# Application Note

# BRT_AN_010

# FT93x User Manual

**Version 1.3**

**Issue Date: 16-10-2023**

This application note provides details about the peripherals of the FT93x as well as the general system registers.

Use of Bridgetek devices in life support and/or safety applications is entirely at the user's risk, and the user agrees to defend, indemnify and hold Bridgetek harmless from any and all damages, claims, suits or expense resulting from such use.

**Bridgetek Pte Ltd (BRT Chip)**
1 Tai Seng Avenue, Tower A, #03-05, Singapore 536464
Tel: +65 6547 4827
Web Site: http://www.brtchip.com
Copyright © Bridgetek Pte Ltd

# Table of Contents

**1 Introduction ..... 12**

**2 FT93x System Architecture ..... 13**

**2.1 Architecture Overview ..... 13**

**2.2 Memory Organization ..... 14**

**2.3 FT93x Boot Control ..... 14**

**2.4 Debugging Support ..... 15**

**3 Register Map ..... 16**

**4 Notations ..... 17**

**4.1 Bit Type Notation ..... 17**

**4.2 Default Bit Value Notation ..... 17**

**4.3 X Default Value ..... 17**

**4.4 Bit Range Symbol in Bit Column ..... 17**

**5 General System Registers ..... 19**

**5.1 Register Summary ..... 19**

**5.2 Register Details ..... 21**

5.2.1 CHIPID - Chip ID Register (address offset: 0x00) ..... 21

5.2.2 CHIPCFG - Chip Configuration Register (address offset: 0x04) ..... 21

5.2.3 CLKCFG - Clock Configuration Register (address offset: 0x08) ..... 22

5.2.4 PMCFG - Power Management Register (address offset: 0x0C) ..... 22

5.2.5 RESERVED (address offset: 0x10) ..... 24

5.2.6 RESERVED (address offset: 0x14) ..... 24

5.2.7 MSC0CFG - Miscellaneous Configuration Register (address offset: 0x18) ..... 24

5.2.8 GPIO Pin Configuration Registers (address offset: 0x1C – 0x5F) ..... 25

5.2.9 GPIO Configuration Registers (address offset: 0x60 – 0x70) ..... 28

5.2.10 GPIO Value Registers (address offset: 0x84 – 0x88) ..... 30

5.2.11 GPIO Interrupt Enable Registers (address offset: 0x90 – 0x94) ..... 30

5.2.12 Interrupt Pending Registers (address offset: 0x9C – 0xA0) ..... 31

5.2.13 DAC_ADC_CONF - ADC/DAC Configuration/Status Register (address offset: 0xB0) ..... 31

5.2.14 DAC_ADC_CNT - ADC/DAC Count Register (address offset: 0xB4) ..... 32

5.2.15 DAC_ADC_DATA - ADC/DAC Data Register (address offset: 0xB8) ..... 33



Copyright © Bridgetek Pte Ltd

5.2.16 GLOBAL_RESET_STATUS – Global Reset and Status Register (address offset: 0xBC) ..... 33

**6 Interrupt Controller ..... 34**

**6.1 Register Summary ..... 34**

**6.2 Register Details ..... 35**

6.2.1 IRQ00-03 Assignment Register (address offset: 0x00) ..... 35

6.2.2 IRQ04-07 Assignment Register (address offset: 0x04) ..... 35

6.2.3 IRQ08-11 Assignment Register (address offset: 0x08) ..... 36

6.2.4 IRQ12-15 Assignment Register (address offset: 0x0C) ..... 36

6.2.5 IRQ16-19 Assignment Register (address offset: 0x10) ..... 36

6.2.6 IRQ20-23 Assignment Register (address offset: 0x14) ..... 37

6.2.7 IRQ24-27 Assignment Register (address offset: 0x18) ..... 37

6.2.8 IRQ28-31 Assignment Register (address offset: 0x1C) ..... 37

6.2.9 IRQ Control Register (address offset: 0x20) ..... 38

**7 USB Device ..... 39**

**7.1 Register Summary ..... 39**

**7.2 Initialization Registers ..... 41**

7.2.1 DC_ADDRESS_ENABLE – Address Register (address offset: 0x18) ..... 41

7.2.2 DC_MODE – Mode Register (address offset: 0x10) ..... 41

7.2.3 DC_INT_ENABLE – Interrupt Enable Register (address offset: 0x08) ..... 41

7.2.4 DC_EP_INT_ENABLE – Endpoints Interrupt Enable Register (address offset: 0x0C) 42

**7.3 Control Endpoint Data Flow Registers ..... 42**

7.3.1 DC_EP0_CONTROL – Endpoint 0 Control Register (address offset: 0x1C) ..... 42

7.3.2 DC_EP0_STATUS – Endpoint 0 Status Register (address offset: 0x20) ..... 42

7.3.3 DC_EP0_BUFFER_LENGTH – Endpoint 0 Buffer Length Register (address offset: 0x24) ..... 43

7.3.4 DC_EP0_BUFFER – Endpoint 0 Buffer Register (address offset: 0x28) ..... 43

**7.4 Other Endpoint Data Flow Registers ..... 43**

7.4.1 DC_EP(x)_CONTROL – Endpoint Control Registers (address offset: 0x2C/0x3C/0x4C/0x5C/0x6C/0x7C/0x8C) ..... 44

7.4.2 DC_EP(x)_STATUS – Endpoint Status Registers (address offset: 0x30/0x40/0x50/0x60/0x70/0x80/0x90) ..... 44

7.4.3 DC_EP(x)_BUFFER_LENGTH_LSB – Endpoint Buffer Length LSB Registers (address offset: 0x34/0x44/0x54/0x64/0x74/0x84/0x94) ..... 45

7.4.4 DC_EP(x)_BUFFER_LENGTH_MSB – Endpoint Buffer Length MSB Registers (address offset: 0x35/0x45/0x55/0x65/0x75/0x85/0x95) ..... 46



Copyright © Bridgetek Pte Ltd

7.4.5 DC_EP(x)_BUFFER – Endpoint Buffer Registers (address offset: 0x38/0x48/0x58/0x68/0x78/0x88/0x98) ..... 46

**7.5 General Registers ..... 46**

7.5.1 DC_INT_STATUS – Interrupt Status Register (address offset: 0x00)..... 46

7.5.2 DC_EP_INT_STATUS – Endpoints Interrupt Status Register (address offset: 0x04) 46

7.5.3 DC_FRAME_NUMBER_LSB – Frame Number LSB Register (address offset: 0x14) .. 47

7.5.4 DC_FRAME_NUMBER_MSB – Frame Number MSB Register (address offset: 0x15). 47

**8 SD Host ..... 48**

**8.1 Register Summary ..... 48**

**8.2 Register Details ..... 50**

8.2.1 SDH_AUTO_CMD23_ARG2 – Auto CMD23 Argument 2 Register (address offset: 0x00) ..... 50

8.2.2 SDH_BLK_SIZE – Block Size Register (address offset: 0x04) ..... 51

8.2.3 SDH_BLK_COUNT – Block Count Register (address offset: 0x06) ..... 51

8.2.4 SDH_ARG_1 – Argument 1 Register (address offset: 0x08) ..... 51

8.2.5 SDH_TNSFER_MODE – Transfer Mode Register (address offset: 0x0C)..... 51

8.2.6 SDH_CMD – Command Register (address offset: 0x0E) ..... 52

8.2.7 SDH_RESPONSE – Response Register (address offset: 0x10-0x1C) ..... 53

8.2.8 SDH_BUF_DATA – Buffer Data Port Register (address offset: 0x20) ..... 53

8.2.9 SDH_PRESENT_STATE – Present State Register (address offset: 0x24)..... 53

8.2.10 SDH_HST_CNTL_1 – Host Control 1 Register (address offset: 0x28)..... 55

8.2.11 SDH_PWR_CNTL – Power Control Register (address offset: 0x29) ..... 55

8.2.12 SDH_BLK_GAP_CNTL – Block Gap Control Register (address offset: 0x2A) ..... 56

8.2.13 SDH_CLK_CNTL – Clock Control Register (address offset: 0x2C) ..... 56

8.2.14 SDH_TIMEOUT_CNTL – Timeout Control Register (address offset: 0x2E)..... 57

8.2.15 SDH_SW_RST – Software Reset Register (address offset: 0x2F) ..... 57

8.2.16 SDH_NRML_INT_STATUS – Normal Interrupt Status Register (address offset: 0x30) ..... 57

8.2.17 SDH_ERR_INT_STATUS – Error Interrupt Status Register (address offset: 0x32) 58

8.2.18 SDH_NRML_INT_ENABLE – Normal Interrupt Status Enable Register (address offset: 0x34) ..... 58

8.2.19 SDH_ERR_INT_ENABLE – Error Interrupt Status Enable Register (address offset: 0x36) ..... 59

8.2.20 SDH_NRML_INT_SGNL_ENABLE – Normal Interrupt Signal Enable Register (address offset: 0x38) ..... 60

8.2.21 SDH_ERR_INT_SGNL_ENABLE – Error Interrupt Signal Enable Register (address offset: 0x3A) ..... 60


Copyright © Bridgetek Pte Ltd

8.2.22 SDH_AUTO_CMD12_ERR_STATUS – Auto CMD12 Error Status Register (address offset: 0x3C) ..... 61
8.2.23 SDH_HOST_CNTL_2 – Host Control 2 Register (address offset: 0x3E)..... 61
8.2.24 SDH_CAP_1 – Capabilities Register 1 (address offset: 0x40) ..... 61
8.2.25 SDH_CAP_2 – Capabilities Register 2 (address offset: 0x44) ..... 62
8.2.26 SDH_RSRV_1 – Reserved 1 Register (address offset: 0x48) ..... 63
8.2.27 SDH_RSRV_2 – Reserved 2 Register (address offset: 0x4C) ..... 63
8.2.28 SDH_FORCE_EVT_CMD_ERR_STATUS – Force Event Register for Auto CMD Error Status (address offset: 0x50) ..... 63
8.2.29 SDH_FORCE_EVT_ERR_INT_STATUS – Force Event for Error Interrupt Status Register (address offset: 0x52) ..... 64
8.2.30 SDH_RSRV_3 – Reserved 3 Register (address offset: 0x54) ..... 64
8.2.31 SDH_RSRV_4 – Reserved 4 Register (address offset: 0x58) ..... 64
8.2.32 SDH_PRST_INIT - Preset value for initialization (address offset: 0x60) ..... 65
8.2.33 SDH_PRST_DFLT_SPD – Preset value for default speed (address offset: 0x62) ... 65
8.2.34 SDH_PRST_HIGH_SPD – Preset value for the high speed (address offset: 0x64) . 65
8.2.35 SDH_PRST_SDR12 – Preset value for SDR12 (address offset: 0x66)..... 66
8.2.36 SDH_PRST_SDR25 – Preset value for SDR25 (address offset: 0x68)..... 66
8.2.37 SDH_PRST_SDR50 – Preset value for SDR50 (address offset: 0x6A)..... 67
8.2.38 SDH_PRST_SDR104 – Preset value for SDR104 (address offset: 0x6C) ..... 67
8.2.39 SDH_PRST_DDR50 – Preset value for DDR50 (address offset: 0x6E) ..... 67
8.2.40 SDH_RSRV_5 – Reserved 5 Register (address offset: 0xFC) ..... 68
8.2.41 SDH_HC_VER – Host Controller Version Register (address offset: 0xFE)..... 68
8.2.42 SDH_VNDR_0 – Vendor-defined 0 Register (address offset: 0x100)..... 68
8.2.43 SDH_VNDR_1 – Vendor-defined 1 Register (address offset: 0x104)..... 69
8.2.44 SDH_VNDR_2 – Vendor-defined 2 Register (address offset: 0x108)..... 69
8.2.45 SDH_VNDR_3 – Vendor-defined 3 Register (address offset: 0x10C) ..... 70
8.2.46 SDH_VNDR_4 – Vendor-defined 4 Register (address offset: 0x110)..... 70
8.2.47 SDH_VNDR_5 – Vendor-defined 5 Register (address offset: 0x114)..... 70
8.2.48 SDH_VNDR_6 – Vendor-defined 6 Register (address offset: 0x118)..... 70
8.2.49 SDH_VNDR_7 – Vendor-defined 7 Register (address offset: 0x11C) ..... 70
8.2.50 SDH_VNDR_8 – Vendor-defined 8 Register (address offset: 0x120)..... 71
8.2.51 SDH_VNDR_9 – Vendor-defined 9 Register (address offset: 0x124)..... 71
8.2.52 SDH_RSRV_6 – Reserved 6 Register (address offset: 0x128)..... 71
8.2.53 SDH_HW_ATTR – Hardware Attributes Register (address offset: 0x178) ..... 71
8.2.54 SDH_CPR_MOD_CNTL – Cipher Mode Control Register (address offset: 0x180)... 71


Copyright © Bridgetek Pte Ltd

8.2.55 SDH_CPR_MOD_STATUS – Cipher Mode Status Register (address offset: 0x184) 72
8.2.56 SDH_CPR_MOD_STATUS_EN – Cipher Mode Status Enable Register (address offset: 0x188) ..... 73
8.2.57 SDH_CPR_MOD_SIG_EN – Cipher Mode Signal Enable Register (address offset: 0x18A) ..... 73
8.2.58 SDH_IN_DATA_LSB – Input Data LSB Register (address offset: 0x18C) ..... 73
8.2.59 SDH_IN_DATA_MSB - Input Data MSB Register (address offset: 0x190) ..... 73
8.2.60 SDH_IN_KEY_LSB – Input Key LSB Register (address offset: 0x194) ..... 73
8.2.61 SDH_IN_KEY_MSB – Input Key MSB Register (address offset: 0x198) ..... 73
8.2.62 SDH_OUT_DATA_LSB – Output Data LSB Register (address offset: 0x19C) ..... 74
8.2.63 SDH_OUT_DATA_MSB – Output Data MSB Register (address offset: 0x1A0) ..... 74
8.2.64 SDH_SCRT_CONS_DATA – Secret Constant Table Data Port (address offset: 0x1A4) ..... 74

**9 UART ..... 75**

**9.1 Register Summary ..... 76**

**9.2 UART MODE SELECTION ..... 78**

**9.3 STANDARD 550 COMPATIBLE REGISTERS ..... 79**

9.3.1 UART_RBR - Receiver Buffer Register (address offset: 0x00 and LCR[7] = 0) ..... 79
9.3.2 UART_THR - Transmitter Holding Register (address offset: 0x00 and LCR[7] = 0) . 80
9.3.3 UART_DIV_LSB - Divisor LSB Register (address offset: 0x00 and LCR[7] = 1) ..... 80
9.3.4 UART_DIV_MSB - Divisor MSB Register (address offset: 0x01 and LCR[7] = 1) .... 80
9.3.5 UART_INT_ENABLE - Interrupt Enable Register (address offset: 0x01) ..... 80
9.3.6 UART_INT_STATUS - Interrupt Status Register (address offset: 0x02)..... 81
9.3.7 UART_FCR - FIFO Control Register (address offset: 0x02) ..... 82
9.3.8 UART_LCR - Line Control Register (address offset: 0x03) ..... 84
9.3.9 UART_MCR - Modem Control Register (address offset: 0x04)..... 84
9.3.10 UART_LSR - Line Status Register (address offset: 0x05) ..... 85
9.3.11 UART_MSR - Modem Status Register (address offset: 0x06) ..... 87
9.3.12 UART_SPR - SPR Register (address offset: 0x07) ..... 87

**9.4 650 COMPATIBLE REGISTERS ..... 87**

9.4.1 UART_EFR - Enhanced Feature Register (address offset: 0x02)..... 88
9.4.2 UART_XON1 - XON1 Register (address offset: 0x04) ..... 88
9.4.3 UART_XON2 - XON2 Register (address offset: 0x05) ..... 88
9.4.4 UART_XOFF1 - XOFF1 Register (address offset: 0x06) ..... 88
9.4.5 UART_XOFF2 - XOFF2 Register (address offset: 0x07) ..... 89


Copyright © Bridgetek Pte Ltd

**9.5 950 COMPATIBLE REGISTERS ..... 89**
9.5.1 UART_ASR - Additional Status Register (address offset: 0x01) ..... 89
9.5.2 UART_RFL - Receiver FIFO Level Register (address offset: 0x03) ..... 89
9.5.3 UART_TFL - Transmitter FIFO Level Register (address offset: 0x04) ..... 90
9.5.4 UART_ICR - ICR Register (address offset: 0x05) ..... 90
**9.6 INDEXED CONTROL REGISTERS ..... 90**
9.6.1 UART_ACR - Additional Control Register (SPR offset: 0x00) ..... 92
9.6.2 UART_CPR - Clock Prescaler Register (SPR offset: 0x01) ..... 92
9.6.3 UART_TCR - Time Clock Register (SPR offset: 0x02) ..... 93
9.6.4 UART_CKS - Clock Select Register (SPR offset: 0x03) ..... 93
9.6.5 UART_TTL - Transmitter Trigger Level Register (SPR offset: 0x04) ..... 94
9.6.6 UART_RTL - Receiver Trigger Level Register (SPR offset: 0x05) ..... 94
9.6.7 UART_FCL - Flow Control Level LSB Register (SPR offset: 0x06) ..... 94
9.6.8 UART_FCH - Flow Control Level Register MSB (SPR offset: 0x07) ..... 94
9.6.9 UART_ID1 - Identification 1 Register (SPR offset: 0x08) ..... 95
9.6.10 UART_ID2 - Identification 2 Register (SPR offset: 0x09) ..... 95
9.6.11 UART_ID3 - Identification 3 Register (SPR offset: 0x0A) ..... 95
9.6.12 UART_REV - Revision Register (SPR offset: 0x0B) ..... 95
9.6.13 UART_CSR - Channel Software Reset Register (SPR offset: 0x0C) ..... 95
9.6.14 UART_NMR - Nine Bit Mode Register (SPR offset: 0x0D) ..... 95
9.6.15 UART_MDM - Modem Disable Mask Register (SPR offset: 0x0E) ..... 96
9.6.16 UART_RFC - Readable FCR Register (SPR offset: 0x0F) ..... 97
9.6.17 UART_GDS - Good Data Status Register (SPR offset: 0x10) ..... 97
9.6.18 UART_RSRV_1 - Reserved 1 Register (SPR offset: 0x11) ..... 97
9.6.19 UART_PIDX - Port Index Register (SPR offset: 0x12) ..... 97
9.6.20 UART_CKA - Clock Alteration Register (SPR offset: 0x13) ..... 97
**10 Timers and Watchdog ..... 98**
**10.1 Register Summary ..... 99**
**10.2 Register Details ..... 99**
10.2.1 TIMER_CONTROL_0 - Timers Control Register 0 (address offset: 0x00) ..... 99
10.2.2 TIMER_CONTROL_1 - Timers Control Register 1 (address offset: 0x01) ..... 99
10.2.3 TIMER_CONTROL_2 - Timers Control Register 2 (address offset: 0x02) ..... 100
10.2.4 TIMER_CONTROL_3 - Timers Control Register 3 (address offset: 0x03) ..... 100
10.2.5 TIMER_CONTROL_4 - Timers Control Register 4 (address offset: 0x04) ..... 100


Copyright © Bridgetek Pte Ltd

10.2.6 TIMER_INT - Timers Interrupt Register (address offset: 0x05) ..... 100
10.2.7 TIMER_SELECT - Timers A..D Select Register (address offset: 0x06) ..... 101
10.2.8 TIMER_WDG - Watchdog Start Value (address offset: 0x07) ..... 101
10.2.9 TIMER_WRITE_LS - Timer A..D Start Value 7:0 (address offset: 0x08) ..... 101
10.2.10 TIMER_WRITE_MS - Timer A..D Start Value 15:8 (address offset: 0x09) ..... 101
10.2.11 TIMER_PRESC_LS - Prescaler Start Value 7:0 (address offset: 0x0A) ..... 101
10.2.12 TIMER_PRESC_MS - Prescaler Start Value 15:8 (address offset: 0x0B) ..... 101
10.2.13 TIMER_READ_LS - Timer A..D Current Value 7:0 (address offset: 0x0C) ..... 102
10.2.14 TIMER_READ_MS - Timer A..D Current Value 15:8 (address offset: 0x0D) ..... 102

**11 SPI Master ..... 103**

**11.1 Register Summary ..... 103**

**11.2 Register Details ..... 104**

11.2.1 SPIM_CNTL – Control Register (address offset: 0x00) ..... 104
11.2.2 SPIM_STATUS – Status Register (address offset: 0x04) ..... 104
11.2.3 SPIM_DATA – Receiver and Transmitter Data Registers (address offset: 0x08) ..105
11.2.4 SPIM_SLV_SEL_CNTL – Slave Select Control Register (address offset: 0x0C) ..... 105
11.2.5 SPIM_FIFO_CNTL – FIFO Control Register (address offset: 0x10) ..... 105
11.2.6 SPIM_TNSFR_FRMT_CNTL – Transfer Format Control Register (address offset: 0x14) ..... 106
11.2.7 SPIM_ALT_DATA – Alternative SPI Master Data Register (address offset: 0x18) .106
11.2.8 SPIM_RX_FIFO_COUNT – SPI Master RX FIFO Count Register (address offset: 0x1C) ..... 106
11.2.9 SPIM_CNTL_2 – Control 2 Register (address offset: 0x20) ..... 107
11.2.10 SPIM_STATUS_2 – Status 2 Register (address offset: 0x30) ..... 107
11.2.11 SPIM_FIFO_CNTL_2 – FIFO Control 2 Register (address offset: 0x34) ..... 107
11.2.12 SPIM_TX_FIFO_COUNT – SPI Master TX FIFO Count Register (address offset: 0x38) ..... 108
11.2.13 SPIM_BAUD – SPI Master Baud Register (address offset: 0x3C) ..... 108

**12 SPI Slave ..... 109**

**12.1 Register Summary ..... 109**

**12.2 Register Details ..... 110**

12.2.1 SPIS_CNTL – Control Register (address offset: 0x00) ..... 110
12.2.2 SPIS_STATUS – Status Register (address offset: 0x04) ..... 110
12.2.3 SPIS_DATA – Receiver and Transmitter Data Registers (address offset: 0x08) ...111
12.2.4 SPIS_SLV_SEL_CNTL – Slave Select Control Register (address offset: 0x0C) ..... 111


Copyright © Bridgetek Pte Ltd

12.2.5 SPIS_FIFO_CNTL – FIFO Control Register (address offset: 0x10) ..... 111
12.2.6 SPIS_TNSFR_FRMT_CNTL – Transfer Format Control Register (address offset: 0x14) ..... 112
12.2.7 SPIS_ALT_DATA – Alternative SPI Slave Data Register (address offset: 0x18) ... 112
12.2.8 SPIS_RX_FIFO_COUNT – SPI Slave RX FIFO Count Register (address offset: 0x1C) ..... 112
12.2.9 SPIS_CNTL_2 – Control 2 Register (address offset: 0x20) ..... 112
12.2.10 SPIS_STATUS_2 – Status 2 Register (address offset: 0x30)..... 113
12.2.11 SPIS_FIFO_CNTL_2 – FIFO Control 2 Register (address offset: 0x34) ..... 114
12.2.12 SPIS_TX_FIFO_COUNT – SPI Slave TX FIFO Count Register (address offset: 0x38) ..... 114
12.2.13 SPIS_BAUD – SPI Slave Baud Register (address offset: 0x3C) ..... 114

**13 I²C Master ..... 115**

**13.1 Register Summary ..... 115**

**13.2 Register Details ..... 116**

13.2.1 I2CM_SLV_ADDR – Slave Address Register (address offset: 0x00) ..... 116
13.2.2 I2CM_CNTL – Control Register (address offset: 0x01)..... 116
13.2.3 I2CM_STATUS – Status Register (address offset: 0x01) ..... 116
13.2.4 I2CM_DATA – Receive / Transmit Data Register (address offset: 0x02)..... 117
13.2.5 I2CM_TIME_PERIOD – Timer Period Register (address offset: 0x03)..... 117
13.2.6 I2CM_HS_TIME_PERIOD – High Speed Timer Period Register (address offset: 0x03) ..... 117
13.2.7 I2CM_FIFO_LEN – FIFO Mode Byte Length (address offset: 0x04) ..... 118
13.2.8 I2CM_FIFO_INT_ENABLE – FIFO Mode Interrupt Enable (address offset: 0x05)..118
13.2.9 I2CM_FIFO_INT_PEND – FIFO Mode Interrupt Pending (address offset: 0x06) ...118
13.2.10 I2CM_FIFO_DATA - FIFO Data Register (address offset: 0x07)..... 119
13.2.11 I2CM_TRIG - Trigger Register (address offset: 0x08)..... 119

**14 I²C Slave ..... 120**

**14.1 Register Summary ..... 120**

**14.2 Register Details ..... 120**

14.2.1 I2CS_OWN_ADDR – Own Address Register (address offset: 0x00) ..... 120
14.2.2 I2CS_CNTL – Control Register (address offset: 0x01) ..... 121
14.2.3 I2CS_STATUS – Status Register (address offset: 0x01) ..... 121
14.2.4 I2CS_DATA – Receive / Transmit Data Register (address offset: 0x02) ..... 122

**15 RTC ..... 123**


Copyright © Bridgetek Pte Ltd

**15.1 Register Summary ..... 123**
**15.2 Register Details ..... 123**
15.2.1 RTC_TIME – Time Register (address offset: 0x00) ..... 123
15.2.2 RTC_DATE – Date Register (address offset: 0x04) ..... 124
15.2.3 RTC_AALARM1 – Auto Alarm 1 Register (address offset: 0x08) ..... 124
15.2.4 RTC_AALARM2 – Auto Alarm 2 Register (address offset: 0x0C) ..... 125
15.2.5 RTC_CTRL – Control Register (address offset: 0x10) ..... 125
15.2.6 RTC_STAT - Status Register (address offset: 0x14) ..... 126
15.2.7 RTC_TRIM – Trimming Register (address offset: 0x18) ..... 127
15.2.8 RTC_SLE_TIME – Sleep Time Register (address offset: 0x1C) ..... 127
15.2.9 RTC_SLE_DATE – Sleep Date Register (address offset: 0x20) ..... 128
15.2.10 RTC_COMP_PER – Compare Period Register (address offset: 0x24) ..... 128
15.2.11 RTC_VREF_TRIM – LVREF Trimming Value (address offset: 0x28) ..... 128
15.2.12 I2CS_FIFO_DATA - FIFO Data Register (address offset: 0x07) ..... 129
15.2.13 I2CS_TRIG - Trigger Register (address offset: 0x08) ..... 129
15.2.14 I2CS_FIFO_LEN – FIFO Mode Byte Length (address offset: 0x04) ..... 129
15.2.15 I2CS_FIFO_INT_ENABLE – FIFO Mode Interrupt Enable (address offset: 0x05) 129
15.2.16 I2CS_FIFO_INT_PEND – FIFO Mode Interrupt Pending (address offset: 0x06) .. 130
**16 PWM ..... 131**
**16.1 Register Summary ..... 131**
**16.2 Register Details ..... 132**
16.2.1 PWM_CTRL0 - PCM Control Register (address offset: 0x00) ..... 132
16.2.2 PWM_CTRL1 - PWM Control Register (address offset: 0x01) ..... 133
16.2.3 PWM_PRESCALER - PWM Prescaler Register (address offset: 0x02) ..... 133
16.2.4 PWM_CNTL - PWM Counter Register (LSB) (address offset: 0x03) ..... 134
16.2.5 PWM_CNTH - PWM Counter Register (MSB) (address offset: 0x04) ..... 134
16.2.6 PWM_CMP0L - Comparator 0 Value Register (LSB) (address offset: 0x05) ..... 134
16.2.7 PWM_CMP0H - Comparator 0 Value Register (MSB) (address offset: 0x06) ..... 134
16.2.8 PWM_CMP1L - Comparator 1 Value Register (LSB) (address offset: 0x07) ..... 134
16.2.9 PWM_CMP1H - Comparator 1 Value Register (MSB) (address offset: 0x08) ..... 134
16.2.10 PWM_CMP2L - Comparator 2 Value Register (LSB) (address offset: 0x09) ..... 135
16.2.11 PWM_CMP2H - Comparator 2 Value Register (MSB) (address offset: 0x0A) ..... 135
16.2.12 PWM_CMP3L - Comparator 3 Value Register (LSB) (address offset: 0x0B) ..... 135
16.2.13 PWM_CMP3H - Comparator 3 Value Register (MSB) (address offset: 0x0C) ..... 135


Copyright © Bridgetek Pte Ltd

16.2.14 PWM_CMP4L - Comparator 4 Value Register (LSB) (address offset: 0x0D) ..... 135
16.2.15 PWM_CMP4H - Comparator 4 Value Register (MSB) (address offset: 0x0E) ..... 135
16.2.16 PWM_CMP5L - Comparator 5 Value Register (LSB) (address offset: 0x0F) ..... 136
16.2.17 PWM_CMP5H - Comparator 5 Value Register (MSB) (address offset: 0x10) ..... 136
16.2.18 PWM_CMP6L - Comparator 6 Value Register (LSB) (address offset: 0x11) ..... 136
16.2.19 PWM_CMP6H - Comparator 6 Value Register (MSB) (address offset: 0x12) ..... 136
16.2.20 PWM_CMP7L - Comparator 7 Value Register (LSB) (address offset: 0x13) ..... 136
16.2.21 PWM_CMP7H - Comparator 7 Value Register (MSB) (address offset: 0x14) ..... 137
16.2.22 PWM_TOGGLE0 - Channel 0 OUT Toggle Comparator Mask Register (address offset: 0x15) ..... 137
16.2.23 PWM_TOGGLE1 - Channel 1 OUT Toggle Comparator Mask Register (address offset: 0x16) ..... 137
16.2.24 PWM_TOGGLE2 - Channel 2 OUT Toggle Comparator Mask Register (address offset: 0x17) ..... 137
16.2.25 PWM_TOGGLE3 - Channel 3 OUT Toggle Comparator Mask Register (address offset: 0x18) ..... 137
16.2.26 PWM_TOGGLE4 - Channel 4 OUT Toggle Comparator Mask Register (address offset: 0x19) ..... 137
16.2.27 PWM_TOGGLE5 - Channel 5 OUT Toggle Comparator Mask Register (address offset: 0x1A) ..... 138
16.2.28 PWM_TOGGLE6 - Channel 6 OUT Toggle Comparator Mask Register (address offset: 0x1B) ..... 138
16.2.29 PWM_TOGGLE7 - Channel 7 OUT Toggle Comparator Mask Register (address offset: 0x1C) ..... 138
16.2.30 PWM_OUT_CLR_EN - PWM OUT Clear Enable Register (address offset: 0x1D) ..138
16.2.31 PWM_CTRL_BL_CMP8 - Control Block CMP8 Value Register (address offset: 0x1E) ..... 138
16.2.32 PWM_INIT - PWM Initialization Register (address offset: 0x1F) ..... 138
16.2.33 PWM_INTMASK - PWM Interrupt Mask Register (address offset: 0x20) ..... 139
16.2.34 PWM_INTSTATUS - PWM Interrupt Status Register (address offset: 0x21) ..... 139
16.2.35 PWM_SAMPLE_FREQ_H - PWM Data Sampling Frequency High Byte Register (address offset: 0x22) ..... 139
16.2.36 PWM_SAMPLE_FREQ_L - PWM Data Sampling Frequency Low Byte Register (address offset: 0x23) ..... 140
16.2.37 PCM_VOLUME - PCM Volume Register (address offset: 0x24) ..... 140
16.2.38 PWM_BUFFER - PCM Buffer Register (address offset: 0x3C) ..... 140

**17 Flash Controller ..... 141**

**17.1 Register Summary ..... 141**


Copyright © Bridgetek Pte Ltd

**17.2 Register Details ..... 142**
17.2.1 RSADDR0 – Memory Start Address Register (LSB) (address offset: 0x00) ..... 142
17.2.2 RSADDR1 – Memory Start Address Register (Byte 1) (address offset: 0x01) ..... 142
17.2.3 RSADDR2 – Memory Start Address Register (MSB) (address offset: 0x02) ..... 142
17.2.4 FSADDR0 – Flash Start Address Register (LSB) (address offset: 0x03) ..... 143
17.2.5 FSADDR1 – Flash Start Address Register (Byte 1) (address offset: 0x04) ..... 143
17.2.6 FSADDR2 – Flash Start Address Register (MSB) (address offset: 0x05) ..... 143
17.2.7 BLENGTH0 – Data Byte Length Register (LSB) (address offset: 0x06) ..... 143
17.2.8 BLENGTH1 – Data Byte Length Register (Byte 1) (address offset: 0x07) ..... 143
17.2.9 BLENGTH2 – Data Byte Length Register (MSB) (address offset: 0x08) ..... 143
17.2.10 COMMAND – Command Register (address offset: 0x09) ..... 144
17.2.11 SEMAPHORE – Semaphore Register (address offset: 0x0B) ..... 144
17.2.12 CONFIG – Configuration Register (address offset: 0x0C) ..... 144
17.2.13 STATUS – Status Register (address offset: 0x0D) ..... 145
17.2.14 CRCL – Flash Content CRC Register (LSB) (address offset: 0x0E) ..... 145
17.2.15 CRCH – Flash Content CRC Register (MSB) (address offset: 0x0F) ..... 145
17.2.16 CHIPID0 – Chip ID Register (LSB) (address offset: 0x7C) ..... 145
17.2.17 CHIPID1 – Chip ID Register (Byte 1) (address offset: 0x7D) ..... 145
17.2.18 CHIPID2 – Chip ID Register (Byte 2) (address offset: 0x7E) ..... 146
17.2.19 CHIPID3 – Chip ID Register (MSB) (address offset: 0x7F) ..... 146
17.2.20 DRWDATA – Data Register (address offset: 0x80) ..... 146
**17.3 Flash Controller Commands ..... 146**
**18 Contact Information ..... 150**
**Appendix A – References ..... 151**
**Document References ..... 151**
**Acronyms and Abbreviations ..... 151**
**Appendix B – List of Tables & Figures ..... 153**
**List of Tables ..... 153**
**List of Figures ..... 161**
**Appendix C – Revision History ..... 162**


Copyright © Bridgetek Pte Ltd

# 1 Introduction

FT93x is a programmable System-on-chip device that sports a 32-bit RISC core (FT32B) that runs at 100MHz and is complemented by a D2xx[1] hardware engine and a wide assortment of connectivity options. It is ideally suited as a high-speed data bridging workhorse. The description of the general system registers, as well as the register set of various peripheral interfaces, is explained in detail in this document.

