/W	-
Reset	0	0	0	0	-	0	0	-

Bit	Name	Description
[7]	P27_PU	P2.7 Pull-up Resistor Enable 0: Disable 1: Enable
[6]	P26_PU	P2.6 Pull-up Resistor Enable 0: Disable 1: Enable
[5]	P25_PU	P2.5 Pull-up Resistor Enable 0: Disable 1: Enable
[4]	P24_PU	P2.4 Pull-up Resistor Enable 0: Disable 1: Enable
[3]	RSV	Reserved
[2]	P22_PU	P2.2 Pull-up Resistor Enable 0: Disable 1: Enable
[1]	P21_PU	P2.1 Pull-up Resistor Enable 0: Disable 1: Enable
[0]	RSV	Reserved

21.3.11 P3_PU (0x4056)

Bit	7	6	5	4	3	2	1	0
Name	RSV						P3_PU	
Type	-	-	-	-	-	-	R/W	R/W
Reset	-	-	-	-	-	-	0	0

Bit	Name	Description
[7:2]	RSV	Reserved
[1:0]	P3_PU	P3.1 ~ P3.0 Pull-up Resistor Enable 0: Disable 1: Enable

21.3.12 P4_PU (0x4057)

Bit	7	6	5	4	3	2	1	0
Name	P47_PU	RSV						
Type	R/W	-	-	-	-	-	-	-
Reset	0	-	-	-	-	-	-	-

Bit	Name	Description

[7]	P47_PU	P47 Pull-up Resistor Enable 0: Disable 1: Enable
[6:0]	RSV	Reserved

21.3.13 PX_PL (0x4048)

Bit	7	6	5	4	3	2	1	0
Name	P47_PL	P06_PL	RSV					
Type	R/W	R/W	-	-	-	-	-	-
Reset	0	0	-	-	-	-	-	-

Bit	Name	Description
[7]	P47_PL	P4.7 Pull-down Resistor Enable 0: Disable 1: Enable
[6]	P06_PL	P0.6 Pull-down Resistor Enable 0: Disable 1: Enable
[5:0]	RSV	Reserved

21.3.14 PH_SEL (0x404C)

Bit	7	6	5	4	3	2	1	0
Name	RSV	UART1EN	UART2EN	T4SEL	T3SEL	T2SEL	RSV	
Type	-	R/W	R/W	R/W	R/W	R/W	-	-
Reset	-	0	0	0	0	0	-	-

Bit	Name	Description
[7]	RSV	Reserved
[6]	UART1EN	Port pins multiplexed as RXD, TXD and UART1 enabled 0: Disable 1: P0.6 and P0.5 multiplexed as RXD and TXD, and UART1 enabled
[5]	UART2EN	Port pins multiplexed as RXD2, TXD2 and UART2 enabled 0: Disable 1: P1.4 and P1.2 multiplexed as TXD2; P4.7 and P1.3 multiplexed as RXD2; UART2 enabled
[4]	T4SEL	Port pins multiplexed as Timer4 or Timer4S 0: Disable 1: P0.5 or P1.2 multiplexed as Timer4 I/O pins
[3]	T3SEL	Port pins multiplexed as Timer3 or Timer3S 0: Disable 1: P0.6 multiplexed as Timer3 I/O pins or P4.7 multiplexed as Timer3 input

[2]	T2SEL	Port pins multiplexed as Timer2 0: Disable 1: P0.3 multiplexed as Timer2 I/O pins
[1:0]	RSV	Reserved

21.3.15 PH_SEL1 (0x404D)

Bit	7	6	5	4	3	2	1	0
Name	UART2CH	RSV			T4CT1	T4CT0	T3CT1	T3CT0
Type	R/W	-	-	-	R/W	R/W	R/W	R/W
Reset	0	-	-	-	0	0	0	0

Bit	Name	Description
[7]	UART2CH	UART2 Function Switching 0: P1.3 serving as RXD, and P1.4 as TXD (P1.3 is an I/O pin of single-wire mode) 1: P4.7 serving as RXD, and P1.2 as TXD (P1.2 is an I/O pin of single-wire mode)
[6:4]	RSV	Reserved
[3:2]	T4CT	Timer4 Function Switching 0: P0.5 as Timer4 I/O pin 1: P1.2 as Timer4 I/O pin
[1:0]	T3CT	Timer3 Function Switching 0: P0.6 as Timer3 I/O pin 1: P4.7 as Timer3 input pin

21.3.16 P0 (0x80)

Port output register P0/1/2/3/4/5 supports read and write access. RMW commands are used to access the register value (see Table 21-1 for RMW commands), and other commands are used to access PORT pin.

Bit	7	6	5	4	3	2	1	0
Name	RSV	GP06	GP05	RSV	GP03	RSV		
Type	-	R/W	R/W	-	R/W	-	-	-
Reset	-	0	0	-	0	-	-	-

Bit	Name	Description
[7]	RSV	Reserved
[6]	GP06	GP06 Pin
[5]	GP05	GP05 Pin
[4]	RSV	Reserved
[3]	GP03	GP03 Pin
[2:0]	RSV	Reserved

21.3.17 P1 (0x90)

Bit	7	6	5	4	3	2	1	0
Name	GP17	GP16	GP15	GP14	GP13	GP12	RSV	
Type	R/W	R/W	R/W	R/W	R/W	R/W	-	-
Reset	0	0	0	0	0	0	-	-

Bit	Name	Description
[7]	GP17	GP17 Pin
[6]	GP16	GP16 Pin
[5]	GP15	GP15 Pin
[4]	GP14	GP14 Pin
[3]	GP13	GP13 Pin
[2]	GP12	GP12 Pin
[1:0]	RSV	Reserved

21.3.18 P2 (0xA0)

Bit	7	6	5	4	3	2	1	0
Name	GP27	GP26	GP25	GP24	RSV	GP22	GP21	RSV
Type	R/W	R/W	R/W	R/W	-	R/W	R/W	-
Reset	0	0	0	0	-	0	0	-

Bit	Name	Description
[7]	GP27	GP27 Pin
[6]	GP26	GP26 Pin
[5]	GP25	GP25 Pin
[4]	GP24	GP24 Pin
[3]	RSV	Reserved
[2]	GP22	GP22 Pin
[1]	GP21	GP21 Pin
[0]	RSV	Reserved

21.3.19 P3 (0xB0)

Bit	7	6	5	4	3	2	1	0
Name	RSV						GP31	GP30
Type	-	-	-	-	-	-	R/W	R/W
Reset	-	-	-	-	-	-	0	0

Bit	Name	Description
[7: 2]	RSV	Reserved
[1]	GP31	GP31 Pin

[0]	GP30	GP30 Pin

21.3.20 P4 (0xB8)

Bit	7	6	5	4	3	2	1	0
Name	GP47	RSV						
Type	R/W	-	-	-	-	-	-	-
Reset	0	-	-	-	-	-	-	-

Bit	Name	Description
[7]	GP47	GP47 Pin
[6:0]	RSV	Reserved

Table 21-1 RMW Commands

Command	Descriptions
ANL	Bitwise logical AND operation
ORL	Bitwise logical OR operation
XRL	Bitwise logical XOR operation
JBC	Jump if the bit is set to “1” and then cleared to “0”
CPL	Bitwise logical converse operation
INC, DEC	+1, -1 logical operation
DJNZ	Jump if the bit is not “0”
MOV Px, y, C	Assign carry bit C to Px, y
CLR Px, y	Px, y is cleared to “0”
SETB Px, y	Px, y is set to “1”

22 ADC

22.1 ADC Introduction

ADC module is a 12-bit successive approximation register ADC with 11 channels inside. The sampling mode supports sequential sampling (that is, from ADC Channel 0 to ADC channel 10 in sequence) and triggered sampling (including FOC triggered sampling mode and Timer1 triggered sampling mode). The results of sequential sampling are stored in ADCx_DR (x = 0 ~ 10) in a right-aligned or left-second-high-aligned format. The result of triggered sampling is sent to FOC module or Timer1 module instead of ADCx_DR for motor control. The relevant registers of the FOC module or Timer1 module are always left-second-high-aligned to store the triggered sample results. Triggered sampling is done automatically by hardware, and sequential sampling is controlled by software. The priority of triggered sampling is higher than that of sequential sampling. If both triggered sampling and sequential sampling are applied at the same time, the triggered sampling is performed first, and ADC automatically restores sequential sampling mode upon completion of triggered sampling.

The clock source for ADC sampling is 12MHz and the sampling time is set by DAC_CR[5:2] and ADC_SCYC. See ADC Electrical Characteristics for sample time and conversion time.

22.2 ADC Block Diagram

Figure 22-1 ADC Multiplexer Block Diagram

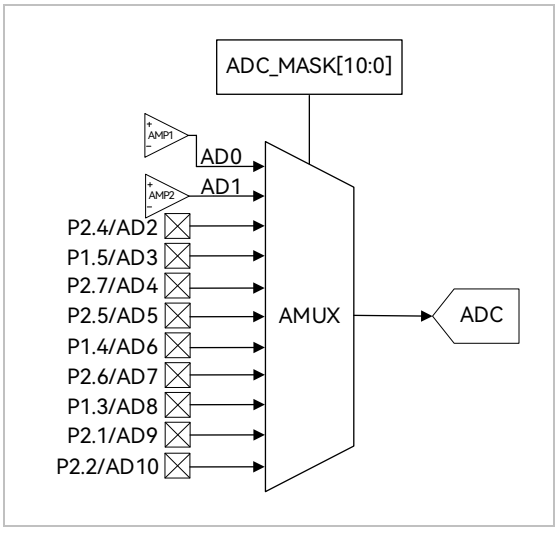
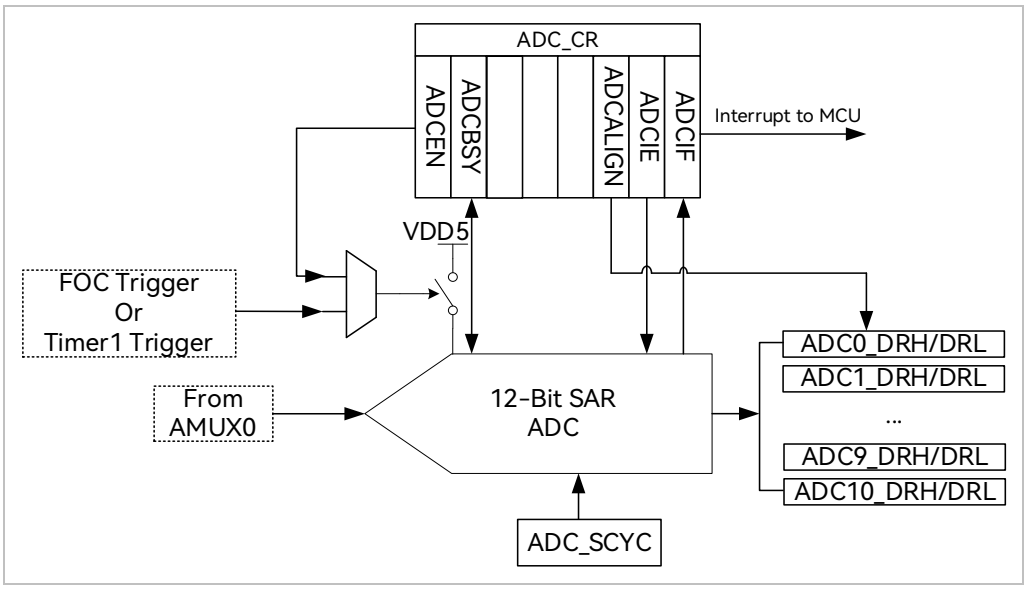


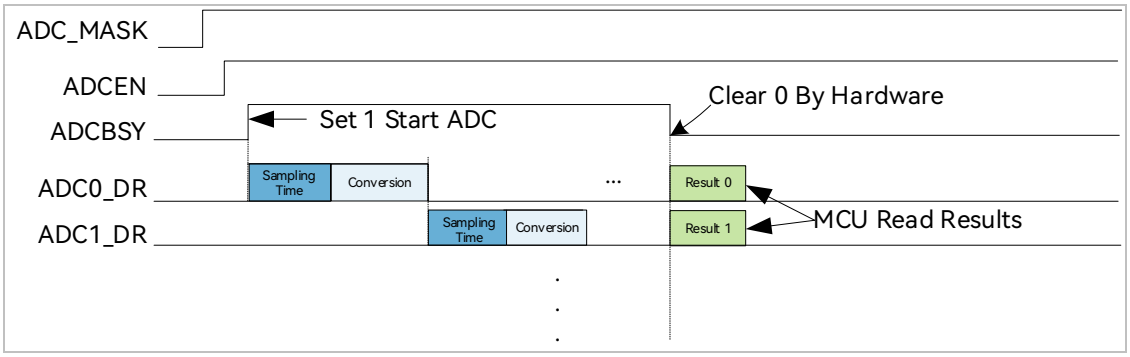
Figure 22-2 ADC Functional Block Diagram



22.3 ADC Operations

22.3.1 Sequential Sampling Mode

Figure 22-3 ADC Sequential Sampling Timing



ADC operations:

1. Configure ADC_MASK to enable the corresponding channel required to sample;
2. Configure ADC_SCYC to select the sampling period of each channel;
3. Configure ADC_CR[ADCEN] = 1 to enable ADC;
4. Configure ADC_CR[ADCBSY] = 1 to start ADC;
5. Read ADCx_DR to obtain ADC results when ADC_CR[ADCBSY] = 0.

Note

The ADC conversion sequence is from low to high based on the enabled channel (i.e., when channel 2/3/4 is enabled, the signal is sampled in order of 2/3/4, and a single conversion result is read after confirming ADC_CR[ADCBSY] = 0)

22.3.2 Triggered Sampling Mode

When FOC module is enabled, ADC channel 0/1/2/4 can be used to FOC triggered sampling. Channel 2 is used for bus voltage triggered sampling. In single-shunt current sampling mode, channel 4 is used for itrip sampling. In dual-shunt current sampling mode, channel 0 is used for ia sampling and channel 1 for ib sampling. In triple-shunt current sampling mode, channel 0 is used for ia sampling, channel 1 for ib sampling, and channel 4 for ic sampling. In PFC mode, channel 0 is used for current sampling.

When Timer1 is enabled, channel 4 is used for bus current sampling. TIM1_CR3[T1TIS] is configured to select ADC as the input source of position detection. In this mode, channel 3 samples phase-U voltage, channel 6 samples phase-V voltage, and channel 8 samples phase-W voltage. Channel 7 can be selected for averaged bus current sampling when AMP0_CR0[CP_EN] is set to “1” (with a 1μF capacitor connected to P2.6). Channels 9 and 10 are reserved for general-purpose applications.

22.3.3 Output Data Format

Registers ADCx_DRH and ADCx_DRL contain the high-order bits and the low-order bits of ADC sampling results. Data can be right-aligned or left-second-high-aligned by configuring ADC_CR[ADCALIGN]. When input voltage ranges from 0 to VDD5, the relation between the input voltage and result data is shown in Table 22-1. The bits, which are not used in ADCx_DRH and ADCx_DRL, are set to “0”.