[1] Used under agreement between BRT and FTDI, D2xx is the popular FTDI USB driver used on its FT2xx USB UART/FIFO Bridge platforms. It can be accessed directly or through a virtual COM port.


Copyright © Bridgetek Pte Ltd

# 2 FT93x System Architecture

## 2.1 Architecture Overview

The diagram below shows the block diagram of FT93x. The FT93x contains the latest variant of the FT32 core, called FT32B. FT32B supports mixed 16-bit and 32-bit instructions which helps to increase code density and relieves flash memory pressure. 128kB of flash storage is provided on-chip and which is fully shadowed into a 128kB zero wait-state program memory area for fast execution. 32kB of data memory is available for stack, global and other data. The one-wire module is used as the debug access port and doubles up as a flash download port. All peripherals in the FT93x are memory mapped.

The features of the FT93x series include:

- 100MHz 16-bit and 32-bit mixed mode instructions capable RISC core
- USB 2.0 Compliant High Speed Device Controller
- Battery Charger Detection conformant to USB Battery Charging Specification Rev 1.2
- Nested Vectored Interrupt Controller (NVIC)
- 4 x UARTs
- 1 x SPI master interface
- 1 x SPI slave interface
- 8 x PWM channels with digital filter on channel 0 and 1
- 1 x I²C master interface
- 1 x I²C slave interface
- 1 x SD-Host Controller conformant to SD Association SD Host Controller Specification V3.0
- 1 x Real Time Clock
- 1 x Watchdog timer
- 4 x 16-bit General purpose timers
- 1 x one-wire debug interface
- 3-channel 8-bit 480kS/s ADC
- 2-channel 10-bit 1MS/s DAC
- 40 multi-purpose GPIOs

The block diagram shown in Figure 2.1 illustrates the main peripherals of the FT93x.

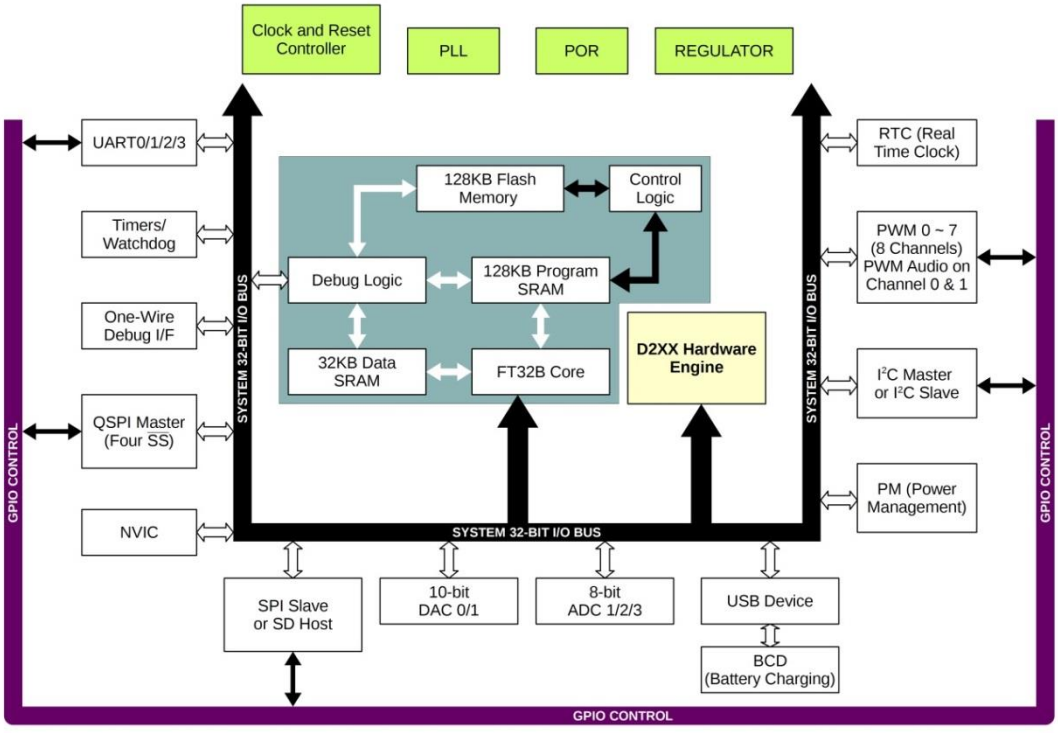



**Figure 2.1 - FT93x System Architecture**


Copyright © Bridgetek Pte Ltd

## 2.2 Memory Organization

The first 144 bytes in the Program Memory contain the following:

- Reset vector
- Watchdog vector
- 32 interrupt vectors
- Program entry point

| Address | Function |
|---|---|
| 0x00 | Reset vector |
| 0x04 | Watchdog vector |
| 0x08 | Interrupt vector 0 |
| 0x0C | Interrupt vector 1 |
| … | … |
| 0x80 | Interrupt vector 30 |
| 0x84 | Interrupt vector 31 |
| 0x88 | Interrupt vector 32 (NMI) |
| 0x8C | Program entry point |

**Table 2-1 - FT93x Program Memory Organization**

## 2.3 FT93x Boot Control

Upon reset, boot control logic takes control of the memory buses and puts the CPU in a reset state.

It automatically transfers the data from the flash memory to the CPU program memory, starting from address 0 on both sides. Boot control calculates a CRC check over the entire contents of flash (128 kB) and the result is placed in CRCH and CRCL registers found in the Flash Controller module.


Copyright © Bridgetek Pte Ltd

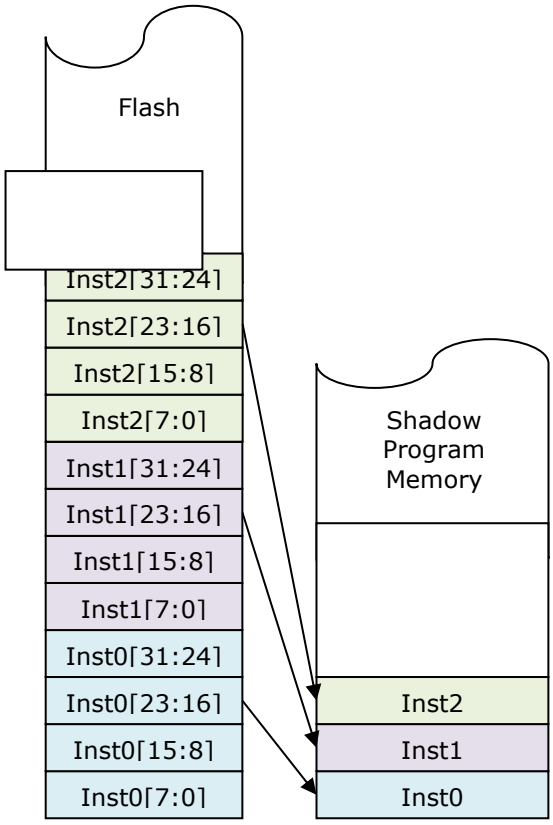



**Figure 2.2 - FT93x Boot Control**

## 2.4 Debugging Support

Debugging the FT93x series is carried out via the proprietary One-Wire interface. The debugging support is implemented in the FT93x bootloader. The protocol used for debugging is the GDB remote protocol and a port of GDB is available in the Bridgetek FT9xx Toolchain. The GDB serial debug protocol commands are interpreted by a debug interpreter in the bootloader.

In addition, the debug interpreter:

- Saves all machine states
- Executes commands received over the debug interface
- Restores all machine states and returns

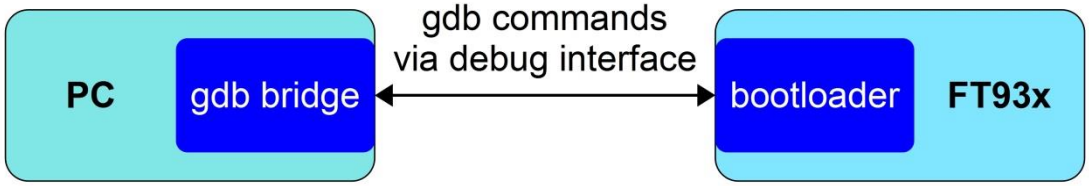



**Figure 2.3 - FT93x Debugging Support**


Copyright © Bridgetek Pte Ltd

# 3 Register Map

This section lists the I/O map for registers / memory in the device. Please note that some peripherals are not available on some models in the FT93x series. The details can be found in the table below. An (X) indicates that the peripheral exists and a minus (-) indicates that the peripheral is not available. All other peripherals that are not mentioned are available on all models in the series.

| | UART0 / 1 | UART 2 / 3 | SPI Master | SPI Slave | I2C Master / Slave | RTC | DAC 0 / 1 | ADC 1 | ADC 2 / 3 | USB Device | PWM 0 - 3 | PWM 4 - 7 | SD Host Controller |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **FT930** | x | x | x | x | x | x | x | x | x | x | x | x | x |
| **FT931** | x | - | x | x | x | x | x | x | x | x | x | x | x |
| **FT932** | x | - | x | x | x | - | x | x | x | x | x | x | x |
| **FT933** | x | - | x | x | x | - | - | - | x | x | x | - | - |

**Table 3-1 - Peripheral Availability on FT93x Series Models**

The register map of the peripherals is as follows:

| Function | Address Base Range | | Access Mechanism* |
|---|---|---|---|
| General System | 0x10000 | 0x100FF | DW, W, B |
| Interrupt Controller | 0x10100 | 0x1013F | DW, W, B |
| USB Device | 0x10200 | 0x1031F | DW, W, B |
| RTC | 0x10400 | 0x1043F | DW |
| SPI Master | 0x10440 | 0x1047F | DW |
| SPI Slave | 0x10480 | 0x104BF | DW |
| I2C Master | 0x10500 | 0x1050F | B |
| I2C Slave | 0x10510 | 0x1051F | B |
| UART 0 | 0x10520 | 0x1052F | B |
| UART 1 | 0x10530 | 0x1053F | B |
| UART 2 | 0x10540 | 0x1054F | B |
| UART 3 | 0x10550 | 0x1055F | B |
| Timers/Watchdog | 0x10560 | 0x1056F | B |
| PWM | 0x105B0 | 0x105EF | B for registers, W for FIFO |
| SD Host | 0x10600 | 0x107FF | DW |
| Flash Controller | 0x10800 | 0x108BF | B |

***** DW (Double-Word): 32-bit; W (Word): 16-bit; B (Byte): 8-bit

**Table 3-2 - Register Map for FT93x Series**


Copyright © Bridgetek Pte Ltd

# 4 Notations

## 4.1 Bit Type Notation

These notations are used in the register descriptions to describe the bit type:

| Terms | Description |
|---|---|
| Reserved | Reserved location, do not use / modify |
| RAC | Read-able with automatic clear |
| RO | Read-only |
| ROC | Read-only / Cleared-when-read |
| RW | Read- and Write-able |
| RW1C | Read and Write-1-to-clear |
| RW1S | Read and Write-1-to-set |
| RWAC | Read- and Write-able with automatic clear |
| W1S | Write-1-to-set |
| W1T | Write-1-to-trigger-event |
| WO | Write-only |

**Table 4.1 - Notations used in Register Description**

## 4.2 Default Bit Value Notation

When several bits form a field in a register, the following notation is used to represent the value of the field.

**<Field length>'<Value Format><Value>**

For example, a field with a default value of **8'h04** means that there are 8 bits in the field and the value of the field is 04 hex (i.e., 0x04).

The formats used for the default values are either hex ('h) or binary ('b).

## 4.3 X Default Value

"X" denotes "don't care" value which means a non-determinable (unknown) default value of a bit at a particular point of time.

## 4.4 Bit Range Symbol in Bit Column

Colon (:) will be used to represent the bit range in "Bit" column of Register tables. Bit range is usually given in descending order in the "Bit" column.

**N:M**

where N is a greater number than M. This indicates a bit range from bit N to bit M (in descending order). For example, 26:20 indicates a bit range from bit 26 to bit 20.

**Example 1:**

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31 | CAN_ACTIVE | RO | X | CAN modules available; the value depends on device model; 1 – available; 0 – not available. |
| 30 | MAC_ACTIVE | RO | X | Ethernet module available; default value depends on device model; 1 – available; 0 – not available. |


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 29 | 100_PIN | RO | X | 1 - the device is a 100-pin device; 0 - the device is a 76-pin (QFN) or 80-pin (LQFP) device. |
| 28 | Reserved | RO | 0 | Always read as '0' |
| 27 | ONE-WIRE_ACTIVE | RO | X | If set, one-wire debug interface is enabled; otherwise, it's permanently disabled. |
| 26:20 | Reserved | RO | X | Reserved |
| 19 | FLASH_WR_B3_ENA | RO | X | 0 - Flash is permanently non-writable/ non-erasable.<br>1 - Allow the Flash to write/erase from 0x30000 to 0x3ffff bytes. |

**Example 2:**

A register is assigned to bits[39:8] of the command field.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:0 | ARG1 | RW | 32'h00000000 | Command argument |


Copyright © Bridgetek Pte Ltd

# 5 General System Registers

This section describes the registers that govern the general behavior of the FT93x.

## 5.1 Register Summary

Listed below are the registers with their offset from the base address (0x10000). All registers can be accessed via Byte (8-bit), Word (16-bit) or Double-Word (32-bit) mode.

| Address Offset | Register | Default Value | Reference |
|---|---|---|---|
| | | | |
| **0x00** | CHIPID - Chip ID Register | 0xXXXXXXXX | Section 5.2.1 |
| **0x04** | CHIPCFG - Chip Configuration Register | 0xXXXXXXXX | Section 5.2.2 |
| **0x08** | CLKCFG - Clock Configuration Register | 0x00000000 | Section 5.2.3 |
| **0x0C** | PMCFG - Power Management Register | 0x00000000 | Section 5.2.4 |
| **0x10** | Reserved | - | Section 5.2.5 |
| **0x14** | Reserved | - | Section 5.2.6 |
| **0x18** | MSC0CFG - Miscellaneous Configuration Register | 0x00000000 | Section 5.2.7 |
| **0x1C** | Pin 00 – 03 Register | 0x04080808 | Section 5.2.8.1 |
| **0x20** | Pin 04 – 07 Register | 0x04040404 | Section 5.2.8.2 |
| **0x24** | Pin 08 – 11 Register | 0x04040404 | Section 5.2.8.3 |
| **0x28** | Pin 12 – 15 Register | 0x04040404 | Section 5.2.8.4 |
| **0x2C** | Pin 16 – 19 Register | 0x04040404 | Section 5.2.8.5 |
| **0x30** | Pin 20 – 23 Register | 0x04040404 | Section 5.2.8.6 |
| **0x34** | Pin 24 – 27 Register | 0x04040404 | Section 5.2.8.7 |
| **0x38** | Pin 28 – 31 Register | 0x04040404 | Section 5.2.8.8 |
| **0x3C** | Pin 32 – 35 Register | 0x04040404 | Section 5.2.8.9 |
| **0x40** | Pin 36 – 39 Register | 0x04040404 | Section 5.2.8.10 |
| **0x60** | GPIO 00 – 07 Configuration Register | 0x00000000 | Section 5.2.9.1 |
| **0x64** | GPIO 08 – 15 Configuration Register | 0x00000000 | Section 5.2.9.2 |
| **0x68** | GPIO 16 – 23 Configuration Register | 0x00000000 | Section 5.2.9.3 |
| **0x6C** | GPIO 24 – 31 Configuration Register | 0x00000000 | Section 5.2.9.4 |
| **0x70** | GPIO 32 – 39 Configuration Register | 0x00000000 | Section 5.2.9.5 |
| **0x84** | GPIO 00 – 31 Value Register | 0x00000000 | Section 5.2.10.1 |
| **0x88** | GPIO 32 – 39 Value Register | 0x00000000 | Section 5.2.10.2 |


Copyright © Bridgetek Pte Ltd

| Address Offset | Register | Default Value | Reference |
|---|---|---|---|
| **0x90** | GPIO 00 – 31 Interrupt Enable Register | 0x00000000 | Section 5.2.11.1 |
| **0x94** | GPIO 32 – 39 Interrupt Enable Register | 0x00000000 | Section 5.2.11.2 |
| **0x9C** | GPIO 00 – 31 Interrupt Pending Register | 0x00000000 | Section 5.2.12.1 |
| **0xA0** | GPIO 32 – 39 Interrupt Pending Register | 0x00000000 | Section 5.2.12.2 |
| **0xB0** | DAC_ADC_CONF - ADC/DAC Configuration/Status Register | 0x00000000 | Section 5.2.13 |
| **0xB4** | DAC_ADC_CNT - ADC/DAC Count Register | 0x63XXXXXX | Section 5.2.14 |
| **0xB8** | DAC_ADC_DATA - ADC/DAC Data Register | 0x00000000 | Section 5.2.15 |
| **0xBC** | 5.2.28 GLOBAL_RESET_STATUS – Global Reset and Status Register | 0x00000000 | Section 5.2.16 |

**Table 5-1 - Overview of General System Registers**


Copyright © Bridgetek Pte Ltd

## 5.2 Register Details

### 5.2.1 CHIPID - Chip ID Register (address offset: 0x00)

This register is read-only.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:0 | Chip ID | RO | 0x093XXXXX | The two MSBs (093X) depict FT93x series and the two LSBs (XXXX) shows the revision of the chip. |

**Table 5-2 - CHIPID - Chip ID Register**

For revision 0001 of the FT93x series, the pre-configured bits of the chip ID register (CHIPID) are listed in the table below. Note that these bits are always read-only.

| | CHIPID [31..16] | CHIPID [15..0] |
|---|---|---|
| **FT930Q** | 0X0930 | 0X0001 |
| **FT931Q** | 0X0931 | 0X0001 |
| **FT932Q** | 0X0932 | 0X0001 |
| **FT933Q** | 0X0933 | 0X0001 |

**Table 5-3 - FT93x Series Chip ID Configuration**

### 5.2.2 CHIPCFG - Chip Configuration Register (address offset: 0x04)

This register contains read-only information.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31-28 | Reserved | - | - | Reserved |
| 27 | ONE-WIRE_ACTIVE | RO | X | If set, the one-wire debug interface is enabled; otherwise, it's permanently disabled. |
| 26:20 | Reserved | - | X | Reserved |
| 19 | Reserved | - | X | Reserved |
| 18 | Reserved | - | X | Reserved |
| 17 | FLASH_WR_B1_ENA | RO | X | 0 - Flash is permanently non-writable/non-erasable.<br><br>1 - Allow the Flash to write/erase from 0x10000 to 0x1ffff bytes. |
| 16 | FLASH_WR_B0_ENA | RO | X | 0 - Flash is permanently non-writable/non-erasable.<br><br>1 - Allow the Flash to write/erase from 0x00000 to 0x0ffff bytes. |
| 15:0 | Reserved | - | | Reserved |

**Table 5-4 - CHIPCFG - Chip Configuration Register**


Copyright © Bridgetek Pte Ltd

### 5.2.3 CLKCFG - Clock Configuration Register (address offset: 0x08)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:20 | Reserved | - | - | Reserved |
| 19:16 | CPU_CLK_DIV | RW | 0 | CPU Clock frequency setting:<br>0 : CPU at full system clock speed<br>1 : CPU at system clock / 2<br>2 : CPU at system clock / 4<br>3 : CPU at system clock / 8<br>4 : CPU at system clock / 64<br>5 : CPU at system clock / 128<br>6 : CPU at system clock / 512 |
| 15 | Reserved | - | - | - |
| 14 | DEV_ENA<br>(only valid for non-D2XX mode) | RW | 0 | Set to "1" to enable USB Device when it is not in D2XX mode (reg_d2xx_mode=0). |
| 13 | Reserved | | - | - |
| 12 | SD _ENA | RW | 0 | 1 : enable SD Host Controller |
| 11:10 | Reserved | | - | - |
| 9 | I2CM _ENA | RW | 0 | 1 : enable I2C Master |
| 8 | I2CS _ENA | RW | 0 | 1 : enable I2C Slave |
| 7 | SPIM _ENA | RW | 0 | 1 : enable SPI Master |
| 6 | SPIS _ENA | RW | 0 | 1 : enable SPI Slave |
| 5 | Reserved | - | - | - |
| 4 | UART0 _ENA | RW | 0 | 1 : enable UART 0 |
| 3 | UART1 _ENA | RW | 0 | 1 : enable UART 1 |
| 2 | PWM _ENA | RW | 0 | 1 : enable PWM |
| 1 | UART3 _ENA | RW | 0 | 1 : enable UART 3 |
| 0 | UART2 _ENA | RW | 0 | 1 : enable UART 2 |

**Table 5-5 - CLKCFG - Clock Configuration Register**

### 5.2.4 PMCFG - Power Management Register (address offset: 0x0C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31 | RTC_ALARM_IRQ_PEND | RW1C | 0 | RTC Alarm Wake Up interrupts pending; write 1 to clear. This bit will be set when the RTC alarm is active while it is in sleep mode. |
| 30:27 | Reserved | - | - | |
| 26 | SLAVE_PERI_IRQ_PEND<br>(only valid for D2XX mode) | | | SLAVE Wake Up interrupt pending; write 1 to clear. This bit will be set when there is a wake-up event from the D2XX Engine |
| 25 | PM_GPIO_IRQ_PEND | RW1C | 0 | GPIO interrupt during system shut down with clock not running; write 1 to clear. |
| 24 | SLOWCLK_5ms_IRQ_PEND<br>(only valid for non-D2XX mode) | RW1C | 0 | Slow clock 5ms timer interrupts pending; write 1 to clear. If enabled, an interrupt will be generated. |
| 23:21 | Reserved | - | - | - |
| 20 | DEV_CONN_DEV<br>(only valid for non-D2XX mode) | RW1C | 0 | Device Connect Interrupt pending to USB Device; write 1 to clear. If enabled, an interrupt will be generated on PM IRQ. |


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 19 | DEV_DIS_DEV (only valid for non-D2XX mode) | RW1C | 0 | Device Disconnect Interrupt pending to USB Device; write 1 to clear. If enabled, an interrupt will be generated on PM IRQ. |
| 18 | HOST_RST_DEV (only valid for non-D2XX mode) | RW1C | 0 | Host Reset Interrupt pending to USB Device; write 1 to clear. If enabled, an interrupt will be generated on PM IRQ. |
| 17 | HOST_RESUME_DEV (only valid for non-D2XX mode) | RW1C | 0 | Host Resume Interrupt pending to USB Device; write 1 to clear. If enabled, an interrupt will be generated on PM IRQ. |
| 16 | Reserved | - | - | - |
| 15 | DIS_DISCONN_DET (only valid for non-D2XX mode) | RW | 0 | 1: To disable Device Disconnection detection.<br>0: To enable Device Disconnection detection. |
| 14:13 | Reserved | - | - | - |
| 12 | RTC_ALARM_IRQ_EN | RW | 0 | RTC Alarm Wake Up Interrupt Enable; if this bit is set, an interrupt will be generated on PM IRQ when RTC_ALARM_IRQ_PEND becomes active. |
| 11 | SLAVE_PERI_IRQ_EN (only valid for D2XX mode) | RW | 0 | SLAVE Wake Up Interrupt Enable; If this bit is set, an interrupt will be generated on PM IRQ when SLAVE_PERI_IRQ_PEND becomes active. |
| 10 | DEV_PHY_EN (only valid for non-D2XX mode) | RW | 0 | 1 : Enable USB Device PHY.<br>0 : USB Device PHY is in Suspend. |
| 9 | PM_PWRDN_MODE | RW | 0 | 1 : disable system oscillator when powering down<br>0 : do not disable system oscillator when powering down |
| 8 | PM_PWRDN | RW | 0 | 1 : power down system.<br>This bit should be cleared after the system wakes up or at least 60-100 µs prior to setting it to 1 again. |
| 7 | SLOWCLOCK_5ms_IRQ_EN (only valid for non-D2XX mode) | RW | 0 | 1 : enable slow clock 5 ms timer interrupt. |
| 6 | SLOWCLOCK_5ms_START (only valid for non-D2XX mode) | RW1 | 0 | 1 : To start the 1-shot slow clock 5ms timer; once started it cannot be stopped. This bit will be cleared automatically when the timer expires. |
| 5 | Reserved | - | - | - |
| 4 | FORCE_DEV_DET (only valid for non-D2XX mode) | RW | 0 | Normally device activity detection is performed only when required, setting this bit will force the PM to check for device connection activities regardless. |
| 3:2 | Reserved | - | - | - |
| 1 | DEV_DETECT_EN (only valid for non-D2XX mode) | RW | 0 | 1 : Enable device connect/disconnect to external host or external host reset detection. Enable interrupt to IRQ when any of DEV_CONN_DEV, DEV_DIS_DEV and HOST_RST_DEV is set. |
| 0 | Reserved | - | - | - |

**Table 5-6 - PMCFG - Power Management Configuration Register**


Copyright © Bridgetek Pte Ltd

### 5.2.5 RESERVED (address offset: 0x10)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:0 | Reserved | - | - | - |

**Table 5-7 - Reserved**

### 5.2.6 RESERVED (address offset: 0x14)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:0 | Reserved | - | - | - |

**Table 5-8 - Reserved**

### 5.2.7 MSC0CFG - Miscellaneous Configuration Register (address offset: 0x18)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31 | PERI_SOFTRESET | RWAC | 0 | Write 1 to cause a soft reset to all peripherals. It is automatically cleared. |
| 30 | PWM_SOFTRESET | RWAC | 0 | Write 1 to cause a soft reset to PWM. It is automatically cleared. |
| 29 | UART3_CLKSEL | RW | 0 | Clock Select for UART3. |
| 28 | UART3_FIFOSEL | RW | 0 | FIFO Selection for UART3. |
| 27 | UART3_INTSEL | RW | 0 | INT Selection for UART3. |
| 26 | UART2_CLKSEL | RW | 0 | Clock Select for UART2. |
| 25 | UART2_FIFOSEL | RW | 0 | FIFO Selection for UART2. |
| 24 | UART2_INTSEL | RW | 0 | INT Selection for UART2. |
| 23 | HIGH_SPEED_MODE (only valid for non-D2XX mode) | RO | 0 | 1: it indicates USB is in high-speed mode. 0: it indicates USB is in full speed mode. |
| 22 | USB_VBUS_EN (only valid for non-D2XX mode) | RW | 0 | Write 0 followed by 1 to enable the VBUS detection circuit. DEV_CONN_DEV is set when VBUS is detected. This bit is reset when USB Device is reset. |
| 21 | UART0_CLKSEL | RW | 0 | Clock Select for UART0. |
| 20 | UART0_FIFOSEL | RW | 0 | FIFO Selection for UART0. |
| 19 | UART0_INTSEL | RW | 0 | INT Selection for UART0. |
| 18 | UART1_CLKSEL | RW | 0 | Clock Select for UART1. |
| 17 | UART1_FIFOSEL | RW | 0 | FIFO Selection for UART1. |
| 16 | UART1_INTSEL | RW | 0 | INT Selection for UART1. |
| 15:13 | Reserved | - | - | - |
| 12 | DEV_RMWAKEUP (only valid for non-D2XX mode) | RW | 0 | 1 : Drive K-state on USB Device port; software must maintain the 1ms requirement before turning it off. |
| 11 | DEV_RESET_ALL (only valid for non-D2XX mode) | RWAC | 0 | Write 1 to reset USB Device Controller and USB Device PHY; it is automatically cleared immediately. |
| 10 | DEV_RESET_CONTROLLER (only valid for non-D2XX mode) | RWAC | 0 | Write 1 to cause USB Dev Controller reset; it is automatically cleared immediately. |


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 9 | DEV_RESET_ATX (only valid for non-D2XX mode) | RWAC | 0 | Write 1 to cause USB Dev PHY reset; it is automatically cleared immediately. |
| 8:5 | Reserved | - | - | - |
| 4 | BCD_SOFTRESET (only valid for non-D2XX mode) | WO | 0 | 1: Generate software reset to<br>- BCD Host if BCDHOST_EN is 1.<br>- BCD Dev if BCDDEV_EN is 1.<br>It is automatically cleared immediately. |
| 4 | BCDDEV_DETECT_RUNNING | RO | 0 | 1 : indicates BCD Device detection is running |
| 3 | BCDDEV_EN (only valid for non-D2XX mode) | WO | 0 | 1 : enable Battery Charger Detection Device. This bit is cleared by writing a 1 to BCD_SOFTRESET. |
| 3 | BCDDEV_DETECT_COMPLETE | RO | 0 | 1 : indicates BCD Device detection is done. |
| 2 | BCDDEV_SD_EN (only valid for non-D2XX mode) | WO | 1 | 1 : enable secondary detection for details. |
| 2 | BCDDEV_SDP_FOUND | RO | 0 | 1 : SDP detected |
| 1 | BCDDEV_VDP_EN_POST_DCP (only valid for non-D2XX mode) | WO | 0 | 1 : enable connection of $V_{DP_SRC}$ after DCP detection |
| 1 | BCDDEV_CDP_FOUND | RO | 0 | 1 : CDP detected |
| 0 | BCDDEV_LGC_COMP_INHIB (only valid for non-D2XX mode) | WO | 1 | 1 : disable logic comparison during BCD detections |
| 0 | BCDDEV_DCP_FOUND | RO | 0 | 1 : DCP detected |

**Table 5-9 - MSC0CFG - Miscellaneous Configuration Register**

### 5.2.8 GPIO Pin Configuration Registers (address offset: 0x1C – 0x5F)

These registers control the pin configurations for the physical pins on the device. Each register houses the configuration for 4 digital pins. Each byte of the register configures 1 digital pin. The pin direction for each of the special functions is fixed and is set automatically. A pin that is configured as a GPIO can be further configured. Refer to the GPIO Configuration Registers in section 5.2.9.

The bit layout for the Pin Configuration Registers is as follows:

| Bit | Description | Value | Configuration |
|---|---|---|---|
| 31:30<br>23:22<br>15:14<br>7:6 | Pin Functionality | 00<br>01<br>10<br>11 | GPIO Function<br>Special Function 1<br>Special Function 2 (if available)<br>Special Function 3 (if available) |
| 29:28<br>21:20<br>13:12<br>5:4 | Output drive capability | 00<br>01<br>10<br>11 | 4mA<br>8mA<br>12mA<br>16mA |
| 27:26<br>19:18<br>11:10<br>3:2 | With Pull-up / Pull-down | 00<br>01<br>10<br>11 | None<br>75kΩ Pull-down<br>75kΩ Pull-up<br>75kΩ Keeper |


Copyright © Bridgetek Pte Ltd

| Bit | Description | Value | Configuration |
|---|---|---|---|
| 25<br>17<br>9<br>1 | Schmitt | 0<br>1 | Normal<br>Schmitt |
| 24<br>16<br>8<br>0 | Slew Rate | 0<br>1 | Fast<br>Slow |

**Table 5-10 - Pin Configuration Register Description**

The following tables give more details about each Pin Configuration Register. The "Pin Functionality Bits" section refers to bits 31:30, 23:22, 15:14, 7:6 in each register for configuring the corresponding pin to perform a specific function. Refer to Table 5.10 above.

### *5.2.8.1 Pin 00 – 03 register (address offset: 0x1C)*

| Bit | Name | Type | Default Value | Pin Functionality Bits | | | |
|---|---|---|---|---|---|---|---|
| | | | | 00 | 01 | 10 | 11 |
| 31:24 | PIN03_CFG | RW | 8'h04 | GPIO 3 | SS(S) | DATA0(SD) | - |
| 23:16 | PIN02_CFG | RW | 8'h04 | GPIO 2 | MOSI(S) | CD(SD) | - |
| 15:8 | PIN01_CFG | RW | 8'h04 | GPIO 1 | MISO(S) | CMD(SD) | - |
| 7:0 | PIN00_CFG | RW | 8'h04 | GPIO 0 | SCK(S) | CLK(SD) | - |

**Table 5-11 - Pin 00 – 03 Register**

### *5.2.8.2 Pin 04 – 07 Register (address offset: 0x20)*

| Bit | Name | Type | Default Value | Pin Functionality Bits | | | |
|---|---|---|---|---|---|---|---|
| | | | | 00 | 01 | 10 | 11 |
| 31:24 | PIN07_CFG | RW | 8'h04 | GPIO 7 | PWM4 | WP(SD) | |
| 23:16 | PIN06_CFG | RW | 8'h04 | GPIO 6 | PWM5 | DATA3(SD) | |
| 15:8 | PIN05_CFG | RW | 8'h04 | GPIO 5 | PWM6 | DATA2(SD) | PWM1 |
| 7:0 | PIN04_CFG | RW | 8'h04 | GPIO 4 | PWM7 | DATA1(SD) | PWM0 |

**Table 5-12 - Pin 04 – 07 Register**

### *5.2.8.3 Pin 08 – 11 Register (address offset: 0x24)*

| Bit | Name | Type | Default Value | Pin Functionality Bits | | | |
|---|---|---|---|---|---|---|---|
| | | | | 00 | 01 | 10 | 11 |
| 31:24 | PIN11_CFG | RW | 8'h04 | GPIO 11 | PWM0 | - | |
| 23:16 | PIN10_CFG | RW | 8'h04 | GPIO 10 | PWM1 | - | |
| 15:8 | PIN09_CFG | RW | 8'h04 | GPIO 9 | PWM2 | - | |
| 7:0 | PIN08_CFG | RW | 8'h04 | GPIO 8 | PWM3 | - | |

**Table 5-13 - Pin 08 – 11 Register**


Copyright © Bridgetek Pte Ltd

#### *5.2.8.4 Pin 12 – 15 Register (address offset: 0x28)*

| Bit | Name | Type | Default Value | Pin Functionality Bits | | | |
|---|---|---|---|---|---|---|---|
| | | | | 00 | 01 | 10 | 11 |
| 31:24 | PIN15_CFG | RW | 8'h04 | GPIO 15 | TXD(U3) | - | - |
| 23:16 | PIN14_CFG | RW | 8'h04 | GPIO 14 | RXD(U3) | - | - |
| 15:8 | PIN13_CFG | RW | 8'h04 | GPIO 13 | SDA(M) | SDA(S) | - |
| 7:0 | PIN12_CFG | RW | 8'h04 | GPIO 12 | SCL(M) | SCL(S) | - |

**Table 5-14 - Pin 12 – 15 Register**

#### *5.2.8.5 Pin 16 – 19 Register (address offset: 0x2C)*

| Bit | Name | Type | Default Value | Pin Functionality Bits | | | |
|---|---|---|---|---|---|---|---|
| | | | | 00 | 01 | 10 | 11 |
| 31:24 | PIN19_CFG | RW | 8'h04 | GPIO 19 | DSR(U3) | TXD(U4) | - |
| 23:16 | PIN18_CFG | RW | 8'h04 | GPIO 18 | DTR(U3) | RXD(U4) | - |
| 15:8 | PIN17_CFG | RW | 8'h04 | GPIO 17 | CTS(U3) | | - |
| 7:0 | PIN16_CFG | RW | 8'h04 | GPIO 16 | RTS(U3) | | - |

**Table 5-15 - Pin 16 – 19 Register**

#### *5.2.8.6 Pin 20 – 23 Register (address offset: 0x30)*

| Bit | Name | Type | Default Value | Pin Functionality Bits | | | |
|---|---|---|---|---|---|---|---|
| | | | | 00 | 01 | 10 | 11 |
| 31:24 | PIN23_CFG | RW | 8'h04 | GPIO 23 | TXD(U1) | PWM2 | - |
| 23:16 | PIN22_CFG | RW | 8'h04 | GPIO 22 | RXD(U1) | PWM3 | - |
| 15:8 | PIN21_CFG | RW | 8'h04 | GPIO 21 | RI(U3) | CTS(U4) | - |
| 7:0 | PIN20_CFG | RW | 8'h04 | GPIO 20 | DCD(U3) | RTS(U4) | - |

**Table 5-16 - Pin 20 – 23 Register**

#### *5.2.8.7 Pin 24 – 27 Register (address offset: 0x34)*

| Bit | Name | Type | Default Value | Pin Functionality Bits | | | |
|---|---|---|---|---|---|---|---|
| | | | | **00** | **01** | **10** | **11** |
| 31:24 | PIN27_CFG | RW | 8'h04 | GPIO 27 | DSR(U1) | TXD(U2) | - |
| 23:16 | PIN26_CFG | RW | 8'h04 | GPIO 26 | DTR(U1) | RXD(U2) | - |
| 15:8 | PIN25_CFG | RW | 8'h04 | GPIO 25 | CTS(U1) | PWM0 | - |
| 7:0 | PIN24_CFG | RW | 8'h04 | GPIO 24 | RTS(U1) | PWM1 | - |

**Table 5-17 - Pin 24 – 27 Register**

#### *5.2.8.8 Pin 28 – 31 Register (address offset: 0x38)*

| Bit | Name | Type | Default Value | Pin Functionality Bits | | | |
|---|---|---|---|---|---|---|---|
| | | | | 00 | 01 | 10 | 11 |
| 31:24 | PIN31_CFG | RW | 8'h04 | GPIO 31 | SS1(SPIM) | - | - |
| 23:16 | PIN30_CFG | RW | 8'h04 | GPIO 30 | SS0(SPIM) | - | - |
| 15:8 | PIN29_CFG | RW | 8'h04 | GPIO 29 | RI(U1) | CTS(U2) | SS0(SPIM) |
| 7:0 | PIN28_CFG | RW | 8'h04 | GPIO 28 | DCD(U1) | RTS(U2) | - |

**Table 5-18 - Pin 28 – 31 Register**

#### *5.2.8.9 Pin 32 – 35 Register (address offset: 0x3C)*

| Bit | Name | Type | Default Value | Pin Functionality Bits | | | |
|---|---|---|---|---|---|---|---|
| | | | | 00 | 01 | 10 | 11 |
| 31:24 | PIN35_CFG | RW | 8'h04 | GPIO 35 | MISO(SPIM) | MISO(SPIS) | - |
| 23:16 | PIN34_CFG | RW | 8'h04 | GPIO 34 | SCK(SPIM) | SCK(SPIS) | - |
| 15:8 | PIN33_CFG | RW | 8'h04 | GPIO 33 | SS3(SPIM) | - | - |
| 7:0 | PIN32_CFG | RW | 8'h04 | GPIO 32 | SS2(SPIM) | - | - |

**Table 5-19 - Pin 32 – 35 Register**


Copyright © Bridgetek Pte Ltd

#### *5.2.8.10 Pin 36 – 39 Register (address offset: 0x40)*

| Bit | Name | Type | Default Value | Pin Functionality Bits | | | |
|---|---|---|---|---|---|---|---|
| | | | | 00 | 01 | 10 | 11 |
| 31:24 | PIN39_CFG | RW | 8'h04 | GPIO 39 | VBUS Detect | - | - |
| 23:16 | PIN38_CFG | RW | 8'h04 | GPIO 38 | IO3O(SPIM) | RTC_REF | - |
| 15:8 | PIN37_CFG | RW | 8'h04 | GPIO 37 | IO2O(SPIM) | SS(SPIS) | - |
| 7:0 | PIN36_CFG | RW | 8'h04 | GPIO 36 | MOSI(SPIM) | MOSI(SPIS) | - |

**Table 5-20 - Pin 36 – 39 Register**

### 5.2.9 GPIO Configuration Registers (address offset: 0x60 – 0x70)

These registers control the GPIO configurations. Each register houses the configuration for 8 digital pins. Each nibble of the register configures 1 digital pin.

All GPIOs can function as an interrupt. The polarity can be either positive edge or negative edge if its interrupt capability is enabled. If this feature is desired, the pin must be configured as a GPIO input. Otherwise unpredictable behavior may result.

There is no debouncing for any GPIOs. If they are used as general inputs, and debouncing is needed, then the software must handle the debouncing routine. If it is to be used as an interrupt, the external interrupt source should be glitch free.

The bit layout for the GPIO Configuration Registers is as follows:

| Bit | Description | Value | Configuration |
|---|---|---|---|
| 31:30<br>27:26<br>23:22<br>19:18<br>15:14<br>11:10<br>7:6<br>3:2 | GPIO Direction | 00<br>01<br>1X | Input<br>Output<br>Open-Drain Output |
| 29<br>25<br>21<br>17<br>13<br>9<br>5<br>1 | Interrupt Capable<br>(Also see section 5.2.11, "GPIO Interrupt Enable Registers") | 0<br>1 | Disabled<br>Enabled |
| 28<br>24<br>20<br>16<br>12<br>8<br>4<br>0 | Interrupt Edge<br>(only if Interrupt Capable bit is 1) | 0<br>1 | Falling Edge<br>Rising Edge |

**Table 5-21 - GPIO Configuration Register Description**

The following tables give more details about which bits control a specific GPIO Pin using the values from Table 5-21.

#### *5.2.9.1 GPIO 00 – 07 Configuration Register (address offset: 0x60)*

| Bit | Name | Type | Default | Description |
|---|---|---|---|---|


Copyright © Bridgetek Pte Ltd

| | | | Value | |
|---|---|---|---|---|
| 31:28 | GPIO07_CFG | RW | 0 | For GPIO 7 |
| 27:24 | GPIO06_CFG | RW | 0 | For GPIO 6 |
| 23:20 | GPIO05_CFG | RW | 0 | For GPIO 5 |
| 19:16 | GPIO04_CFG | RW | 0 | For GPIO 4 |
| 15:12 | GPIO03_CFG | RW | 0 | For GPIO 3 |
| 11:8 | GPIO02_CFG | RW | 0 | For GPIO 2 |
| 7:4 | GPIO01_CFG | RW | 0 | For GPIO 1 |
| 3:0 | GPIO00_CFG | RW | 0 | For GPIO 0 |

**Table 5-22 - GPIO 00 – 07 Configuration Register**

#### *5.2.9.2 GPIO 08 – 15 Configuration Register (address offset: 0x64)*

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:28 | GPIO15_CFG | RW | 0 | For GPIO 15 |
| 27:24 | GPIO14_CFG | RW | 0 | For GPIO 14 |
| 23:20 | GPIO13_CFG | RW | 0 | For GPIO 13 |
| 19:16 | GPIO12_CFG | RW | 0 | For GPIO 12 |
| 15:12 | GPIO11_CFG | RW | 0 | For GPIO 11 |
| 11:8 | GPIO10_CFG | RW | 0 | For GPIO 10 |
| 7:4 | GPIO09_CFG | RW | 0 | For GPIO 9 |
| 3:0 | GPIO08_CFG | RW | 0 | For GPIO 8 |

**Table 5-23 - GPIO 08 – 15 Configuration Register**

#### *5.2.9.3 GPIO 16 – 23 Configuration Register (address offset: 0x68)*

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:28 | GPIO23_CFG | RW | 0 | For GPIO 23 |
| 27:24 | GPIO22_CFG | RW | 0 | For GPIO 22 |
| 23:20 | GPIO21_CFG | RW | 0 | For GPIO 21 |
| 19:16 | GPIO20_CFG | RW | 0 | For GPIO 20 |
| 15:12 | GPIO19_CFG | RW | 0 | For GPIO 19 |
| 11:8 | GPIO18_CFG | RW | 0 | For GPIO 18 |
| 7:4 | GPIO17_CFG | RW | 0 | For GPIO 17 |
| 3:0 | GPIO16_CFG | RW | 0 | For GPIO 16 |

**Table 5-24 - GPIO 16 – 23 Configuration Register**

#### *5.2.9.4 GPIO 24 – 31 Configuration Register (address offset: 0x6C)*

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:28 | GPIO31_CFG | RW | 0 | For GPIO 31 |
| 27:24 | GPIO30_CFG | RW | 0 | For GPIO 30 |
| 23:20 | GPIO29_CFG | RW | 0 | For GPIO 29 |
| 19:16 | GPIO28_CFG | RW | 0 | For GPIO 28 |
| 15:12 | GPIO27_CFG | RW | 0 | For GPIO 27 |
| 11:8 | GPIO26_CFG | RW | 0 | For GPIO 26 |
| 7:4 | GPIO25_CFG | RW | 0 | For GPIO 25 |
| 3:0 | GPIO24_CFG | RW | 0 | For GPIO 24 |

**Table 5-25 - GPIO 24 – 31 Configuration Register**


Copyright © Bridgetek Pte Ltd

#### *5.2.9.5 GPIO 32 – 39 Configuration Register (address offset: 0x70)*

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:28 | GPIO39_CFG | RW | 0 | For GPIO 39 |
| 27:24 | GPIO38_CFG | RW | 0 | For GPIO 38 |
| 23:20 | GPIO37_CFG | RW | 0 | For GPIO 37 |
| 19:16 | GPIO36_CFG | RW | 0 | For GPIO 36 |
| 15:12 | GPIO35_CFG | RW | 0 | For GPIO 35 |
| 11:8 | GPIO34_CFG | RW | 0 | For GPIO 34 |
| 7:4 | GPIO33_CFG | RW | 0 | For GPIO 33 |
| 3:0 | GPIO32_CFG | RW | 0 | For GPIO 32 |

**Table 5-26 - GPIO 32 – 39 Configuration Register**

### 5.2.10 GPIO Value Registers (address offset: 0x84 – 0x88)

These registers contain the values for the GPIO pins. Each register contains the value of 32 digital pins except the last register, which only contains the value of 8 pins (32 to 39). Each bit of the register maps to the corresponding digital pin.

#### *5.2.10.1 GPIO 00 – 31 Value Register (address offset: 0x84)*

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:0 | GPIO_VAL_IN[31:0] | RO | X | Input values of GPIO 31 – GPIO 0 |
| 31:0 | GPIO_VAL_OUT[31:0] | WO | 0 | Output values of GPIO 31 – GPIO 0 |

**Table 5-27 - GPIO 00 – 31 Value Register**

#### *5.2.10.2 GPIO 32 – 39 Value Register (address offset: 0x88)*

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:8 | Reserved | - | - | - |
| 7:0 | GPIO_VAL_IN[39:32] | RO | X | Input values of GPIO 39 – GPIO 32 |
| 7:0 | GPIO_VAL_OUT[39:32] | WO | 0 | Output values of GPIO 39 – GPIO 32 |

**Table 5-28 - GPIO 32 – 39 Value Register**

### 5.2.11 GPIO Interrupt Enable Registers (address offset: 0x90 – 0x94)

When a pin has been configured as an input with interrupt capability, the GPIO Interrupt Enable Register can be used to enable interrupt generation. Each register enables interrupt generation for 32 digital pins except the last register, which only enables interrupt generation for 8 pins (32 to 39). Each bit of the register enables interrupt generation for 1 digital pin. These should be used only for pins that have been properly configured as GPIO interrupt inputs.

#### *5.2.11.1 GPIO 00 – 31 Interrupt Enable Register (address offset: 0x90)*

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:0 | GPIO_INT_EN[31:0] | RW | 0 | GPIO input 31-0 interrupt enable when set. |

**Table 5-29 - GPIO 00 – 31 Interrupt Enable Register**

#### *5.2.11.2 GPIO 32 – 39 Interrupt Enable Register (address offset: 0x94)*

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:8 | Reserved | - | - | - |
| 7:0 | GPIO_INT_EN[39:32] | RW | 0 | GPIO input 39-32 interrupt enable when set. |

**Table 5-30 - GPIO 32 – 39 Interrupt Enable Register**


Copyright © Bridgetek Pte Ltd

## 5.2.12 Interrupt Pending Registers (address offset: 0x9C – 0xA0)

These registers hold the interrupt pending flags for the GPIO pins. Each register holds the flags for 32 digital pins except the last register, which only holds the flags for 8 pins (32 to 39). Each bit of the register holds the flag for 1 digital pin.

### *5.2.12.1 GPIO 00 – 31 Interrupt Pending Register (address offset: 0x9C)*

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:0 | GPIO_INT_PEND[31:0] | RW1C | 0 | GPIO input 31-0 interrupt pending when set; write a 1 to clear. |

**Table 5-31 - GPIO 00 – 31 Interrupt Pending Register**

### *5.2.12.2 GPIO 32 – 39 Interrupt Pending Register (address offset: 0xA0)*

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:8 | Reserved | - | - | - |
| 7:0 | GPIO_INT_PEND[39:32] | RW1C | 0 | GPIO input 39-32 interrupt pending when set; write a 1 to clear. |

**Table 5-32 - GPIO 32 – 39 Interrupt Pending Register**

## 5.2.13 DAC_ADC_CONF - ADC/DAC Configuration/Status Register (address offset: 0xB0)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31 | Reserved | - | - | - |
| 30 | ADC_BUF_ERR | RO | 0 | Buffer underflow/overflow error; this is sticky. Cleared by ADC reset or ADC power down. |
| 29 | DAC1_BUF_ERR | RO | 0 | Buffer underflow/overflow error; this is sticky. Cleared by DAC 1 reset or DAC 1 power down. |
| 28 | DAC0_BUF_ERR | RO | 0 | Buffer underflow/overflow error; this is sticky. Cleared by DAC 0 reset or DAC 0 power down. |
| 27 | Reserved | - | - | - |
| 26 | ADC_IRQ_PEND | RW1C | 0 | ADC Interrupt Pending when set.<br>Write a "1" to clear this bit. |
| 25 | DAC1_IRQ_PEND | RW1C | 0 | DAC 1 Interrupt Pending when set.<br>Write a 1 to clear this bit. |
| 24 | DAC0_IRQ_PEND | RW1C | 0 | DAC 0 Interrupt Pending when set.<br>Write a 1 to clear this bit. |
| 23 | Reserved | - | - | - |
| 22:19 | Reserved | - | - | - |
| 18 | ADC_IRQ_EN | RW | 0 | 1 : Enable ADC interrupt |
| 17 | DAC1_IRQ_EN | RW | 0 | 1 : Enable DAC 1 interrupt. |
| 16 | DAC0_IRQ_EN | RW | 0 | 1 : Enable DAC 0 interrupt. |
| 15 | ADC_START | RWAC | 0 | Write 1 to start ADC operations. This bit will be automatically cleared if ADC_CONT = 0; otherwise when ADC_CONT is set, ADC operation runs till ADC_START is cleared. |
| 14 | Reserved | - | - | - |
| 13 | ADC_EXT_VREF | RW | 0 | 1 : Enable Rail-Rail Reference. The voltage reference supplied externally to the chip. i.e. the voltage on ADC_VREFP. |
| 12 | ADC_CONT | RW | 0 | 1 : Enable ADC continuous mode; When continuous mode is enabled, samples are loaded into the FIFO. |


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 11 | ADC_PDB | RW | 0 | 0 : power down ADC<br>1 : power up ADC<br>Set to 0 if ADC is not used in the chip configuration. |
| 10:8 | ADC_CHANNEL | RW | 0 | If ADC is not used, set these bits to 0.<br>0 : No channel is selected<br>1 : Channel 1 selected<br>2 : Channel 2 selected<br>3 : Channel 3 selected |
| 7 | DAC1_START | RWAC | 0 | Write 1 to start DAC 1 operations. This bit will be automatically cleared if DAC_CONT1 = 0; otherwise, if DAC_CONT1=1, DAC 1 operation runs till DAC1_START is cleared. |
| 6 | Reserved | - | - | - |
| 5 | DAC1_CONT | RW | 0 | 1 : Enable DAC 1 continuous mode; fifo base |
| 4 | DAC1_PDB | RW | 0 | 0 : power down DAC 1<br>Set to 0 if DAC 1 Is not used in the chip configuration. |
| 3 | DAC0_START | RW | 0 | Write 1 to start DAC 0 operations. This bit will be automatically cleared if DAC_CONT0 = 0; otherwise, if DAC_CONT0=1, DAC 0 operation runs till DAC0_START is cleared. |
| 2 | Reserved | - | - | - |
| 1 | DAC0_CONT | RW | 0 | 1 : Enable DAC 0 continuous mode; FIFO base |
| 0 | DAC0_PDB | RW | 0 | 0 : power down DAC 0<br>Set to 0 if DAC 0 Is not used in the chip configuration. |

**Table 5-33 - DAC_ADC_CONF - ADC/DAC Configuration/Status Register**

## 5.2.14 DAC_ADC_CNT - ADC/DAC Count Register (address offset: 0xB4)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31 | Reserved | - | - | - |
| 30:24 | DAC_DIVIDER | RW | 7'h63 | This determines the DAC 1 and DAC 0 conversion rate. The rate is determined by Peripheral clock freq / (DAC_DIVIDER+1)<br>The maximum conversion rate is 1MHz. |
| 23:16 | ADC_DATA_COUNT | RO | X | The amount of data available for reading in the ADC FIFO at the most recent interrupt. Note this value does not reflect the amount of data available in real time. |
| 15:8 | DAC1_DATA_COUNT | RO | X | The FIFO contains the numbers of samples for conversion. This is equal to DAC1_DATA_COUNT plus one.<br>**DACn_DATA_COUNT** / **Samples left to convert** / **Note**<br>0 / 1 /<br>1 / 2 /<br>... / ... /<br>127 / 128 / full<br>255 / 0 / empty |
| 7:0 | DAC0_DATA_COUNT | RO | X | The FIFO contains the numbers of samples for conversion. This is equal to DAC1_DATA_COUNT plus one. |

**Table 5-34 - DAC_ADC_CNT - ADC/DAC Count Register**


Copyright © Bridgetek Pte Ltd

### 5.2.15 DAC_ADC_DATA - ADC/DAC Data Register (address offset: 0xB8)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:26 | Reserved | RO | 0 | |
| 25:16 | DAC1_DATA | WO | 0 | DAC 1 Data write window for DAC 1 FIFO;<br>If byte access is used, a write to the FIFO occurs only when the high byte is written. Hence the upper bits should be written last in this case. |
| 15:10 | Reserved | RO | 0 | |
| 9:8 | Reserved | RO | 0 | |
| 7:0 | ADC_DATA | RO | 0 | ADC Data read window from FIFO for ADC. |
| 9:0 | DAC0_DATA | WO | 0 | DAC 0 Data write window for DAC 0 FIFO;<br>If byte access is used, a write to the FIFO occurs only when the high byte is written. Hence the upper bits should be written last in this case. |

**Table 5-35 - DAC_ADC_DATA - ADC/DAC Data Register**

### 5.2.16 GLOBAL_RESET_STATUS – Global Reset and Status Register (address offset: 0xBC)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:3 | - | - | - | - |
| 2 | WDG_RESET | RW1C | 0 | Watchdog Second Level Reset; if this bit is set it indicates the second level watchdog reset has occurred in the previous operation. Write "1" to clear this bit. Please note this bit must be cleared to enable this function again. |
| 1 | SOFT_RESET | RW | 0 | Soft Global Reset<br>Write "1" to reset the system to POR state.<br>Write "0" to clear this bit. This bit must be cleared before it can be enabled again.<br>When "1" is read from this bit, it indicates the soft global reset has been performed from the previous operation.<br>NOTE: This bit must be cleared before it can be enabled again. |
| 0 | HW_RESET | RW1C | 0 | External Hardware Reset; if it is set it indicates the external reset has been activated. Write "1" to clear it. |

**Table 5-36 - GLOBAL_RESET_STATUS - Global Reset and Status Register**


Copyright © Bridgetek Pte Ltd

# 6 Interrupt Controller

The interrupt controller takes in 32 interrupts, and based on the interrupt priorities assigned generates the interrupt to the FT93x together with an ISR address. Nested interrupts are allowed if enabled. By default, it is disabled. Up to 16 levels of nesting is allowed which defaults to only 1 level if nesting is enabled.

When nesting is enabled, only interrupts with higher priorities can interrupt the current interrupt. Interrupts of same or lower priorities will be queued as long as the interrupt sources are not cleared.

The ISR vectors range from 0 to 31, corresponding to interrupts 0 to 31. The actual ISR address corresponds to program memory addresses 2 to 33.

The highest priority interrupt by default is interrupt input 0, and the lowest interrupt input is 31. The priorities however can be rearranged by setting the appropriate registers. Each interrupt input is assigned an interrupt priority position that can be changed. Note it's possible to assign multiple interrupts to the same priority. By default, the interrupts 0 to 31 are assigned interrupt priorities 0 to 31 respectively, with a lower number indicating a higher priority.

**A global interrupt mask bit is also available. Setting it to 1 will temporarily block all interrupts except the interrupt assigned as interrupt 0 which is non-maskable by this global mask.**

In the FT93x, the interrupt connections from the peripherals are listed in the table below. Interrupts 23 to 31 are unused by default.

| Peripheral Interrupt | Interrupt Controller Interrupt Input Number | Default Priority |
|---|---|---|
| Power Management | 0 | 0 (Highest) – Non-Maskable. |
| Reserved | 1 | 1 |
| USB Dev | 2 | 2 |
| Reserved | 3 | 3 |
| SD Card | 4 | 4 |
| Reserved | 6:5 | 6:5 |
| UART 3 | 7 | 7 |
| SPI Master | 8 | 8 |
| SPI Slave | 9 | 9 |
| Reserved | 10 | 10 |
| I2C Master | 11 | 11 |
| I2C Slave | 12 | 12 |
| UART 0 | 13 | 13 |
| UART 1 | 14 | 14 |
| UART 2 | 15 | 15 |
| PWM | 16 | 16 |
| TIMERS | 17 | 17 |
| GPIO | 18 | 18 |
| RTC | 19 | 19 |
| ADC | 20 | 20 |
| DAC | 21 | 21 |
| SLOWCLOCK Timer | 22 | 22 |
| Reserved | 31:23 | 31:23 |

**Table 6-1 - Interrupt Assignment Table**

## 6.1 Register Summary

The base address for the interrupt assignment registers is 0x10100. All registers and RAM locations can be accessed via Byte (8-bit), Word (16-bit) or Double-Word (32-bit) mode.


Copyright © Bridgetek Pte Ltd

| Address Offset | Register | Default Value | Reference |
|---|---|---|---|
| | | | |
| **0x00** | IRQ00-03 Assignment Register | 0x03020100 | Section 6.2.1 |
| **0x04** | IRQ04-07 Assignment Register | 0x07060504 | Section 6.2.2 |
| **0x08** | IRQ08-11 Assignment Register | 0x0B0A0908 | Section 6.2.3 |
| **0x0C** | IRQ12-15 Assignment Register | 0x0F0E0D0C | Section 6.2.4 |
| **0x10** | IRQ16-19 Assignment Register | 0x13121110 | Section 6.2.5 |
| **0x14** | IRQ20-23 Assignment Register | 0x17161514 | Section 6.2.6 |
| **0x18** | IRQ24-27 Assignment Register | 0x1B1A1918 | Section 6.2.7 |
| **0x1C** | IRQ28-31 Assignment Register | 0x1F1E1D1C | Section 6.2.8 |
| **0x20** | IRQ Control Register | 0x80000000 | Section 6.2.9 |

**Table 6-2 - Overview of Interrupt Control Registers**

## 6.2 Register Details

### 6.2.1 IRQ00-03 Assignment Register (address offset: 0x00)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:29 | Reserved | - | - | - |
| 28:24 | PR03ASSIGN | RW | 5'h03 | Priority assignment for interrupt 3 |
| 23:21 | Reserved | - | - | - |
| 20:16 | PR02ASSIGN | RW | 5'h02 | Priority assignment for interrupt 2 |
| 15:13 | Reserved | - | - | - |
| 12:8 | PR01ASSIGN | RW | 5'h01 | Priority assignment for interrupt 1 |
| 7:5 | Reserved | - | - | - |
| 4:0 | PR00ASSIGN | RW | 5'h00 | Priority assignment for interrupt 0 |

**Table 6-3 - IRQ00-03 Assignment Register**

### 6.2.2 IRQ04-07 Assignment Register (address offset: 0x04)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:29 | Reserved | - | - | - |
| 28:24 | PR07ASSIGN | RW | 5'h07 | Priority assignment for interrupt 7 |
| 23:21 | Reserved | - | - | - |
| 20:16 | PR06ASSIGN | RW | 5'h06 | Priority assignment for interrupt 6 |
| 15:13 | Reserved | - | - | - |
| 12:8 | PR05ASSIGN | RW | 5'h05 | Priority assignment for interrupt 5 |
| 7:5 | Reserved | - | - | - |
| 4:0 | PR04ASSIGN | RW | 5'h04 | Priority assignment for interrupt 4 |

**Table 6-4 - IRQ04-07 Assignment Register**


Copyright © Bridgetek Pte Ltd

### 6.2.3 IRQ08-11 Assignment Register (address offset: 0x08)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:29 | Reserved | - | - | - |
| 28:24 | PR11ASSIGN | RW | 5'h0B | Priority assignment for interrupt 11 |
| 23:21 | Reserved | - | - | - |
| 20:16 | PR10ASSIGN | RW | 5'h0A | Priority assignment for interrupt 10 |
| 15:13 | Reserved | - | - | - |
| 12:8 | PR09ASSIGN | RW | 5'h09 | Priority assignment for interrupt 9 |
| 7:5 | Reserved | - | - | - |
| 4:0 | PR08ASSIGN | RW | 5'h08 | Priority assignment for interrupt 8 |

**Table 6-5 - IRQ08-11 Assignment Register**

### 6.2.4 IRQ12-15 Assignment Register (address offset: 0x0C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:29 | Reserved | - | - | - |
| 28:24 | PR15ASSIGN | RW | 5'h0F | Priority assignment for interrupt 15 |
| 23:21 | Reserved | - | - | - |
| 20:16 | PR14ASSIGN | RW | 5'h0E | Priority assignment for interrupt 14 |
| 15:13 | Reserved | - | - | - |
| 12:8 | PR13ASSIGN | RW | 5'h0D | Priority assignment for interrupt 13 |
| 7:5 | Reserved | - | - | - |
| 4:0 | PR12ASSIGN | RW | 5'h0C | Priority assignment for interrupt 12 |

**Table 6-6 - IRQ12-15 Assignment Register**

### 6.2.5 IRQ16-19 Assignment Register (address offset: 0x10)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:29 | Reserved | - | - | - |
| 28:24 | PR19ASSIGN | RW | 5'h13 | Priority assignment for interrupt 19 |
| 23:21 | Reserved | - | - | - |
| 20:16 | PR18ASSIGN | RW | 5'h12 | Priority assignment for interrupt 18 |
| 15:13 | Reserved | - | - | - |
| 12:8 | PR17ASSIGN | RW | 5'h11 | Priority assignment for interrupt 17 |
| 7:5 | Reserved | - | - | - |
| 4:0 | PR16ASSIGN | RW | 5'h10 | Priority assignment for interrupt 16 |

**Table 6-7 - IRQ16-19 Assignment Register**


Copyright © Bridgetek Pte Ltd

### 6.2.6 IRQ20-23 Assignment Register (address offset: 0x14)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:29 | Reserved | - | - | - |
| 28:24 | PR23ASSIGN | RW | 5'h17 | Priority assignment for interrupt 23 |
| 23:21 | Reserved | - | - | - |
| 20:16 | PR22ASSIGN | RW | 5'h16 | Priority assignment for interrupt 22 |
| 15:13 | Reserved | - | - | - |
| 12:8 | PR21ASSIGN | RW | 5'h15 | Priority assignment for interrupt 21 |
| 7:5 | Reserved | - | - | - |
| 4:0 | PR20ASSIGN | RW | 5'h14 | Priority assignment for interrupt 20 |

**Table 6-8 - IRQ20-23 Assignment Register**

### 6.2.7 IRQ24-27 Assignment Register (address offset: 0x18)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:29 | Reserved | - | - | - |
| 28:24 | PR27ASSIGN | RW | 5'h1B | Priority assignment for interrupt 27 |
| 23:21 | Reserved | - | - | - |
| 20:16 | PR26ASSIGN | RW | 5'h1A | Priority assignment for interrupt 26 |
| 15:13 | Reserved | - | - | - |
| 12:8 | PR25ASSIGN | RW | 5'h19 | Priority assignment for interrupt 25 |
| 7:5 | Reserved | - | - | - |
| 4:0 | PR24ASSIGN | RW | 5'h18 | Priority assignment for interrupt 24 |

**Table 6-9 - IRQ24-27 Assignment Register**

### 6.2.8 IRQ28-31 Assignment Register (address offset: 0x1C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:29 | Reserved | - | - | - |
| 28:24 | PR31ASSIGN | RW | 5'h1F | Priority assignment for interrupt 31 |
| 23:21 | Reserved | - | - | - |
| 20:16 | PR30ASSIGN | RW | 5'h1E | Priority assignment for interrupt 30 |
| 15:13 | Reserved | - | - | - |
| 12:8 | PR29ASSIGN | RW | 5'h1D | Priority assignment for interrupt 29 |
| 7:5 | Reserved | - | - | - |
| 4:0 | PR28ASSIGN | RW | 5'h1C | Priority assignment for interrupt 28 |

**Table 6-10 - IRQ28-31 Assignment Register**


Copyright © Bridgetek Pte Ltd

### 6.2.9 IRQ Control Register (address offset: 0x20)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31 | Global Interrupt Mask | RW | 1 | Set to 1 to mask all interrupts. |
| 30:8 | Reserved | - | - | - |
| 7 | Nested Interrupt | RW | 0 | Set to 1 to enable nested interrupts |
| 6:4 | Reserved | - | - | - |
| 3:0 | Nested Depth | RW | 4'h00 | Maximum number of nested interrupts permitted (Nested Depth + 1); minimum 1 level, and maximum 16 levels. |

**Table 6-11 - IRQ Control Register**


Copyright © Bridgetek Pte Ltd

# 7 USB Device

This is a USB device controller fully compliant with the USB 2.0 specification. It supports a control end point - End Point 0 (EP0) - and up to 7 other End Points (EP1-7).