Table 22-1 Relation between Input Voltage and Result Data

Input Voltage	Right-aligned	Left-second-high-aligned
0	0x0000	0x0000
VDD5/2	0x0800	0x4000
VDD5	0x0FFF	0x7FF8

22.4 ADC Registers

22.4.1 ADC_CR (0x4039)

Bit	7	6	5	4	3	2	1	0
Name	ADCEN	ADCBSY	RSV			ADCALIGN	ADCIE	ADCIF
Type	R/W	R/W1	-	-	-	R/W	R/W	R/W0
Reset	0	0	-	-	-	0	0	0

Bit	Name	Description
[7]	ADCEN	ADC Enable 0: Disable 1: Enable
[6]	ADCBSY	ADC Start & ADC Busy Flag Read: 0: ADC Idle 1: ADC Busy Write: 0: No effect 1: ADC conversion starts Note Writing “1” to this bit has no effect when ADC_MASK = 0
[5:3]	RSV	Reserved
[2]	ADCALIGN	ADC Data Format Selection 0: ADC output is right-aligned, and ADC result = ADCx_DR[11:0] 1: ADC output is left-second-high-aligned, and ADC result = ADCx_DR[14:3] Note The results of triggered sampling mode are always left-second-high-aligned
[1]	ADCIE	ADC Interrupt Enable (excluding triggered sampling mode interrupt) 0: Disable 1: Enable
[0]	ADCIF	ADC Interrupt Flag This bit is set to “1” by hardware when ADC conversion is completed. Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect

22.4.2 ADC_MASK (0x4036, 0x4037)

ADC_MASKH(0x4036)								
Bit	15	14	13	12	11	10	9	8
Name	ADC_SCYC[15:12]				RSV	CH10EN	CH9EN	CH8EN
Type	R/W	R/W	R/W	R/W	-	R/W	R/W	R/W
Reset	0	0	1	1	-	0	0	0
ADC_MASKL(0x4037)								
Bit	7	6	5	4	3	2	1	0
Name	CH7EN	CH6EN	CH5EN	CH4EN	CH3EN	CH2EN	CH1EN	CH0EN
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:12]	ADC_SCYC[15:12]	ADC Sampling Cycle for ADC Channel 9 ~ 10 ADC_SCYCH[3] = 0: Sampling cycle is ADC_SCYCH[2:0] ADC clock cycles. ADC_SCYCH[3] = 1: Sampling cycle is (ADC_SCYCH[2:0]*8 + 7) ADC clock cycles
[11]	RSV	Reserved
[10]	CH10EN	ADC Channel 10 Enable
[9]	CH9EN	ADC Channel 9 Enable
[8]	CH8EN	ADC Channel 8 Enable
[7]	CH7EN	ADC Channel 7 Enable
[6]	CH6EN	ADC Channel 6 Enable
[5]	CH5EN	ADC Channel 5 Enable
[4]	CH4EN	ADC Channel 4 Enable
[3]	CH3EN	ADC Channel 3 Enable
[2]	CH2EN	ADC Channel 2 Enable
[1]	CH1EN	ADC Channel 1 Enable
[0]	CH0EN	ADC Channel 0 Enable

Note

In triggered sampling mode, it is not required to configure ADC_MASK

22.4.3 DAC_CR (0x4035)

Bit	7	6	5	4	3	2	1	0
Name	DAC0_1EN	DACMOD	ADC_SCYCH[11:8]				DAC2EN	RSV
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	-
Reset	0	0	0	0	1	1	0	-

Bit	Name	Description
[7]	DAC0_1EN	See section DAC_CR (0x4035) in chapter DAC

[6]	DACMOD	See section DAC_CR (0x4035) in chapter DAC
[5:2]	ADC_SCYCH[11:8]	ADC Sampling Cycle for ADC Channel 3, 6 and 8 ADC_SCYCH[3] = 0: Sampling cycle is ADC_SCYCH[2:0] ADC clock cycles. ADC_SCYCH[3] = 1: Sampling cycle is (ADC_SCYCH[2:0]*8 + 7) ADC clock cycles.
[1]	DAC2EN	See section DAC_CR (0x4035) in chapter DAC
[0]	RSV	Reserved

22.4.4 ADC_SCYC (0x4038)

Bit	7	6	5	4	3	2	1	0
Name	ADC_SCYC[7:4]				ADC_SCYC[3:0]			
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	1	1	0	0	1	1

Bit	Name	Description
[7:4]	ADC_SCYC[7:4]	ADC Sampling Cycle for ADC Channel 2 and 5 ADC_SCYC[7] = 0: Sampling cycle is ADC_SCYC[6:4] ADC clock cycles. ADC_SCYC[7] = 1: Sampling cycle is (ADC_SCYC[6:4]*8 + 7) ADC clock cycles.
[3:0]	ADC_SCYC[3:0]	ADC Sampling Cycle for ADC Channel 0, 1, 4 and 7 ADC_SCYC[3] = 0: Sampling cycle is ADC_SCYC[2:0] ADC clock cycles. ADC_SCYC[3] = 1: Sampling cycle is (ADC_SCYC[2:0]*8 + 7) ADC clock cycles.

22.4.5 ADC0_DR (0x0FD8, 0x0FD9)

ADC0_DRH(0x0FD8)								
Bit	15	14	13	12	11	10	9	8
Name	ADC0_DR[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
ADC0_DRL(0x0FD9)								
Bit	7	6	5	4	3	2	1	0
Name	ADC0_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	ADC0_DR	The conversion results of ADC channel 0 upon completion of ADC conversion in the Sequential Sampling Mode. The data is aligned according to ADC_CR[ADCALIGN]. Note ADC results of Triggered Sampling Mode are not updated to this register

22.4.6 ADC1_DR (0x0FDA, 0x0FDB)

ADC1_DRH(0x0FDA)								
Bit	15	14	13	12	11	10	9	8
Name	ADC1_DR[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
ADC1_DRL(0x0FDB)								
Bit	7	6	5	4	3	2	1	0
Name	ADC1_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	ADC1_DR	The conversion results of ADC channel 1 upon completion of ADC conversion in the Sequential Sampling Mode. The data is aligned according to ADC_CR[ADCALIGN]. Note ADC results of Triggered Sampling Mode are not updated to this register

22.4.7 ADC2_DR (0x0FDC, 0x0FDD)

ADC2_DRH(0x0FDC)								
Bit	15	14	13	12	11	10	9	8
Name	ADC2_DR[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
ADC2_DRL(0x0FDD)								
Bit	7	6	5	4	3	2	1	0
Name	ADC2_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	ADC2_DR	The conversion results of ADC channel 2 upon completion of ADC conversion in the Sequential Sampling Mode. The data is aligned according to ADC_CR[ADCALIGN]. Note ADC results of Triggered Sampling Mode are not updated to this register

22.4.8 ADC3_DR (0x0FDE, 0x0FDF)

ADC3_DRH(0x0FDE)								
Bit	15	14	13	12	11	10	9	8
Name	ADC3_DR[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
ADC3_DRL(0x0FDF)								
Bit	7	6	5	4	3	2	1	0
Name	ADC3_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	ADC3_DR	The conversion results of ADC channel 3 upon completion of ADC conversion in the Sequential Sampling Mode. The data is aligned according to ADC_CR[ADCALIGN]. Note ADC results of Triggered Sampling Mode are not updated to this register

22.4.9 ADC4_DR (0x0FE0, 0x0FE1)

ADC4_DRH(0x0FE0)								
Bit	15	14	13	12	11	10	9	8
Name	ADC4_DR[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
ADC4_DRL(0x0FE1)								
Bit	7	6	5	4	3	2	1	0
Name	ADC4_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	ADC4_DR	The conversion results of ADC channel 4 upon completion of ADC conversion in the Sequential Sampling Mode. The data is aligned according to ADC_CR[ADCALIGN]. Note ADC results of Triggered Sampling Mode are not updated to this register

22.4.10 ADC5_DR (0x0FE2, 0x0FE3)

ADC5_DRH(0x0FE2)								
Bit	15	14	13	12	11	10	9	8
Name	ADC5_DR[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
ADC5_DRL(0x0FE3)								
Bit	7	6	5	4	3	2	1	0
Name	ADC5_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	ADC5_DR	The conversion results of ADC channel 5 upon completion of ADC conversion in the Sequential Sampling Mode. The data is aligned according to ADC_CR[ADCALIGN]. Note ADC results of Triggered Sampling Mode are not updated to this register

22.4.11 ADC6_DR (0x0FE4, 0x0FE5)

ADC6_DRH(0x0FE4)								
Bit	15	14	13	12	11	10	9	8
Name	ADC6_DR[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
ADC6_DRL(0x0FE5)								
Bit	7	6	5	4	3	2	1	0
Name	ADC6_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	ADC6_DR	The conversion results of ADC channel 6 upon completion of ADC conversion in the Sequential Sampling Mode. The data is aligned according to ADC_CR[ADCALIGN]. Note ADC results of Triggered Sampling Mode are not updated to this register

22.4.12 ADC7_DR (0x0FE6, 0x0FE7)

ADC7_DRH(0x0FE6)								
Bit	15	14	13	12	11	10	9	8
Name	ADC7_DR[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
ADC7_DRL(0x0FE7)								
Bit	7	6	5	4	3	2	1	0
Name	ADC7_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	ADC7_DR	The conversion results of ADC channel 7 upon completion of ADC conversion in the Sequential Sampling Mode. The data is aligned according to ADC_CR[ADCALIGN]. Note ADC results of Triggered Sampling Mode are not updated to this register

22.4.13 ADC8_DR (0x0FE8, 0x0FE9)

ADC8_DRH(0x0FE8)								
Bit	15	14	13	12	11	10	9	8
Name	ADC8_DR[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
ADC8_DRL(0x0FE9)								
Bit	7	6	5	4	3	2	1	0
Name	ADC8_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	ADC8_DR	The conversion results of ADC channel 8 upon completion of ADC conversion in the Sequential Sampling Mode. The data is aligned according to ADC_CR[ADCALIGN]. Note ADC results of Triggered Sampling Mode are not updated to this register

22.4.14 ADC9_DR (0x0FEA, 0x0FEB)

ADC9_DRH(0x0FEA)								
Bit	15	14	13	12	11	10	9	8
Name	ADC9_DR[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
ADC9_DRL(0x0FEB)								
Bit	7	6	5	4	3	2	1	0
Name	ADC9_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	ADC9_DR	The conversion results of ADC channel 9 upon completion of ADC conversion in the Sequential Sampling Mode. The data is aligned according to ADC_CR[ADCALIGN]. Note ADC results of Triggered Sampling Mode are not updated to this register

22.4.15 ADC10_DR (0x0FEC, 0x0FED)

ADC10_DRH(0x0FEC)								
Bit	15	14	13	12	11	10	9	8
Name	ADC10_DR[15:8]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
ADC10_DRL(0x0FED)								
Bit	7	6	5	4	3	2	1	0
Name	ADC10_DR[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:0]	ADC10_DR	The conversion results of ADC channel 10 upon completion of ADC conversion in the Sequential Sampling Mode. The data is aligned according to ADC_CR[ADCALIGN]. Note ADC results of Triggered Sampling Mode are not updated to this register

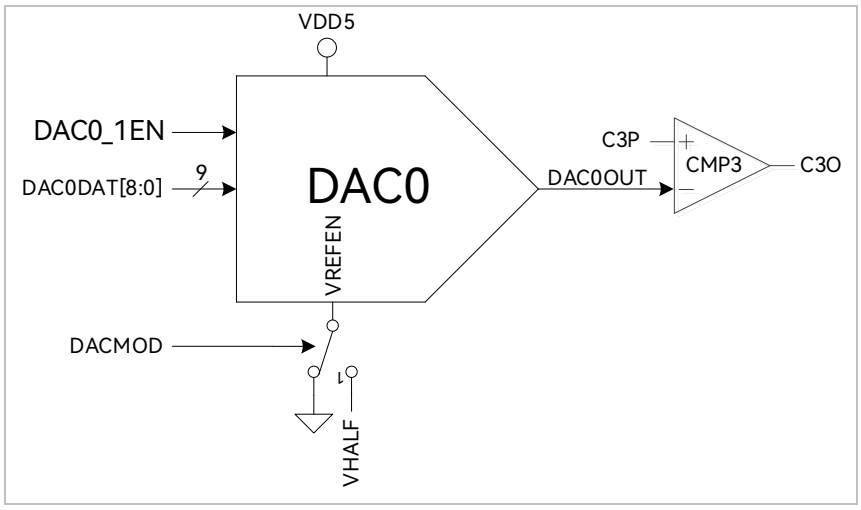
23 DAC

23.1 DAC Introduction

The chip integrates three DAC modules, where DAC0 is a 9-bit digital-to-analog converter, DAC1 is a 6-bit digital-to-analog converter and DAC2 is a 8-bit digital-to-analog converter.

23.2 DAC0 Functional Block Diagram

Figure 23-1 DAC0 Functional Block Diagram



As shown in Figure 23-1, DAC0 converts 9-bit digital data into analog voltage and sends the voltage to CMP3 negative input for bus over-current protection.

DAC0 operations are as follows:

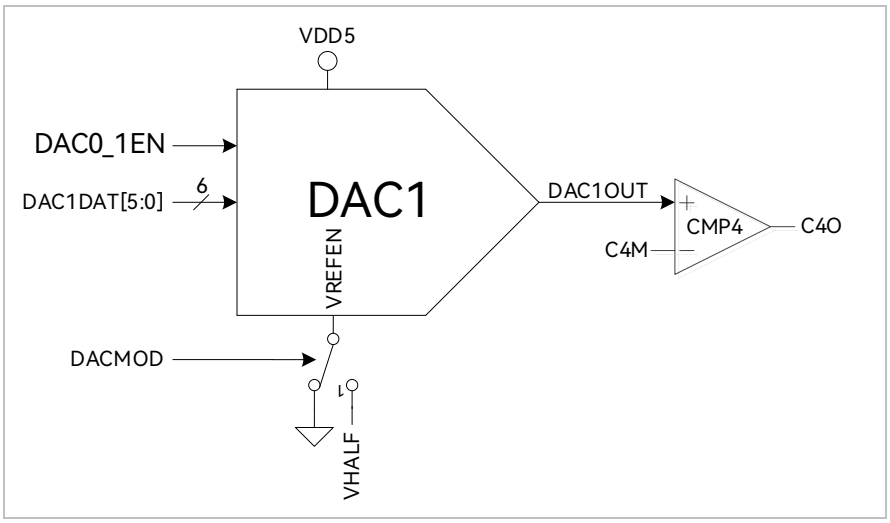
1. Configure DAC_CR[DAC0_1EN] = 1 to enable DAC0;
2. The range of output voltage is set by DAC_CR[DACMOD]. When DAC_CR[DACMOD] = 0, full-voltage output mode is active, and the range of output voltage is 0 ~ VDD5. When DAC_CR[DACMOD] = 1, half-voltage output mode is active, the range of output voltage is VHALF ~ VDD5. Output voltage of DAC0DAT under different configurations is shown in Table 23-1.

Table 23-1 Output Voltage of DAC0 under Different Configurations

DAC0DAT[8:0]	DAC Output Voltage (DAC_CR[DACMOD] = 0)	DAC Output Voltage (DAC_CR[DACMOD] = 1)
0x000	0	VHALF
0x100	VDD5/2	(VDD5 - VHALF)/2 + VHALF
0x1FF	VDD5*511/512	(VDD5 - VHALF)*511/512 + VHALF

23.3 DAC1 Functional Block Diagram

Figure 23-2 DAC1 Functional Block Diagram



As shown in Figure 23-2, DAC1 converts 6-bit digital data into analog voltage, and sends the voltage to CMP4 positive input for cycle-by-cycle current limiting.