The EP0 control endpoint buffer size ranges from 8 to 64 bytes, configurable via software. EP1-7 support optional double buffering. The number of end points supported can be set by software, as well as their individual direction, type (Interrupt, Bulk, and Isochronous) and buffer size (8-1024 bytes). The total buffer size to be shared by all end points is 8kB.

## 7.1 Register Summary

Listed below are the registers with their offset from the base address (0x10200). All registers can only be accessed via Byte mode, except all endpoints' Data Buffer Registers are accessed via Double-Word, Word and Byte modes.

| Address Offset | Register | Default Value | Reference |
|---|---|---|---|
| | **Initialization Registers** | | |
| **0x18** | DC_ADDRESS_ENABLE – Address Register | 0x00 | Section 7.2.1 |
| **0x10** | DC_MODE – Mode Register | 0x00 | Section 7.2.2 |
| **0x08** | DC_INT_ENABLE – Interrupt Enable Register | 0x00 | Section 7.2.3 |
| **0x0C** | DC_EP_INT_ENABLE – Endpoints Interrupt Enable Register | 0x00 | Section 7.2.4 |
| | **Control Endpoint Data Flow Registers** | | |
| **0x1C** | DC_EP0_CONTROL – Endpoint 0 Control Register | 0x00 | Section 7.3.1 |
| **0x20** | DC_EP0_STATUS – Endpoint 0 Status Register | 0x00 | Section 7.3.2 |
| **0x24** | DC_EP0_BUFFER_LENGTH – Endpoint 0 Buffer Length Register | 0x00 | Section 7.3.3 |
| **0x28** | DC_EP0_BUFFER – Endpoint 0 Buffer Register | 0x00 | Section 7.3.4 |
| | **Other Endpoints Data Flow Registers** | | |
| **0x2C** | DC_EP1_CONTROL – Endpoint 1 Control Register | 0x00 | Section 7.4.1 |
| **0x30** | DC_EP1_STATUS – Endpoint 1 Status Register | 0x00 | Section 7.4.2 |
| **0x34** | DC_EP1_BUFFER_LENGTH_LSB – Endpoint 1 Buffer Length LSB Register | 0x00 | Section 7.4.3 |
| **0x35** | DC_EP1_BUFFER_LENGTH_MSB – Endpoint 1 Buffer Length MSB Register | 0x00 | Section 7.4.4 |
| **0x38** | DC_EP1_BUFFER – Endpoint 1 Buffer Register | 0x00 | Section 7.4.5 |
| **0x3C** | DC_EP2_CONTROL – Endpoint 2 Control Register | 0x00 | Section 7.4.1 |
| **0x40** | DC_EP2_STATUS – Endpoint 2 Status Register | 0x00 | Section 7.4.2 |
| **0x44** | DC_EP2_BUFFER_LENGTH_LSB – Endpoint 2 Buffer Length LSB Register | 0x00 | Section 7.4.3 |
| **0x45** | DC_EP2_BUFFER_LENGTH_MSB – Endpoint 2 Buffer Length MSB Register | 0x00 | Section 7.4.4 |
| **0x48** | DC_EP2_BUFFER – Endpoint 2 Buffer Register | 0x00 | Section 7.4.5 |


Copyright © Bridgetek Pte Ltd

| Address Offset | Register | Default Value | Reference |
|---|---|---|---|
| | | | |
| **0x4C** | DC_EP3_CONTROL – Endpoint 3 Control Register | 0x00 | Section 7.4.1 |
| **0x50** | DC_EP3_STATUS – Endpoint 3 Status Register | 0x00 | Section 7.4.2 |
| **0x54** | DC_EP3_BUFFER_LENGTH_LSB – Endpoint 3 Buffer Length LSB Register | 0x00 | Section 7.4.3 |
| **0x55** | DC_EP3_BUFFER_LENGTH_MSB – Endpoint 3 Buffer Length MSB Register | 0x00 | Section 7.4.4 |
| **0x58** | DC_EP3_BUFFER – Endpoint 3 Buffer Register | 0x00 | Section 7.4.5 |
| **0x5C** | DC_EP4_CONTROL – Endpoint 4 Control Register | 0x00 | Section 7.4.1 |
| **0x60** | DC_EP4_STATUS – Endpoint 4 Status Register | 0x00 | Section 7.4.2 |
| **0x64** | DC_EP4_BUFFER_LENGTH_LSB – Endpoint 4 Buffer Length LSB Register | 0x00 | Section 7.4.3 |
| **0x65** | DC_EP4_BUFFER_LENGTH_MSB – Endpoint 4 Buffer Length MSB Register | 0x00 | Section 7.4.4 |
| **0x68** | DC_EP4_BUFFER – Endpoint 4 Buffer Register | 0x00 | Section 7.4.5 |
| **0x6C** | DC_EP5_CONTROL – Endpoint 5 Control Register | 0x00 | Section 7.4.1 |
| **0x70** | DC_EP5_STATUS – Endpoint 5 Status Register | 0x00 | Section 7.4.2 |
| **0x74** | DC_EP5_BUFFER_LENGTH_LSB – Endpoint 5 Buffer Length LSB Register | 0x00 | Section 7.4.3 |
| **0x75** | DC_EP5_BUFFER_LENGTH_MSB – Endpoint 5 Buffer Length MSB Register | 0x00 | Section 7.4.4 |
| **0x78** | DC_EP5_BUFFER – Endpoint 5 Buffer Register | 0x00 | Section 7.4.5 |
| **0x7C** | DC_EP6_CONTROL – Endpoint 6 Control Register | 0x00 | Section 7.4.1 |
| **0x80** | DC_EP6_STATUS – Endpoint 6 Status Register | 0x00 | Section 7.4.2 |
| **0x84** | DC_EP6_BUFFER_LENGTH_LSB – Endpoint 6 Buffer Length LSB Register | 0x00 | Section 7.4.3 |
| **0x85** | DC_EP6_BUFFER_LENGTH_MSB – Endpoint 6 Buffer Length MSB Register | 0x00 | Section 7.4.4 |
| **0x88** | DC_EP6_BUFFER – Endpoint 6 Buffer Register | 0x00 | Section 7.4.5 |
| **0x8C** | DC_EP7_CONTROL – Endpoint 7 Control Register | 0x00 | Section 7.4.1 |
| **0x90** | DC_EP7_STATUS – Endpoint 7 Status Register | 0x00 | Section 7.4.2 |
| **0x94** | DC_EP7_BUFFER_LENGTH_LSB – Endpoint 7 Buffer Length LSB Register | 0x00 | Section 7.4.3 |
| **0x95** | DC_EP7_BUFFER_LENGTH_MSB – Endpoint 7 Buffer Length MSB Register | 0x00 | Section 7.4.4 |
| **0x98** | DC_EP7_BUFFER – Endpoint 7 Buffer Register | 0x00 | Section 7.4.5 |
| | **General Registers** | | |
| **0x00** | DC_INT_STATUS – Interrupt Status Register | 0x00 | Section 7.5.1 |
| **0x04** | DC_EP_INT_STATUS – Endpoints Interrupt Status Register | 0x00 | Section 7.5.2 |
| **0x14** | DC_FRAME_NUMBER_LSB – Frame Number LSB Register | 0x00 | Section 7.5.3 |


Copyright © Bridgetek Pte Ltd

| Address Offset | Register | Default Value | Reference |
|---|---|---|---|
| **0x15** | DC_FRAME_NUMBER_MSB – Frame Number MSB Register | 0x00 | Section 7.5.4 |

**Table 7-1 - Overview of USB Device Registers**

# 7.2 Initialization Registers

## 7.2.1 DC_ADDRESS_ENABLE – Address Register (address offset: 0x18)

This register sets the USB assigned address sent from the USB Host and enables the USB Device. In response to the standard USB request SET_ADDRESS, the firmware must write the peripheral address to this register.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | ENABLE | RO | 1'b0 | Hardware sets this to 1 when software writes a new address to this register. It is cleared by hardware at the end of the current transfer when the new address will take effect. |
| 6:0 | ADDR | RW | 7'h00 | Function Address |

**Table 7-2 - DC_ADDRESS_ENABLE – Address Register**

## 7.2.2 DC_MODE – Mode Register (address offset: 0x10)

This register allows the firmware to select the different test modes and enables the USB Device function.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | TST_MODE_ENABLE | RW | 1'b0 | Test Mode Enable. Setting this to 1 to enter the test mode. It can only be cleared by hardware reset. |
| 6:5 | TST_MODE_SELECT | RW | 2'h0 | Test Mode select (writeable only if MODE_ENABLE is 0)<br>2'h0: SE0_NAK<br>2'h1: J<br>2'h2: K<br>2'h3: Packet |
| 4:2 | Reserved | - | - | - |
| 1 | FS_ONLY | RW | 1'b0 | Setting this to 1 disables HS detection handshake |
| 0 | USB_DEV_EN | RW | 1'b0 | USB function enables. Setting this to 1 enables the USB Device |

**Table 7-3 - DC_MODE – Mode Register**

## 7.2.3 DC_INT_ENABLE – Interrupt Enable Register (address offset: 0x08)

This register enables the different interrupt sources by writing a 1 to the corresponding bit.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | PHY_IE | RW | 1'b0 | PHY receive error interrupt enable |
| 6 | PID_IE | RW | 1'b0 | Package ID error interrupt enable |
| 5 | CRC16_IE | RW | 1'b0 | CRC16 error interrupt enable |
| 4 | CRC5_IE | RW | 1'b0 | CRC5 error interrupt enable |
| 3 | RESM_IE | RW | 1'b0 | Resume interrupt enable |
| 2 | SUS_IE | RW | 1'b0 | Suspend interrupt enable |
| 1 | RST_IE | RW | 1'b0 | Reset interrupt enable |
| 0 | SOF_IE | RW | 1'b0 | Start of Frame interrupt enable |

**Table 7-4 - DC_INT_ENABLE – Interrupt Enable Register**


Copyright © Bridgetek Pte Ltd

### 7.2.4 DC_EP_INT_ENABLE – Endpoints Interrupt Enable Register (address offset: 0x0C)

This register enables the different interrupt sources based on the specific endpoints by writing 1 to the corresponding bit.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | EP7_IE | RW | 1'b0 | Endpoint 7 interrupt enable |
| 6 | EP6_IE | RW | 1'b0 | Endpoint 6 interrupt enable |
| 5 | EP5_IE | RW | 1'b0 | Endpoint 5 interrupt enable |
| 4 | EP4_IE | RW | 1'b0 | Endpoint 4 interrupt enable |
| 3 | EP3_IE | RW | 1'b0 | Endpoint 3 interrupt enable |
| 2 | EP2_IE | RW | 1'b0 | Endpoint 2 interrupt enable |
| 1 | EP1_IE | RW | 1'b0 | Endpoint 1 interrupt enable |
| 0 | EP0_IE | RW | 1'b0 | Endpoint 0 interrupt enable |

**Table 7-5 - DC_EP_INT_ENABLE – Endpoints Interrupt Enable Register**

## 7.3 Control Endpoint Data Flow Registers

Control Endpoint Data flow registers are used for configuring the control endpoint and handle the sending and receiving data to the control endpoint.

### 7.3.1 DC_EP0_CONTROL – Endpoint 0 Control Register (address offset: 0x1C)

This register configures the maximum packet size of the control endpoint 0. It is also used to stall the control endpoint.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:3 | Reserved | - | - | - |
| 2:1 | EP_SIZE | RW | 2'h0 | Endpoint Maximum packet size<br><br>2'h0: 8 bytes<br>2'h1: 16 bytes<br>2'h2: 32 bytes<br>2'h3: 64 bytes |
| 0 | STALL | RW | 1'b0 | Send STALL<br><br>Software writes 1 to send a STALL handshake in response to an IN token. Software writes 0 to terminate the STALL signalling. |

**Table 7-6 - DC_EP0_CONTROL – Endpoint 0 Control Register**

### 7.3.2 DC_EP0_STATUS – Endpoint 0 Status Register (address offset: 0x20)

This register is used by the hardware to report the status of the control endpoint 0. Software writes 1 to the corresponding register bit to clear the status.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:5 | Reserved | - | - | - |
| 4 | DATA_END | RW1S | 1'b0 | Data End.<br><br>Software should update this bit to 1 when writing:<br>• 1 to IN_PKT_RDY for last outgoing data packet<br>• 1 to IN_PKT_RDY for a zero-length data packet<br>• 1 to OUT_PKT_RDY after servicing the |


Copyright © Bridgetek Pte Ltd

| | | | | |
|---|---|---|---|---|
| | | | | last incoming data packet |
| 3 | STALL | RW1C | 1'b0 | Sent STALL.<br>Hardware sets this to 1 when the STALL handshake has been transmitted. Software writes 1 to clear it. |
| 2 | SETUP | RW1C | 1'b0 | SETUP token received.<br>Hardware sets this to 1 and interrupts when the SETUP token has been received. Software writes 1 to clear it. |
| 1 | IN_PKT_RDY | RW1S | 1'b0 | IN packet ready.<br>Software should write 1 to it after loading a data packet into the endpoint 0 IN FIFO. Hardware clears it and generates an interrupt when the data packet has been successfully transmitted. |
| 0 | OUT_PKT_RDY | RW1C | 1'b0 | OUT packet ready.<br>Hardware sets this bit to 1 and generates an interrupt when a data packet has been received. Software writes 1 to clear it after unloading the data packet from the endpoint 0 OUT FIFO. |

**Table 7-7 - DC_EP0_STATUS – Endpoint 0 Status Register**

### 7.3.3 DC_EP0_BUFFER_LENGTH – Endpoint 0 Buffer Length Register (address offset: 0x24)

This register is used by the hardware to report the length of the packet received in the endpoint 0 OUT buffer.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | Reserved | - | - | - |
| 6:0 | BUF_LEN | RO | 7'h00 | Indicates the number of data bytes received. Valid only if OUT_PKT_RDY is 1. |

**Table 7-8 - DC_EP0_BUFFER_LENGTH – Endpoint 0 Buffer Length Register**

### 7.3.4 DC_EP0_BUFFER – Endpoint 0 Buffer Register (address offset: 0x28)

This register is the used to access the endpoint 0 FIFO.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | Data | R/W | 8'h00 | Endpoint 0 FIFO window register. A read unloads one data byte from the Endpoint 0 OUT FIFO. A write loads one data byte into the Endpoint 0 IN FIFO. |

**Table 7-9 - DC_EP0_BUFFER – Endpoint 0 Buffer Register**

## 7.4 Other Endpoint Data Flow Registers

This is a set of register used to access the other endpoints from endpoint 1 to endpoint 7. The register definitions are the same for the registers from endpoint 1 to endpoint 7 but they are located at different offset addresses.


Copyright © Bridgetek Pte Ltd

### 7.4.1 DC_EP(x)_CONTROL – Endpoint Control Registers (address offset: 0x2C/0x3C/0x4C/0x5C/0x6C/0x7C/0x8C)

The different endpoints are configured by writing to the corresponding endpoint control register selected based on the corresponding offset address. This register is used to configure the endpoint type, direction, maximum packet size and double buffering. It is also used to stall the corresponding data endpoint.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | DBL_BUF | RW | 1'b0 | Endpoint double buffering Enable.<br><br>0: double buffering disabled<br>1: double buffering enabled |
| 6:4 | EP_SIZE | RW | 3'h0 | Endpoint Maximum packet size. This parameter has to be fixed during the Set Configuration request.<br><br>0x0: 8 bytes<br>0x1: 16 bytes<br>0x2: 32 bytes<br>0x3: 64 bytes<br>0x4: 128 bytes<br>0x5: 256 bytes<br>0x6: 512 bytes<br>0x7: 1024 bytes |
| 3 | STALL | RW | 1'b0 | Send STALL. Valid only when the Endpoint is in bulk or interrupt mode. Software should write a 1 to this bit to send a STALL handshake in response to IN token, PING token and data phase of OUT transaction. Software should write a 0 to this bit to terminate the STALL signalling. |
| 2:1 | EP_MODE | RW | 2'h0 | Endpoint Mode. This parameter has to be fixed during the Set Configuration request.<br><br>0x0: EP disabled<br>0x1: EP configured for bulk transfers<br>0x2: EP configured for interrupt transfers<br>0x3: EP configured for isochronous transfers |
| 0 | EP_DIR | RW | 1'b0 | Endpoint Direction. This parameter has to be fixed during the Set Configuration request.<br><br>0x0: EP direction selected as OUT<br>0x1: EP direction selected as IN |

**Table 7-10 - DC_EP(x)_CONTROL – Endpoint Control Registers**

### 7.4.2 DC_EP(x)_STATUS – Endpoint Status Registers (address offset: 0x30/0x40/0x50/0x60/0x70/0x80/0x90)

The different endpoint statuses are read by accessing the corresponding status register. The hardware reports the endpoint status in this register. Write a 1 to the corresponding register bit to clear the bit.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | CLR_TOGGLE | RW1S | 1'b0 | Clear data toggle. Software can write a 1 to this bit to reset the Endpoint data toggle to 0. This bit is always read as 0. |
| 6 | FIFO_FLUSH | RW1S | 1'b0 | FIFO Flush. Valid only when the EP direction is IN. Writing 1 to this bit flushes the next packet to be transmitted from the Endpoint IN FIFO. The FIFO pointer is reset and the IN_PKT_RDY bit is cleared. Hardware resets the FLUSH bit to 0 when the FIFO flush is complete. |
| 5 | DATA_ERR | RW1C | 1'b0 | Data error. Valid only when the Endpoint is |


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| | | | | in isochronous mode and the direction is OUT. This flag is set to 1 by hardware if a received packet has a CRC-16 error. It is automatically cleared when software clears the OUT_PKT_RDY bit. |
| 4 | STALL | RW1C | 1'b0 | Sent STALL. Valid only when the Endpoint is in bulk or interrupt mode. Hardware sets this bit to 1 when the STALL handshake is transmitted. Software can clear this bit by writing a 1 to this bit. |
| 3 | UNDER_RUN | RW1C | 1'b0 | Data underrun. Valid only when the Endpoint direction is IN. Its function is dependent upon the Endpoint mode:<br>Isochronous: Hardware sets this to 1 when a zero-length packet is sent in response to an IN token while IN_PKT_RDY is 0.<br>Bulk/Interrupt: Hardware sets this to 1 when a NAK packet is sent in response to an IN token while IN_PKT_RDY is 0.<br>Software can clear this bit by writing a 1 to this bit. |
| 2 | OVER_RUN | RW1C | 1'b0 | Data overrun. Valid only when the Endpoint is in isochronous mode and the direction is OUT.<br>Hardware sets this bit to 1 if a received packet cannot be loaded into the Endpoint FIFO.<br>Software can clear this bit by writing a 1 to this bit. |
| 1 | IN_PKT_RDY | RW1S | 1'b0 | IN packet ready. Valid only when the Endpoint direction is IN. Software should write a 1 to this bit after loading a data packet into the Endpoint IN FIFO.<br>Hardware clears this bit and generates an interrupt when the data packet has been successfully transmitted |
| 0 | OUT_PKT_RDY | RW1C | 1'b0 | OUT packet ready. Valid only when the Endpoint direction is OUT. Hardware sets this bit to 1 and generates an interrupt when a data packet has been received. Software writes a 1 to clear it after unloading the data packet from the Endpoint OUT FIFO. |

**Table 7-11 - DC_EP(x)_STATUS – Endpoint Status Registers**

### 7.4.3 DC_EP(x)_BUFFER_LENGTH_LSB – Endpoint Buffer Length LSB Registers (address offset: 0x34/0x44/0x54/0x64/0x74/0x84/0x94)

The different endpoint buffer length is read by accessing the corresponding buffer length register. This register reports the LSB of the OUT-buffer length.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | BUF_LEN_LSB | RO | 8'h00 | Indicates the low byte of the number of received data bytes in the Endpoint FIFO. Valid only if OUT_PKT_RDY is 1. |

**Table 7-12 - DC_EP(x)_BUFFER_LENGTH_LSB – Endpoint Buffer Length LSB Registers**


Copyright © Bridgetek Pte Ltd

### 7.4.4 DC_EP(x)_BUFFER_LENGTH_MSB – Endpoint Buffer Length MSB Registers (address offset: 0x35/0x45/0x55/0x65/0x75/0x85/0x95)

The different endpoint buffer length is read by accessing the corresponding buffer length register. This register reports the MSB of the OUT-buffer length.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:3 | Reserved | RO | 5'h00 | - |
| 2:0 | BUF_LEN_MSB | RO | 3'h0 | Indicates the high 3-bit of the number of received data bytes in the Endpoint FIFO. Valid only if OUT_PKT_RDY is 1. |

**Table 7-13 - DC_EP(x)_BUFFER_LENGTH_MSB – Endpoint Buffer Length MSB Registers**

### 7.4.5 DC_EP(x)_BUFFER – Endpoint Buffer Registers (address offset: 0x38/0x48/0x58/0x68/0x78/0x88/0x98)

The different endpoint buffer is accessed by accessing the corresponding buffer register.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | buffer | RW | 8'h00 | Endpoint FIFO window register. A read unloads one data byte from the Endpoint OUT FIFO. A write loads one data byte into the Endpoint IN FIFO. |

**Table 7-14 - DC_EP(x)_BUFFER – Endpoint Buffer Registers**

## 7.5 General Registers

### 7.5.1 DC_INT_STATUS – Interrupt Status Register (address offset: 0x00)

This register indicates that the hardware condition of the corresponding interrupt has occurred. Write a 1 to clear the corresponding bit.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | PHY | RW1C | 1'b0 | PHY receive error interrupt status |
| 6 | PID | RW1C | 1'b0 | Package ID error interrupt status |
| 5 | CRC16 | RW1C | 1'b0 | CRC16 error interrupt status |
| 4 | CRC5 | RW1C | 1'b0 | CRC5 error interrupt status |
| 3 | RESM | RW1C | 1'b0 | Resume interrupt status |
| 2 | SUS | RW1C | 1'b0 | Suspend interrupt status |
| 1 | RST | RW1C | 1'b0 | Reset interrupt status |
| 0 | SOF | RW1C | 1'b0 | Start of Frame interrupt status |

**Table 7-15 - DC_INT_STATUS – Interrupt Status Register**

### 7.5.2 DC_EP_INT_STATUS – Endpoints Interrupt Status Register (address offset: 0x04)

This register indicates the corresponding endpoint hardware condition has occurred. Write a 1 to clear the corresponding bit.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | EP7 | RW1C | 1'b0 | Endpoint 7 interrupt status |
| 6 | EP6 | RW1C | 1'b0 | Endpoint 6 interrupt status |
| 5 | EP5 | RW1C | 1'b0 | Endpoint 5 interrupt status |
| 4 | EP4 | RW1C | 1'b0 | Endpoint 4 interrupt status |
| 3 | EP3 | RW1C | 1'b0 | Endpoint 3 interrupt status |
| 2 | EP2 | RW1C | 1'b0 | Endpoint 2 interrupt status |


Copyright © Bridgetek Pte Ltd

| | | | | |
|---|---|---|---|---|
| 1 | EP1 | RW1C | 1'b0 | Endpoint 1 interrupt status |
| 0 | EP0 | RW1C | 1'b0 | Endpoint 0 interrupt status |

**Table 7-16 - DC_EP_INT_STATUS – Endpoints Interrupt Status Register**

### 7.5.3 DC_FRAME_NUMBER_LSB – Frame Number LSB Register (address offset: 0x14)

This register has the LSB of the last successfully received SOF.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | FRAME_LSB | RO | 8'h00 | Frame Number for last received SOF, least significant byte |

**Table 7-17 - DC_FRAME_NUMBER_LSB – Frame Number LSB Register**

### 7.5.4 DC_FRAME_NUMBER_MSB – Frame Number MSB Register (address offset: 0x15)

This register has the MSB of the last successfully received SOF.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:3 | Reserved | RO | 5'h00 | - |
| 2:0 | FRAME_MSB | RO | 3'h0 | Frame Number for last received SOF, Most significant byte |

**Table 7-18 - DC_FRAME_NUMBER_MSB – Frame Number MSB Register**


Copyright © Bridgetek Pte Ltd

# 8 SD Host

The device supports an SD Host with the following features.

- Supports PIO data transfers
- Supports configurable SD bus modes: 4-bit mode and 8-bit mode
- Supports configurable 8-word/16-word register for data FIFO
- Supports configurable 1K/2K/4K SRAM for data FIFO
- Supports configurable 1-bit/4-bit SD card bus and 1-bit/4-bit/8-bit MMC card bus
- Configurable CPRM function for security
- Built-in generation and check for 7-bit and 16-bit CRC data
- Card detection (Insertion/Removal)
- Supports Read Wait mechanism for SDIO function
- Supports Suspend/Resume mechanism for SDIO function

## 8.1 Register Summary

Listed below are the registers with their offset from the base address (0x10600). All registers can only be accessed via Double-Word (32-bit) mode. Note that some registers are not 32-bit long. In that case, several registers are combined into one 32-bit location. If one such register is accessed, all other registers in the same 32-bit location will also be affected. The user needs to take care not to modify the content of the other registers.

| Address Offset | Register | Default Value | Reference |
|---|---|---|---|
| | | | |
| **0x00** | SDH_AUTO_CMD23_ARG2 – Auto CMD23 Argument 2 Register | 0x00000000 | Section 12.2.1 |
| **0x04** | SDH_BLK_SIZE – Block Size Register | 0x0000 | Section 12.2.2 |
| **0x06** | SDH_BLK_COUNT – Block Count Register | 0x0000 | Section 12.2.3 |
| **0x08** | SDH_ARG_1 – Argument 1 Register | 0x00000000 | Section 12.2.4 |
| **0x0C** | SDH_TNSFER_MODE – Transfer Mode Register | 0x0000 | Section 12.2.5 |
| **0x0E** | SDH_CMD – Command Register | 0x0000 | Section 12.2.6 |
| **0x10 - 0x1C** | SDH_RESPONSE – Response Register | 0x00000000 | Section 12.2.7 |
| **0x20** | SDH_BUF_DATA – Buffer Data Port Register | 0x00000000 | Section 12.2.8 |
| **0x24** | SDH_PRESENT_STATE – Present State Register | 0x00000000 | Section 12.2.9 |
| **0x28** | SDH_HST_CNTL_1 – Host Control 1 Register | 0x00 | Section 12.2.10 |
| **0x29** | SDH_PWR_CNTL – Power Control Register | 0x00 | Section 12.2.11 |
| **0x2A** | SDH_BLK_GAP_CNTL – Block Gap Control Register | 0x0000 | Section 12.2.12 |
| **0x2C** | SDH_CLK_GAP_CNTL – Clock Control Register | 0x0000 | Section 12.2.13 |


Copyright © Bridgetek Pte Ltd

| | | | |
|---|---|---|---|
| **0x2E** | SDH_TIMEOUT_CNTL – Timeout Control Register | 0x00 | Section 12.2.14 |
| **0x2F** | SDH_SW_RST – Software Reset Register | 0x00 | Section 12.2.15 |
| **0x30** | SDH_NRML_INT_STATUS – Normal Interrupt Status Register | 0x0000 | Section 12.2.16 |
| **0x32** | SDH_ERR_INT_STATUS – Error Interrupt Status Register | 0x0000 | Section 12.2.17 |
| **0x34** | SDH_NRML_INT_ENABLE – Normal Interrupt Status Enable Register | 0x0000 | Section 12.2.18 |
| **0x36** | SDH_ERR_INT_ENABLE – Error Interrupt Status Enable Register | 0x0000 | Section 12.2.19 |
| **0x38** | SDH_NRML_INT_SGNL_ENABLE – Normal Interrupt Signal Enable Register | 0x0000 | Section 12.2.20 |
| **0x3A** | SDH_ERR_INT_SGNL_ENABLE – Error Interrupt Signal Enable Register | 0x0000 | Section 12.2.21 |
| **0x3C** | SDH_AUTO_CMD12_ERR_STATUS – Auto CMD12 Error Status Register | 0x0000 | Section 12.2.22 |
| **0x3E** | SDH_HOST_CNTL_2 – Host Control 2 Register | 0x0000 | Section 12.2.23 |
| **0x40** | SDH_CAP_1 – Capabilities Register 1 | 0x016A0080 | Section 12.2.24 |
| **0x44** | SDH_CAP_2 – Capabilities Register 2 | 0x00000F75 | Section 12.2.25 |
| **0x48** | SDH_RSRV_1 – Reserved 1 Register | 0x00000000 | Section 12.2.26 |
| **0x4C** | SDH_RSRV_2 – Reserved 2 Register | 0x00000000 | Section 12.2.27 |
| **0x50** | SDH_FORCE_EVT_CMD_ERR_STATUS – Force Event for Auto CMD Error Status Register | 0x0000 | Section 12.2.28 |
| **0x52** | SDH_FORCE_EVT_ERR_INT_STATUS – Force Event for Error Interrupt Status Register | 0x0000 | Section 12.2.29 |
| **0x54** | SDH_RSRV_3 – Reserved 3 Register | 0x00000000 | Section 12.2.30 |
| **0x58** | SDH_RSRV_4 – Reserved 4 Register | 0x00000000 | Section 12.2.31 |
| **0x60** | SDH_PRST_INIT – Preset value for initialization Register | 0x0000 | Section 12.2.32 |
| **0x62** | SDH_PRST_DFLT_SPD – Preset value for default speed Register | 0x0000 | Section 12.2.33 |
| **0x64** | SDH_PRST_HIGH_SPD – Preset value for the high-speed Register | 0x0000 | Section 12.2.34 |
| **0x66** | SDH_PRST_SDR12 – Preset value for SDR12 Register | 0x0000 | Section 12.2.35 |
| **0x68** | SDH_PRST_SDR25 – Preset value for SDR25 Register | 0x0000 | Section 12.2.36 |
| **0x6A** | SDH_PRST_SDR50 – Preset value for SDR50 Register | 0x0000 | Section 12.2.37 |
| **0x6C** | SDH_PRST_SDR104 – Preset value for SDR104 Register | 0x0000 | Section 12.2.38 |
| **0x6E** | SDH_PRST_DDR50 – Preset value for DDR50 Register | 0x0000 | Section 12.2.39 |
| **0xFC** | SDH_RSRV_5 – Reserved 5 Register | 0x0000 | Section 12.2.40 |
| **0xFE** | SDH_HC_VER – Host Controller Version Register | 0x0002 | Section 12.2.41 |
| **0x100** | SDH_VNDR_0 – Vendor defined 0 Register | 0x00000001 | Section 12.2.42 |
| **0x104** | SDH_VNDR_1 – Vendor defined 1 Register | 0x00000000 | Section 12.2.43 |
| **0x108** | SDH_VNDR_2 – Vendor defined 2 Register | 0x00000000 | Section 12.2.44 |
| **0x10C** | SDH_VNDR_3 – Vendor defined 3 Register | 0x1FXXXXXX | Section 12.2.45 |