DAC1 operations are as follows:

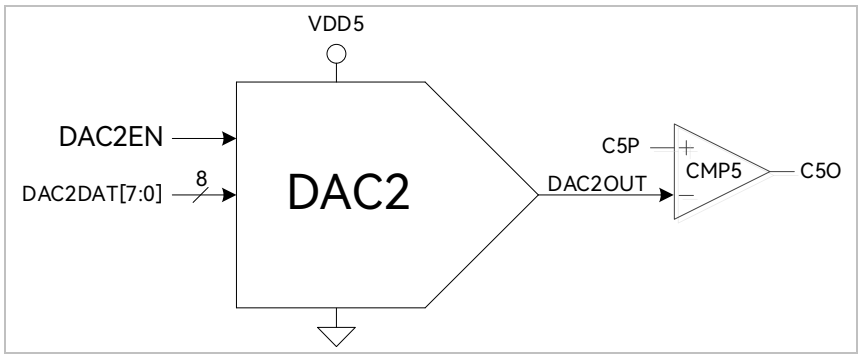
1. Configure DAC_CR[DAC0_1EN] = 1 to enable DAC1;
2. The range of output voltage is set by DAC_CR[DACMOD]. When DAC_CR[DACMOD] = 0, full-voltage output mode is active, and the range of output voltage is 0 ~ VDD5. When DAC_CR[DACMOD] = 1, half-voltage output mode is active, and the range of output voltage is VHALF ~ VDD5. Output voltage of DAC1 under different configurations is shown in Table 23-2.

Table 23-2 Output Voltage of DAC1 under Different Configurations

DAC1DAT[5:0]	DAC Output Voltage (DAC_CR[DACMOD] = 0)	DAC Output Voltage (DAC_CR[DACMOD] = 1)
0x00	0	VHALF
0x20	VDD5/2	(VDD5 - VHALF)/2 + VHALF
0x3F	VDD5*63/64	(VDD5 - VHALF)*63/64 + VHALF

23.4 DAC2 Functional Block Diagram

Figure 23-3 DAC2 Functional Block Diagram



As shown in Figure 23-3, DAC2 converts 8-bit digital data into analog voltage, and sends the voltage to CMP5 positive input for cycle-by-cycle current limiting.

Configure DAC_CR[DAC2EN] = 1 to enable DAC2.

Table 23-3 Output Voltage of DAC2 under Different Configurations

DAC2DAT[7:0]	DAC Output Voltage
0x00	0
0x80	VDD5/2
0xFF	VDD5*255/256

23.5 DAC Registers

23.5.1 DAC_CR (0x4035)

Bit	7	6	5	4	3	2	1	0
Name	DAC0_1EN	DACMOD	ADC_SCYCH[3:0]				DAC2EN	RSV
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	-
Reset	0	0	0	0	1	1	0	-

Bit	Name	Description
[7]	DAC0_1EN	DAC0 and DAC1Enable 0: Disable 1: Enable
[6]	DACMOD	DAC Mode Setting 0: Full-voltage Output Mode 1: Half-voltage Output Mode
[5:2]	ADC_SCYCH[3:0]	See section DAC_CR (0x4035) in ADC chapter
[1]	DAC2EN	DAC2 Enable 0: Disable 1: Enable
[0]	RSV	Reserved

23.5.2 DAC0_DR (0x404B)

Bit	7	6	5	4	3	2	1	0
Name	DAC0_DR							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	DAC0_DR	8 high-order bits input of DAC0 controller

23.5.3 DAC1_DR (0x404A)

Bit	7	6	5	4	3	2	1	0
Name	DAC0_DR_0	RSV	DAC1DAT					
Type	R/W	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	-	0	0	0	0	0	0

Bit	Name	Description

[7]	DAC0_DR_0	LSB input of DAC0 controller
[6]	RSV	Reserved
[5:0]	DAC1DAT	6-bit data input of DAC1 controller

23.5.4 DAC2_DR (0x4049)

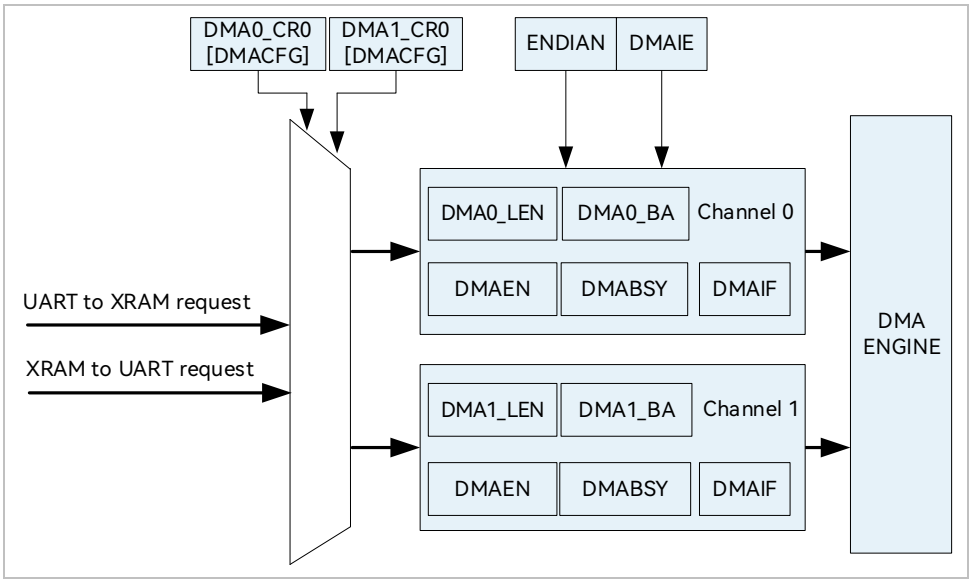
Bit	7	6	5	4	3	2	1	0
Name	DAC2_DR							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	DAC2_DR	8-bit data input of DAC2 controller

24 DMA

24.1 DMA Instructions

Figure 24-1 DMA Functional Block Diagram



The DMA module is a dual-channel DMA controller, which performs direct data transfer between peripherals and XRAM (IRAM data invalid). DMA accessing to XRAM does not interfere with the normal CPU read/write operation to XRAM. The length of the transferred data and the start address of XRAM access are configurable. Data transfer mode is configurable and interrupt can be enabled.

DMA operations are as follows:

1. Configure and enable the peripheral, and set input and output channels taken over by DMA through DMAx_CR0[DMACFG];
2. Configure DMA interrupt enable, transfer order, transfer length and XRAM start address. Write “1” to DMAx_CR0[DMAEN] and DMAx_CR0[DMABSY] to start DMA;
3. After data transfer, the interrupt flag bit DMAx_CR0[DMAIF] is set to “1” by hardware and it is cleared to “0” by software;
4. Set DMAx_CR0[DMABSY] to “1” to start DMA again.

24.2 DMA Registers

24.2.1 DMA0_CR0 (0x403A)

Bit	7	6	5	4	3	2	1	0
Name	DMAEN	DMABSY	DMACFG			DMAIE	ENDIAN	DMAIF
Type	R/W	R/W1	R/W	R/W	R/W	R/W	R/W	R/W0
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	DMAEN	DMA Channel 0 Enable 0: Disable 1: Enable
[6]	DMABSY	DMA Channel 0 Start/Busy Flag Read: 0: Channel 0 Idle 1: Channel 0 Busy Write: 0: No effect 1: Channel 0 starts for data transfer
[5:3]	DMACFG	DMA Channel 0 Peripherals and Transfer Direction Selection 000: From UART1 to XRAM 001: From XRAM to UART1 010: Reserved 011: Reserved 100: Reserved 101: Reserved 110: From UART2 to XRAM 111: From XRAM to UART2 Note It cannot be configured when Channel 0 is busy
[2]	DMAIE	DMA Channel Interrupt Enable 0: Disable 1: Enable
[1]	ENDIAN	DMA Data Transfer Sequence 0: High-order bytes are received or sent first 1: Low-order bytes are received or sent first Note This bit is set for 16-bit data mode, and shall be configured to “0” for 8-bit data mode. It cannot be configured when Channel 0 or 1 is busy

[0]	DMAIF	DMA Channel 0 Transfer Interrupt Event Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: The interrupt event is generated.

24.2.2 DMA1_CR0 (0x403B)

Bit	7	6	5	4	3	2	1	0
Name	DMAEN	DMABSY	DMACFG			DBGSW	DBGEN	DMAIF
Type	R/W	R/W1	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	DMAEN	DMA Channel 1 Enable 0: Disable 1: Enable
[6]	DMABSY	DMA Channel 1 Start/Busy Read: 0: Channel 1 Idle 1: Channel 1 Busy Write: 0: No effect 1: Channel 1 starts for data transfer
[5:3]	DMACFG	DMA Channel 1 Peripherals and Direction Selection 000: From UART1 to XRAM 001: From XRAM to UART1 010: Reserved 011: Reserved 100: Reserved 101: Reserved 110: From UART2 to XRAM 111: From XRAM to UART2 Note It cannot configured when Channel 1 is busy
[2]	DBGSW	Sector Targeted in Debug Mode 0: XSFR as the Debug area 1: XRAM as the Debug area

[1]	DBGEN	Debug Mode Enable DMA works in Debug mode when DMA1_CR0[DMACFG] is set to “101” and DMA1_CR0[DBGEN] to “1”. After SPI is enabled, DMA automatically sends relevant data in the sector defined by DMA1_CR0[DBGSW] via MOSI. DMA1_BA/DMA1_LEN defines the start address and range of the relevant data. 0: Disable 1: Enable  Note DMA Channel 1 Interrupt is automatically disabled in Debug mode
[0]	DMAIF	DMA Channel 1 Transfer Interrupt Event Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: The interrupt event is generated.

24.2.3 DMA0_LEN (0x403C)

Bit	7	6	5	4	3	2	1	0
Name	RSV		DMA0_LEN					
Type	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	-	0	0	0	0	0	0

Bit	Name	Description
[7:6]	RSV	Reserved
[5:0]	DMA0_LEN	Transfer Length of DMA Channel 0 Read: The number of the byte that is currently transferred by DMA Channel 0 (0 denotes the first byte) Write: XRAM data transfer length of DMA Channel 0  Note It cannot be configured when Channel 0 is busy. When DMA0_CR0[ENDIAN] = 1 (low bytes are received or transmitted first) , it is recommended that DMA0_LEN be set to an odd number

24.2.4 DMA0_BA (0x403E, 0x403F)

DMA0_BAH(0x403E)								
Bit	15	14	13	12	11	10	9	8
Name	RSV				DMA0_BA[11:8]			
Type	-	-	-	-	R/W	R/W	R/W	R/W
Reset	-	-	-	-	0	0	0	0
DMA0_BAL(0x403F)								
Bit	7	6	5	4	3	2	1	0
Name	DMA0_BA[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:12]	RSV	Reserved
[11:0]	DMA0_BA	Start address of data transfer by DMA Channel 0 Start address of XRAM data transfer by DMA Channel 0 It cannot be configured when Channel 0 is busy.  Note XRAM address space for data transfer by Channel 0: DMA0_BA[11:0] ~ (DMA0_BA[11:0] + DMA0_LEN[5:0])

24.2.5 DMA1_LEN (0x403D)

Bit	7	6	5	4	3	2	1	0
Name	RSV		DMA1_LEN					
Type	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	-	0	0	0	0	0	0

Bit	Name	Description
[7:6]	RSV	Reserved
[5:0]	DMA1_LEN	Transfer length of DMA Channel 1 Read: The number of the bytes that is currently transferred by DMA Channel 1 (0 denotes the first byte) Write: XRAM data transfer length of DMA Channel 1  Note It cannot be configured when Channel 1 is busy. When DMA0_CR0[ENDIAN] = 1 (low bytes are received or transmitted first), it is recommended that DMA1_LEN be set to an odd number

24.2.6 DMA1_BA (0x4040, 0x4041)

DMA1_BAH(0x4040)								
Bit	15	14	13	12	11	10	9	8
Name	RSV				DMA1_BA[11:8]			
Type	-	-	-	-	R/W	R/W	R/W	R/W
Reset	-	-	-	-	0	0	0	0
DMA1_BAL(0x4041)								
Bit	7	6	5	4	3	2	1	0
Name	DMA1_BA[7:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[15:12]	RSV	Reserved
[11:0]	DMA1_BA	Start address of data transfer by DMA Channel 1 Start address of XRAM data transfer by DMA Channel 1 It cannot be configured when Channel 1 is busy. Note XRAM address space for data transfer by Channel 1: DMA1_BA[11:0] ~ (DMA1_BA[11:0] + DMA1_LEN[5:0])

25 VHALF

25.1 VHALF Instructions

VHALF module generates the voltage reference. Its voltage value is controlled by register VHALF_CR [VHALFSEL], as detailed: 00: VDD5/8, 01: VDD5/4, 10: VDD5*25/64 and 11: VDD5/2 (Default)

VHALF is enabled when VHALF_CR[VHALFEN] is set to “1”.

25.2 VHALF Register

25.2.1 VHALF_CR (0x404F)

Bit	7	6	5	4	3	2	1	0
Name	RSV					VHALFSEL		VHALFEN
Type	-	-	-		-	R/W	R/W	R/W
Reset	-	-	-		-	1	1	0

Bit	Name	Description
[7:3]	RSV	Reserved
[2:1]	VHALFSEL	VHALF Operating Voltage Selection (VDD5 Coefficient) 00: 1/8 01: 1/4 10: 25/64 11: 1/2 (Default)
[0]	VHALFEN	VHALF Enable 0: Disable 1: Enable

26 Operational Amplifier

26.1 Operational Amplifier Introduction

The chip integrates three high-speed independent operational amplifiers, AMP0, AMP1 and AMP2. Each operational amplifier has a separate enable bit, and can be configured as PGA.

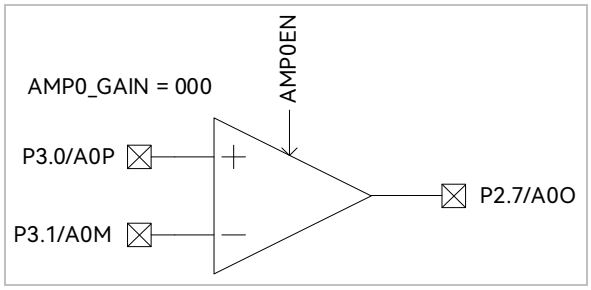
26.2 Operational Amplifier Instructions

26.2.1 Bus Current Sampling Operational Amplifier (AMP0)

AMP0 operates in two modes: normal mode and PGA differential input mode.

26.2.1.1 AMP0 Normal Mode

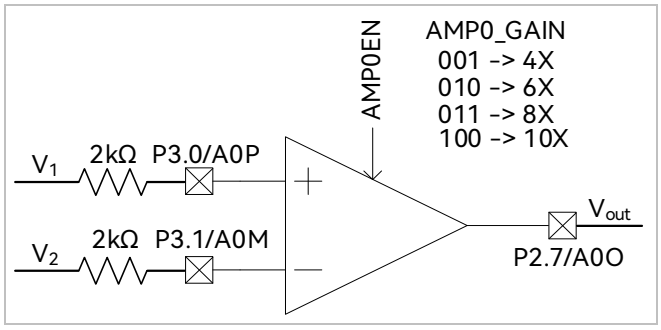
Figure 26-1 Bus Current AMP0



The I/O pins of AMP0 are shown in Figure 26-1. AMP0 is enabled when AMP_CR0[AMP0EN] = 1, and P2.7, P3.0 and P3.1 are automatically configured to analog signal mode by the hardware.

26.2.1.2 AMP0 PGA Differential Input Mode

Figure 26-2 AMP0 Operating in PGA Differential Input Mode



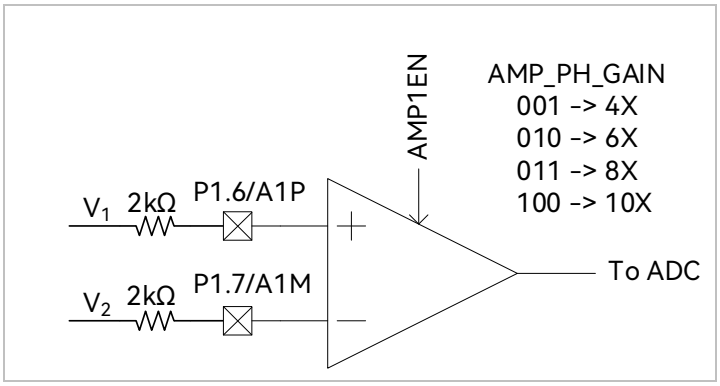
As shown in Figure 26-2, positive and negative inputs of AMP0 are connected with a 2kΩ resistor in the external circuit respectively.