Copyright © Bridgetek Pte Ltd

| | | | |
|---|---|---|---|
| **0x110** | SDH_VNDR_4 – Vendor defined 4 Register | 0x00000000 | Section 12.2.46 |
| **0x114** | SDH_VNDR_5 – Vendor defined 5 Register | 0x00000000 | Section 12.2.47 |
| **0x118** | SDH_VNDR_6 – Vendor defined 6 Register | 0x00000001 | Section 12.2.48 |
| **0x11C** | SDH_VNDR_7 – Vendor defined 7 Register | 0x00000000 | Section 12.2.49 |
| **0x120** | SDH_VNDR_8 – Vendor defined 8 Register | 0x00000000 | Section 12.2.50 |
| **0x124** | SDH_VNDR_9 – Vendor defined 9 Register | 0x00000000 | Section 12.2.51 |
| **0x128** | SDH_RSRV_6 – Reserved 6 Register | 0x00000000 | Section 12.2.52 |
| **0x178** | SDH_HW_ATTR – Hardware Attributes Register | 0x00000150 | Section 12.2.53 |
| **0x180** | SDH_CPR_MOD_CNTL – Cipher Mode Control Register | 0x00000000 | Section 12.2.54 |
| **0x184** | SDH_CPR_MOD_STATUS – Cipher Mode Status Register | 0x00000000 | Section 12.2.55 |
| **0x188** | SDH_CPR_MOD_STATUS_EN – Cipher Mode Status Enable Register | 0x0000 | Section 12.2.56 |
| **0x18A** | SDH_CPR_MOD_SIG_EN – Cipher Mode Signal Enable Register | 0x0000 | Section 12.2.57 |
| **0x18C** | SDH_IN_DATA_LSB –Input Data LSB Register | 0x00000000 | Section 12.2.58 |
| **0x190** | SDH_IN_DATA_MSB –Input Data MSB Register | 0x00000000 | Section 12.2.59 |
| **0x194** | SDH_IN_KEY_LSB – Input Key LSB Register | 0x00000000 | Section 12.2.60 |
| **0x198** | SDH_IN_KEY_MSB – Input Key MSB Register | 0x00000000 | Section 12.2.61 |
| **0x19C** | SDH_OUT_DATA_LSB – Output Data LSB Register | 0x00000000 | Section 12.2.62 |
| **0x1A0** | SDH_OUT_DATA_MSB – Output Data MSB Register | 0x00000000 | Section 12.2.63 |
| **0x1A4** | SDH_SCRT_CONS_DATA – Secret Constant Table Data Port | 0x00000000 | Section 12.2.64 |

**Table 8-1 - Overview of SD Host Registers**

## 8.2 Register Details

### 8.2.1 SDH_AUTO_CMD23_ARG2 – Auto CMD23 Argument 2 Register (address offset: 0x00)

This register sets a 32-bit block count to the argument of CMD23 while executing Auto CMD23. The available block count will be limited by BLK_CNT. In this case, 65535 blocks are the maximum value.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:0 | ARG_2 | RW | 32'h00000000 | Auto CMD23 argument 2 |

**Table 8-2 - 11.2.1 SDH_AUTO_CMD23_ARG2 – Auto CMD23 Argument 2 Register**


Copyright © Bridgetek Pte Ltd

### 8.2.2 SDH_BLK_SIZE – Block Size Register (address offset: 0x04)

This register is used to configure the number of bytes in a data block.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 15:12 | Reserved | - | - | - |
| 11:0 | BLK_SIZE | RW | 12'h000 | This register specifies the block size of data transfers for CMD17/18/24/25/53 and can be set with values ranging from 1 up to the maximum buffer size. |

**Table 8-3 - 11.2.2 SDH_BLK_SIZE – Block Size Register**

### 8.2.3 SDH_BLK_COUNT – Block Count Register (address offset: 0x06)

The block count register is set when the BLK_CNT_EN bit is set to 1. This register is used only for the multi-block transfers. The host controller will decrease the counting number during the data transfer and stop counting when it counts down to zero. When a suspend command is completed in the SDIO transfer, the remaining block counts can be determined by reading this register. Before issuing a resume command to start a re-transfer, the host driver should restore the block counts that are previously saved.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:16 | BLK_CNT | RW | 16'h000 | Block count of the current transfer.<br>16'hFFFF: 65535 blocks<br>…<br>16'h0002: 2 blocks<br>16'h0001: 1 block<br><br>16'h0000: Stop counting |

**Table 8-4 - 11.2.3 SDH_BLK_COUNT – Block Count Register**

### 8.2.4 SDH_ARG_1 – Argument 1 Register (address offset: 0x08)

This register is assigned to bits[39:8] of the command field.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:0 | ARG1 | RW | 32'h00000000 | Command argument |

**Table 8-5 - 11.2.4 SDH_ARG_1 – Argument 1 Register**

### 8.2.5 SDH_TNSFER_MODE – Transfer Mode Register (address offset: 0x0C)

The host driver should set this register before issuing the data transfer command or resume command. When in the SDIO transfer, the values of this register should be preserved after the suspend command and should be restored before the resume command.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 15:6 | Reserved | - | - | - |
| 5 | MULTI_BLK | RW | 1'b0 | 1: Multiple blocks selection<br>0: Single block selection |
| 4 | TRAN_DIR_SEL | RW | 1'b0 | 1: Read from the card to host<br>0: Write from the host to card |
| 3:2 | AUTO_CMD_EN | RW | 2'h0 | Auto CMD enable<br>There are two methods to stop the read and |


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| | | | | write operations of multiple blocks:<br>01: Auto CMD12 Enable<br>When this field is set to 01, the host controller will issue a CMD12 when the last block transfer is completed.<br>10: Auto CMD23 Enable<br>When this bit field is set to 10, the host controller will issue a CMD23 before issuing a command specified in the Command Register<br>11: Reserved<br>00: Auto Command Disabled |
| 1 | BLK_CNT_EN | RW | 1'b0 | Block count enable. This bit is only valid for a multi-block transfer. When set to 0 the BLK_CNT register will be disabled. The multi-block transfer will be an infinite transfer |
| 0 | Reserved | RW | 1'b0 | Write 0 to this bit |

**Table 8-6 - 11.2.5 SDH_TNSFER_MODE – Transfer Mode Register**

## 8.2.6 SDH_CMD – Command Register (address offset: 0x0E)

The host driver should check the Command Inhibit (CMD) and Command Inhibit (DAT) bits in the present state register to determine whether the SD bus is free to transfer.

| Bit | Name | Type | Default Value | Description | | |
|---|---|---|---|---|---|---|
| 31:30 | Reserved | - | - | - | | |
| 29:24 | CMD_IDX | RW | 6'h00 | Command Index<br>These bits should be assigned to bits [45:40] of the command field | | |
| 23:22 | CMD_TYPE | RW | 2'h0 | 11 | Abort | CMD12/52 for writing I/O Abort in CCCR |
| | | | | 10 | Resume | CMD52 for writing Function Select in CCCR |
| | | | | 01 | Suspend | CMD52 for writing Bus Suspend in CCCR |
| | | | | 00 | Normal | Other commands |
| 21 | DATA_PRE_SEL | RW | 1'b0 | Data Present Select.<br>1: Indicates that data is present and data transfer is enabled for the command.<br>0: Commands only using the CMD line, commands with no data transfer but set the busy signal on DAT[0], the resume command. | | |
| 20 | CMD_IDX_CHK_EN | RW | 1'b0 | Command Index Check Enable<br>1: the host controller will check the index field response to determine if the values are CMD_IDX. If they are not the same, CMD_IDX_ERR will be triggered | | |
| 19 | CMD_CRC_CHK_EN | RW | 1'b0 | Command CRC Check Enable<br>1: the host controller will check the CRC field response to determine whether the CRC is correct. CMD_CRC_ERR will be triggered if an error is detected. | | |


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 18 | Reserved | - | - | - |
| 17:16 | RSP_TYPE_SEL | RW | 2'h0 | Response Type Select<br>11: Response length 48 with busy check after response<br>10: Response length 48<br>01: Response length 136<br>00: No response |

**Table 8-7 - 11.2.6 SDH_CMD – Command Register**

### 8.2.7 SDH_RESPONSE – Response Register (address offset: 0x10-0x1C)

The host controller preserves the Auto CMD12 response in the upper word of the response register to prevent the value being overwritten by responses to subsequent commands.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 127:0 | RSP | RO | 0 | Command Response |

**Table 8-8 - 11.2.7 SDH_RESPONSE – Response Register**

### 8.2.8 SDH_BUF_DATA – Buffer Data Port Register (address offset: 0x20)

This register uses the 32-bit Data Port Register to access the internal buffer.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:0 | DATA_PORT | RO | 32'h00000000 | Buffer Data Port Register |

**Table 8-9 - 11.2.8 SDH_BUF_DATA – Buffer Data Port Register**

### 8.2.9 SDH_PRESENT_STATE – Present State Register (address offset: 0x24)

The host driver can access the status from this read-only register.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:25 | Reserved | - | - | - |
| 24 | CMD_LIN_LV | RO | X | Command Line Signal Level |
| 23:20 | DATA_LIN_LV | RO | X | Data[3:0] Line Signal Level |
| 19 | WR_PROP_LV | RO | X | Write Protect Pin Level<br>1: Write enabled<br>0: Write protected |
| 18 | CD_PIN_LV | RO | X | Card Detect Pin Level<br>1: Card is detected<br>0: Card is not detected |
| 17 | SYS_CARD_STABLE | RO | X | Card State Stable<br>1: No card or card is inserted<br>0: Reset or de-bounce |
| 16 | SYS_CARD_INSERT | RO | X | Card Inserted<br>1: Card inserted<br>0: Reset, debouncing, or no card is detected |
| 15:12 | Reserved | - | - | - |


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 11 | BUF_RD_EN | ROC | - | Buffer Read Enable<br>1: Read Enable<br>0: Read Disable |
| 10 | BUF_WR_EN | ROC | - | Buffer Write Enable<br>1: Write Enable<br>0: Write Disable |
| 9 | RD_TRAN_ACT | ROC | - | Read Transfer Active<br>1: under the following conditions:<br>(1) After the end bit of a read command<br>(2) When CONT_REQ in the block gap control register is set to restart a transfer.<br>0: under the following conditions:<br>(1) When all data blocks specified by the block length are transferred to the system.<br>(2) When SP_BLK_GAP in the block gap control register is set to 1 and the host controller has transferred all the valid data blocks to the system.<br>The TRAN_CMPLT interrupt is generated when this bit changes from 1 to 0 |
| 8 | WR_TRAN_ACT | ROC | - | Write Transfer Active<br>1: under the following conditions:<br>(1) After the end bit of a write command<br>(2) When CONT_REQ in the block gap control register is set to restart a transfer.<br>0: under the following conditions:<br>(1) After getting the CRC status of the last data block specified by the transfer count.<br>(2) After getting the CRC status of any block where data transmission is stopped by SP_BLK_GAP.<br>A BLK_GAP_EVT interrupt will be generated when SP_BLK_GAP is set to 1 and this bit changes to 0. This bit is useful in the command with a busy data line. |
| 7:3 | Reserved | - | - | - |
| 2 | DATA_LIN_ACT | ROC | - | Data Line Active<br>In a read transfer, this status bit is used to check whether a read transfer is executing on the bus. Changing this bit from 1 to 0 will generate a BLK_GAP_EVT interrupt when SP_BLK_GAP is set to 1.<br>1: under the following conditions:<br>(1) After the end bit of a read command<br>(2) When CONT_REQ in the block gap control register is set to restart a transfer.<br>0: under the following conditions:<br>(1) When the end bit of the last data block is sent from the SD bus to the host controller.<br>(2) When SP_BLK_GAP is set to 1 and a read transfer is stopped at the block gap.<br>In a write transfer, this status bit is used to check whether a write transfer is executing on the bus. Changing this bit from 1 to 0 will generate a TRAN_CMPLT interrupt in the normal interrupt status register.<br>1: under the following conditions:<br>(1) After the end bit of a read command<br>(2) When CONT_REQ in the block gap control register is set to restart a transfer. |


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| | | | | 0: under the following conditions:<br>(1) When the card releases the busy signal of the last data block.<br>(2) When SP_BLK_GAP is set to 1 and the card releases the write busy at the block gap.<br>In the command with busy data line, this bit indicates whether a command with busy is executing on the bus. This bit will be set after the end bit of the command with busy and will be cleared when busy is de-asserted or busy is not detected after the end of a response. |
| 1 | CMD_INHIBIT_DAT | ROC | - | Command Inhibit (DAT)<br>1: Cannot issue new commands to use the data line<br>0: Issue new commands to use the data line |
| 0 | CMD_INHIBIT_CMD | ROC | - | Command Inhibit (CMD)<br>1: Cannot issue command<br>0: Issue command only with the command line |

**Table 8-10 - 11.2.9 SDH_PRESENT_STATE – Present State Register**

## 8.2.10 SDH_HST_CNTL_1 – Host Control 1 Register (address offset: 0x28)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | CD_SEL | RW | 1'b0 | Card Detect Signal Selection<br>1: The test level for the card detection<br>0: The card detect pin is selected |
| 6 | CD_TEST_LV | RW | 1'b0 | Card Detect Test Level<br>1: Card is inserted<br>0: Card cannot be found |
| 5 | EXT_DATA_WIDTH | RW | 1'b0 | Extended Data Transfer Width<br>1: 8-bit bus width<br>0: Bus width is selected by the data transfer width |
| 4:2 | Reserved | - | - | - |
| 1 | DATA_WIDTH | RW | 1'b0 | Data Width<br>1: 4-bit mode<br>0: 1-bit mode |
| 0 | Reserved | - | - | - |

**Table 8-11 - SDH_HST_CNTL_1 – Host Control 1 Register**

## 8.2.11 SDH_PWR_CNTL – Power Control Register (address offset: 0x29)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 15:12 | Reserved | - | - | - |
| 11:9 | SD_BUS_VOL | RW | 3'h0 | SD Bus Voltage Select<br>111: 3.3V (Typical)<br>110: 3.0V (Typical)<br>101: 1.8V (Typical)<br>Others: Reserved |


Copyright © Bridgetek Pte Ltd

| | | | | |
|---|---|---|---|---|
| 8:0 | Reserved | - | - | - |

**Table 8-12 - SDH_PWR_CNTL – Power Control Register**

### 8.2.12 SDH_BLK_GAP_CNTL – Block Gap Control Register (address offset: 0x2A)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 23:20 | Reserved | - | - | - |
| 19 | INT_BLK_GAP | RW | 1'b0 | 1: Check the interrupt at block gap enabled |
| 18 | READ_WAIT | RW | 1'b0 | 1: Enable Read Wait |
| 17 | CONT_REQ | RWAC | 1'b0 | 1: To restart a transaction (SP_BLK_GAP must also be 1; if 0 this will be aborted)<br>It is cleared automatically by the host controller when:<br>In a read transfer, DATA_LIN_ACT changes from 0 to 1 to start a read transfer.<br>In a write transfer, WR_TRAN_ACT changes from 0 to 1 to start a write transfer. |
| 16 | SP_BLK_GAP | RW | 1'b0 | 1: Stop at block gap request; the host controller will stop at the block gap by using READ_WAIT or stop IO_SD_CLK in a read transaction<br>0: the host controller will not write data to DATA_PORT |

**Table 8-13 - SDH_BLK_GAP_CNTL – Block Gap Control Register**

### 8.2.13 SDH_CLK_CNTL – Clock Control Register (address offset: 0x2C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 15:8 | LOW_BIT_SD_CLK_SEL | RW | 8'h00 | SD clock frequency value 7:0 for the 10-bit divided clock mode. These are used to select the frequency of the IO_SD_CLK pin. The base clock is ½ of chip system clock<br>N: chip system clock * (1 / 2N)<br>0: not supported |
| 7:6 | UPPER_BIT_SD_CLK_SEL | RW | 2'h0 | SD clock frequency bits 9:8 for 10-bit divided clock mode. |
| 5 | CLK_GEN_SEL | ROC | 1'b0 | This bit is always set to zero<br>0: 10-bit divided clock mode |
| 4:3 | Reserved | - | - | - |
| 2 | SD_CLK_EN | RW | 1'b0 | 1: IO_SD_CLK will be output |
| 1 | CLK_STABLE | ROC | 1'b0 | This bit is set to 1 when the internal clock is stable |
| 0 | INTER_CLK_EN | RW | 1'b0 | 1: Internal clock will start oscillating |

**Table 8-14 - SDH_CLK_CNTL – Clock Control Register**


Copyright © Bridgetek Pte Ltd

### 8.2.14 SDH_TIMEOUT_CNTL – Timeout Control Register (address offset: 0x2E)

This host driver should set the timeout value according to the capabilities register. The value of DATA_TIMER indicates the data line timeout times.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 23:20 | Reserved | - | - | - |
| 19:16 | DATA_TIMER | RW | 4'hE | 1111: Reserved<br>1110: Chip system clk x 2^27<br>1101: Chip system clk x 2^26<br>…….<br>0000: Chip system clk x 2^13 |

**Table 8-15 - SDH_TIMEOUT_CNTL – Timeout Control Register**

### 8.2.15 SDH_SW_RST – Software Reset Register (address offset: 0x2F)

A reset pulse will be generated when this bit is set to 1. This bit will be automatically cleared once the reset pulse is issued.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:27 | Reserved | - | - | - |
| 26 | SOFT_RST_DAT | RWAC | 1'b0 | 1: Software reset for data line |
| 25 | SOFT_RST_CMD | RWAC | 1'b0 | 1: Software reset for command line |
| 24 | SOFT_RST_ALL | RWAC | 1'b0 | 1: Software reset for all |

**Table 8-16 - SDH_SW_RST – Software Reset Register**

### 8.2.16 SDH_NRML_INT_STATUS – Normal Interrupt Status Register (address offset: 0x30)

The interrupt status can be latched by setting the Normal Interrupt Status Enable register corresponding bit to 1.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 15 | ERR_INT | ROC | 1'b0 | 1: Error Interrupt |
| 14:9 | Reserved | - | - | - |
| 8 | CARD_INT | ROC | 1'b0 | 1: Card Interrupt |
| 7 | CARD_REMOVE | RW1C | 1'b0 | 1: Card Remove |
| 6 | CARD_INSERT | RW1C | 1'b0 | 1: Card Inserted |
| 5 | BUF_RD_RDY | RW1C | 1'b0 | 1: Buffer Read Ready |
| 4 | BUF_WR_RDY | RW1C | 1'b0 | 1: Buffer Write Ready |
| 3 | Reserved | RW1C | 1'b0 | Write 0 to this bit |


Copyright © Bridgetek Pte Ltd

| | | | | |
|---|---|---|---|---|
| 2 | BLK_GAP_EVT | RW1C | 1'b0 | 1: Block Gap Event |
| 1 | TRAN_CMPLT | RW1C | 1'b0 | 1: Transfer Complete |
| 0 | CMD_CMPLT | RW1C | 1'b0 | 1: Command Complete |

**Table 8-17 - SDH_NRML_INT_STATUS – Normal Interrupt Status Register**

### 8.2.17 SDH_ERR_INT_STATUS – Error Interrupt Status Register (address offset: 0x32)

The interrupt status can be latched by setting the corresponding bit in the Error Interrupt Status Enable Register.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:27 | Reserved | - | - | - |
| 26 | Reserved | RW1C | 1'b0 | Write 0 to this bit |
| 25 | Reserved | RW1C | 1'b0 | Write 0 to this bit |
| 24 | AUTO_CMD12_ERR | RW1C | 1'b0 | Auto CMD12 error |
| 23 | CUR_LIM_ERR | RW1C | 1'b0 | Current limit error |
| 22 | DATA_END_BIT_ERR | RW1C | 1'b0 | Data End Bit error |
| 21 | DATA_CRC_ERR | RW1C | 1'b0 | Data CRC error |
| 20 | DATA_TIMEOUT_ERR | RW1C | 1'b0 | Data Timeout error |
| 19 | CMD_IDX_ERR | RW1C | 1'b0 | Command Index error |
| 18 | CMD_ERR_BIT_ERR | RW1C | 1'b0 | Command End Bit error |
| 17 | CMD_CRC_ERR | RW1C | 1'b0 | Command CRC error |
| 16 | CMD_TIMEOUT_ERR | RW1C | 1'b0 | Command Timeout error |

**Table 8-18 - SDH_ERR_INT_STATUS – Error Interrupt Status Register**

### 8.2.18 SDH_NRML_INT_ENABLE – Normal Interrupt Status Enable Register (address offset: 0x34)

If the corresponding bit of the interrupt source in the normal interrupt status enable register is set to 1, the interrupt becomes active, which is latched and available for the host driver in the normal interrupt status register.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 15 | Reserved | - | - | - |
| 14:9 | Reserved | - | - | - |


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 8 | CARD_INT_ST_EN | RW | 1'b0 | Card Interrupt status enable |
| 7 | CARD_REMOVE_ST_EN | RW | 1'b0 | Card Remove status enable |
| 6 | CARD_INSERT_ST_EN | RW | 1'b0 | Card Insert status enable |
| 5 | BUF_RD_RDY_ST_EN | RW | 1'b0 | Buffer Read Ready status enable |
| 4 | BUF_WR_RDY_ST_EN | RW | 1'b0 | Buffer Write Ready status enable |
| 3 | Reserved | RW | 1'b0 | Write 0 to this bit |
| 2 | BLK_GAP_EVT_ST_EN | RW | 1'b0 | Block Gap Event status enable |
| 1 | TRAN_CMPLT_ST_EN | RW | 1'b0 | Transfer Complete status enable |
| 0 | CMD_CMPLT_ST_EN | RW | 1'b0 | Command Complete status enable |

**Table 8-19 - SDH_NRML_INT_ENABLE – Normal Interrupt Status Enable Register**

### 8.2.19 SDH_ERR_INT_ENABLE – Error Interrupt Status Enable Register (address offset: 0x36)

If the corresponding bit of the interrupt source in the Error Interrupt Status Enable Register is set to 1 and if the interrupt becomes active, the active state will be latched and will be available for the host driver in this register.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:26 | Reserved | - | - | - |
| 25 | Reserved | RW | 1'b0 | Write 0 to this bit |
| 24 | AUTO_CMD12_ERR_ST_EN | RW | 1'b0 | Auto CMD12 error status enable |
| 23 | CUR_LIM_ERR_ST_EN | RW | 1'b0 | Current limit error status enable |
| 22 | DATA_END_BIT_ERR_ST_EN | RW | 1'b0 | Data End Bit error status enable |
| 21 | DATA_CRC_ERR_ST_EN | RW | 1'b0 | Data CRC error status enable |
| 20 | DATA_TIMEOUT_ERR_ST_EN | RW | 1'b0 | Data Timeout error status enable |
| 19 | CMD_IDX_ERR_ST_EN | RW | 1'b0 | Command Index error status enable |
| 18 | CMD_END_BIT_ERR_ST_EN | RW | 1'b0 | Command End Bit error status enable |
| 17 | CMD_CRC_ERR_ST_EN | RW | 1'b0 | Command CRC error status enable |
| 16 | CMD_TIMEOUT_ERR_ST_EN | RW | 1'b0 | Command Timeout error status enable |

**Table 8-20 - SDH_ERR_INT_ENABLE – Error Interrupt Status Enable Register**


Copyright © Bridgetek Pte Ltd

### 8.2.20 SDH_NRML_INT_SGNL_ENABLE – Normal Interrupt Signal Enable Register (address offset: 0x38)

This register is used to select which interrupts are notified to the host system.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 15 | Reserved | - | - | - |
| 14:9 | Reserved | - | - | - |
| 8 | CARD_INT_SIG_EN | RW | 1'b0 | Card Interrupt signal enable |
| 7 | CARD_REMOVE_SIG_EN | RW | 1'b0 | Card Remove signal enable |
| 6 | CARD_INSERT_SIG_EN | RW | 1'b0 | Card Insert signal enable |
| 5 | BUF_RD_RDY_SIG_EN | RW | 1'b0 | Buffer Read Ready signal enable |
| 4 | BUF_WR_RDY_SIG_EN | RW | 1'b0 | Buffer Write Ready signal enable |
| 3 | Reserved | RW | 1'b0 | Write 0 to this bit |
| 2 | BLK_GAP_EVT_SIG_EN | RW | 1'b0 | Block Gap Event signal enable |
| 1 | TRAN_CMPLT_SIG_EN | RW | 1'b0 | Transfer Complete signal enable |
| 0 | CMD_CMPLT_SIG_EN | RW | 1'b0 | Command Complete signal enable |

**Table 8-21 - SDH_NRML_INT_SGNL_ENABLE – Normal Interrupt Signal Enable Register**

### 8.2.21 SDH_ERR_INT_SGNL_ENABLE – Error Interrupt Signal Enable Register (address offset: 0x3A)

This register is used to select which interrupts are notified to the host system.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:26 | Reserved | - | - | - |
| 25 | Reserved | RW | 1'b0 | Write 0 to this bit |
| 24 | AUTO_CMD12_ERR_SIG_EN | RW | 1'b0 | Auto CMD12 error signal enable |
| 23 | CUR_LIM_ERR_SIG_EN | RW | 1'b0 | Current limit error signal enable |
| 22 | DATA_END_BIT_ERR_SIG_EN | RW | 1'b0 | Data End Bit error signal enable |
| 21 | DATA_CRC_ERR_SIG_EN | RW | 1'b0 | Data CRC error signal enable |
| 20 | DATA_TIMEOUT_ERR_SIG_EN | RW | 1'b0 | Data Timeout error signal enable |
| 19 | CMD_IDX_ERR_SIG_EN | RW | 1'b0 | Command Index error signal enable |
| 18 | CMD_END_BIT_ERR_SIG_EN | RW | 1'b0 | Command End Bit error signal enable |


Copyright © Bridgetek Pte Ltd

| | | | | |
|---|---|---|---|---|
| 17 | CMD_CRC _ERR_SIG_EN | RW | 1'b0 | Command CRC error signal enable |
| 16 | CMD_TIMEOUT_ERR_SIG_EN | RW | 1'b0 | Command Timeout error signal enable |

**Table 8-22 - SDH_ERR_INT_SGNL_ENABLE – Error Interrupt Signal Enable Register**

### 8.2.22 SDH_AUTO_CMD12_ERR_STATUS – Auto CMD12 Error Status Register (address offset: 0x3C)

When the auto_cmd12_en register is set to 1 and the auto cmd12 error status register is set, the host driver will check this register to identify what kind of error happens during executing AUTO CMD12. This register is valid only when the auto_cmd12_err is set to 1.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 15:8 | Reserved | - | - | - |
| 7 | CMD_NO_EX_BY_CMD12 | ROC | 1'b0 | Command not executed by Auto CMD12 error |
| 6:5 | Reserved | - | - | - |
| 4 | AUTO_CMD_IDX_ERR | ROC | 1'b0 | Auto CMD index error |
| 3 | AUTO_CMD_END_BIT_ERR | ROC | 1'b0 | Auto CMD end bit error |
| 2 | AUTO_CMD_CRC_ERR | ROC | 1'b0 | Auto CMD CRC error |
| 1 | AUTO_CMD_TIMEOUT_ERR | ROC | 1'b0 | Auto CMD timeout error |
| 0 | AUTO_CMD12_NO_EX | ROC | 1'b0 | Auto CMD12 not executed |

**Table 8-23 - SDH_AUTO_CMD12_ERR_STATUS – Auto CMD12 Error Status Register**

### 8.2.23 SDH_HOST_CNTL_2 – Host Control 2 Register (address offset: 0x3E)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31 | PRESET_VAL_EN | RW | 1'b0 | 0: SDCLK and driver strength are controlled by the host driver<br>1: automatic selection by the pre-set value |
| 30 | ASYN_INT_EN | R/W | 1'b0 | Asynchronous Interrupt Enable |
| 29:24 | Reserved | - | - | - |
| 23 | SAMPLE_CLK_SEL | RW | 1'b0 | Sampling clock select<br>0: Fixed clock is used to sample data |
| 22:16 | Reserved | - | - | - |

**Table 8-24 - SDH_HOST_CNTL_2 – Host Control 2 Register**

### 8.2.24 SDH_CAP_1 – Capabilities Register 1 (address offset: 0x40)

The host controller may implement these values during initialization.


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:30 | SLOT_TYPE | RO | 2'h0 | Removable Card Slot |
| 29 | ASYNC_INT_SUPPORT | RO | 1'b0 | 0: not supported |
| 28 | BUT_64_SYPPORT | RO | 1'b0 | 0: not supported |
| 27 | Reserved | - | - | - |
| 26:25 | Reserved | - | - | - |
| 24 | VOLTAGE_3_3_SUPPORT | RO | 1'b1 | Voltage supports 3.3V |
| 23 | SUSPEND_RESUME_SUPPORT | RO | 1'b0 | 0: Suspend / Resume not supported |
| 22 | Reserved | RO | 1'b1 | - |
| 21 | HI_SPEED_SUPPORT | RO | 1'b1 | 1: High speed supported |
| 20 | Reserved | RO | 1'b0 | - |
| 19 | Reserved | RO | 1'b1 | - |
| 18 | 8BIT_SUPPORT | RO | 1'b0 | 0: 8-bit not supported |
| 17:16 | MAX_BLK_LEN | R0 | 2'h2 | 2: 2048 bytes |
| 15:8 | BASE_CLK_FOR_SD_CLK | RO | 8'h00 | 0: Get information via another method |
| 7 | TIMEOUT_CLK_UNIT | RO | 1'b1 | 1: MHz |
| 6 | Reserved | - | - | - |
| 5:0 | TIMEOUT_CLK_FREQ | RO | 6'h00 | 0: Get information via another method |

**Table 8-25 - SDH_CAP_1 – Capabilities Register 1**

### 8.2.25 SDH_CAP_2 – Capabilities Register 2 (address offset: 0x44)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:24 | Reserved | - | - | - |
| 23:16 | CLK_MULTI | RO | 8'h00 | not supported |
| 15:12 | Reserved | - | - | - |
| 11:8 | Reserved | RO | 4'hF | - |
| 7 | Reserved | - | - | - |
| 6 | DRIVER_D_SUPPORT | RO | 1'b1 | not supported |


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 5 | DRIVER_C_SUPPORT | RO | 1'b1 | not supported |
| 4 | DRIVER_A_SUPPORT | RO | 1'b1 | not supported |
| 3 | Reserved | - | - | - |
| 2 | DDR50_SUPPORT | RO | 1'b1 | not supported |
| 1 | SDR104_SUPPORT | RO | 1'b0 | not supported |
| 0 | SDR50_SUPPORT | RO | 1'b1 | not supported |

**Table 8-26 - SDH_CAP_2 – Capabilities Register 2**

### 8.2.26 SDH_RSRV_1 – Reserved 1 Register (address offset: 0x48)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:0 | Reserved | - | - | - |

**Table 8-27 - SDH_RSRV_1 – Reserved 1 Register**

### 8.2.27 SDH_RSRV_2 – Reserved 2 Register (address offset: 0x4C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:0 | Reserved | - | - | - |

**Table 8-28 - SDH_RSRV_2 – Reserved 2 Register**

### 8.2.28 SDH_FORCE_EVT_CMD_ERR_STATUS – Force Event Register for Auto CMD Error Status (address offset: 0x50)

The Force Event register is not a physical register. It is an address to which the Auto CMD error status register can be written. The force event register is only for debugging.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 15:8 | Reserved | - | - | - |
| 7 | R_CMD_NO_EX_BY_CMD 12 | WO | 1'b0 | Force event for the Command Not Executed by Auto CMD12 Error |
| 6:5 | Reserved | - | - | - |
| 4 | R_CMD_IDX_ERR | WO | 1'b0 | Force event for the Auto CMD Index Error |
| 3 | R_CMD_END_BIT_ERR | WO | 1'b0 | Force event for the Auto CMD End Bit Error |
| 2 | R_CMD_CRC_ERR | WO | 1'b0 | Force event for the Auto CMD CRC Error |
| 1 | R_CMD_TIMEOUT_ERR | WO | 1'b0 | Force event for the Auto CMD Timeout Error |


Copyright © Bridgetek Pte Ltd

| | | | | |
|---|---|---|---|---|
| 0 | R_CMD12_NO_EX | WO | 1'b0 | Force event for the Auto CMD12 Not Executed |

**Table 8-29 - SDH_FORCE_EVT_CMD_ERR_STATUS – Force Event Register for Auto CMD Error Status**

### 8.2.29 SDH_FORCE_EVT_ERR_INT_STATUS – Force Event for Error Interrupt Status Register (address offset: 0x52)

The Force Event register is not a physical register. It is an address to which the error interrupt status register can be written. This Force Event register is for debugging only. The effect of writing to this address will be reflected in the error interrupt status register if the corresponding bit of the error interrupt status enable register is set.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:25 | Reserved | - | - | Write 0 to these bits |
| 24 | R_AUTP_CMD_ERR | WO | 1'b0 | Force Event for the Auto CMD Error |
| 23 | R_CUR_LIMIT_ERR | WO | 1'b0 | Force Event for the Current Limit Error |
| 22 | R_DATA_END_BIT_ERR | WO | 1'b0 | Force Event for the Data End Bit Error |
| 21 | R_DATA_CRC_ERR | WO | 1'b0 | Force Event for the Data CRC Error |
| 20 | R_DATA_TIMEOUT_ERR | WO | 1'b0 | Force Event for the Data Timeout Error |
| 19 | R_CMD_IDX_ERR | WO | 1'b0 | Force Event for the Command Index Error |
| 18 | R_CMD_END_BIT_ERR | WO | 1'b0 | Force Event for the Command End Bit Error |
| 17 | R_CMD_CRC_ERR | WO | 1'b0 | Force Event for the Command CRC Error |
| 16 | R_CMD_TIMEOUT_ERR | WO | 1'b0 | Force Event for the Command Timeout Error |

**Table 8-30 - SDH_FORCE_EVT_ERR_INT_STATUS – Force Event for Error Interrupt Status Register**

### 8.2.30 SDH_RSRV_3 – Reserved 3 Register (address offset: 0x54)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:0 | Reserved | - | - | - |

**Table 8-31 - SDH_RSRV_3 – Reserved 3 Register**

### 8.2.31 SDH_RSRV_4 – Reserved 4 Register (address offset: 0x58)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|


Copyright © Bridgetek Pte Ltd

| | | | | |
|---|---|---|---|---|
| 31:0 | Reserved | - | - | - |

**Table 8-32 - SDH_RSRV_4 – Reserved 4 Register**

### 8.2.32 SDH_PRST_INIT - Preset value for initialization (address offset: 0x60)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 15:14 | DRIVER_STR_SEL | RO | | Driver Strength Select Value<br>Driver Strength is supported by the 1.8-V signalling bus speed modes. This field is meaningless for the 3.3-V signalling<br>11: Driver type D is selected<br>10: Driver type C is selected<br>01: Driver type A is selected<br>00: Driver type B is selected |
| 13:11 | Reserved | - | - | - |
| 10 | CLK_GEN_SEL | RO | 1'b0 | Clock Generator Select Value<br>The version does not support the programmable clock generator and is fixed to 0 |
| 9:0 | SDCLK_FREQ_SEL | RO | | SDCLK Frequency Select Value<br>The 10-bit pre-set value for setting the SDCLK Frequency Select in the Clock Control. The register is described by a host system. |

**Table 8-33 - SDH_PRST_INIT – Preset value for initialization**

### 8.2.33 SDH_PRST_DFLT_SPD – Preset value for default speed (address offset: 0x62)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 15:11 | Reserved | - | - | - |
| 10 | CLK_GEN_SEL | RO | 1'b0 | Clock Generator Select Value<br>The version does not support the programmable clock generator and is fixed to 0 |
| 9:0 | SDCLK_FREQ_SEL | RO | | SDCLK Frequency Select Value<br>The 10-bit pre-set value for setting the SDCLK Frequency Select in the Clock Control. The register is described by a host system. |

**Table 8-34 - SDH_PRST_DFLT_SPD – Preset value for default speed**

### 8.2.34 SDH_PRST_HIGH_SPD – Preset value for the high speed (address offset: 0x64)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 15:11 | Reserved | - | - | - |


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 10 | CLK_GEN_SEL | RO | | Clock Generator Select Value<br>The version does not support the programmable clock generator and is fixed to 0 |
| 9:0 | SDCLK_FREQ_SEL | RO | | SDCLK Frequency Select Value<br>The 10-bit pre-set value for setting the SDCLK Frequency Select in the Clock Control. The register is described by a host system. |

**Table 8-35 - SDH_PRST_HIGH_SPD – Preset value for the high speed**

### 8.2.35 SDH_PRST_SDR12 – Preset value for SDR12 (address offset: 0x66)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 15:14 | DRIVER_STR_SEL | RO | | Driver Strength Select Value<br>Driver Strength is supported by the 1.8-V signalling bus speed modes. This field is meaningless for the 3.3-V signalling<br>11: Driver type D is selected<br>10: Driver type C is selected<br>01: Driver type A is selected<br>00: Driver type B is selected |
| 13:11 | Reserved | - | - | - |
| 10 | CLK_GEN_SEL | RO | 1'b0 | Clock Generator Select Value<br>The version does not support the programmable clock generator and is fixed to 0 |
| 9:0 | SDCLK_FREQ_SEL | RO | | SDCLK Frequency Select Value<br>The 10-bit pre-set value for setting the SDCLK Frequency Select in the Clock Control. The register is described by a host system. |

**Table 8-36 - SDH_PRST_SDR12 – Preset value for SDR12**

### 8.2.36 SDH_PRST_SDR25 – Preset value for SDR25 (address offset: 0x68)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 15:14 | DRIVER_STR_SEL | RO | | Driver Strength Select Value<br>Driver Strength is supported by the 1.8-V signalling bus speed modes. This field is meaningless for the 3.3-V signalling<br>11: Driver type D is selected<br>10: Driver type C is selected<br>01: Driver type A is selected<br>00: Driver type B is selected |
| 13:11 | Reserved | - | - | - |
| 10 | CLK_GEN_SEL | RO | 1'b0 | Clock Generator Select Value<br>The version does not support the programmable clock generator and is fixed to 0 |
| 9:0 | SDCLK_FREQ_SEL | RO | | SDCLK Frequency Select Value<br>The 10-bit pre-set value for setting the SDCLK Frequency Select in the Clock Control. The register is described by a host system. |


Copyright © Bridgetek Pte Ltd

**Table 8-37 - SDH_PRST_SDR25 – Preset value for SDR25**

### 8.2.37 SDH_PRST_SDR50 – Preset value for SDR50 (address offset: 0x6A)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 15:14 | DRIVER_STR_SEL | RO | | Driver Strength Select Value<br>Driver Strength is supported by the 1.8-V signalling bus speed modes. This field is meaningless for the 3.3-V signalling<br>11: Driver type D is selected<br>10: Driver type C is selected<br>01: Driver type A is selected<br>00: Driver type B is selected |
| 13:11 | Reserved | - | - | - |
| 10 | CLK_GEN_SEL | RO | 1'b0 | Clock Generator Select Value<br>The version does not support the programmable clock generator and is fixed to 0 |
| 9:0 | SDCLK_FREQ_SEL | RO | | SDCLK Frequency Select Value<br>The 10-bit pre-set value for setting the SDCLK Frequency Select in the Clock Control. The register is described by a host system. |

**Table 8-38 - SDH_PRST_SDR50 – Preset value for SDR50**

### 8.2.38 SDH_PRST_SDR104 – Preset value for SDR104 (address offset: 0x6C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 15:14 | DRIVER_STR_SEL | RO | | Driver Strength Select Value<br>Driver Strength is supported by the 1.8-V signalling bus speed modes. This field is meaningless for the 3.3-V signalling<br>11: Driver type D is selected<br>10: Driver type C is selected<br>01: Driver type A is selected<br>00: Driver type B is selected |
| 13:11 | Reserved | - | - | - |
| 10 | CLK_GEN_SEL | RO | 1'b0 | Clock Generator Select Value<br>The version does not support the programmable clock generator and is fixed to 0 |
| 9:0 | SDCLK_FREQ_SEL | RO | | SDCLK Frequency Select Value<br>The 10-bit pre-set value for setting the SDCLK Frequency Select in the Clock Control. The register is described by a host system. |

**Table 8-39 - SDH_PRST_SDR104 – Preset value for SDR104**

### 8.2.39 SDH_PRST_DDR50 – Preset value for DDR50 (address offset: 0x6E)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 15:14 | DRIVER_STR_SEL | RO | | Driver Strength Select Value<br>Driver Strength is supported by the 1.8- |


Copyright © Bridgetek Pte Ltd

| | | | | |
|---|---|---|---|---|
| | | | | V signalling bus speed modes. This field is meaningless for the 3.3-V signalling<br>11: Driver type D is selected<br>10: Driver type C is selected<br>01: Driver type A is selected<br>00: Driver type B is selected |
| 13:11 | Reserved | - | - | - |
| 10 | CLK_GEN_SEL | RO | 1'b0 | Clock Generator Select Value<br>The version does not support the programmable clock generator and is fixed to 0 |
| 9:0 | SDCLK_FREQ_SEL | RO | | SDCLK Frequency Select Value<br>The 10-bit pre-set value for setting the SDCLK Frequency Select in the Clock Control. The register is described by a host system. |

**Table 8-40 - SDH_PRST_DDR50 – Preset value for DDR50**

### 8.2.40 SDH_RSRV_5 – Reserved 5 Register (address offset: 0xFC)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 15:0 | Reserved | - | - | - |

**Table 8-41 - SDH_RSRV_5 – Reserved 5 Register**

### 8.2.41 SDH_HC_VER – Host Controller Version Register (address offset: 0xFE)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:24 | VNDR_VER_NUM | RO | 8'h00 | Vendor Version Number |
| 23:16 | SPEC_VER_NUM | RO | 8'h02 | Specification Version Number |

**Table 8-42 - SDH_HC_VER – Host Controller Version Register**

### 8.2.42 SDH_VNDR_0 – Vendor-defined 0 Register (address offset: 0x100)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:28 | Reserved | - | - | - |
| 27:24 | N_CRC | RW | 4'h0 | Write CRC Status Wait Cycle<br>The host controller is used to set 5 SCLK clock cycles for the specifications and round-chip effect. Users can add the wait cycle for other factors. |
| 23:17 | Reserved | - | - | - |
| 16 | INT_EDGE_SEL | RW | 1'b0 | 1: The CMD and DAT line output at the rising edge of SCLK<br>0: The CMD and DAT line output at the falling edge of SCLK |
| 15:14 | Reserved | - | - | - |


Copyright © Bridgetek Pte Ltd

| | | | | |
|---|---|---|---|---|
| 13:8 | P_LAT_OFF | RW | 6'h00 | Pulse latch offset<br>When the host controller uses the pulse latch to sample the read data and response, users need to set the latch offset to correctly sample the value. The values set should be smaller than the SDCLK Frequency Select<br>0x3F: Latch value at the 63rd ½ chip frequency clock rising edge after the SCLK edge<br>….<br>0x01: Latch value at the 1st<br>0x00: Latch value at SCLK edge |
| 7:1 | Reserved | - | - | - |
| 0 | P_LAT_EN | RW | 1'b1 | 1: Use the pulse latching function for the read data and response. Should always be set to 1. |

**Table 8-43 - SDH_VNDR_0 – Vendor-defined 0 Register**

### 8.2.43 SDH_VNDR_1 – Vendor-defined 1 Register (address offset: 0x104)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:25 | Reserved | - | - | - |
| 24 | CMD_CONFLICT_EN | RW | 1'b0 | 1: Enable host controller to check the CMD line conflict error |
| 23:19 | Reserved | - | - | - |
| 18:16 | N_SB | RW | 3'h0 | N_SB Timing: Users can add the busy wait cycle for other factors. The host controller is set to 5 SCLK clock cycles |
| 15:12 | Reserved | - | - | - |
| 11:8 | N_CR | RW | 3'h0 | N_CR Timing: Users can add the response wait cycle for other factors. The host controller is set to 64 SCLK clock cycles |
| 7:3 | Reserved | - | - | - |
| 2 | MMC_BOOT_ACK_EN | RW | 1'b0 | MMC Booting Mode Acknowledge Enable |
| 1:0 | MMC_BOOT | RW | 2'h0 | MMC Booting Mode Selection<br>11: MMC Bus Test mode<br>10: MMC Alternative Boot Mode<br>01: MMC Boot mode<br>00: Normal mode |

**Table 8-44 - SDH_VNDR_1 – Vendor-defined 1 Register**

### 8.2.44 SDH_VNDR_2 – Vendor-defined 2 Register (address offset: 0x108)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:1 | Reserved | - | - | - |
| 0 | CLK_CTRL_SW_RST | RWAC | 1'b0 | 1: To reset the clock control of the host controller |


Copyright © Bridgetek Pte Ltd

**Table 8-45 - SDH_VNDR_2 – Vendor-defined 2 Register**

### 8.2.45 SDH_VNDR_3 – Vendor-defined 3 Register (address offset: 0x10C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:29 | Reserved | - | - | - |
| 28:24 | Reserved | - | 5'h1F | - |
| 23:0 | Reserved | - | - | - |

**Table 8-46 - SDH_VNDR_3 – Vendor-defined 3 Register**

### 8.2.46 SDH_VNDR_4 – Vendor-defined 4 Register (address offset: 0x110)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:0 | Reserved | RO | - | - |

**Table 8-47 - SDH_VNDR_4 – Vendor-defined 4 Register**

### 8.2.47 SDH_VNDR_5 – Vendor-defined 5 Register (address offset: 0x114)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:4 | Reserved | - | - | - |
| 3:0 | DB_TIMEOUT | RW | 4'h0 | Card Insertion De-bounce Cycle<br>0: $2^9$ chip system clock cycles<br>1: $2^{10}$ chip system clock cycles<br>….<br>15: $2^{24}$ chip system clock cycles |

**Table 8-48 - SDH_VNDR_5 – Vendor-defined 5 Register**

### 8.2.48 SDH_VNDR_6 – Vendor-defined 6 Register (address offset: 0x118)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:1 | Reserved | - | - | - |
| 0 | HBURST_INCR | RW | 1'b1 | 0: AHB master uses SINGLE and INCR4 as the AHB burst type |

**Table 8-49 - SDH_VNDR_6 – Vendor-defined 6 Register**

### 8.2.49 SDH_VNDR_7 – Vendor-defined 7 Register (address offset: 0x11C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:1 | Reserved | - | - | - |
| 0 | AHB_RESP_ERR_STS | RW1C | 1'b0 | This bit is set when the AHB master receives an error type response |

**Table 8-50 - SDH_VNDR_7 – Vendor-defined 7 Register**


Copyright © Bridgetek Pte Ltd

### 8.2.50 SDH_VNDR_8 – Vendor-defined 8 Register (address offset: 0x120)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:1 | Reserved | - | - | - |
| 0 | AHB_RESP_ERR_STS_EN | RW | 1'b0 | 1: Enable the AHB master response error status |

**Table 8-51 - SDH_VNDR_8 – Vendor-defined 8 Register**

### 8.2.51 SDH_VNDR_9 – Vendor-defined 9 Register (address offset: 0x124)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:1 | Reserved | - | - | - |
| 0 | AHB_RESP_ERR_SIG_EN | RW | 1'b0 | 1: Enable the interrupt generation when the AHB master response status is set |

**Table 8-52 - SDH_VNDR_9 – Vendor-defined 9 Register**

### 8.2.52 SDH_RSRV_6 – Reserved 6 Register (address offset: 0x128)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:0 | Reserved | - | - | - |

**Table 8-53 - SDH_RSRV_6 – Reserved 6 Register**

### 8.2.53 SDH_HW_ATTR – Hardware Attributes Register (address offset: 0x178)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:9 | Reserved | - | - | - |
| 8:0 | HW_CONFIG | RO | 9'h150 | 8: Async<br>7: 4-bit SD data bus<br>6: CPRM present<br>5: DLL absent<br>4:0 DATA FIFO is 4k SRAM |

**Table 8-54 - SDH_HW_ATTR – Hardware Attributes Register**

### 8.2.54 SDH_CPR_MOD_CNTL – Cipher Mode Control Register (address offset: 0x180)

This register is the configurable register for the CPRM function. When the CPRM function is used, this register will be used to select the mode of the cipher function to encrypt or decrypt.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:11 | Reserved | - | - | - |
| 10 | SWAP_HL | RW | 1'b0 | 1: Swap the high/low word of the encrypted data to TX FIFO<br>The high-word and low-word of the encrypted data will be swapped before being written to TX FIFO. The high-word |


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| | | | | and low-word of the encrypted data will be swapped before decryption. |
| 9 | CH_ENDIAN | RW | 1'b0 | Change Endianness<br>In this mode, the endianness of the encrypted data will be changed before being written to the TX FIFO.<br>In this mode, the endianness of data from the RX FIFO will be changed before decryption. |
| 8 | SEC_ACCESS_EN | RW | 1'b0 | Secret Constant Table Access Enable<br>This bit must be enabled before writing or reading the secret constant table. Once this bit is enabled, the firmware will always access from the very beginning of the secret constant table. |
| 7 | AUTO_C2_DCBC_EN | RW | 1'b0 | Auto C2 Decryption with C-CBC Mode Enable<br>In this mode, data will be automatically decrypted and sent to the buffer. The data lengths should be multiples of 8 bytes. |
| 6 | AUTO_C2_ECBC_EN | RW | 1'b0 | Auto C2 Encryption with C-CBC Mode Enable<br>In this mode, data written to the buffer will be automatically encrypted and sent to the TX FIFO. The data lengths should be multiples of 8 bytes. |
| 5 | RNGC2_G_EN | RW | 1'b0 | C2 Random Number Generator Enable |
| 4 | C2_DCBC_EN | RW | 1'b0 | C2 Decryption with C-CBC Mode Enable |
| 3 | C2_D_EN | RW | 1'b0 | C2 Decryption with EBC Mode Enable |
| 2 | C2_ECBC_EN | RW | 1'b0 | C2 Encryption with C-CBC Mode Enable |
| 1 | C2_E_EN | RW | 1'b0 | C2 Encryption with EBC Mode Enable |
| 0 | C2_G_EN | RW | 1'b0 | C2 One Way Function Enable |

**Table 8-55 - SDH_CPR_MOD_CNTL – Cipher Mode Control Register**

### 8.2.55 SDH_CPR_MOD_STATUS – Cipher Mode Status Register (address offset: 0x184)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:1 | Reserved | - | - | - |
| 0 | CP_RDY | RW1C | 1'b0 | Cipher is ready<br>When this bit is set to 1, reading 0x19C and 0x1A0 will retrieve cipher or plain text |

**Table 8-56 - SDH_CPR_MOD_STATUS – Cipher Mode Status Register**


Copyright © Bridgetek Pte Ltd

### 8.2.56 SDH_CPR_MOD_STATUS_EN – Cipher Mode Status Enable Register (address offset: 0x188)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:1 | Reserved | - | - | - |
| 0 | RDY_SIG_EN | RW | 1'b0 | Cipher Ready Signal Enable |

**Table 8-57 - SDH_CPR_MOD_STATUS_EN – Cipher Mode Status Enable Register**

### 8.2.57 SDH_CPR_MOD_SIG_EN – Cipher Mode Signal Enable Register (address offset: 0x18A)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:1 | Reserved | - | - | - |
| 0 | RDY_SIG_EN | RW | 1'b0 | Cipher Ready Signal Enable |

**Table 8-58 - SDH_CPR_MOD_SIG_EN – Cipher Mode Signal Enable Register**

### 8.2.58 SDH_IN_DATA_LSB – Input Data LSB Register (address offset: 0x18C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:0 | DATA | RW | 32'h0000 0000 | Input port for the input data bits 31:0 |

**Table 8-59 - SDH_IN_DATA_LSB – Input Data LSB Register**

### 8.2.59 SDH_IN_DATA_MSB - Input Data MSB Register (address offset: 0x190)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:0 | DATA | RW | 32'h0000 0000 | Input port for the input data bits 63:32 |

**Table 8-60 - SDH_IN_DATA_MSB – Input Data MSB Register**

### 8.2.60 SDH_IN_KEY_LSB – Input Key LSB Register (address offset: 0x194)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:0 | KEY | RW | 32'h0000 0000 | Input port for the input key bits 31:0 |

**Table 8-61 - SDH_IN_KEY_LSB – Input Key LSB Register**

### 8.2.61 SDH_IN_KEY_MSB – Input Key MSB Register (address offset: 0x198)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:0 | KEY | RW | 32'h0000 0000 | Input port for the input key bits 63:32 |


Copyright © Bridgetek Pte Ltd

**Table 8-62 - SDH_IN_KEY_MSB – Input Key MSB Register**

### 8.2.62 SDH_OUT_DATA_LSB – Output Data LSB Register (address offset: 0x19C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:0 | DATA | RW | 32'h0000 0000 | Output port for the output data bits 31:0 |

**Table 8-63 - SDH_OUT_DATA_LSB – Output Data LSB Register**

### 8.2.63 SDH_OUT_DATA_MSB – Output Data MSB Register (address offset: 0x1A0)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:0 | DATA | RW | 32'h00000 000 | Output port for the output data bits 63:32 |

**Table 8-64 - SDH_OUT_DATA_MSB – Output Data MSB Register**

### 8.2.64 SDH_SCRT_CONS_DATA – Secret Constant Table Data Port (address offset: 0x1A4)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 31:8 | Reserved | - | - | - |
| 7:0 | DATA_PORT | RW | 8'h00 | Secret constant table data port.<br>256 bytes are needed; this port should be written 256 times to initialize the secret constant table |

**Table 8-65 - SDH_SCRT_CONS_DATA – Secret Constant Table Data Port**


Copyright © Bridgetek Pte Ltd

# 9 UART

The device supports 2 UARTs with the following features:

- Software compatible with 16450 ("450"), 16550 (“550”), 16550A (“Extended 550”), 16650 (“650”), 16750 (“750”) and 16C950 (“950”) UARTs
- Separate configurable BAUD clock line
- Configuration capability
- Two modes of operation: UART mode and FIFO mode
- Majority voting logic
- 16 / 128 bytes FIFO for TX and RX in FIFO mode to reduce the interrupt frequency
- adds or deletes standard asynchronous communication bits (start, stop and parity) to or from the serial data
- Double buffering for both TX and RX in UART mode
- Independently controlled transmit, receive, line status and data set interrupts
- Programmable baud generator
- MODEM control functions (CTS, RTS, DSR, DTR, RI and DCD)
- Programmable automatic out-of-band Flow Control logic through Auto-RTS and Auto-CTS
- Programmable automatic Flow Control logic using DTR and DSR
- Programmable automatic in-band Flow Control logic using XON / XOFF characters
- Programmable special characters detection
- Trigger levels for TX and RX FIFO interrupts, automatic in-band and out-of-band flow control
- RS-485 buffer enable signals
- TX and RX disable capability
- Fully programmable serial interface characteristics:
    - 5-, 6-, 7-, 8- or 9-bit characters
    - Even, odd, or no-parity bit generation and detection
    - 1-, 1.5, or 2-stop bit generation
    - Baud generation
    - Detection of bad data in receive FIFO
- Clock prescaler from 1 to 31875
- Enhanced isochronous clock option
- Complete status reporting capabilities
- False start bit detection


Copyright © Bridgetek Pte Ltd

- Line-break generation and detection. Internal diagnostic capabilities:
    - Loop-back controls for communications link fault isolation
    - Break, parity, overrun, framing error simulation
- Full prioritised interrupt system controls
- Software reset

## 9.1 Register Summary

The registers offset from the base addresses are:

- 0x10520 for UART 0
- 0x10530 for UART 1
- 0x10540 for UART 2
- 0x10550 for UART 3

All registers can only be accessed via Byte (8-bit) mode.

| Offset | Register | Default Value | Reference |
|---|---|---|---|
| | | | |
| | **STANDARD 550 COMPATIBLE REGISTERS** | | |
| **0x00** | UART_RBR - Receiver Buffer Register | 0x00 | Section 13.3.1 |
| **0x00** | UART_THR - Transmitter Holding Register | 0x00 | Section 13.3.2 |
| **0x00** | UART_DIV_LSB - Divisor LSB Register | 0x00 | Section 13.3.3 |
| **0x01** | UART_DIV_MSB - Divisor MSB Register | 0x00 | Section 13.3.4 |
| **0x01** | UART_INT_ENABLE - Interrupt Enable Register | 0x00 | Section 13.3.5 |
| **0x02** | UART_INT_STATUS - Interrupt Status Register | 0x01 | Section 13.3.6 |
| **0x02** | UART_FCR FIFO - Control Register | 0x00 | Section 13.3.7 |
| **0x03** | UART_LCR Line - Control Register | 0x00 | Section 13.3.8 |
| **0x04** | UART_MCR - Modem Control Register | 0x80 | Section 13.3.9 |
| **0x05** | UART_LSR - Line Status Register | 0x60 | Section 13.3.10 |
| **0x06** | UART_MSR - Modem Status Register | 0xX0 | Section 13.3.11 |
| **0x07** | UART_SPR - SPR Register | 0x00 | Section 13.3.12 |
| | | | |
| | **650 COMPATIBLE REGISTERS** | | |
| | | | |
| **0x02** | UART_EFR - Enhanced Feature Register | 0x00 | Section 13.4.1 |
| **0x04** | UART_XON1 - XON1 Register | 0x00 | Section 13.4.2 |
| **0x05** | UART_XON2 - XON2 Register | 0x00 | Section 13.4.3 |
| **0x06** | UART_XOFF1 - XOFF1 Register | 0x00 | Section 13.4.4 |
| **0x07** | UART_XOFF2 - XOFF2 Register | 0x00 | Section 13.4.5 |


Copyright © Bridgetek Pte Ltd

| | | | |
|---|---|---|---|
| | **950 COMPATIBLE REGISTERS** | | |
| **0x01** | UART_ASR - Additional Status Register | 0x00 | Section 13.5.1 |
| **0x03** | UART_RFL - Receiver FIFO Level Register | 0x00 | Section 13.5.2 |
| **0x04** | UART_TFL - Transmitter FIFO Level Register | 0x00 | Section 13.5.3 |
| **0x05** | UART_ICR - ICR Register | 0x00 | Section 13.5.4 |
| | **INDEXED CONTROL REGISTERS** | | |
| **0x00** | UART_ACR - Additional Control Register | 0x00 | Section 13.6.1 |
| **0x01** | UART_CPR - Clock Prescaler Register | 0x00 | Section 13.6.2 |
| **0x02** | UART_TCR - Time Clock Register | 0x00 | Section 13.6.3 |
| **0x03** | UART_CKS - Clock Select Register | 0x00 | Section 13.6.4 |
| **0x04** | UART_TTL - Transmitter Trigger Level Register | 0x00 | Section 13.6.5 |
| **0x05** | UART_RTL - Receiver Trigger Level Register | 0x00 | Section 13.6.6 |
| **0x06** | UART_FCL - Flow Control Level LSB Register | 0x00 | Section 13.6.7 |
| **0x07** | UART_FCH - Flow Control Level Register MSB | 0x00 | Section 13.6.8 |
| **0x08** | UART_ID1 - Identification 1 Register | 0x00 | Section 13.6.9 |
| **0x09** | UART_ID2 - Identification 2 Register | 0x00 | Section 13.6.10 |
| **0x0A** | UART_ID3 - Identification 3 Register | 0x00 | Section 13.6.11 |
| **0x0B** | UART_REV - Revision Register | 0x00 | Section 13.6.12 |
| **0x0C** | UART_CSR - Channel Software Reset Register | 0x00 | Section 13.6.13 |
| **0x0D** | UART_NMR - Nine Bit Mode Register | 0x00 | Section 13.6.14 |
| **0x0E** | UART_MDM - Modem Disable Mask Register | 0x00 | Section 13.6.15 |
| **0x0F** | UART_RFC - Readable FCR Register | 0x00 | Section 13.6.16 |
| **0x10** | UART_GDS - Good Data Status Register | 0x00 | Section 13.6.17 |
| **0x11** | UART_RSRV_1 - Reserved 1 Register | 0x00 | Section 13.6.18 |
| **0x12** | UART_PIDX - Port Index Register | 0x00 | Section 13.6.19 |
| **0x13** | UART_CKA - Clock Alteration Register | 0x00 | Section 13.6.20 |

**Table 9-1 - Overview of UART Registers**


Copyright © Bridgetek Pte Ltd

## 9.2 UART MODE SELECTION

The operation of the UART depends on a number of standard mode settings. These modes are referred to throughout this section. The compatibility modes are tabulated below.

| UART Mode | FIFO Size | FCR(0) | Enhanced Mode EFR(4) = 1 | FCR(5) Guarded with LCR(7) = 1 | FIFOSEL |
|---|---|---|---|---|---|
| 450 | 1 | 0 | X | X | X |
| 550 | 16 | 1 | 0 | 0 | 0 |
| Extended 550 | 128 | 1 | 0 | X | 1 |
| 650 | 128 | 1 | 1 | X | X |
| 750 | 128 | 1 | 0 | 1 | 0 |
| 950 | 128 | 1 | 1 | X | X |

**Table 9-2 - UART mode selection**

**450 Mode**

The 450 mode is the default mode set after a hardware reset. In the 450 Mode the FIFO is disabled, and the UART operates in BYTE mode. With FCR[0] cleared, (FIFO disabled) all other mode setting are ignored.

**550 Mode**

In the 550 mode the FIFO's are enabled, and can accept up to 16 bytes of data (reception and transmission directions). To put the UART into 550 mode, the FCR[0] should be set high. In this mode the FIFOSEL pin should be tied low.

**Extended 550 Mode**

The extended 550 mode is enabled by connection of the FIFOSEL to a HIGH state. In this mode the FIFO size is increased to 128 bytes.

**750 Mode**

The 750 mode is enabled by writing the FCR[0] with 1 and FCR[5] with 1. In the 750 mode the FIFO size is set to 128 bytes. Please note, that writes to FCR[5] are protected by FCR[7]. To write FCR[5], first set the FCR[7] high, then write FCR[5], and clear FCR[7] to activate protection. In the 750 mode the FIFOSEL pin should be tied low (0).

750 mode enhancements over 550 mode:

- Deeper FIFO size
- Automatic RTS/CTS out-of-band flow control
- Sleep mode

**650 Mode**

The 650 mode is active when EFR[4] is set (enhanced mode is enabled). As 650 software drivers usually put the device into enhanced mode, running 650 drivers on the UART device will result in 650 compatibilities with 128 deep FIFO's, as long as FCR[0] is set.

The FIFOSEL state is ignored in the 650 mode.

The 650 mode enhancements over 550 mode:


Copyright © Bridgetek Pte Ltd

- Deeper FIFO size
- Automatic RTS/CTS out-of-band flow control
- Sleep mode
- Automatic in-bank flow control
- Special character detection
- IRDA-format transmit and receive mode
- Transmit trigger levels
- Optional clock prescaler

**950 Mode**

The additional features of 950 mode apply only when UART is in Enhanced mode (EFR[4] = 1). FCR[0] set in Enhanced mode enables the 128 Bytes FIFO mode.

Configuration of the UART in 950 Mode is identical with the 650 Mode. Additional specific features of 950 Mode's, are enabled using the Additional Control Register ACR. In addition to larger FIFO's, higher baud rates the enhancements of 950 over 650 mode are:

- Selectable arbitrary trigger levels for the receiver and transmitter FIFO interrupts
- Improved automatic flow control using selectable arbitrary thresholds
- DSR/DTR automatic flow control
- Transmitter and receiver can be optionally disabled
- Software reset of device
- Readable FIFO fill levels
- Optional generation of an RS-485 buffer enable signal
- Four-byte device identification
- Readable status for automatic in-band and out-of-band flow control
- External 1x clock modes
- Flexible M N/8 clock prescaler
- 9-bit data mode

The 950 trigger levels are enabled when ACR[5] is set (FCR[7:4] are ignored). The arbitrary trigger levels can be defined in RTL, TTL, FCL and FCH registers. The Additional Status Register (ASR) offers flow control status for the local and remote transmitters. FIFO levels are readable using RFL and TFL registers.

The user may apply an external 1x (or Nx) clock for the transmitter and receiver to the RI and DSR pins respectively. The transmitter clock may be asserted on the DTR pin. The external clock options are selected through the CKS register.

It is also possible to define the over-sampling rate used by the transmitter and receiver clocks. The 450/550/750 and compatible devices employ 16 times over-sampling. There are 16 clock cycles per bit. The UART can employ over-sampling rate from 4 to 16 by programming the TCR register. This allows the data rates to be increased. Default value after reset for this register is 0x00, which corresponds to a 16-cycle sampling clock. Writing 0x01, 0x02 or 0x03 will also result in 16 cycle sampling clock. To program the value to any value from 4 to 15 it is necessary to write this value into TCR, to set the device to a 13-cycle sampling clock it would be necessary to write 0x0D to TCR.

The UART also offers 9-bit data frames for multi-drop industrial applications.