When PGA differential Input Mode is selected for AMP0, the amplification gain is set by AMP_CR1[AMP0_GAIN], and AMP0 is enabled when AMP_CR0[AMP0EN] = 1. The relation between output and input of operational amplifier: Vout = VHALF + (V_1 - V_2)*AMP0_GAIN.

26.2.2 Phase Current Operational Amplifier (AMP1)

26.2.2.1 AMP1 PGA Differential Input Mode

Figure 26-3 AMP1 Operating in PGA Differential Input Mode



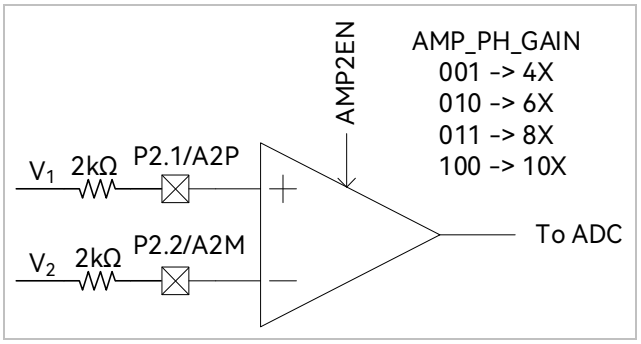
The I/O pins of AMP1 are shown in Figure 26-3. AMP1 is enabled when AMP_CR0[AMP1EN] = 1, and 1.6 and P1.7 are automatically configured to analog signal mode by the hardware. Positive and negative inputs of AMP1 are connected with a 2kΩ resistor in the external circuit respectively.

When PGA Differential Input Mode is selected for AMP1, the amplification gain is set by AMP_CR1 [AMP_PH_GAIN], and AMP1 is enabled when AMP_CR0[AMP1EN] = 1. The relation between output and input of operational amplifier: Vout = VHALF + (V_1 - V_2)*AMP_PH_GAIN.

26.2.3 Phase Current Operational Amplifier (AMP2)

26.2.3.1 AMP2 PGA Differential Input Mode

Figure 26-4 AMP2 Operating in PGA Differential Input Mode



The I/O pins of AMP2 are shown in Figure 26-4. AMP2 is enabled when AMP_CR0[AMP2EN] = 1, and P2.1 and P2.2 are automatically configured to analog signal mode by the hardware. Positive and negative inputs of AMP2 are connected with a 2kΩ resistor in the external circuit respectively.

When PGA Differential Input Mode is selected for AMP2, the amplification gain is set by AMP_CR1 [AMP_PH_GAIN], and AMP2 is enabled when AMP_CR0[AMP2EN] = 1. The relation between output and input of operational amplifier: Vout = VHALF + (V1 - V2) * AMP_PH_GAIN.

26.3 Operational Amplifier Registers

26.3.1 AMP_CR0 (0x0404E)

Bit	7	6	5	4	3	2	1	0
Name	RSV			CP_EN	AMP_FS	AMP2EN	AMP1EN	AMP0EN
Type	-	-	-	R/W	R/W	R/W	R/W	R/W
Reset	-	-	-	0	0	0	0	0

Bit	Name	Description
[7:5]	RSV	Reserved
[4]	CP_EN	Constant Power Configuration Enable 0: Disable 1: Enable Note After this bit is enabled, AMP0 output (P2.7) is routed to P2.6 via an internal resistor. A 1μF external capacitor is required at P2.6 for averaged bus average current sampling.
[3]	AMP_FS	ADC Channel 0/4 Selection for AMP0/AMP1 Output 0: ADC channel 4 is selected for AMP0 output, and ADC channel 0 for AMP1 output. 1: ADC channel 4 is selected for AMP1 output, and ADC channel 0 for AMP0 output.
[2]	AMP2EN	AMP2 Enable 0: Disable 1: Enable
[1]	AMP1EN	AMP1 Enable 0: Disable 1: Enable
[0]	AMP0EN	AMP0 Enable 0: Disable 1: Enable

26.3.2 AMP_CR1 (0x4034)

Bit	7	6	5	4	3	2	1	0
Name	AMP_PH_GAIN			RSV		AMP0_GAIN		
Type	R/W	R/W	R/W	-	-	R/W	R/W	R/W
Reset	0	0	0	-	-	0	0	0

Bit	Name	Description
[7:5]	AMP_PH_GAIN	Amplification gain setting for AMP1&AMP2 See descriptions on AMP_CR1[AMP0_GAIN] in section AMP_CR1 (0x4034)
[4:3]	RSV	Reserved

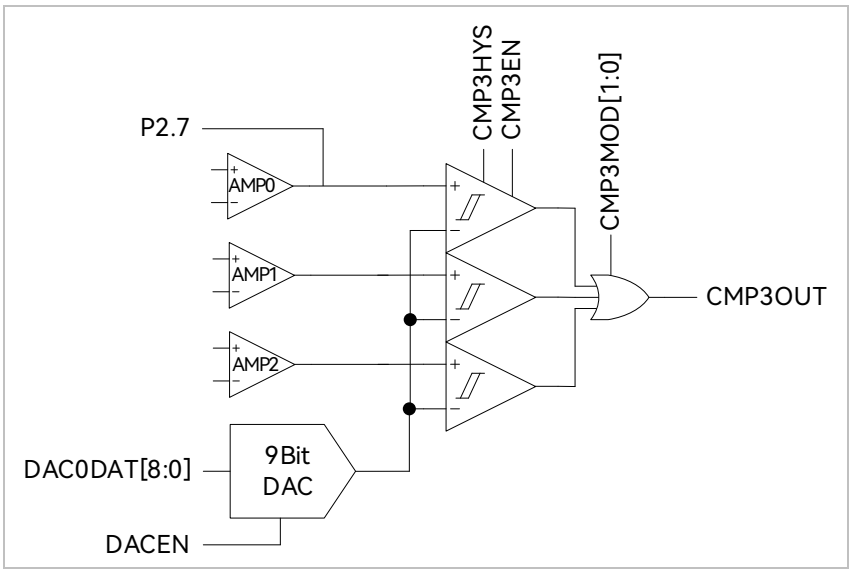
[2:0]	AMP0_GAIN	Amplification gain setting for AMP0 000: The gain is configured by external circuit 001: 4x 010: 6x 011: 8x 100: 10x 101: Reserved 110: Reserved 111: Reserved Note The built-in amplification is isotropic amplification. When the difference of input voltage is 0, the output voltage is VHALF. For other applications, AMP0_GAIN is set to “000” to select external circuit to configure the gain.

27 Comparator

27.1 Comparator Operations

27.1.1 Comparator CMP3

Figure 27-1 CMP3 I/O Pins



The I/O pins of CMP3 are shown in Figure 27-1.

CMP3 configurations are as follows:

1. Configure CMP_CR1[CMP3MOD] to select single-comparator input mode, dual-comparator input mode, or triple-comparator input mode;

> When CMP_CR1[CMP3MOD] = 00, CMP3 works in single-comparator input mode. The connection of input and output pins are shown in Figure 27-2.

> When CMP_CR1[CMP3MOD] = 01, CMP3 works in dual-comparator input mode. The connection of input and output pins are shown in Figure 27-3.

> When CMP_CR1[CMP3MOD] = 1X, CMP3 works in triple-comparator input mode. The connection of input and output pins are as shown in Figure 27-4.

2. Configure CMP_CR1[CMP3HYS] to enable or disable hysteresis;

3. Set CMP_CR1[CMP3EN] = 1 to enable CMP3.

Figure 27-2 Single-comparator Input Mode

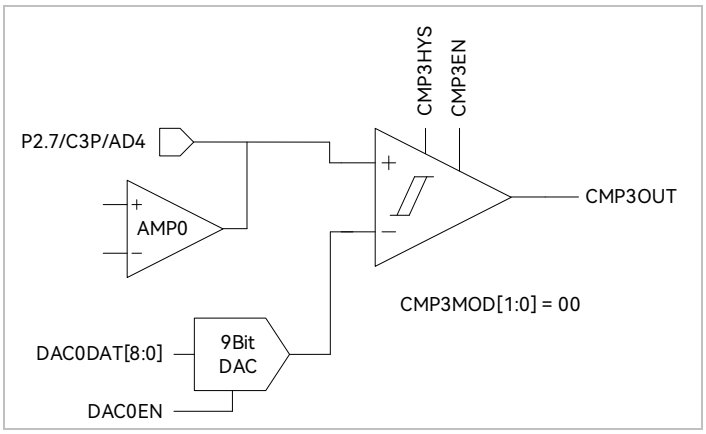


Figure 27-3 Dual-comparator Input Mode

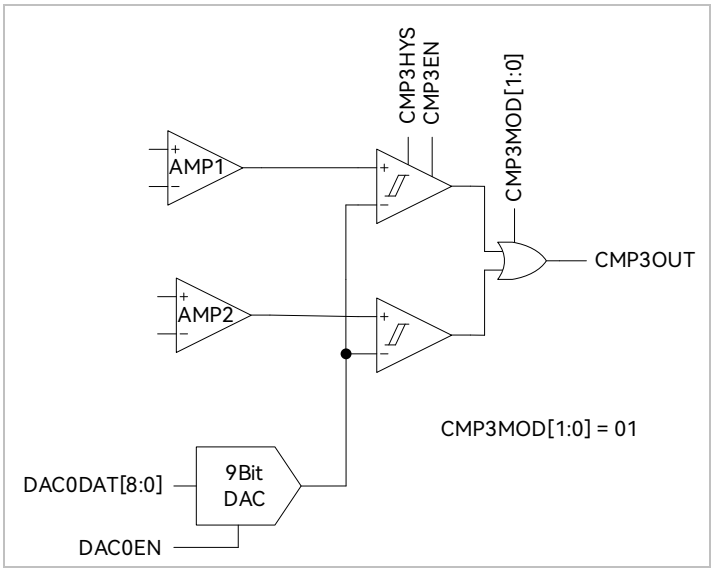
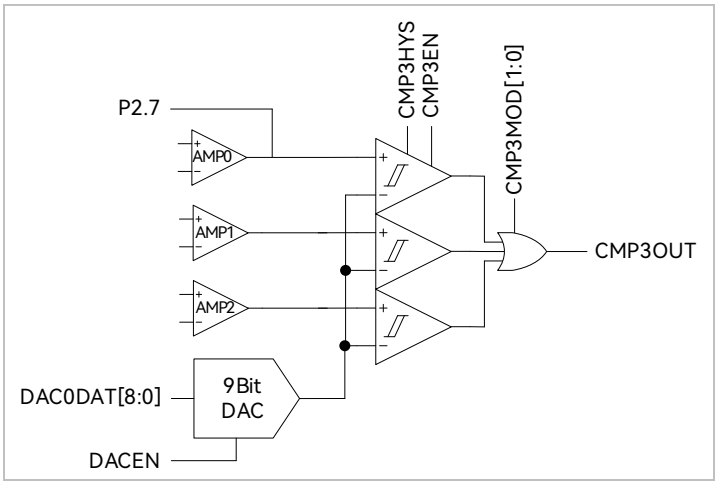


Figure 27-4 Triple-comparator Input Mode



27.1.1.1 Over-current Protection (OCP)

When an over-current protection signal is generated, DRV_OUT[MOE] is automatically cleared to output idle voltage to stop motor drive for chip and motor protection. OCP feature is enabled when EVT_FILT[MOEMD] = 01, which automatically turns off the output and generates an OCP interrupt request if the current exceeds the threshold. When EVT_FILT[MOEMD]=00, the output is not automatically turned off if the current exceeds the threshold. However, an OCP request is generated by the hardware.

The source of OCP interrupt is selected by configuring EVT_FILT[MOEMD] ≠ 00 or EVT_FILT[INT0_MOE_EN] = 1, namely CMP3 interrupt or external interrupt INT0. When EVT_FILT[INT0_MOE_EN] = 1, TCON[IT0] is programmed to select the trigger edge of the external interrupt INT0 which generates an OCP output. At this time, the source of OCP interrupt is INT0. When EVT_FILT[INT0_MOE_EN] = 0 and CMP_CR0[CMP3IM] = 01, the OCP output is generated on the raising edge of CMP3. At this time, the source of OCP interrupt is CMP3. In triple-shunt current sampling mode, CMP_CR1[CMP3MOD] is configured to select triple-comparator input mode. When current of any phase is over the threshold, CMP3 generates an OCP signal. For other sampling modes, CMP_CR1[CMP3MOD] is configured to choose single-comparator input mode. When bus current is over the threshold, CMP3 generates an OCP signal.

Configuring EVT_FILT[EFDIV] enables the filtering of interrupt signals for OCP, and programming EVT_FILT[EFDIV] = 01/10/11 selects filter width of 6/12/24 clock cycles. When the filtering feature is enabled, the filtered signal is delayed by 6/12/24 clock cycles compared to the signal before filtering.

27.1.1.2 Cycle-by-cycle Current Limiting

The cycle-by-cycle current limiting feature is applied to square-wave-based drive control of BLDC motors. When an OCP event occurs, DRV_OUT[MOE] is set to “1” by hardware after it has been cleared to “0” for a period of time, so that the motor drive is automatically restored. When CMP_CR0[CMP3IM] = 11, DRV_OUT[MOE] is cleared to “0” on the rising edge of CMP3OUT to protect the motor. When EVT_FILT[MOEMD] = 10, the outputs are automatically turned off upon an OCP interrupt. DRV_OUT[MOE] is enabled automatically upon Driver timer overflow/underflow event or after 10μs to restore motor drive. When EVT_FILT[MOEMD] = 11, the outputs are automatically turned off upon an OCP interrupt. DRV_OUT[MOE] is enabled automatically upon Driver timer overflow/underflow event or after 5μs to

restore motor drive.

Figure 27-5 Cycle-by-cycle Current Limiting Waveform (t2 - t1 = 10μs)

when EVT_FILT[MOEMD] = 10

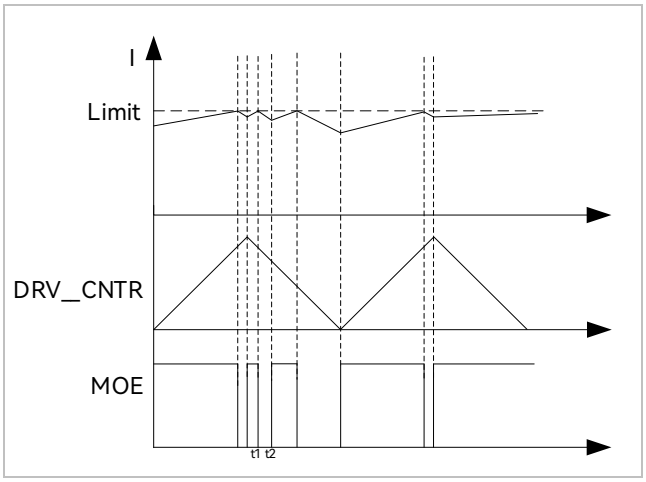
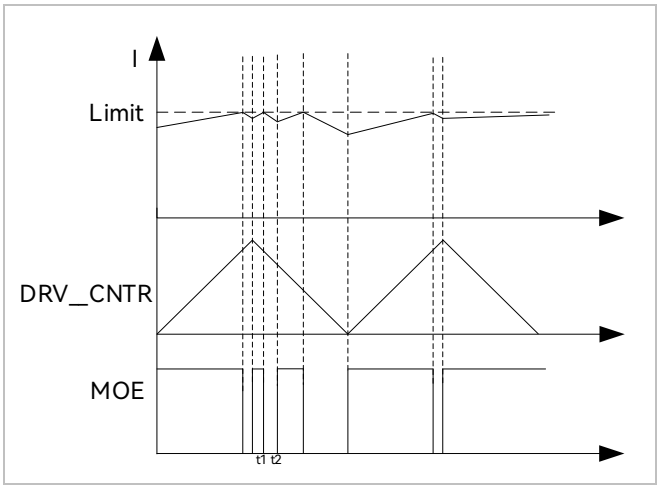


Figure 27-6 Cycle-by-cycle Current Limiting Waveform (t2 - t1 = 5μs)

when EVT_FILT[MOEMD] = 11



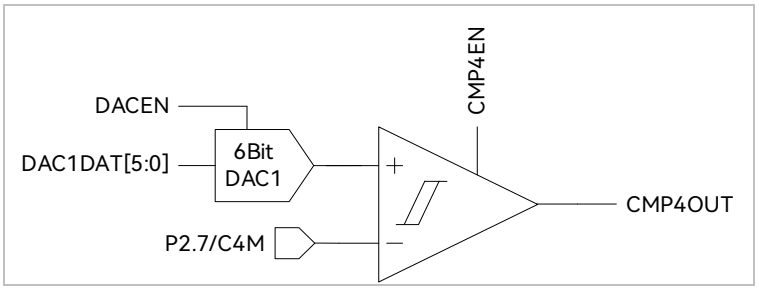
27.1.2 Comparator CMP4

The I/O pins of CMP4 are shown in Figure 27-7. CMP4OUT can be read by software or reversed upon external interrupt INT0. When CMP3 is used for cycle-by-cycle current limiting protection, CMP4 is used for bus current protection. When bus current OCP feature is triggered, CMP4 output must be turned off by software.

CMP4 configurations are as follows:

1. Configure P2_AN[7] = 1 to assign P2.7 pin to analog signal;
2. Configure CMP_CR2[CMP4EN] = 1 to enable CMP4;
3. Clear INT0 flag bit to enable INT0;
4. Set LVSR[EXT0CFG] = 111 to select CMP4 as the source of INT0;
5. Configure TCON[IT0] = 01 to select falling edge triggered INT0

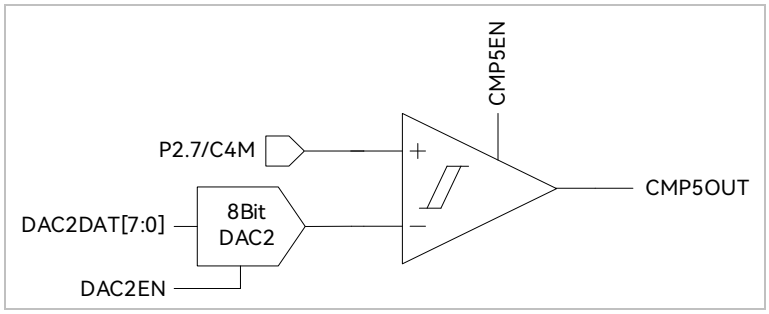
Figure 27-7 Schematic Diagram of CMP4 Module



27.1.3 Comparator CMP5

CMP5 is a hysteresis comparator, as shown in Figure 27-8. CMP5OUT can be read by software. CMP5 is used for cycle-by-cycle current limiting or current protection of PFC module. Configuring CMP_CR4[CMP5EN] = 1 enables CMP5.

Figure 27-8 Schematic Diagram of CMP5 Module



27.1.4 Comparator Group (CMPG)

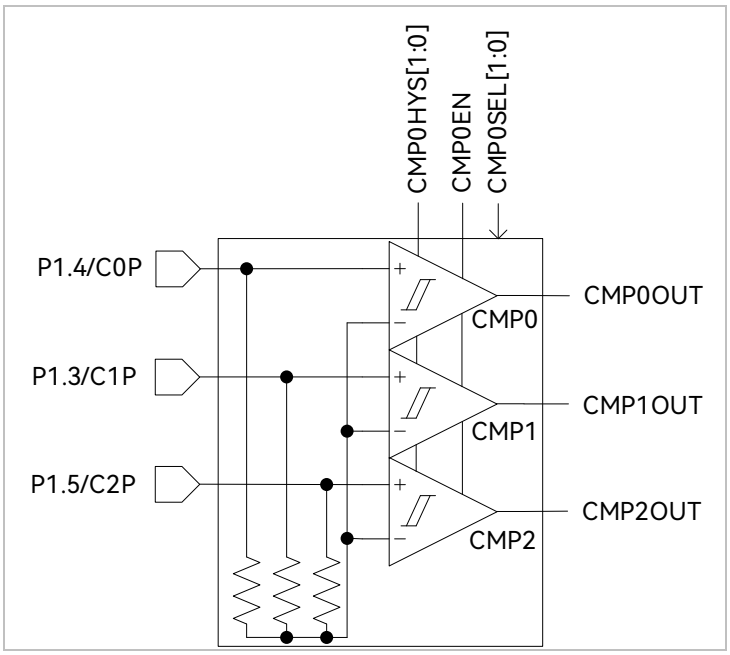
Comparator Group (CMPG) is a collection of CMP0, CMP1 and CMP2, with two comparison modes for different applications.

27.1.4.1 Built-in Three Comparators and Resistors Mode

When CMP_CR2[CMP0MOD] = 0, CMPG works in built-in three comparators and resistors mode. It is used for BEMF detection with the internal virtual neutral point resistors. The number of comparators operating in this mode is defined by CMP_CR2[CMP0SEL]. When CMP_CR2[CMP0SEL] = 00, CMP0, CMP1 and CMP2 work simultaneously, which is the recommended configuration. When CMP_CR2[CMP0SEL] = 01, only CMP0 works. When CMP_CR2[CMP0SEL] = 10, only CMP1 works. When CMP_CR2[CMP0SEL] = 11, only CMP2 works.

The I/O pins are shown in Figure 27-9. The negative inputs of the three comparators are connected together to the center point of the built-in resistor. The positive inputs are connected to P1.4, P1.3 and P1.5 respectively, and the outputs are CMP0OUT, CMP1OUT and CMP2OUT respectively.

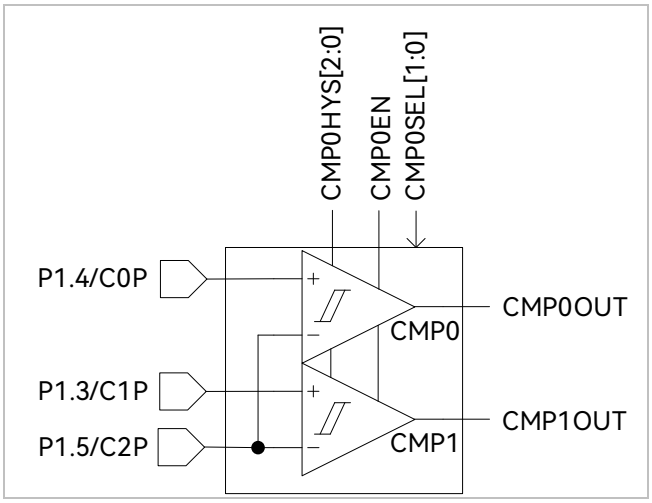
Figure 27-9 CMPG in Built-in Three Comparators and Resistors



27.1.4.2 Dual-comparator Mode

When CMP_CR2[CMP0MOD] = 1, CMPG works in dual-comparator mode for motor speed detection. The I/O pins are shown in Figure 27-10. The negative inputs of the two comparators are connected together to P1.5, and the positive inputs are connected to P1.4 and P1.3 respectively. The outputs are CMP0OUT and CMP1OUT respectively. The number of comparators in this mode is defined by CMP_CR2[CMP0SEL]. When CMP_CR2[CMP0SEL] = 00, CMP0 and CMP1 work simultaneously, which is the recommended configuration. When CMP_CR2[CMP0SEL] = 01, only CMP0 works. When CMP_CR2[CMP0SEL] = 10, only CMP1 works.

Figure 27-10 CMPG in Dual-comparator Mode

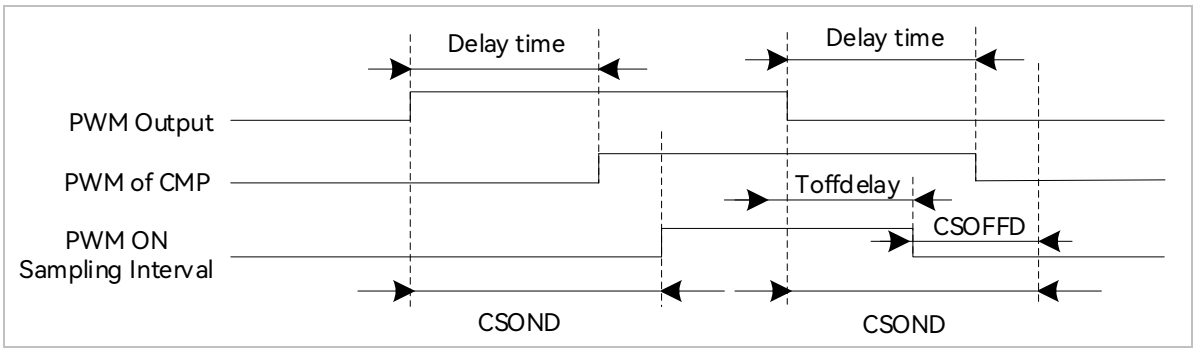


The output signals of CMP0/CMP1/CMP2 are sent to Timer1 after filtering and sampling modules.

27.1.5 Comparator Sampling

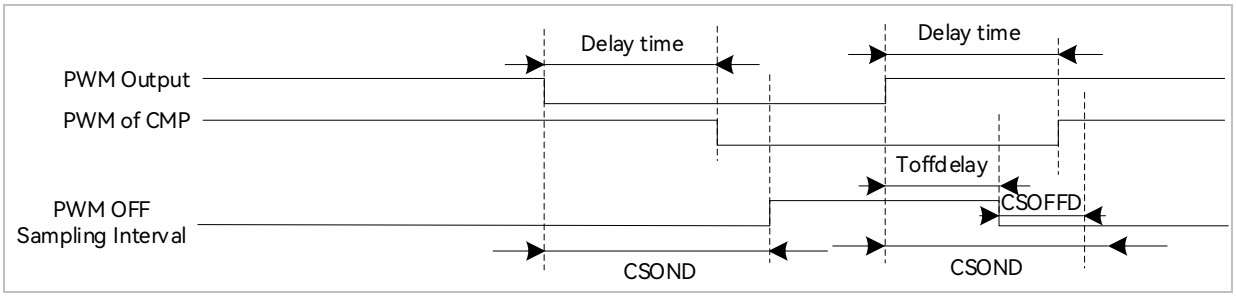
The comparator sampling feature is mainly used for the square-wave control, which eliminates the switching interference from driving circuit. See section 0 for square-wave control.

Figure 27-11 PWM ON Sampling Mode



There is a delay from the PWM output to the output of the comparator, which is mainly affected by the following factors: resistance value of drive resistor, switching speed of the power IC, and input delay and hysteresis settings of the comparator. As shown in the figure, the delay time is from the chip output to the comparator output. When high-level sampling is performed, the sampling interval shall be enveloped by actual high-level output of the comparator. First, the ON-delayed sampling time CMP_SAMR[CSOND] is set to overcome the output delay and avoid ringing at the switch node on power IC. At the end of the sampling interval, CMP_SAMR[CSOND] is delayed after the falling edge of PWM, at which time the actual sampling window has exceeded the corresponding high-level interval. The OFF-lead sampling time CMP_SAMR[CSOFFD] is set to stop sampling Toffdelay after the PWM output falling edge, where Toffdelay = CMP_SAMR[CSOND] – CMP_SAMR[CSOFFD]. By configuring CMP_SAMR[CSOND] and CMP_SAMR[CSOFDD], the sampling interval can be located in the high-level interval of the actual output of the comparator.

Figure 27-12 PWM OFF Sampling Mode



Similarly, when low-level sampling is performed, the ON-delayed sampling time CMP_SAMR[CSOND] and the OFF-lead sampling time CMP_SAMR[CSOFFD] are set reasonably to ensure that the actual sampling interval is located in the low-level output interval of the comparator.

Method for measuring the delay of PWM output to comparator: Set CMP_CR3[SAMSEL] = 00 to disable the comparator sampling delay feature. Enable the PWM output and comparator, manually rotate the motor to change the comparator value, and measure the delay between the PWM output and the comparator output.

27.2 Comparator Registers

27.2.1 CMP_CR0 (0xD5)

Bit	7	6	5	4	3	2	1	0
Name	CMP3IM		CMP2IM		CMP1IM		CMP0IM	
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:6]	CMP3IM	CMP3 Interrupt Mode 00: No interrupt is generated. 01: An interrupt is generated upon rising edge. 10: An interrupt is generated upon falling edge. 11: When a rising edge is detected, DRV_OUT[MOE] is cleared to “0”, and the interrupt event flag CMP_SR[CMP3IF] is set to “1”. However, the interrupt is disabled. Note In cycle-by-cycle current limiting mode, EVT_FILT[MOEMD] must be set to 10/11
[5:4]	CMP2IM	CMP2 Interrupt Mode See descriptions on CMP_CR0[CMP0IM]
[3:2]	CMP1IM	CMP1 Interrupt Mode See descriptions on CMP_CR0[CMP0IM]
[1:0]	CMP0IM	CMP0 Interrupt Mode 00: No interrupt is generated. 01: An interrupt is generated upon rising edge. 10: An interrupt is generated upon falling edge. 11: An interrupt is generated upon both rising/falling edges.