## 9.3 STANDARD 550 COMPATIBLE REGISTERS

### 9.3.1 UART_RBR - Receiver Buffer Register (address offset: 0x00 and LCR[7] = 0)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | RO | 8'h00 | FIFO Data Read from RX Buffer |

**Table 9-3 - UART_RBR - Receiver Buffer Register**


Copyright © Bridgetek Pte Ltd

### 9.3.2 UART_THR - Transmitter Holding Register (address offset: 0x00 and LCR[7] = 0)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | WO | 8'h00 | FIFO Data Write to TX Buffer |

**Table 9-4 - UART_THR - Transmitter Holding Register**

### 9.3.3 UART_DIV_LSB - Divisor LSB Register (address offset: 0x00 and LCR[7] = 1)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | WO | 8'h00 | The 8 least-significant bits (LSBs) of the 16-bit divisor for generation of the baud clock in the baud rate generator.<br>BaudRate = InputClock/ (SC * Divisor * prescaler) |

**Table 9-5 - UART_DIV_LSB - Divisor LSB Register**

### 9.3.4 UART_DIV_MSB - Divisor MSB Register (address offset: 0x01 and LCR[7] = 1)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | WO | 8'h00 | The 8 most-significant bits (MSBs) of the 16-bit divisor for generation of the baud clock in the baud rate generator.<br>BaudRate = InputClock/ (SC * Divisor * prescaler) |

**Table 9-6 - UART_DIV_MSB - Divisor MSB Register**

### 9.3.5 UART_INT_ENABLE - Interrupt Enable Register (address offset: 0x01)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | CTS_EN | RW | 1'h0 | Enable CTS interrupt when EFR[4] = 1 |
| 6 | RTS_EN | RW | 1'h0 | Enable RTS interrupt (when EFR[4]=1) |
| 5 | SC_EN | RW | 1'h0 | Enable Special Character interrupt Mask or Alternate Sleep Mode |
| 4 | SM_EN | RW | 1'h0 | Enable Sleep Mode |
| 3 | MOD_STS_EN | RW | 1'h0 | Enable Modem Status Interrupt |
| 2 | LINE_STS_EN | RW | 1'h0 | Enable Receiver Line Status Interrupt |
| 1 | TX_EMTY_EN | RW | 1'h0 | Enable Transmitter Holding Register Empty Interrupt |
| 0 | RX_AVL_EN | RW | 1'h0 | Enable Received Data Available Interrupt |

**Table 9-7 - UART_INT_ENABLE - Interrupt Enable Register**


Copyright © Bridgetek Pte Ltd

### 9.3.6 UART_INT_STATUS - Interrupt Status Register (address offset: 0x02)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:6 | Reserved | - | - | Reserved |
| 5:4 | Interrupt priority (Enhanced mode) | RO | 2'h0 | <Refer to Table 9-9 below> |
| 3:1 | Interrupt priority (All modes) | RO | 3'h0 | <Refer to Table 9-9 below> |
| 0 | Interrupt pending | RO | 1'h1 | <Refer to Table 9-9 below> |

**Table 9-8 - UART_INT_STATUS - Interrupt Status Register**

In order to provide minimum software overhead during data character transfers, the UART prioritizes interrupts into six levels and records these in the interrupt Status Register. When the CPU accesses the Interrupt status register, the UART freezes all interrupts and indicates the highest priority pending interrupt to the CPU. While this CPU access is occurring, the UART records new interrupt, but do not change its current indication until the access is complete. The table below shows the contents of the Interrupt Status Register.

| UART_INT_STATUS [5:0] | Interrupt Function | | |
|---|---|---|---|
| | INT TYPE | INT Source | Level |
| 000001 | none | No interrupt pending1 | - |
| 000110 | Receiver Line Status or address-bit detected in 9-bit mode | Overrun Error or Parity Error or Framing Error or Break Interrupt or address-bit detected in 9-bit mode | 1 |
| 000100 | Receiver Data Available | The receiver FIFO level is above the interrupt trigger level | 2a |
| 001100 | Receiver time-out | There has been no read of UART_RBR or a period of time greater than the time-out period. There has been no new data received and written into the UART_RBR for a period of time greater than the time-out period. | 2b |
| 000010 | Transmitter THR empty | Transmitter Holding Register Empty | 3 |
| 000000 | Modem status change | Clear to Send or Data Set Ready or Ring Indicator or Data Carrier Detect | 4 |
| 010000 | In-band flow control XOFF or special character (XOFF2) or special character 1,2,3 or 4 or bit 9 set in 9-bit mode | Valid XOFF, valid XOFF2 or matches special character 1,2,3 or 4(only in Enhanced mode) | 5 |
| 100000 | CTS or RTS change of state | When CTS or RTS bits will change | 6 |

**Table 9-9 - Interrupt Status Register Software Handling**

***Notes:***
[1] ISR[0] indicates whether any interrupts are pending.
[2] Interrupts of priority levels 5 and 6 cannot occur unless the UART is in Enhanced mode.
[3] ISR[5] is only used in 650 & 950 modes. In 750 mode, it is 0 when FIFO size is 16 and 1 when FIFO size is 128. In all other modes it is permanently set to 0.


Copyright © Bridgetek Pte Ltd

### 9.3.7 UART_FCR - FIFO Control Register (address offset: 0x02)

**550 AND 750 MODE**

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:6 | RCVR_TRIG | WO | 2'h0 | Receiver FIFO Trigger |
| 5 | FIFO_SIZE | WO | 1'h0 | Enable UART support for 128 Byte deep FIFO's |
| 4:3 | Reserved | WO | 2'h0 | |
| 2 | TXMT_RST | WO | 1'h0 | Transmitter FIFO reset. Writing a 1 to FCR2 clears all bytes in the XMIT FIFO and resets its counter logic to 0. The shift register is not cleared. The 1 that is written to this bit position is self-clearing. |
| 1 | RCVR_RST | WO | 1'h0 | Receiver FIFO reset. Writing a 1 to FCR1 clears all bytes in the RCVR FIFO and resets its counter logic to 0. The shift register is not cleared. The 1 that is written to this bit position is self-clearing. |
| 0 | FIFO_EN | WO | 1'h0 | Receiver and Transmitter FIFO's enable. |

**Table 9-10 - UART_FCR - FIFO Control Register – 550 mode**

This is a write only register at the same location as the ISR (the ISR is a read only register). Readable contents of this register are placed in ICR space on 0x0F offset. This register is used to enable the FIFOs, clear the FIFOs, set the FIFO triggers levels, and select the type of DMA signaling. When changing from the FIFO Mode to UART Mode and vice versa, data is automatically cleared from the FIFOs. This bit must be a 1 when other FCR bits are written to or they will not be programmed.

RCVR TRIG(1:0) - are used to set the trigger level for the RCVR FIFO interrupt.

| UART_FCR[7:6] | RCVR FIFO TRIGGER LEVEL | |
|---|---|---|
| | Standard FIFO*mode (16 B) | Extended FIFO*mode*(128B) |
| 00 | 1 | 1 |
| 01 | 4 | 32 |
| 10 | 8 | 64 |
| 11 | 14 | 112 |

**Table 9-11 - UART_RCVR - FIFO Trigger Level – 550 mode**

* - depends on Ext FIFO enable (FCR(5)) bit value

In this mode the transmitter trigger level is equal to 1.

**650 MODE**

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:6 | RCVR_TRIG | WO | 2'h0 | Receiver FIFO Trigger |
| 5:4 | THR_TRIG | WO | 1'h0 | Transmitter FIFO Trigger |
| 3 | Reserved | - | 2'h0 | - |


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 2 | TXMT_RST | WO | 1'h0 | Transmitter FIFO reset. Writing a 1 to FCR2 clears all bytes in the XMIT FIFO and resets its counter logic to 0. The shift register is not cleared. The 1 that is written to this bit position is self-clearing |
| 1 | RCVR_RST | WO | 1'h0 | Receiver FIFO reset. Writing a 1 to FCR1 clears all bytes in the RCVR FIFO and resets its counter logic to 0. The shift register is not cleared. The 1 that is written to this bit position is self-clearing |
| 0 | FIFO_EN | WO | 1'h0 | Receiver and Transmitter FIFO's enable |

**Table 9-12 - UART_FCR - FIFO Control Register – 650 mode**

This is a write only register at the same location as the ISR (the ISR is a read only register). This register is used to enable the FIFOs, clear the FIFOs, set the RCVR FIFO and THR FIFO triggers level, and select the type of DMA signaling. When changing from the FIFO Mode to UART Mode and vice versa, data is automatically cleared from the FIFOs. This bit must be a 1 when other FCR bits are written to or they will not be programmed.

RCVR TRIG(1:0) - are used to set the trigger level for the RCVR FIFO interrupt and flow control.

| UART_FCR[7:6] | RCVR FIFO TRIGGER LEVEL | |
|---|---|---|
| | Lower trigger for flow control | Interrupt trigger and upper trigger for flow control |
| 00 | 1 | 16 |
| 01 | 16 | 32 |
| 10 | 32 | 112 |
| 11 | 112 | 120 |

**Table 9-13 - UART_RCVR - FIFO Trigger Level – 650 mode**

THR TRGIGG(1:0) - are used to set the trigger level for the XMIT FIFO interrupt.

| UART_FCR[5:4] | TRANSMIT INTERRUPT TRIGGER LEVEL |
|---|---|
| 00 | 16 |
| 01 | 32 |
| 10 | 64 |
| 11 | 112 |

**Table 9-14 - XMIT FIFO Trigger Level**


Copyright © Bridgetek Pte Ltd

**950 MODE**

When ACR[5]=1, bits FCR[5:4] and FCR[7:6] are ignored and the transmitter trigger level can be defined by TTL(transmitter) and RTL(receiver) registers. The trigger level determined by TTL and RTL may be from 0 to 127. There are also FCH and FCL registers used for specifying triggers for the flow control feature.

Setting 0x00 to the TTL register causes an interrupt to occur when the FIFO and the transmitter shift register are both empty and the SO is in idle state.

### 9.3.8 UART_LCR - Line Control Register (address offset: 0x03)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | DLA | RW | 1'h0 | Divisor Latch Access Bit<br>0: Receiver and Transmitter Buffers enable<br>1: Divisor Latch Enable |
| 6 | SET_BRK | RW | 1'h0 | When set the transmitter is switched into break state, The SO Serial Output pin is driven into logic 0 state |
| 5 | SET_PARITY | RW | 1'h0 | Stick Parity<br>0: Disable Parity Stick<br>1: Enable Parity Stick<br><br>When bits 3, 4 and 5 are logic 1 the Parity bit is transmitted and checked as logic 0. If bits 3 and 5 are 1 and bit 4 is logic 0 then the Parity bit is transmitted and checked as logic 1. If bit 5 is logic 0, Stick Parity is disabled. |
| 4 | EVEN_PARITY | RW | 1'h0 | Even Parity Select<br>0: An odd number of logic ones is checked for in the transmitted and received character<br>1: An even number of logic ones is checked for in the transmitted and received character |
| 3 | PARITY_EN | RW | 1'h0 | Parity Enable<br>1: parity enabled<br>0: parity disabled |
| 2 | STOP_BITS | RW | 1'h0 | Number of STOP bits<br>0: 1 Stop bit generated<br>1: 1.5 stop bits generated for 5 data bits or 2 STOP bits generated for 6/7/8 data bits |
| 1:0 | WORD_LEN | RW | 2'h0 | Word Length select bits<br>00: 5 data bits<br>01: 6 data bits<br>10: 7 data bits<br>11: 8 data bits |

**Table 9-15 - UART_LCR - Line Control Register**

### 9.3.9 UART_MCR - Modem Control Register (address offset: 0x04)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | BAUD_PSCL* | RW | 1'h1 | Baud rate prescaler select<br>Writing a 0 to this bit sets the clock divider in the baud generator to 1, else the divider is an M where: M=CPR[7:3]. |
| 6 | IRDA_MODE* | RW | 1'h0 | IrDA mode<br>This bit is only available in 650 or 950 mode. A '1' on this bit enables IrDA mode, which transfers received and transmitted data in special format. |


Copyright © Bridgetek Pte Ltd

<table>
<tr><th>Bit</th><th>Name</th><th>Type</th><th>Default Value</th><th>Description</th></tr>
<tr><td>5</td><td>AFE_XON*</td><td>RW</td><td>1'h0</td><td>XON-any disabled/enabled. Auto Flow Control enable<br>1: auto flow control enabled<br>0: disable auto flow control<br>In Enhanced mode this bit is used for enabling the "XON any" feature. This feature allows re-enabling transmission in case of receiving any character in Automatic In-band Flow Control mode
<table>
<tr><th>AFE BIT(5)</th><th>RTS BIT(1)</th><th>FLOW CONFIGURATION</th></tr>
<tr><td>1</td><td>1</td><td>Auto-RTS and Auto-CTS enabled</td></tr>
<tr><td>1</td><td>0</td><td>Auto-CTS only enabled</td></tr>
<tr><td>0</td><td>-</td><td>Both Auto-RTS and Auto-CTS disabled</td></tr>
</table></td></tr>
<tr><td>4</td><td>LOOPBK_EN</td><td>RW</td><td>1'h0</td><td>Enable Loopback bit<br>1: This bit provides a local loop-back feature for diagnostic testing of the UART</td></tr>
<tr><td>3</td><td>OUT2</td><td>RW</td><td>1'h0</td><td>Output 2 (OUT2)<br>1: the OUT2 output is forced to a logic 0<br>0: the OUT2 output is forced to a logic 1</td></tr>
<tr><td>2</td><td>OUT1</td><td>RW</td><td>1'h0</td><td>Output1 (OUT1)<br>1: the OUT1 output is forced to a logic 0<br>0: the OUT1 output is forced to a logic 1</td></tr>
<tr><td>1</td><td>RTS</td><td>RW</td><td>1'h0</td><td>Request to send<br>1: the RTS output is forced to a logic 0<br>0: the RTS output is forced to a logic 1</td></tr>
<tr><td>0</td><td>DTR</td><td>RW</td><td>1'h0</td><td>Data Terminal Ready<br>1: the DTR output is forced to a logic 0<br>0: the DTR output is forced to a logic 1</td></tr>
</table>

*Only 650/950 mode.

**Table 9-16 - UART_MCR - Modem Control Register**

### 9.3.10 UART_LSR - Line Status Register (address offset: 0x05)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | RBR_ERR | RO | 1'h0 | Error in UART_RBR / Error in RCVR FIFO<br>In the UART Mode this is a 0.<br>In the FIFO mode,<br>1: when there is at least one parity error, framing error or break indication in the FIFO.<br>0: when LSR is read.<br>In 450 mode this bit is permanently cleared. In 9-bit mode this bit is not affected by LSR[2]. |
| 6 | TX_EMTY | RO | 1'h1 | Transmitter Empty<br>1: whenever the Transmitter Holding Register (THR) and the Transmitter Shift Register are both empty.<br>0: whenever either the THR or TSR contains a data character.<br>In the FIFO mode<br>1: whenever the transmitter FIFO and shift register are both empty |
| 5 | TXH_EMTY | RO | 1'h1 | Transmitter Holding Register Empty<br>1: Transmitter Holding Register is Empty<br>0: Transmitter Holding Register has data |


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| | | | | In the FIFO mode,<br>1: XMIT FIFO is empty<br>0: at least 1 byte is written to the XMIT FIFO |
| 4 | BRK_INT | RO | 1'h0 | Break Interrupt<br>1: whenever the received data input is held in the Space (logic 0) state for longer than a full word transmission time (that is, the total time of Start bit, data bits, a Parity and Stop bits).<br>0: whenever the CPU reads the contents of the Line Status Register.<br>In the FIFO mode this error is associated with the particular character in the FIFO it applies to. This error is revealed to the CPU when its associated character is at the top of the FIFO.<br>When break occurs only one zero character is loaded into the FIFO. The next character transfer is enabled after SI goes to the mark state and receives the next valid start bit. |
| 3 | FRM_ERR | RO | 1'h0 | Framing Error, indicates that the received character did not have a valid Stop bit.<br>1: whenever the Stop bit following the last data bit or parity bit is detected as a logic 0 bit (Spacing level).<br>0: whenever the CPU reads the contents of the Line Status Register.<br>In the FIFO mode this error is associated with the particular character in the FIFO it applies to. This error is revealed to the CPU when its associated character is at the top of the FIFO. The UART will try to resynchronize after a framing error. |
| 2 | PRTY_ERR_RX_9BIT | RO | 1'h0 | Parity Error/ 9-bit of received data in UART-RBR<br>1: upon detection of a parity error<br>0: whenever the CPU reads the contents of the Line Status Register.<br>In the FIFO mode this error is associated with the particular character in the FIFO it applies to. This error is revealed to the CPU when its associated character is at the top of the FIFO.<br>In 9-bit mode, this bit is the 9-th bit of the received data, in addition to the 8 bits in UART-RBR. |
| 1 | OVRN_ERR | RO | 1'h0 | Overrun Error<br>1: upon detection of an overrun condition and reset whenever the CPU reads the contents of the Line Status Register. If the FIFO mode data continues to fill the FIFO beyond the trigger level, an overrun error will occur only after the FIFO is full and the next character has been completely received in the shift register. OVRN_ERR is indicated to CPU as soon as it happens. The character in the shift register is overwritten, but it is not transferred to the FIFO. |
| 0 | DATA_RDY | RO | 1'h0 | Data Ready<br>1: whenever a complete incoming character has been received and transferred into the Receiver Buffer Register or the FIFO.<br>0: by reading all of the data in the Receiver Buffer Register or the FIFO |

**Table 9-17 - UART_LSR - Line Status Register**


Copyright © Bridgetek Pte Ltd

### 9.3.11 UART_MSR - Modem Status Register (address offset: 0x06)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | DCD | RO | 1'hX | Data Carrier Detect<br>This bit is the complement of the Data Carrier Detect (DCD) input. If bit 4 of the Modem Control Register is set to a 1, this bit is equivalent to OUT 2 in the Modem Control Register. |
| 6 | RI | RO | 1'hX | Ring Indicator<br>This bit is the complement of the Ring Indicator (RI) input. If bit 4 of the Modem Control Register is set to a 1, this bit is equivalent to OUT 1 in the Modem Control Register. |
| 5 | DSR | RO | 1'hX | Data Set Ready<br>This bit is the complement of the Data Set Ready (DSR) input.<br>If bit 4 of the Modem Control Register is set to a 1, this bit is equivalent to DTR in the Modem Control Register. |
| 4 | CTS | RO | 1'hX | Clear to Send<br>This bit is the complement of the Clear to Send (CTS) input.<br>If bit 4 (LOOPBK_EN) of the Modem control register is set to a 1, this bit is equivalent to RTS in the Modem Control Register. |
| 3 | D_DCD | RO | 1'h0 | Delta Data Carrier Detect<br>1: indicates that the DCD input to the chip has changed state. |
| 2 | TERI | RO | 1'h0 | Trailing Edge Ring Indicator<br>1: Indicates that the RI input to the chip has changed from a low to a high state. |
| 1 | D_DSR | RO | 1'h0 | Delta Data Set ready<br>1: Indicates that the DSR input to the chip has changed state since the last time it was read by the CPU. |
| 0 | D_CTS | RO | 1'h0 | Delta Clear To Send<br>1: Indicates that the CTS input to the chip have changed state since the last time it was read by the CPU. |

**Table 9-18 - UART_MSR - Modem Status Register**

### 9.3.12 UART_SPR - SPR Register (address offset: 0x07)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | RW | 8'h00 | This is the command port for the indexed control register. Writing and reading to indexed control register is addressed by offset written to this register |

**Table 9-19 - UART_SPR - SPR Register**

## 9.4 650 COMPATIBLE REGISTERS

To access these registers LCR must be set to 0xBF.


Copyright © Bridgetek Pte Ltd

### 9.4.1 UART_EFR - Enhanced Feature Register (address offset: 0x02)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | CTS_FC | RO | 1'h0 | 1: Automatic CTS flow control enable |
| 6 | RTS_FC | RO | 1'h0 | 1: Automatic RTS flow control enable |
| 5 | SPL_CHAR | RO | 1'h0 | 1: Special Character Detection mode enable |
| 4 | EM | RO | 1'h0 | 1: Enhanced mode enable |
| 3:2 | IBT_FCM | RO | 1'h0 | In-band transmit flow control mode<br>00: Disable in-band flow control<br>01: Enable single character in-band transmit flow control. Recognizing XON2 as the XON character and XOFF2 as the XOFF character<br>10: Enable single character in-band transmit flow control. Recognizing XON1 as the XON character and XOFF1 as the XOFF character<br>11: Reserved |
| 1:0 | IBR_FCM | RO | 1'h0 | In-band receive flow control mode<br>00: Disable in-band flow control<br>01: Enable single character in-band receive flow control. Recognizing XON2 as the XON character and XOFF2 as the XOFF character<br>10: Enable single character in-band receive flow control. Recognizing XON1 as the XON character and XOFF1 as the XOFF character<br>11: Reserved |

**Table 9-20 - UART_EFR - Enhanced Feature Register**

### 9.4.2 UART_XON1 - XON1 Register (address offset: 0x04)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | RW | 8'h00 | Value of XON1 |

**Table 9-21 - UART_XON1 - XON1 Register**

### 9.4.3 UART_XON2 - XON2 Register (address offset: 0x05)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | RW | 8'h00 | Value of XON2 |

**Table 9-22 - UART_XON2 - XON2 Register**

### 9.4.4 UART_XOFF1 - XOFF1 Register (address offset: 0x06)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | RW | 8'h00 | Value of XOFF1 |

**Table 9-23 - UART_XOFF1 - XOFF1 Register**


Copyright © Bridgetek Pte Ltd

### 9.4.5 UART_XOFF2 - XOFF2 Register (address offset: 0x07)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | RW | 8'h00 | Value of XOFF2 |

**Table 9-24 - UART_XOFF2 - XOFF2 Register**

## 9.5 950 COMPATIBLE REGISTERS

To access these registers ACR[7] must be set to 1.

### 9.5.1 UART_ASR - Additional Status Register (address offset: 0x01)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | TX_IDLE | RO | 1'h0 | 1: transmitter is idle (transmitter FIFO and shift register are empty)<br>0: transmitter is transmitting. |
| 6 | FIFO_SIZE | RO | 1'h0 | 1: FIFO are 16 deep if FCR[0]=1;<br>0: FIFO are 128 deep if FCR[0]=1 |
| 5 | FIFO_SEL | RO | 1'h0 | Actual state of FIFOSEL pin |
| 4 | SPECIAL_CHAR_DETECT | RO | 1'h0 | 1: special character detect and is stored in UART_RBR;<br>0: no special character detect<br>The flag is cleared by reading ASR. |
| 3 | DTR | RO | 1'h0 | Complement state of DTR pin |
| 2 | RTS | RO | 1'h0 | Complement state of the RTS pin |
| 1 | REMOTE_TX_DSBL | RW | 1'h0 | 1: transmitter has sent an XOFF character<br>0: the remote transmitter is not disabled by in-band flow control.<br>This bit may be cleared by software to re-enable remote transmitter (XON is sent) |
| 0 | TX_DSBL | RW | 1'h0 | 1: transmitter disabled (receiver detect XOFF)<br>0: transmitter is not disabled by in- band flow control.<br>This bit may be cleared by software to re-enable transmission if it was disabled by in-band flow control. |

**Table 9-25 - UART_ASR - Additional Status Register**

### 9.5.2 UART_RFL - Receiver FIFO Level Register (address offset: 0x03)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | RO | 8'h00 | Number of characters in the receiver FIFO |

**Table 9-26 - UART_RFL - Receiver FIFO Level Register**

**Note:** Reading from this register requires ACR[7] = 1


Copyright © Bridgetek Pte Ltd

### 9.5.3 UART_TFL - Transmitter FIFO Level Register (address offset: 0x04)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | RO | 8'h00 | Number of characters in the transmitter FIFO |

**Table 9-27 - UART_TFL - Transmitter FIFO Level Register**

**Note:** Reading from this register requires that last value written to LCR was not 0xBF and ACR[7]=1.

### 9.5.4 UART_ICR - ICR Register (address offset: 0x05)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | RW | 8'h00 | Data port to the indexed control register. Writing data to the indexed control register and reading data from the indexed control register is done using this register. |

**Table 9-28 - UART_ICR - ICR Register**

## 9.6 INDEXED CONTROL REGISTERS

Writing to Indexed Control Registers (ICRs) is addressed by the SPR offset, and data is loaded through the ICR register. Before writing, be sure that the LCR was not loaded with value '0xBF' (it enables access to 650 compatible registers). To write ICRs follow the steps in Figure 9.1.

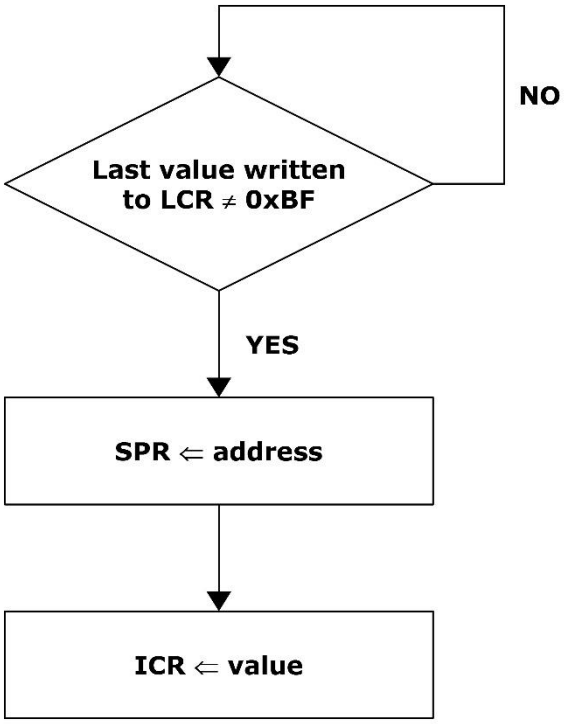



**Figure 9.1 - ICR registers write access**

Reading from Indexed Control Registers is addressed by the SPR offset, and data is read through the ICR register. Before reading be sure that LCR was not loaded with the value '0xBF' (it enables access to 650 compatible registers). To read from ICRs please follow the steps in Figure 9.2.


Copyright © Bridgetek Pte Ltd

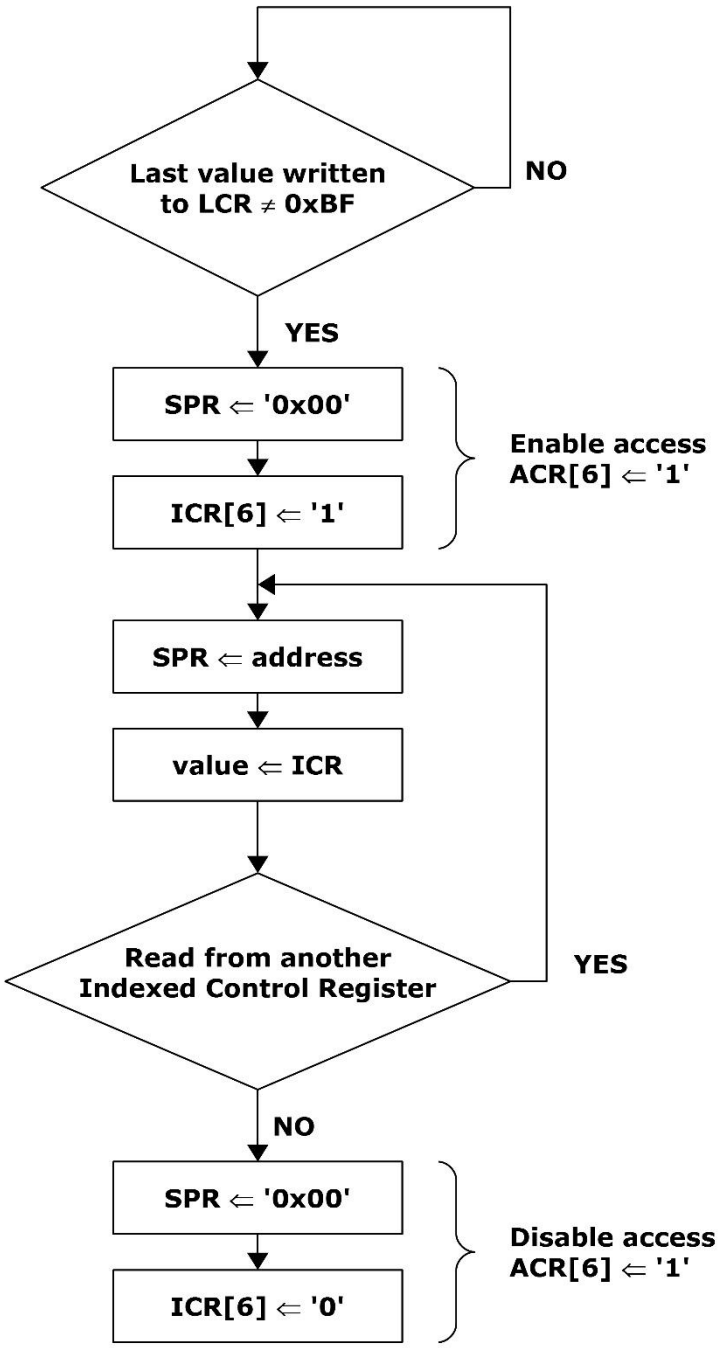



**Figure 9.2 - ICR registers read access**


Copyright © Bridgetek Pte Ltd

### 9.6.1 UART_ACR - Additional Control Register (SPR offset: 0x00)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | ADL_STS_EN | RW | 1'h0 | Additional status enable<br>1: ASR, TFL and RFL are enabled<br>0: disabled |
| 6 | ICR_RD_EN | RW | 1'h0 | ICR read enable<br>0: LSR is readable<br>1: ICR registers are readable |
| 5 | 950_TLE | RW | 1'h0 | 0: Interrupts and flow control trigger levels are described in the FCR register<br>1: The triggers are set by RTL, TTL, FCL and FCH registers |
| 4:3 | DTR | RW | 1'h0 | DTR line configuration<br>When CKS[4] or CKS[5] are set, the transmitter 1x clock or the output of the baud rate generator (Nx clock) are asserted on the DTR pin, otherwise the pin is defined as follows:<br>00: DTR pin is compatible with 450, 550, 650 and 750 (i.e., normal).<br>01: DTR pin is used for out-of-band flow control. It will be forced high, when the receiver FIFO level reaches the upper flow control trigger (FCH). It will be forced low when the receiver FIFO level falls below the lower flow control trigger (FCL).<br>10: DTR is configured to drive the active low enable pin of an external RS485 buffer. The pin will be forced low whenever the transmitter is not empty (LSR[6]=0), otherwise the pin is high.<br>11: DTR is configured to drive the active high enable pin of the external RS485 buffer. The pin will be forced high whenever the transmitter is not empty (LSR[6]=0), otherwise the pin is low. |
| 2 | DSR | RW | 1'h0 | 1: enables automatic out-of-band flow control using the DSR pin |
| 1 | TX | RW | 1'h0 | Transmitters disable.<br>0: Transmitter is enabled.<br>1: Transmitter is disabled. Data in THR are not transmitted. In-band flow control characters may still be transmitted. |
| 0 | RX | RW | 1'h0 | Receiver disable.<br>0: Receiver is enabled and data are stored in UART_RBR.<br>1: Receiver is disabled. The receiver continues to operate as normal to maintain frame synchronization but received data are not stored. In-band control characters continue to be detected and acted upon. Special characters will not be detected. |

**Table 9-29 - UART_ACR- Additional Control Register**

### 9.6.2 UART_CPR - Clock Prescaler Register (SPR offset: 0x01)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:3 | PSCL | RW | 5'h00 | clock prescaler |
| 2:0 | Reserved | - | - | - |

**Table 9-30 - UART_CPR - Clock Prescaler Register**


Copyright © Bridgetek Pte Ltd

### 9.6.3 UART_TCR - Time Clock Register (SPR offset: 0x02)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:4 | Reserved | - | - | - |
| 3:0 | N_TIMES_CLOCK | RW | 4'h00 | Bits N-times clock |

**Table 9-31 - UART_TCR - Time Clock Register**

### 9.6.4 UART_CKS - Clock Select Register (SPR offset: 0x03)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | TX_CLK_MD | RW | 1'h0 | Transmitter 1x clock mode selector<br>0: transmitter clock is in Nx clock mode<br>1: transmitter is in isochronous 1x clock mode |
| 6 | TX_CLK_SRC | RW | 1'h0 | Transmitter clock source selector<br>0: transmitter clock source is the output of the baud rate generator<br>1: transmitter uses external clock applied to the RI pin |
| 5:4 | TX_CLK_GEN | RW | 2'h0 | Transmitter 1x clock or baud rate generator output (BAUD_OUT) on the DTR pin.<br>00: The function of the DTR pin is defined by the setting of ACR[4:3].<br>01: The transmitter 1x clock is asserted on the DTR pin and setting of ACR[4:3] is ignored.<br>10: The output of baud rate generator (Nx clock) is asserted on the DTR pin and the setting of ACR[4:3] is ignored.<br>11: Reserved |
| 3 | RX_CLK_MD | RW | 1'h0 | Receiver 1x clock mode selector<br>0: Receiver clock is in Nx clock mode<br>1: Receiver is in isochronous 1x clock mode |
| 2 | BAUD_OUT | RW | 1'h0 | Disable BAUD_OUT pin<br>0: BAUD_OUT is enabled and connected to the baud rate generator which is Nx clock. By default, it is the 16x clock but using the TCR register it may be configured within a range from 4x to 16x clock.<br>1: BAUD_OUT is disabled and permanently set to logic 0 |
| 1:0 | RX_CLK_SRC | RW | 2'h0 | Receiver Clock source selector<br>00: The RCLK pin is selected for the receiver clock<br>01: The DSR pin is selected for the receiver clock<br>10: The baud rate generator output is selected for the receiver clock (internal BAUD_OUT connection)<br>11: The transmitter clock is selected for the receiver clock |

**Table 9-32 - UART_CKS Clock Select Register**


Copyright © Bridgetek Pte Ltd

### 9.6.5 UART_TTL - Transmitter Trigger Level Register (SPR offset: 0x04)

This register is located at offset 0x04 of the Indexed Control Register. This register is used for storing the interrupt trigger level for the transmitter in 950 mode (ACR[5] = 1). The interrupt occurs (if enabled) when the transmitter FIFO level falls below the value of the TTL register. If the TTL=0, then an interrupt will occur when both FIFO and shift register are empty and the SO line is marked to be in the idle state.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | Reserved | - | - | - |
| 6:0 | TRIG_LVL | RW | 7'h0 | Transmitter Trigger Level<br>The interrupt trigger level for the transmitter in 950 mode (ACR[5] = 1) |

**Table 9-33 - UART_TTL - Transmitter Trigger Level Register**

### 9.6.6 UART_RTL - Receiver Trigger Level Register (SPR offset: 0x05)

The RTL register is located at offset 0x05 of the ICR. This register is used for storing the interrupt trigger level for the receiver in 950 mode (ACR[5] = 1). The interrupt occurs (if enabled) when the receiver FIFO level reaches the value stored in this register.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | Reserved | - | - | - |
| 6:0 | TRIG_LVL | RW | 7'h0 | Receiver Trigger Level<br>The interrupt trigger level for the receiver in 950 mode (ACR[5] = 1) |

**Table 9-34 - UART_RTL - Receiver Trigger Level Register**

### 9.6.7 UART_FCL - Flow Control Level LSB Register (SPR offset: 0x06)

Automatic flow control is supported by FCL and FCH registers. These registers are active only in Enhanced mode, when FCR[6:7] bits are disabled (ACR[5] = 1). The FCL stores the lower trigger level and FCH stores the upper trigger level. Both registers are able to store level values from 0 to 127.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | Reserved | - | - | - |
| 6:0 | FLW_CNTL | RW | 7'h0 | Value of Flow Control LSB |

**Table 9-35 - UART_FCL - Flow Control Level LSB Register**

### 9.6.8 UART_FCH - Flow Control Level Register MSB (SPR offset: 0x07)

Automatic flow control is supported by FCL and FCH registers. These registers are active only in Enhanced mode, when FCR[6:7] bits are disabled (ACR[5] = 1). The FCL stores the lower trigger level and FCH stores the upper trigger level. Both registers are able to store level values from 0 to 127.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | Reserved | - | - | - |
| 6:0 | FLW_CNTL | RW | 7'h0 | Value of Flow Control MSB |

**Table 9-36 - UART_FCH - Flow Control Level Register MSB**



Copyright © Bridgetek Pte Ltd

### 9.6.9 UART_ID1 - Identification 1 Register (SPR offset: 0x08)

To identify the device type, use ID1,ID2,ID3 and REV registers. In ID1,ID2 and ID3 registers a hexadecimal ID of the device is written. The REV register includes a hardware revision sign.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | RW | 8'h0 | Identification value 1 |

**Table 9-37 - UART_ID1 - Identification 1 Register**

### 9.6.10 UART_ID2 - Identification 2 Register (SPR offset: 0x09)

To identify the device type, use ID1,ID2,ID3 and REV registers. In ID1,ID2 and ID3 registers a hexadecimal ID of the device is written.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | RW | 8'h0 | Identification value 2 |

**Table 9-38 - UART_ID2 - Identification 2 Register**

### 9.6.11 UART_ID3 - Identification 3 Register (SPR offset: 0x0A)

To identify the device type, use ID1,ID2,ID3 and REV registers. In ID1,ID2 and ID3 registers a hexadecimal ID of the device is written.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | RW | 8'h0 | Identification value 3 |

**Table 9-39 - UART_ID3 - Identification 3 Register**

### 9.6.12 UART_REV - Revision Register (SPR offset: 0x0B)

To identify the device type, use ID1,ID2,ID3 and REV registers. In ID1,ID2 and ID3 registers a hexadecimal ID of the device is written.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | RW | 8'h0 | hardware revision sign |

**Table 9-40 - UART_REV - Revision Register**

### 9.6.13 UART_CSR - Channel Software Reset Register (SPR offset: 0x0C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | RW | 8'h0 | 00: To reset the UART write 0x00 to the Channel Software Reset register |

**Table 9-41 - UART_CSR - Channel Software Reset Register**

### 9.6.14 UART_NMR - Nine Bit Mode Register (SPR offset: 0x0D)

To enable 9-bit data mode NMR[0] bit must be logic 0. In this mode data are nine bits wide, and the 9-th bit is stored in LSR[2] for receiving. In transmission the 9-th bit should be written in SPR[0] bit before writing 8-bit data to THR.