27.2.2 CMP_CR1 (0xD6)

Bit	7	6	5	4	3	2	1	0
Name	RSV	CMP3MOD		CMP3EN	CMP3HYS	RSV	CMP0HYS	
Type	-	R/W	R/W	R/W	R/W	-	R/W	R/W
Reset	-	0	0	0	0	-	0	0

Bit	Name	Description
[7]	RSV	Reserved
[6:5]	CMP3MOD	CMP3 Mode Selection 00: Single-comparator mode, where P2.7 is connected to the positive input, as shown in Figure 27-2

		01: Dual-comparator mode, where AMP1 and AMP2 are connected to the positive inputs, as shown in Figure 27-3 1X: Triple-comparator mode, where AMP1, AMP2 and P2.7 are connected to the positive inputs, as shown in Figure 27-4
[4]	CMP3EN	CMP3 Enable 0: Disable 1: Enable
[3]	CMP3HYS	CMP3 Hysteresis Voltage Selection 0: No hysteresis 1: Hysteresis voltage is selected
[2]	RSV	Reserved
[1:0]	CMP0HYS	CMP0/1/2 Hysteresis Voltage Selection 00: Disable 01: ± 3mV 10: ± 6mV 11: ± 12mV

27.2.3 CMP_CR2 (0xDA)

Bit	7	6	5	4	3	2	1	0
Name	CMP4EN	RSV	CMP0MOD	CMP0SEL		CMP0CSEL		CMP0EN
Type	R/W	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	-	0	0	0	0	0	0

Bit	Name	Description
[7]	CMP4EN	CMP4 Enable 0: Disable 1: Enable
[6]	RSV	Reserved
[5]	CMP0MOD	CMPG Mode Setting 0: Built-in three comparators and resistors mode, as shown in Figure 27-9 1: Dual-comparator mode, where only CMP0 and CMP1 work, as shown in Figure 27-10
[4:3]	CMP0SEL	CMPG Pin Combination Selection, used with CMP_CR2[CMP0MOD]. It is set to 00 by default. In square-wave drive applications, TIM1_DBRx[T1CPE] automatically controls CMP_CR2[CMP0SEL] to enable or disable each comparator. Table 27-1 Function Description of CMPG Port and CMP_CR2[CMP0MOD] Combination

CMP0MOD	CMP0SEL	Descriptions
0	00	CMP0/1/2 work simultaneously, as shown in Figure

				27-9. The negative input of these comparators are connected to the center of the built-in resistor. The hardware automatically compares the positive inputs C0P, C1P and C2P with C0M, and the output results are transferred to CMP0OUT, CMP1OUT and CMP2OUT respectively.
			01	Only CMP0 works. The positive input is connected to C0P, and the negative input to the center of built-in BEMF resistor. The output results are transferred to CMP0OUT.
			10	Only CMP1 works. The positive input is connected to C1P, and the negative input to the center of built-in BEMF resistor. The output results are transferred to CMP1OUT.
			11	Only CMP2 works. The positive input is connected to C2P, and the negative input to the center of built-in BEMF resistor. The output results are transferred to CMP2OUT.
		1	00	CMP0/1 work simultaneously, as shown in Figure 27-10. The positive inputs are connected to C0P and C1P respectively, and the negative inputs to C2P. The outputs results are transferred to CMP0OUT and CMP1OUT respectively.
			01	Only CMP0 works. The positive input is connected to C0P, and the negative input to C2P. The output results are transferred to CMP0OUT.
			10	Only CMP1 works. The positive input is connected to C1P, and the negative input to C2P. The output results are transferred to CMP1OUT.
			11	Reserved
[2:1]	CMP0CSEL	CMP0/1/2 Polling Speed Selection 00: Normal Speed 01: Fast Speed 10: Lower Speed 11: Slow Speed		
[0]	CMP0EN	CMP0/1/2 Enable 0: Disable		

		1: Enable

27.2.4 CMP_CR3 (0xDC)

Bit	7	6	5	4	3	2	1	0
Name	CMPDTEN	DBGSEL		SAMSEL		RSV		
Type	R/W	R/W	R/W	R/W	R/W	-	-	-
Reset	0	0	0	0	0	-	-	-

Bit	Name	Description
[7]	CMPDTEN	Comparator Deadtime Sampling Enable 0: Disable 1: Enable
[6:5]	DBGSEL	Debug Output Selection (routed to P0.5) 00: Disable 01: Freewheeling shielding is completed and ZCP signal is detected 10: ADC Trigger Signal 11: Comparator Sampling Interval
[4:3]	SAMSEL	Sampling Delay Enable of CMP0, CMP1, CMP2 and ADC in PWM ON/OFF Modes 00: Sampling at both PWM ON and OFF modes without time delay 01: Sampling at PWM OFF mode, with time delay according to CMP_SAMR 10: Sampling at PWM ON mode, with time delay according to CMP_SAMR 11: Sampling at both PWM ON and OFF, with time delay according to CMP_SAMR
[2:0]	RSV	Reserved

27.2.5 CMP_CR4 (0xE1)

Bit	7	6	5	4	3	2	1	0
Name	CMP4OUT	CMP5OUT	RSV	CMP3P4M_FS	CMP5HYS	FAEN	RSV	CMP5EN
Type	R	R	-	R/W	R/W	R/W	-	R/W
Reset	0	0	-	0	0	0	-	0

Bit	Name	Description
[7]	CMP4OUT	CMP4 Output
[6]	CMP5OUT	CMP5 Output
[5]	RSV	Reserved
[4]	CMP3P4M_FS	C3P and C4M are switched to output at AMP1 for bus current sampling. 0: Disable 1: Function switching to output at AMP1, where CMP3 has only one input channel at the positive input.

[3]	CMP5_HYS	CMP5 Hysteresis Configuration 0: Disable 1: Enable
[2]	FAEN	Filtered Signal Sampling Coefficient Scale-up Enable With it enabled, basic clock rates of TIM1_CR3[T1INM] and CMP_SAMR are scaled up by 4 times. 0: Disable 1: Enable
[1]	RSV	Reserved
[0]	CMP5EN	CMP5 Enable 0: Disable 1: Enable

27.2.6 CMP_SAMR (0x40AD)

Bit	7	6	5	4	3	2	1	0
Name	CSOND				CSOFFD			
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	1

Bit	Name	Description
[7:4]	CSOND	CMP0/CMP1/CMP2 ON-delayed Sampling Time When PWM module switches from OFF to ON or from ON to OFF, turn-on/off of the power IC affects input signal of the comparator. In this case, CMP_SAMR[CSOND] is configured to delay the sampling of CMP0/CMP1/CMP2. The On-delayed sampling time can be multiplied by 4 times by setting CMP_CR4[FAEN]. CMP_CR4[FAEN] = 0: ON-delayed Sampling Time = 8*CMP_SAMR[CSOND] * T CMP_CR4[FAEN] = 1: ON-delayed Sampling Time = 32*CMP_SAMR[CSOND] * T Note › CMP_SAMR[CSOND] must be greater than or equal to CMP_SAMR[CSOFFD] › See section Sampling for BLDC drive applications
[3:0]	CSOFFD	CMP0/CMP1/CMP2 OFF-lead Sampling Time CMP_SAMR[CSOND] is configured to end the sampling CMP_SAMR[CSOND] - CMP_SAMR[CSOFFD] after the back edge of PWM output to ensure sampling interval is enveloped by the PWM interval. OFF-lead sampling time can be multiplied by 4 times by setting CMP_CR4[FAEN]. CMP_CR4[FAEN] = 0: OFF-lead Sampling Time = 8*CMP_SAMR[CSOFFD] * T CMP_CR4[FAEN] = 1: OFF-lead Sampling Time = 32*CMP_SAMR[CSOFFD] * T

		Note › CMP_SAMR[CSOND] must be greater than or equal to CMP_SAMR[CSOFFD] › See section Sampling for BLDC drive applications

27.2.7 CMP_SR (0xD7)

Bit	7	6	5	4	3	2	1	0
Name	CMP3IF	CMP2IF	CMP1IF	CMP0IF	CMP3OUT	CMP2OUT	CMP1OUT	CMP0OUT
Type	R/W0	R/W0	R/W0	R/W0	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7]	CMP3IF	CMP3 Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[6]	CMP2IF	CMP2 Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[5]	CMP1IF	CMP1 Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[4]	CMP0IF	CMP0 Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect
[3]	CMP3OUT	CMP3 Output
[2]	CMP2OUT	CMP2 Output

[1]	CMP1OUT	CMP1 Output
[0]	CMP0OUT	CMP0 Output

27.2.8 HALL_CR (0xE2)

Bit	7	6	5	4	3	2	1	0
Name	HALL_IF	HALL_IE	RSV			HALL2	HALL1	HALL0
Type	R/W0	R/W	-	-	-	R	R	R
Reset	0	0	-	-	-	0	0	0

Bit	Name	Description
[7]	HALL_IF	Hall Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt pending Write: 0: This bit is cleared to “0” 1: No effect
[6]	HALL_IE	Hall Interrupt Enable 0: Disable 1: Enable
[5:3]	RSV	Reserved
[2]	HALL2	Hall2 Level 0: Hall2 level = 0 1: Hall2 level = 1
[1]	HALL1	Hall1 Level 0: Hall1 level = 0 1: Hall1 level = 1
[0]	HALL0	Hall0 Level 0: Hall0 level = 0 1: Hall0 level = 1

27.2.9 EVT_FILT (0xD9)

Bit	7	6	5	4	3	2	1	0
Name	RSV			MOEMD		INT0_MOE_EN	EFDIV	
Type	-	-	-	R/W	R/W	R/W	R/W	R/W
Reset	-	-	-	0	0	0	0	0

Bit	Name	Description
[7:5]	RSV	Reserved

[4:3]	MOEMD	MOE Cleared and Enabled by Hardware MOE is cleared and enabled by hardware upon over-/under-current protection event. 00: MOE is not automatically cleared. 01: MOE is automatically cleared. 10: MOE is automatically cleared and enabled by hardware upon Driver timer overflow/underflow events or after 10 μs (for square-wave drive applications). 11: MOE is automatically cleared and enabled automatically upon Driver timer overflow/underflow events or after 5 μs (for square-wave drive).
[2]	INT0_MOE_EN	INT0 OFF triggered by INT0 Interrupt 0: Disable 1: Enable
[1:0]	EFDIV	Filter Width for Current Protection 00: Disable 01: 6 system clock cycles 10: 12 system clock cycles 11: 24 system clock cycles

27.2.10 TSD_CR (0x402F)

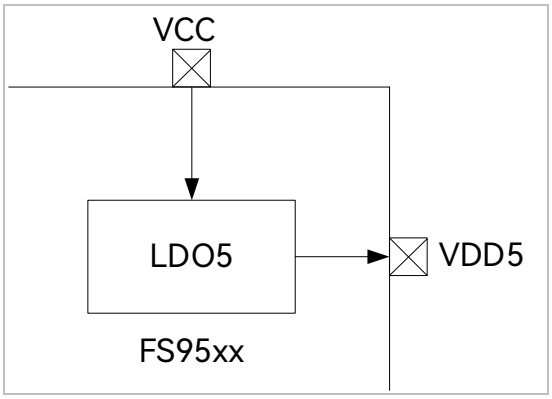
Bit	7	6	5	4	3	2	1	0
Name	TSDEN	RSV			TSDADJ			
Type	R/W	-	-	-	R/W	R/W	R/W	R/W
Reset	0	-	-	-	0	1	1	0

Bit	Name	Description
[7]	TSDEN	Temperature Detection Feature Enable 0: Disable 1: Enable
[6:4]	RSV	Reserved
[3:0]	TSDADJ	Overtemperature Protection Temperature (Chip Junction Temperature) 0000: 70℃ 0001: 76℃ 0010: 82℃ 0011: 87℃ 0100: 92℃ 0101: 97℃ 0110: 102℃ 0111: 108℃ 1000: 114℃ 1001: 120℃ 1010: 126℃ 1011: 132℃ 1100: 139℃ 1101: 145℃ 1110: 152℃ 1111: 159℃

28 Power Supply

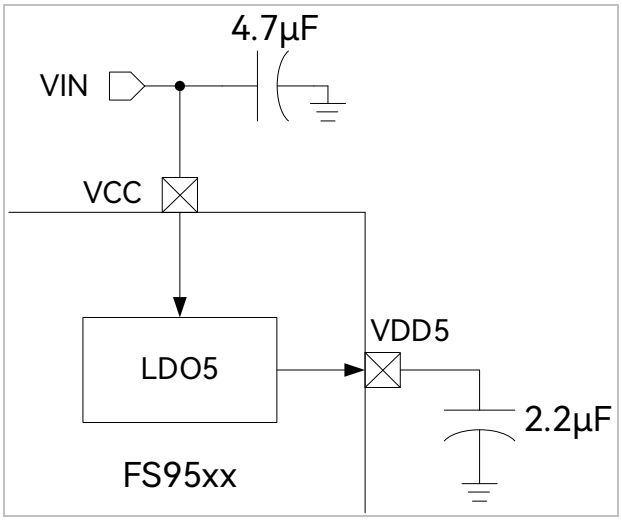
28.1 LDO Instructions

Figure 28-1 Functional Block Diagram of Power Supply



The I/O pins of LDO module is shown in Figure 28-1. The LDO module converts the input supply voltage to 5V (VDD5) as the power supply for built-in analog module. For the power connection, see Figure 28-2, where VCC = 13V ~ 20V.

Figure 28-2 Power Connection



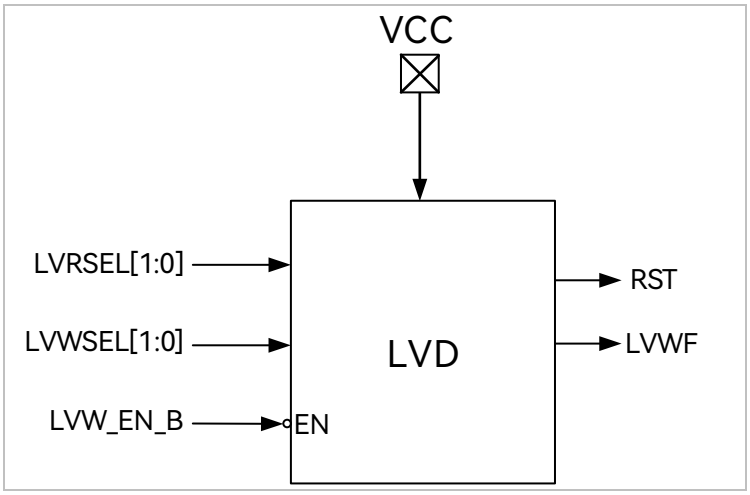
28.2 Low Voltage Detection (LVD)

28.2.1 LVD Introduction

The low voltage detection includes low voltage warning and low voltage reset.

28.2.2 LVD Operations

Figure 28-3 LV Detection Module



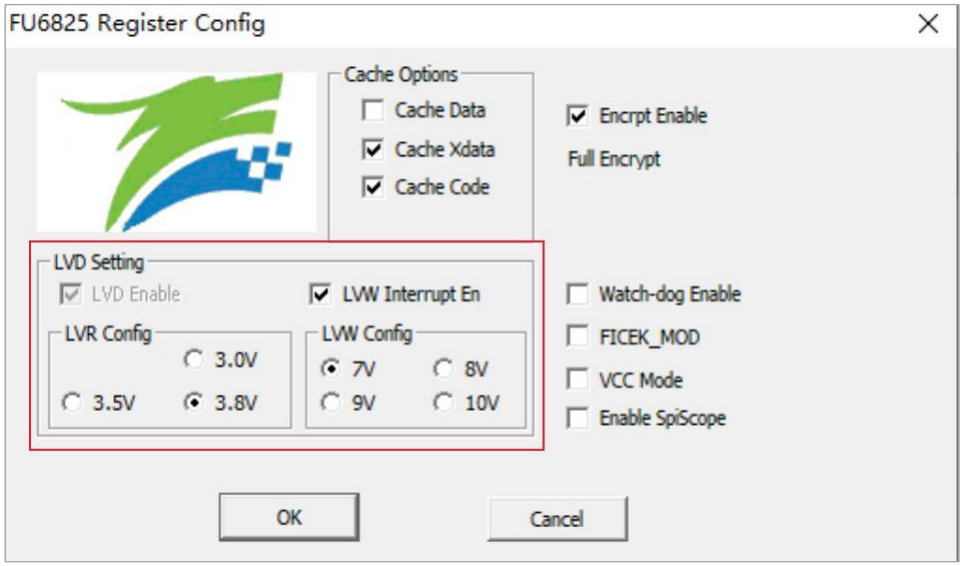
The operating instructions for LVD are as follows:

- LV warning and LV reset are always enabled by default.
- 7/8/9/10V can be selected for LV warning threshold. When the interrupt feature is enabled, an interrupt is triggered if VCC voltage is lower than the LV warning threshold.
- 3.0/3.5/3.8V can be selected for the LV reset threshold. The chip resets when VCC voltage is lower than the LV reset voltage threshold.

LV warning threshold, interrupt settings and LV reset threshold are configured through the debug tool, as shown in Figure 28-4.

LVR Config sets low voltage reset threshold, LVW Interrupt En enables low voltage interrupt, and LVW Config sets low voltage warning threshold.

Figure 28-4 Configurations of LV Warning Threshold, LV Interrupt and LV Reset Threshold



28.2.3 LVD Registers

28.2.3.1 LVSR (0xDB)

Bit	7	6	5	4	3	2	1	0
Name	RSV		EXT0CFG			TSDF	LVWF	LVWIF
Type	-	-	R/W	R/W	R/W	R	R	R/W0
Reset	-	-	0	0	0	0	0	0

Bit	Name	Description
[7:6]	RSV	Reserved
[5:3]	EXT0CFG	INT0 Pin Selection 000: P1.3 001: P1.4 010: P2.5 011: P0.3 100: P4.7 101: P0.5 110: P0.6 111: CMP4 Output
[2]	TSDF	Over Temperature State Indicator 0: The chip is not in the LV warning state. 1: The chip is in the LV warning state. Note This flag bit often works with TSD interrupt flag TCON[TSDIF]
[1]	LVWF	VCC Low Voltage(LV) Flag This bit indicates whether the chip is in the low voltage state. 0: The chip is not in the LV warning state. 1: The chip is in the LV warning state.
[0]	LVWIF	VCC LV Interrupt Flag Read: 0: No Interrupt Pending 1: Interrupt Pending Write: 0: This bit is cleared to “0” 1: No effect Note This bit is not set to “1” by hardware when LVD interrupt is disabled

29 Flash

29.1 Flash Introduction

The chip provides 32k bytes of Flash space. It supports page erasure, page pre-programing and write.

Main features:

- 128 sectors in total, each with a size of 256 bytes
- 16 pages in total, each with 8 sectors
- Last sector (address range: 0x7F00 ~ 0x7FFF) cannot be erased at any time
- 120ms ~ 150ms for page erase
- Programming is enabled when FLA_CR[FLAEN] is set to “1”, where page pre-programming, page erase or write and other Flash operations are activated with MOVX instructions.

29.2 Flash Operations

- Flash memory must be unlocked before erase and programming operations. The Flash software programming feature is activated after “0x5A” and “0x1F” are written to register FLA_KEY in sequence. If the sequence is incorrect or other values are written, Flash space is frozen until the next reset. After unlocking, any write to the FLA_CR register lock FLA_KEY again.
- CRC results change if Flash memory is rewritten during program execution.
- Page pre-programming must be done before page erase.
- Configuring FLA_CR = 0x23 enables page erase, FLA_CR = 0x25 enables page pre-programming and FLA_CR = 0x21 enables write operations.



Note

All interrupts must be disabled before self-programming to ensure the security of Flash operations and avoid mis-operation of Flash using MOVX instruction during interrupt processing

29.3 Flash Registers

29.3.1 FLA_CR (0x85)

Bit	7	6	5	4	3	2	1	0
Name	RSV		PAGE_EN	FLAERR	RSV	FLAPRE	FLAERS	FLAEN
Type	-	-	R/W	R	-	R/W	R/W	R/W
Reset	-	-	0	0	-	0	0	0

Bit	Name	Description
[7:6]	RSV	Reserved
[5]	PAGE_EN	Page Operation Enable 0: Disable 1: Enable
[4]	FLAERR	Programming Error Flag 0: Programming or pre-programming succeeds. 1: Programming or pre-programming fails.
[3]	RSV	Reserved
[2]	FLAPRE	Pre-programming Enable 0: Disable 1: Enable Note FLA_CR[FLAPRE] is valid only when FLA_CR[FLAEN] = 1
[1]	FLAERS	Erase Enable 0: Disable 1: Enable Note FLA_CR[FLAERS] is valid only when FLA_CR[FLAEN] = 1
[0]	FLAEN	Programming Enable 0: Disable 1: Enable

29.3.2 FLA_KEY (0x84)

Bit	7	6	5	4	3	2	1	0
Name	FLA_KEY							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	FLA_KEY	Write: Write “0x5A” and “0x1F” in sequence to unlock Flash operations; Write any value to FLA_CR bit to lock Flash operations.

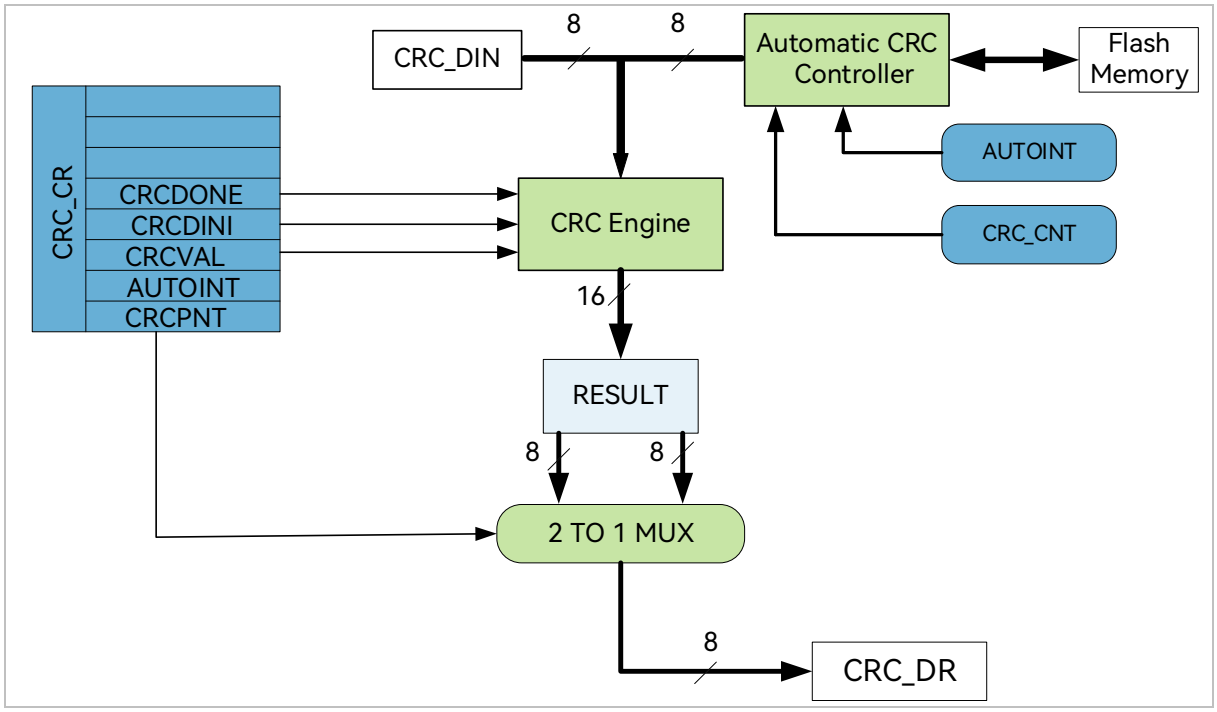
Bit	7	6	5	4	3	2	1	0
Name	RSV						FLAKSTA	
Type	-	-	-	-	-	-	R	R
Reset	-	-	-	-	-	-	0	0

Bit	Name	Description
[7:2]	RSV	Reserved
[1:0]	FLAKSTA	Read: Flash Release Status 00: Locked 01: Write of 0x5A is done, waiting for 0x1F 10: Frozen 11: Released

30 CRC

30.1 CRC Functional Block Diagram

Figure 30-1 CRC Functional Block Diagram



CRC module outputs the result of CRC calculation for any 8-bit data based on a fixed polynomial. As shown in Figure 30-1, CRC receives the 8-bit data from CRC_DIN and sends the 16-bit result to the internal register after the calculation is completed. The result can be indirectly accessed through CRC_CR[CRCPNT] and CRC_DR.

30.2 CRC16 Polynomial

The chip uses CRC16/CCITT-FALSE polynomial.

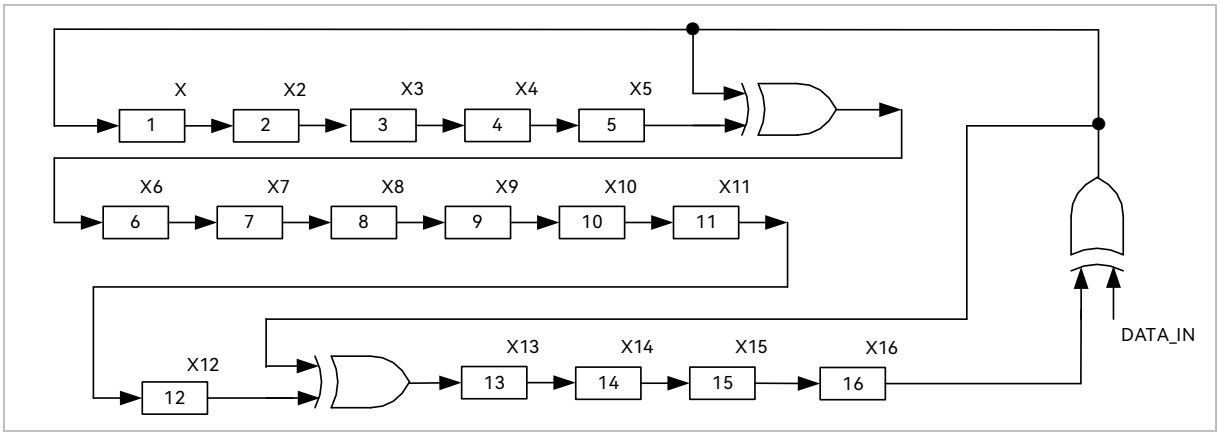
Table 30-1 CRC Criteria and Polynomials

CRC Criteria	Polynomial	Hexadecimal Representation
CRC16-CCITT-FALSE	x^16+x^12+x^5+1	0x1021

30.3 CRC16 Logic Diagram

The schematic diagram of CRC16 is shown in Figure 30-2. The chip implementation is based on parallel algorithm. For each input byte, MCU calculates the results within one system clock cycle.

Figure 30-2 CRC16 Schematic Diagram



30.4 CRC Operations

30.4.1 CRC Calculation of Single Byte

CRC of a single byte is calculated as follows:

1. Initialize CRC_DR with two options: Configure CRC_CR[CRCVAL] and set CRC_CR[CRCDINI] to “1”, with an initial value of 0x0000 or 0xFFFF. Or configure CRC_CR[CRCPNT] and CRC_DR, where any initial value can be set;
2. Write data to CRC_DIN, and the CRC calculation is completed in the next clock cycle;
3. Read CRC results: Configure CRC_CR[CRCPNT] = 1, and read off CRC_DR in software to gets the high-order bytes. Configure CRC_CR[CRCPNT] = 0, and read off CRC_DR to get the low-order bytes.

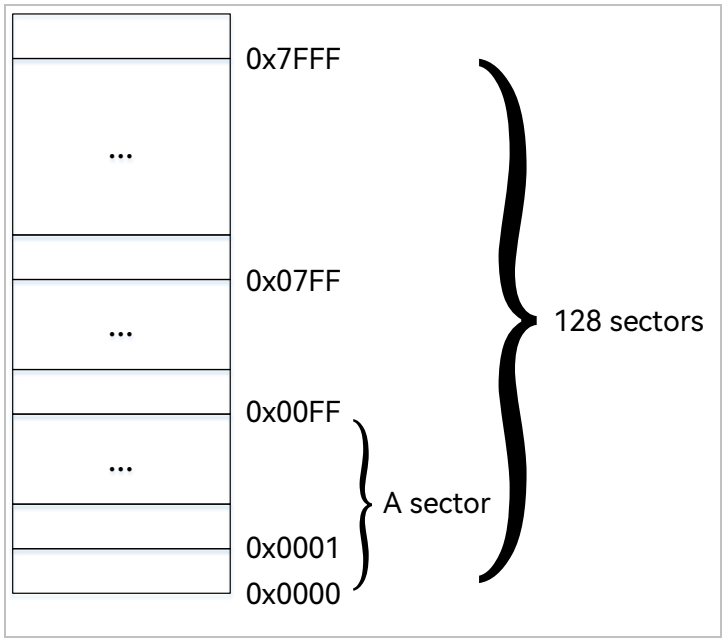
30.4.2 CRC Calculation of ROM Sector

CRC of a continuous area of data in the ROM is calculated as follows:

1. Initialize CRC_DR, in the same way as that of single-byte CRC calculation;
2. Configure CRC_BEG to define starting sector of the ROM to be calculated;

3. Configure CRC_CNT to set the offset from the starting sector to the ending sector;
4. Write “1” to CRC_CR[AUTOINT] and keep other bits unchanged. The calculation starts automatically;
5. Read the CRC results.

Figure 30-3 ROM Sectors



As shown in Figure 30-3, ROM contains 32k bytes and is divided into 128 sectors, numbered from sector0 to sector127. Each sector contains 256 bytes. For CRC calculation of sectors, the value of CRC_BEG (the starting sector) can be any value falling between 0x00 and 0xFF, including 0x00 and 0x7F. The CRC_CNT (total number of sectors to be calculated) can be any value between 0x00 ~ 0x7F, including 0x00 and 0xFF.

As CRC_BEG increases, CRC_CNT decreases accordingly. For example, if CRC_BEG is 0x7F, CRC_CNT can be 0x00 only, i.e., the CRC value of the data in the last sector is calculated. In this case, if CRC_CNT is large, CRC controller will automatically limits the number of sectors to be calculated. Finally, CRC module only calculates CRC value of the last sector.

30.5 CRC Registers

30.5.1 CRC_CR (0x4022)

Bit	7	6	5	4	3	2	1	0
Name	RSV			CRCDONE	CRCDINI	CRCVAL	AUTOINT	CRCPNT
Type	-	-	-	R	W1	R/W	W1	R/W
Reset	-	-	-	1	0	0	0	0

Bit	Name	Description
[7:5]	RSV	Reserved
[4]	CRCDONE	CRC Sector Calculation Completion Flag During the calculation, this bit is automatically set to “0” by hardware and the software program stops. In other cases, this bit is automatically set to “1” by the hardware, so the software always returns “1” when reading this bit.
[3]	CRCDINI	CRC Result Initialization Trigger 0: No effect. 1: CRC result initialization is triggered.
[2]	CRCVAL	CRC Result Initialization Selection 0: CRC result is initialized to 0x0000. 1: CRC result is initialized to 0xFFFF.
[1]	AUTOINT	CRC Sector Calculation Launch 0: No effect 1: Launch CRC Batch calculation See section CRC Calculation of ROM Sector
[0]	CRCPNT	CRC Result Pointer 0: Read CRC_DR to access 8 low-order bits of the 16-bit CRC result 1: Read CRC_DR to access 8 high-order bits of the 16-bit CRC result

Note

CRC_CR[AUTOINT] is set to “0” to perform single-byte CRC checksum

30.5.2 CRC_DIN (0x4021)

Bit	7	6	5	4	3	2	1	0
Name	CRC_DIN							
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	CRC_DIN	CRC Input Data Each time a data frame is written to this register, CRC module automatically

		calculates a new CRC result based on the existing CRC result, and overwrites the original one.  Note It is a virtual register, so the written data is not saved. 0x00 is returned when the address is accessed

30.5.3 CRC_DR (0x4023)

Bit	7	6	5	4	3	2	1	0
Name	CRC_DR							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	Name	Description
[7:0]	CRC_DR	CRC Result Output Each time this register is read or written, the configuration of CRC_CR[CRCPNT] determines whether to access the 8 high-order or low-order bits of the CRC result.

30.5.4 CRC_BEG (0x4024)

Bit	7	6	5	4	3	2	1	0
Name	RSV	CRC_BEG						
Type	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	0	0	0	0	0	0	0

Bit	Name	Description
[7]	RSV	Reserved
[6:0]	CRC_BEG	First ROM Sector Pending Automatic CRC Calculation Example: If CRC_BEG is set to “1”, CRC calculation starts from location 1*256 = 256, or rather from the first byte of sector 2.

30.5.5 CRC_CNT (0x4025)

Bit	7	6	5	4	3	2	1	0
Name	RSV	CRC_CNT						
Type	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	0	0	0	0	0	0	0

Bit	Name	Description
[7]	RSV	Reserved
[6:0]	CRC_CNT	Offset of Sector Pending Automatic CRC Calculation This bit defines the offset of ROM sector for CRC calculation and determines the last sector pending CRC calculation.

31 Sleep Mode

31.1 Introduction

The chip operates in three modes: normal mode, standby mode and sleep mode. These modes are selected by setting PCON[IDLE] and PCON[STOP].

The operating states of the module under different power modes are summarized in Table 31-1.

Table 31-1 Power Consumption Modes

Power Mode	Description	Wakeup Source	Power Consumption Performance
Normal	All modules work at full speed except for peripherals that are disabled	NA	High power consumption with best performance
Standby	CPU clock stops and other functional modules are either enabled or disabled depending on their control bit setting. WDT stops.	Any interrupt; Reset/Debug on external interrupt	Low power performance with flexible performance
Sleep	Flash Deep Sleep. The analog fast clock circuit is disconnected and MCU software shall ensure that ADC, FOC, and driver modules are idle before the chip enters the Sleep Mode. WDT is disabled.	External interrupt; RTC interrupt; Level changes of P4.7 in IO mode; Reset/Debug on external interrupt	Extremely low power performance with flexible performance

Note

It is recommended to insert 3 null statements in the sleep mode

```
PCON = 0x02;
_nop_();
_nop_();
_nop_();
```

31.2 Sleep Mode Register

31.2.1 PCON(0x87)

Bit	7	6	5	4	3	2	1	0
Name	RSV		GF3	GF2	GF1	RSV	STOP	IDLE
Type	-	-	R/W	R/W	R/W	-	R/W	R/W
Reset	-	-	0	0	0	-	0	0

Bit	Name	Description
[7:6]	RSV	Reserved
[5]	GF3	General-purpose flag bit 3
[4]	GF2	General-purpose flag bit 2
[3]	GF1	General-purpose flag bit 1
[2]	RSV	Reserved
[1]	STOP	A write of “1” makes the chip enter the sleep mode. This bit is automatically cleared to “0” by hardware after wakeup.
[0]	IDLE	A write of “1” makes the chip enter the standby mode. This bit is automatically cleared to “0” by hardware after wakeup.

Power Consumption Mode PCON[STOP: IDLE]:

00: Normal

01: Standby

1X: Sleep

32 Code Protection

32.1 Introduction

The chip supports Flash space encryption to protect your software intellectual property and avoid unauthorized access. When Flash memory is encrypted, the data inside cannot be read, and data consistency can be verified by CRC module only.

32.2 Operating Instructions

Figure 32-1 Code Protection Configurations

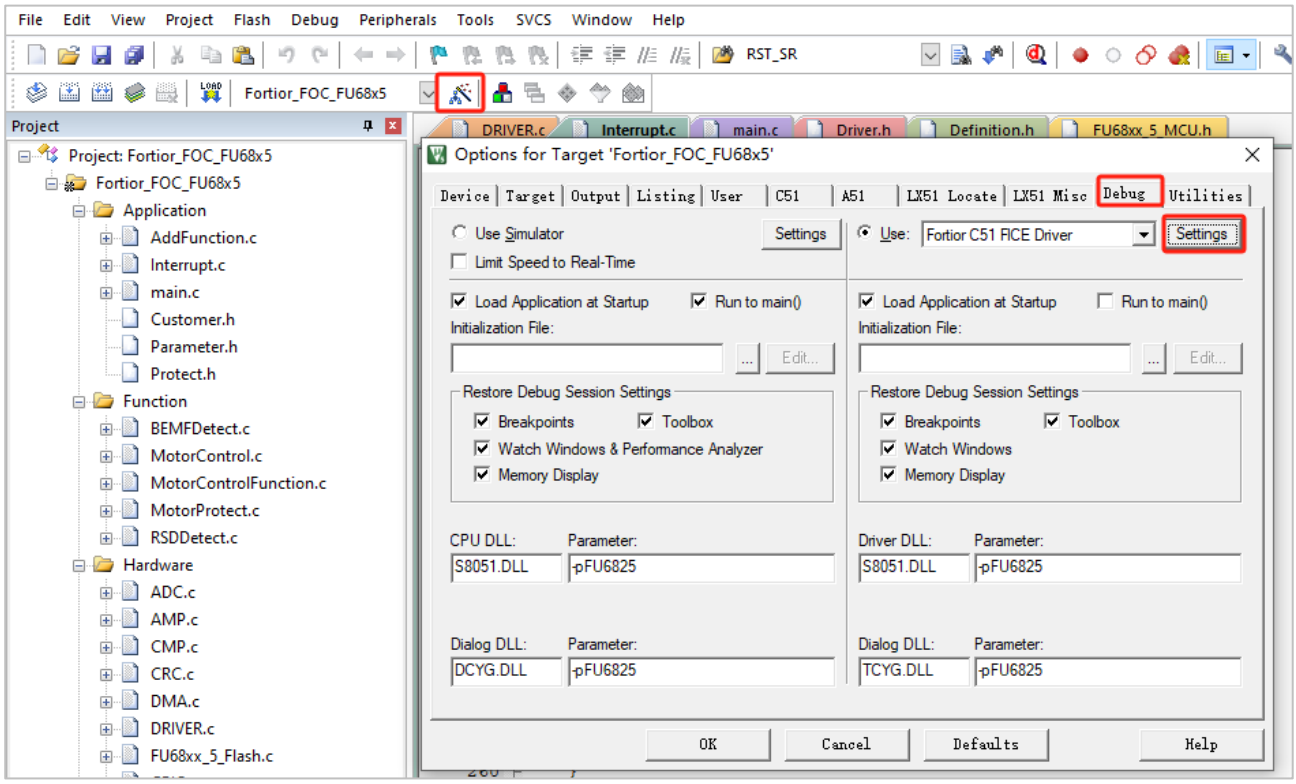
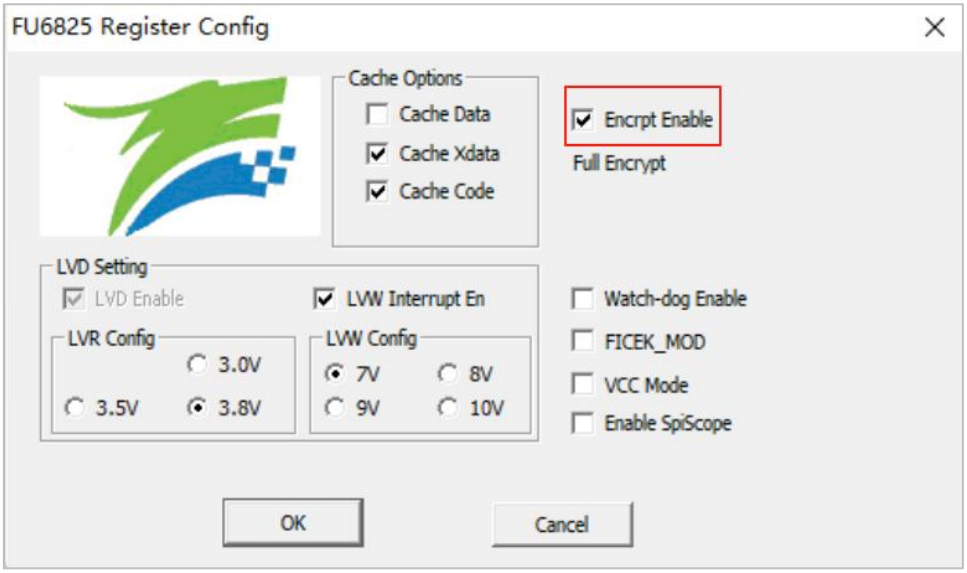


Figure 32-2 Full Code Protection Mode



Operation steps are as follows:

1. Start 8051 IDE, enter Target Options and select Debug tab. As shown in Figure 32-1, click Settings to proceed with the setting;

2. Select the options as shown in Figure 32-2, and click OK. Then compile the project and download it. Get the BIN file and program it to Flash.

33 Revision History

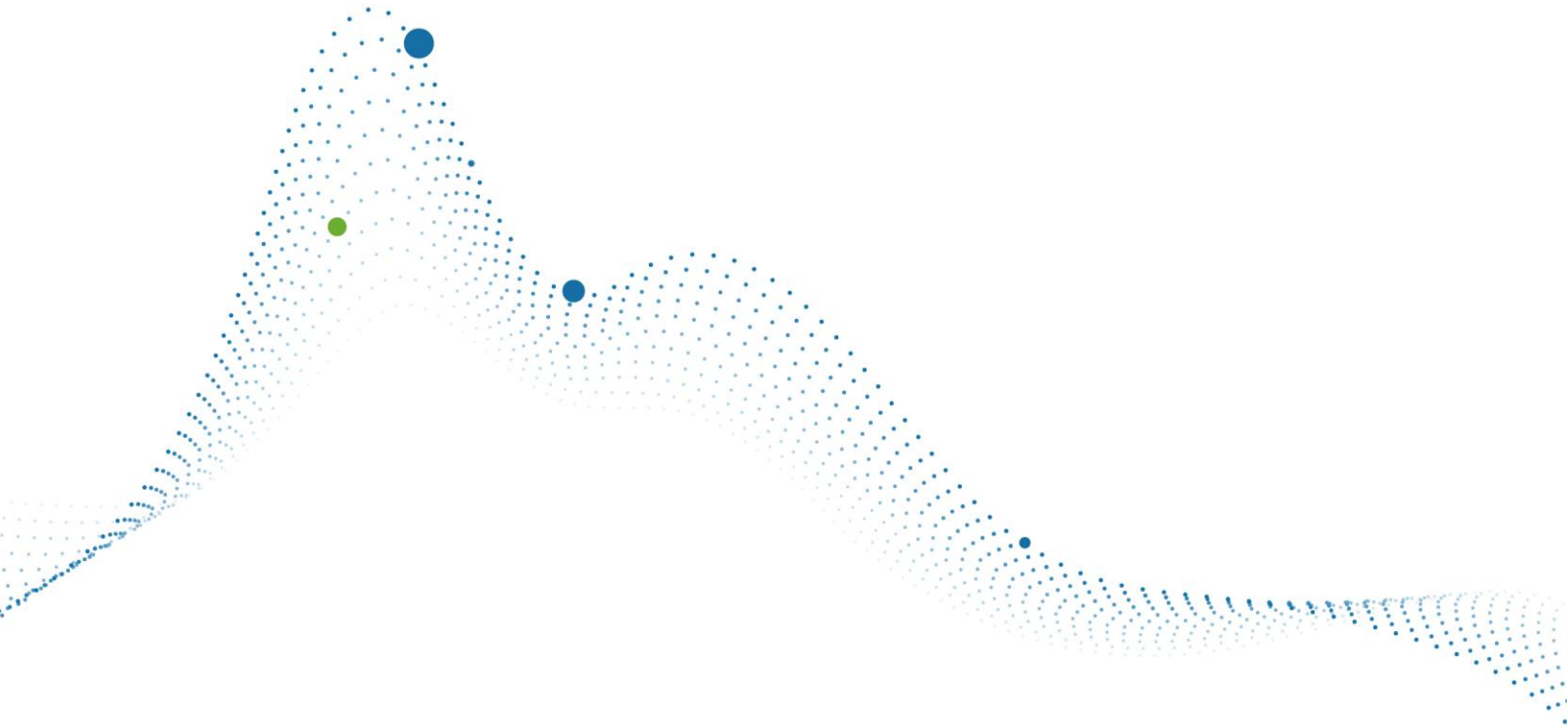
Rev.	Description	Date	Prepared By
V0.1	Preliminary datasheet	2025/07/18	Eric Deng
V0.2	1. Modified the functional block diagram in 1.4.1 FS95A6AS; 2. Modified description of pin 1, 5, and 27 in 2.1 FS95A6AS SSOP A54-38 Pins; Modified "FOC_CR2[CSOC]" as "FOC_CH2[CSOC]" in 12.1.8.3 Current Sampling Offset.	2025/08/13	Freya Fu
V0.3	1. Added 5.12 Mounting Torque; 2. Updated baud rate calculation formula in 9.3.3 UT_BAUD (0x9A, 0x9B) and 9.4.3 UT2_BAUD (0x4042, 0x4043); 3. Deleted descriptions of CXO pins.	2025/10/31	Freya Fu
V1.0	1. Deleted P4_OE in 1.5.3 SFR; 2. Deleted "FG generation" in Descriptions of TIM4 of P0.5; modified Description of TIM3 of P0.6 as "Timer3 input/out" in 2.1 FS95A6AS SSOP A54-38 Pins ; 3. Updated 2.2 FS95A6AS SSOP A54-38 Pinout Diagram; 4. Added Min. "-40℃" to Junction Temperature T_J in 5.1 Absolute Maximum Ratings; 5. Modified Min. and Max. of VDD5 Voltage as "4.8" and "5.2" in 5.10 LDO Electrical Characteristics; 6. Added 5.11 Switching Electrical Characteristics; 7. Modified baud rate calculation formula as "Baud rate = UARTCLK/(16/(1 + UT_BAUD[BAUD_SEL]))/(UT_BAUD[BAUD] + 1)" in 9.3.3 UT_BAUD (0x9A, 0x9B)[11:0]; 8. Modified baud rate calculation formula as " Baud rate = = UARTCLK/(16/(1 + UT_BAUD[BAUD_SEL]))/(UT_BAUD[BAUD] + 1) in 9.4.3 UT2_BAUD (0x4042, 0x4043)"; 9. Modified UAVG as PFC_UAVG, UAC as PFC_UAC, IAC as PFC_IAC, UTHR as PFC_CR1[UTHR], and UCTHR as PFC_CR1[UCTHR] in 11 PFC; 10. Modified the Typ. of 11.2.11 UDC_KP (0x40EC, 0x40ED) ~ 11.2.21 IAC_UKMIN (0x40FE, 0x40FF), 12.2.17 FOC_DMAX (0x4078), and 12.2.18 FOC_DMIN (0x4079) as R/W;	2026/01/16	Freya Fu

Rev.	Description	Date	Prepared By
	11. Deleted “PIRAN” in Figure 12-13 Schematic Block Diagram of Estimator; 12. Added 12.2.25 FOC_ID (0x40BC, 0x40BD) and 12.2.26 FOC_IQ (0x40BE, 0x40BF); 13. Optimized 13 Timer1 and simplified 13.3.10 TIM1_DBR1 (0x4074, 0x4075) ~ 13.3.16 TIM1_DBR7 (0x4080, 0x4081); 14. Modified “SYSCLK” as “T2CLK” and “TxCLK” in 14 Timer2 and 15 Timer3/Timer4; 15. Updated Figure 17-1 Block Diagram of FS95A6AS Driver Module; 16. Modified “CAL_CR0” as “CAL_CR” in 19.2 Clock Calibration, “CSOFDD” as “CSOFFD” in 27.1.5 Comparator Sampling and Description of bit [2:1] in 27.2.3 CMP_CR2 (0xDA) from “CMP0” as “CMP0/1/2”; 17. Modified bit [5:4] as “RSV”, description of bit [3:2] as “0: P0.5 as Timer4 I/O pin, 1: P1.2 as Timer4 I/O pin”, and deleted “P0.0 as Timer4 input/output”; Modified description of bit [1:0] as “0: P0.6 as Timer3 I/O pin 1: P1.2 as Timer4 I/O pin” and deleted “P0.1 as Timer3 I/O pin” in 21.3.15 PH_SEL1 (0x404D) 18. Modified description of bit [2] from “DMA Channel 0 Interrupt Enable ” as “ DMA Channel Interrupt Enable ” in 24.2.1 DMA0_CR0 (0x403A) 19. Modified “Vout” as “Vout” in 26 Operational Amplifier 20. Modified “When EVT_FILT[INT0_MOE_EN] = 0 and CMP_CR0[CMP3IM] = 11” as “When EVT_FILT[INT0_MOE_EN] = 0 and CMP_CR0[CMP3IM] = 01” in 27.1.1.1 Over-current Protection (OCP); 21. Turned to the public version.		

Fortior Technology (Shenzhen) Co., Ltd

Room 203, 2/F, Building No.11, Keji Central Road 2,
Software Park, High-Tech Industrial Park, Shenzhen, P.R.
China

Tel.: 0755-26867710
Fax: 0755-26867715
P.C.: 518057
Web: http://www.fortiortech.com



The content of this document
Copyright by Fortior Technology (Shenzhen) Co., Ltd. All Rights Reserved.