Copyright © Bridgetek Pte Ltd

In 9-bit mode the setting in LCR[1:0] are ignored. Furthermore, as parity is permanently disabled, the setting of LCR[5:3] is also ignored.

In 9-bit mode, in-band flow control is disabled.

When the UART is configured for both Enhanced and 9-bit data mode, setting IER[5] will enable detection of up to four 'address' characters. The eight least significant bits of these characters are stored in XON1, XON2, XOFF1 and XOFF2 registers. The 9-th bit of these characters is stored in NMR[5] to NMR[2].

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:6 | Reserved | - | - | - |
| 5 | 9_SC4 | RW | 1'h0 | 9-th bit of special characters |
| 4 | 9_SC3 | RW | 1'h0 | 9-th bit of special characters |
| 3 | 9_SC2 | RW | 1'h0 | 9-th bit of special characters |
| 2 | 9_SC1 | RW | 1'h0 | 9-th bit of special characters |
| 1 | 9_INT_EN | RW | 1'h0 | 9-bit data mode interrupt enable<br>0: interrupt for detection of an 'address' character is disabled<br>1: interrupt for detection of an 'address' character is enabled |
| 0 | 9_EN | RW | 1'h0 | 9-bit data mode enable |

**Table 9-42 - UART_NMR - Nine Bit Mode Register**

### 9.6.15 UART_MDM - Modem Disable Mask Register (SPR offset: 0x0E)

MDM is used to mask selected interrupts of modem lines.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:4 | Reserved | - | - | - |
| 3 | DCD_MASK | RW | 1'h0 | Delta DCD disable.<br>0: Enables level 4 interrupt from delta DCD when IER[3]=1<br>1: Disables level 4 interrupt from delta DCD. |
| 2 | RI_MASK | RW | 1'h0 | Trailing edge RI disable.<br>0: Enables level 4 interrupt from trailing edge RI when IER[3]=1<br>1: Disables level 4 interrupt from trailing edge RI. |
| 1 | DSR_MASK | RW | 1'h0 | Delta DSR disable.<br>0: Enables level 4 interrupt from delta DSR when IER[3]=1<br>1: Disables level 4 interrupt from delta DSR. |
| 0 | CTS_MASK | RW | 1'h0 | Delta CTS disable.<br>0: Enables level 4 interrupt from delta CTS when IER[3]=1<br>1: Disables level 4 interrupt from delta CTS. |

**Table 9-43 - UART_MDM - Modem Disable Mask Register**


Copyright © Bridgetek Pte Ltd

### 9.6.16 UART_RFC - Readable FCR Register (SPR offset: 0x0F)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | FCR | RO | 8'h00 | read the state of FCR register |

**Table 9-44 - UART_RFC - Readable FCR Register**

### 9.6.17 UART_GDS - Good Data Status Register (SPR offset: 0x10)

Good data status is set when the following conditions are true:

- ISR reads level 0 (no interrupt), level 2 or 2a (receiver data) or level 3 (THR empty) interrupt.
- LSR[7] is clear.
- LSR[1] is clear.

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | RO | 8'h00 | contains 'good data status' bit on least significant position. |

**Table 9-45 - UART_GDS - Good Data Status Register**

### 9.6.18 UART_RSRV_1 - Reserved 1 Register (SPR offset: 0x11)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | Reserved | - | - | - |

**Table 9-46 - UART_RSRV_1 - Reserved 1 Register**

### 9.6.19 UART_PIDX - Port Index Register (SPR offset: 0x12)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | RO | 8'h00 | The value of UART index is 0. |

**Table 9-47 - UART_PIDX - Port Index Register**

### 9.6.20 UART_CKA - Clock Alteration Register (SPR offset: 0x13)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:5 | Reserved | - | - | - |
| 4 | CLK_TXRDY | RW | 1'h0 | Output system clock on TxRdy pin |
| 3 | CLK_SEL | RW | 1'h0 | Use CLKSEL pin for system clock |
| 2 | INV_DTR | RW | 1'h0 | Invert DTR signal |
| 1 | INV_TX_CLK | RW | 1'h0 | Invert internal Tx clock |
| 0 | INV_RX_CLK | RW | 1'h0 | Invert internal Rx clock |

**Table 9-48 - UART_CKA - Clock Alteration Register**


Copyright © Bridgetek Pte Ltd

# 10 Timers and Watchdog

FT93x consists of a 32-bit watchdog timer and four 16-bit users' timers.

The watchdog timer is clocked off the main clock. The watchdog can be initialized with a 5-bit register. The value of this register points to a bit of the 32-bit counter which will be set. A timer decrements and signals an interrupt when it rolls over. Once started and initialized the watchdog cannot be stopped. It can be cleared by writing into a register.

Four user timers can be clocked off the main clock; each timer has its own 16-bit prescaler. These timers can be started, stopped and cleared/initialized. The prescalers can be cleared/initialized the same way. Current values of all four user timers can be read from registers (one at a time - common register, multiplexed access). All timers count up or down and signal an interrupt when rolling over. Each of the timers can be configured to be one-shot or in continuous mode. They are initialized from a common register and only one at a time (multiplexed access).

If the user timer has already started using its prescaler it cannot be cleared and the command is ignored. Each of the prescalers automatically stops after it is cleared individually. It also starts automatically when the corresponding user timer starts using it.

Normal operation:

| **User timers** | **Prescaler** | **Watchdog** |
|---|---|---|
| Select the timer to initialize by writing into the **TIMER_SELECT** register.<br><br>Write initial/final value into **TIMER_WRITE_LS** and **TIMER_WRITE_MS** registers. | The corresponding prescaler is selected by **TIMER_**SELECT register.<br><br>Write initial value into **TIMER_PRESC_LS** and **TIMER_PRESC_MS** registers. | Write initial value into **TIMER_WDG** register to initialize one of the 32 bits of the timer. |
| Write into **direction** bit in **TIMER_CONTROL_3** register to select up/down counting. | N/A | N/A |
| Write into **mode** bit in **TIMER_CONTROL_3** register to select mode. | N/A | N/A |
| Write into **clear_x** bits in **TIMER_CONTROL_4** register to initialize the timer. | Write into **clear_presc** bit in **TIMER_CONTROL_4** register to initialize the prescaler (if possible) | Write into **clear_wdg** bit in **TIMER_CONTROL_2** register to clear watchdog. |
| Write into **start_x** bits in **TIMER_CONTROL_1** register to start the timer. | Write into **prescaler_en** bit in **TIMER_CONTROL_2** register to enable prescaler and it will automatically start when the timer/timers using it starts. | Write into **start_wdg** bit in **TIMER_CONTROL_2** register to start watchdog. |
| Select timer you want to read from by writing into **timer_read_sel** bit in **TIMER_SELECT** register.<br><br>Current value can be read from **TIMER_READ_LS** and **TIMER_READ_MS** registers. | N/A | N/A |
| Write into **stop_x** bit in **TIMER_CONTROL_1** register to stop the timer. | N/A | N/A |

**Table 10-1 - Timers/Watchdog Operation**


Copyright © Bridgetek Pte Ltd

## 10.1 Register Summary

Listed below are the registers with their offset from the base address (0x10340). All registers can only be accessed via Byte (8-bit) mode.

| Offset | Register | Default Value | Reference |
|---|---|---|---|
| | | | |
| **0x00** | TIMER_CONTROL_0 - Timers Control Register 0 | 0x00 | Section 14.2.1 |
| **0x01** | TIMER_CONTROL_1 - Timers Control Register 1 | 0x00 | Section 14.2.2 |
| **0x02** | TIMER_CONTROL_2 - Timers Control Register 2 | 0x00 | Section 14.2.3 |
| **0x03** | TIMER_CONTROL_3 - Timers Control Register 3 | 0x00 | Section 14.2.4 |
| **0x04** | TIMER_CONTROL_4 - Timers Control Register 4 | 0x00 | Section 14.2.5 |
| **0x05** | TIMER_INT - Timers Interrupt Register | 0x00 | Section 14.2.6 |
| **0x06** | TIMER_SELECT - Timers A..D Select Register | 0x00 | Section 14.2.7 |
| **0x07** | TIMER_WDG - Watchdog Start Value | 0x00 | Section 14.2.8 |
| **0x08** | TIMER_WRITE_LS - Timer A..D Start Value 7:0 | 0x00 | Section 14.2.9 |
| **0x09** | TIMER_WRITE_MS - Timer A..D Start Value 15:8 | 0x00 | Section 14.2.10 |
| **0x0A** | TIMER_PRESC_LS - Prescaler Start Value 7:0 | 0x00 | Section 14.2.11 |
| **0x0B** | TIMER_PRESC_MS - Prescaler Start Value 15:8 | 0x00 | Section 14.2.12 |
| **0x0C** | TIMER_READ_LS - Timer A..D Current Value 7:0 | 0x00 | Section 14.2.13 |
| **0x0D** | TIMER_READ_MS - Timer A..D Current Value 15:8 | 0x00 | Section 14.2.14 |

**Table 10-2 - Overview of Timers/Watchdog Registers**

## 10.2 Register Details

### 10.2.1 TIMER_CONTROL_0 - Timers Control Register 0 (address offset: 0x00)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 1 | block_en | RW | 1'b0 | 1: To enable the timer module |
| 0 | soft_reset | W1T | 1'b0 | 1: Write 1 to trigger the reset |

**Table 10-3 - TIMER_CONTROL_0 - Timers Control Register 0**

### 10.2.2 TIMER_CONTROL_1 - Timers Control Register 1 (address offset: 0x01)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | stop_d | W1T | 1'b0 | 1: To trigger stopping timer D |
| 6 | stop_c | W1T | 1'b0 | 1: To trigger stopping timer C |
| 5 | stop_b | W1T | 1'b0 | 1: To trigger stopping timer B |
| 4 | stop_a | W1T | 1'b0 | 1: To trigger stopping timer A |
| 3 | start_d | W1T | 1'b0 | 1: To trigger starting timer D |
| 2 | start_c | W1T | 1'b0 | 1: To trigger starting timer C |
| 1 | start_b | W1T | 1'b0 | 1: To trigger starting timer B |
| 0 | start_a | W1T | 1'b0 | 1: To trigger starting timer A |

**Table 10-4 - TIMER_CONTROL_1 - Timers Control Register 1**


Copyright © Bridgetek Pte Ltd

### 10.2.3 TIMER_CONTROL_2 - Timers Control Register 2 (address offset: 0x02)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:4 | prescaler_en | RW | 4'h0 | Enable prescaler bits for timers D/C/B/A respectively |
| 3 | wdg_int_ien | RW | 1'b0 | 1: enable watchdog interrupt |
| 2 | wdg_int | RW1C | 1'b0 | 1: watchdog interrupt pending |
| 1 | clear_wdg | W1T | 1'b0 | 1: To trigger clearing watchdog timer |
| 0 | start_wdg | W1T | 1'b0 | 1: To trigger starting watchdog timer |

**Table 10-5 - TIMER_CONTROL_2 - Timers Control Register 2**

### 10.2.4 TIMER_CONTROL_3 - Timers Control Register 3 (address offset: 0x03)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | direction_d | RW | 1'b0 | Counter direction bits for timer D<br>1: Up<br>0: Down |
| 6 | direction_c | RW | 1'b0 | Counter direction bits for timer C |
| 5 | direction_b | RW | 1'b0 | Counter direction bits for timer B |
| 4 | direction_a | RW | 1'b0 | Counter direction bits for timer A |
| 3 | mode_d | RW | 1'b0 | Continuous/1-shot mode bits for timer D<br>1: 1-shot<br>0: Continuous |
| 2 | mode_c | RW | 1'b0 | Continuous/1-shot mode bits for timer C |
| 1 | mode_b | RW | 1'b0 | Continuous/1-shot mode bits for timer B |
| 0 | mode_a | RW | 1'b0 | Continuous/1-shot mode bits for timer A |

**Table 10-6 - TIMER_CONTROL_3 - Timers Control Register 3**

### 10.2.5 TIMER_CONTROL_4 - Timers Control Register 4 (address offset: 0x04)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:5 | Reserved | | | |
| 4 | presc_clear | W1T | 1'b0 | 1: To trigger clearing prescaler |
| 3 | clear_d | W1T | 1'b0 | 1: To trigger clearing timer D |
| 2 | clear_c | W1T | 1'b0 | 1: To trigger clearing timer C |
| 1 | clear_b | W1T | 1'b0 | 1: To trigger clearing timer B |
| 0 | clear_a | W1T | 1'b0 | 1: To trigger clearing timer A |

**Table 10-7 - TIMER_CONTROL_4 - Timers Control Register 4**

### 10.2.6 TIMER_INT - Timers Interrupt Register (address offset: 0x05)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | timer_int_d_en | RW | 1'b0 | 1: Enable timer D interrupt |
| 6 | timer_int_d | RW1C | 1'b0 | 1: Timer D interrupt pending |
| 5 | timer_int_c_en | RW | 1'b0 | 1: Enable timer C interrupt |
| 4 | timer_int_c | RW1C | 1'b0 | 1: Timer C interrupt pending |
| 3 | timer_int_b_en | RW | 1'b0 | 1: Enable timer B interrupt |
| 2 | timer_int_b | RW1C | 1'b0 | 1: Timer B interrupt pending |
| 1 | timer_int_a_en | RW | 1'b0 | 1: Enable timer A interrupt |
| 0 | timer_int_a | RW1C | 1'b0 | 1: Timer A interrupt pending |

**Table 10-8 - TIMER_INT - Timers Interrupt Register**


Copyright © Bridgetek Pte Ltd

### 10.2.7 TIMER_SELECT - Timers A..D Select Register (address offset: 0x06)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 3:2 | timer_read_sel | RW | 2'h0 | Select one of Timers/prescalers A/B/C/D to read (value 0/1/2/3) |
| 1:0 | timer_write_sel | RW | 2'h0 | Select one of Timers/prescalers A/B/C/D to write (value 0/1/2/3) |

**Table 10-9 - TIMER_SELECT - Timers A..D Select Register**

### 10.2.8 TIMER_WDG - Watchdog Start Value (address offset: 0x07)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 4:0 | timer_wdg_write | RW | 5'h00 | Setting watchdog value<br>5'h00 32'h00000001<br>5'h01 32'h00000002<br>… …<br>5'h1E 32'h40000000<br>5'h1F 32'h80000000 |

**Table 10-10 - TIMER_WDG - Watchdog Start Value**

### 10.2.9 TIMER_WRITE_LS - Timer A..D Start Value 7:0 (address offset: 0x08)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | timer_write_7_0 | RW | 8'h00 | Write low byte of the timer start value |

**Table 10-11 - TIMER_WRITE_LS - Timer A..D Start Value 7:0**

### 10.2.10 TIMER_WRITE_MS - Timer A..D Start Value 15:8 (address offset: 0x09)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | timer_write_15_8 | RW | 8'h00 | Write high byte of the timer start value |

**Table 10-12 - TIMER_WRITE_MS - Timer A..D Start Value 15:8**

### 10.2.11 TIMER_PRESC_LS - Prescaler Start Value 7:0 (address offset: 0x0A)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | timer_presc_7_0 | RW | 8'h00 | Write low byte of the timer prescaler start value |

**Table 10-13 - TIMER_PRESC_LS - Prescaler Start Value 7:0**

### 10.2.12 TIMER_PRESC_MS - Prescaler Start Value 15:8 (address offset: 0x0B)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | timer_presc_15_8 | RW | 8'h00 | Write high byte of the timer prescaler start value |

**Table 10-14 - TIMER_PRESC_MS - Prescaler Start Value 15:8**


Copyright © Bridgetek Pte Ltd

### 10.2.13 TIMER_READ_LS - Timer A..D Current Value 7:0 (address offset: 0x0C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | timer_read_7_0 | RO | 8'h00 | Read low byte of the timer start value |

**Table 10-15 - TIMER_READ_LS - Timer A..D Current Value 7:0**

### 10.2.14 TIMER_READ_MS - Timer A..D Current Value 15:8 (address offset: 0x0D)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | timer_read_15_8 | RO | 8'h00 | Read high byte of the timer start value |

**Table 10-16 - TIMER_READ_MS - Timer A..D Current Value 15:8**


Copyright © Bridgetek Pte Ltd

# 11 SPI Master

There is a SPI Master module in the device. Listed below are the key features of this SPI master:

- Full duplex synchronous serial data transfer
- Single, Dual and Quad SPI transfer
- Master operation
- Multimaster system supported
- Two modes of operations: SPI mode and FIFO mode
- FIFO size of 64 bytes
- Support up to 8 SPI slaves
- System error detection
- Interrupt generation
- Bit rates generated 1/4, 1/8, 1/16, 1/32, 1/64, 1/128, 1/256, 1/512 of system clock
- Four transfer formats supported

## 11.1 Register Summary

Listed below are the registers with their offset from the base address (0x10440). All registers can only be accessed via Double-Word (32-bit) mode. However, the least significant byte of the FIFO can be accessed via Byte (8-bit) mode. This facilitates fast byte-oriented operations.

| Address Offset | Register | Default Value | Reference |
|---|---|---|---|
| | | | |
| **0x00** | SPIM_CNTL – Control Register | 0x04 | Section 11.2.1 |
| **0x04** | SPIM_STATUS – Status Register | 0x0C | Section 11.2.2 |
| **0x08** | SPIM_DATA – Receiver and Transmitter Data Registers | 0x00 | Section 11.2.3 |
| **0x0C** | SPIM_SLV_SEL_CNTL – Slave Select Control Register | 0xFF | Section 11.2.4 |
| **0x10** | SPIM_FIFO_CNTL – FIFO Control Register | 0x00 | Section 11.2.5 |
| **0x14** | SPIM_TNSFR_FRMT_CNTL – Transfer Format Control Register | 0x00 | Section 11.2.6 |
| **0x18** | SPIM_ALT_DATA – Alternative SPI Master Data Register | 0x00 | Section 11.2.7 |
| **0x1C** | SPIM_RX_FIFO_COUNT – SPI Master RX FIFO Count Register | 0x00 | Section 11.2.8 |
| **0x20** | SPIM_CNTL_2 – Control 2 Register | 0x01 | Section 11.2.9 |
| **0x30** | SPIM_STATUS_2 – Status 2 Register | 0x45 | Section 11.2.10 |
| **0x34** | SPIM_FIFO_CNTL_2 – FIFO Control 2 Register | 0x01 | Section 11.2.11 |
| **0x38** | SPIM_TX_FIFO_COUNT – SPI Master TX FIFO Count Register | 0x10 | Section 11.2.12 |
| **0x3C** | SPIM_BAUD – SPI Master Baud Register | 0x04 | Section 11.2.13 |

**Table 11-1 - Overview of SPI Master Registers**


Copyright © Bridgetek Pte Ltd

## 11.2 Register Details

### 11.2.1 SPIM_CNTL – Control Register (address offset: 0x00)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | SP_IE | RW | 1'b0 | 1: To enable SPI Master interrupt |
| 6 | SP_E | RW | 1'b0 | SPI System enable; 1 to enable. |
| 5 | SP_R2 | RW | 1'b0 | See table at SP_R1 and SP_R0 |
| 4 | MSTR | RW | 1'b0 | 1: To enable this SPI Master |
| 3 | CLK_POL | RW | 1'b0 | Clock polarity select<br>0: High level; SCK idles Low<br>1: Low level; SCK idles high |
| 2 | CLK_PHA | RW | 1'b1 | Clock phase<br>0: Shift Data Out on Falling edge; capture Data In on Rising edge<br>1: Shift Data Out on Rising edge; capture Data In on Falling edge |
| 1:0 | SP_R[1:0] | RW | 1'b0 | Together with SP_R2, they define the SPI clock rate |

| SP_R2 | SP_R1 | SP_R0 | System Clock divided by |
|---|---|---|---|
| 0 | 0 | 0 | 4 |
| 0 | 0 | 1 | 8 |
| 0 | 1 | 0 | 16 |
| 0 | 1 | 1 | 32 |
| 1 | 0 | 0 | 64 |
| 1 | 0 | 1 | 128 |
| 1 | 1 | 0 | 256 |
| 1 | 1 | 1 | 512 |

**Table 11-2 - SPIM_CNTL – Control Register**

### 11.2.2 SPIM_STATUS – Status Register (address offset: 0x04)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | SPI_FLAG | RW | 1'b0 | Interrupt request; this flag is automatically set to one at the end of an SPI transfer |
| 6 | WR_COL | RW | 1'b0 | Write collision error status flag. The flag is automatically set if the SPDR is written when the TX register is full (in FIFO Mode when the TX FIFO is full) |
| 5 | SPI_BIS | RW | 1'b0 | Indicates end of transmission from SPIM_ALT_DATA register. This flag can generate an interrupt if enabled by SPIM_TNSFR_FRMT_CNTL[6]=1 |
| 4 | MOD_FAULT | RW | 1'b0 | SPI mode-fault error status flag. This flag is set if the SS pin goes to active low. |
| 3 | THRE | RW | 1'b1 | SPI in IDLE state with TX FIFO or THR register empty<br>0: Transmission is in progress |


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 2 | TX_EMPTY | RW | 1'b1 | Transmitter Empty<br>0: TX FIFO contains at least one byte.<br>1: TX FIFO is empty |
| 1 | RX_FIFOFULL | RW | 1'b0 | Receiver FIFO Full |
| 0 | SSC_EN | RW | 1'b0 | Slave Select Control Enable<br>1: auto SS assertions enabled<br>0: auto SS assertions disabled – SS always shows contents of Slave Select Control Register |

**Table 11-3 - SPIM_STATUS – Status Register**

### 11.2.3 SPIM_DATA – Receiver and Transmitter Data Registers (address offset: 0x08)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | SPDR_RX | RO | 8'h00 | Data from last Receive operation |
| 7:0 | SPDR_TX | WO | 8'h00 | Data for next Transmit operation |

**Table 11-4 - SPIM_DATA – Receiver and Transmitter Data Registers**

### 11.2.4 SPIM_SLV_SEL_CNTL – Slave Select Control Register (address offset: 0x0C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | RW | 8'hFF | SS7-SS0 pin select<br>0: assigned SS while Master transfer is active<br>1: SS is forced to logic 1 |

**Table 11-5 - SPIM_SLV_SEL_CNTL – Slave Select Control Register**

### 11.2.5 SPIM_FIFO_CNTL – FIFO Control Register (address offset: 0x10)

<table>
<tr><th>Bit</th><th>Name</th><th>Type</th><th>Default Value</th><th colspan="3">Description</th></tr>
<tr><td rowspan="7">7:6</td><td rowspan="7">RCVR_TRIG</td><td rowspan="7">RW</td><td rowspan="7">2'h0</td><td></td><td colspan="2">RCVR FIFO Trigger Level</td></tr>
<tr><td>Bits</td><td>Standard FIFO Mode (16W)*</td><td>Extended FIFO Mode (64W)*</td></tr>
<tr><td>00</td><td>01</td><td>01</td></tr>
<tr><td>01</td><td>04</td><td>16</td></tr>
<tr><td>10</td><td>08</td><td>32</td></tr>
<tr><td>11</td><td>14</td><td>56</td></tr>
<tr><td colspan="3">* Depends on SPIM_FIFO_CNTL[5] value</td></tr>
<tr><td>5</td><td>64_BYTE</td><td>RW</td><td>1'b0</td><td colspan="3">1: 64 Byte deep FIFOs enabled</td></tr>
<tr><td>4</td><td>TIMEOUT</td><td>RW</td><td>1'b0</td><td colspan="3">1: Enable Timeout interrupt</td></tr>
<tr><td>3</td><td>Reserved</td><td>-</td><td>-</td><td colspan="3">-</td></tr>
<tr><td>2</td><td>TX_RST</td><td>RW</td><td>1'b0</td><td colspan="3">Write 1 to TX FIFO and its logic; The shift register is not affected; This bit will clear itself</td></tr>
</table>


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 1 | RCVR_RST | RW | 1'b0 | Write 1 to RX FIFO and its logic; The shift register is not affected; This bit will clear itself |
| 0 | FIFO_EN | RW | 1'b0 | RX and TX FIFO's enable |

**Table 11-6 - SPIM_FIFO_CNTL – FIFO Control Register**

### 11.2.6 SPIM_TNSFR_FRMT_CNTL – Transfer Format Control Register (address offset: 0x14)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | FIFO_EXT | RW | 1'b0 | 1: Enable FIFO extension and allow 16 bits data transfer to / from FIFO |
| 6 | BISINT_EN | RW | 1'b0 | 1: Enable interrupt generation after the transfer is complete from SPIM_ALT_DATA register |
| 5 | MULTI_REC | RW | 1'b0 | 1: Allow continuous reception of data without the necessity of loading the TX FIFO |
| 4 | Reserved | - | - | - |
| 3 | TX_IEN | RW | 1'b0 | 1: Transmitter FIFO Empty interrupt enabled |
| 2 | DIR | RW | 1'b0 | 1: Performs multichannel READ<br>(Requires DUAL / QUAD mode to be enabled) |
| 1 | QUAD_SPI | RW | 1'b0 | 1: Enable QUAD SPI transfer |
| 0 | DUAL_SPI | RW | 1'b0 | 0: Enable DUAL SPI transfer |

**Table 11-7 - SPIM_TNSFR_FRMT_CNTL – Transfer Format Control Register**

### 11.2.7 SPIM_ALT_DATA – Alternative SPI Master Data Register (address offset: 0x18)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | RW | 8'h00 | Alternative SPI Mode Data register. Data transmitted through this register are done as a single channel SPI only regardless of QUAD/DUAL mode setting |

**Table 11-8 - SPIM_ALT_DATA – Alternative SPI Master Data Register**

### 11.2.8 SPIM_RX_FIFO_COUNT – SPI Master RX FIFO Count Register (address offset: 0x1C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | Reserved | - | - | - |
| 6:0 | DATA | RW | 7'h00 | The number of bytes available in the RX FIFO.<br><br>Note that the counter can only count up to 63, then rolls back to 0. If it is 0 and the SPI_FLAG bit is 1, it means there are 64 bytes in the FIFO |

**Table 11-9 - SPIM_RX_FIFO_COUNT – SPI Master RX FIFO Count Register**


Copyright © Bridgetek Pte Ltd

### 11.2.9 SPIM_CNTL_2 – Control 2 Register (address offset: 0x20)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | BAUD_REG_EN | RW | 1'b0 | 1 : Use BAUD Register to generate SCK |
| 6:4 | Reserved | - | - | - |
| 3 | CLSBF | RW | 1'b0 | 1 : LSB (0) of Command is sent/received first; this bit affects only TX operations. |
| 2 | LSBF | RW | 1'b0 | 1 : LSB (0) of Data is sent/received first; this bit affects both TX and RX operations. |
| 1 | Reserved | - | - | - |
| 0 | RX_EMPTY | RO | 1'b1 | 1 : RX data buffer or RX FIFO are empty |

**Table 11-10 - SPIM_CNTL_2 – Control 2 Register**

### 11.2.10 SPIM_STATUS_2 – Status 2 Register (address offset: 0x30)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | Reserved | - | - | - |
| 6 | TEMT | RO | 1'b1 | 1 : Indicates TX FIFO or THR register are empty |
| 5 | TXSPIF | RO | 1'b0 | 1 : Indicates that the vacant entries in TX FIFO are more than the TX FIFO Trigger level |
| 4 | RX_FULLOVERRIDDEN | RO | 1'b0 | 1 : RX FIFO overflows and data has been overridden. Clear this bit by reading SPDR |
| 3 | RX_FIFOFULL | RO | 1'b0 | 1 : RX FIFO is full |
| 2 | RX_EMPTY | RO | 1'b1 | 1 : RX FIFO is empty |
| 1 | TX_FIFOFULL | RO | 1'b0 | 1 : TX FIFO is full |
| 0 | TX_FIFOEMPTY | RO | 1'b1 | 1 : TX FIFO is empty |

**Table 11-11 - SPIM_STATUS_2 – Status 2 Register**

### 11.2.11 SPIM_FIFO_CNTL_2 – FIFO Control 2 Register (address offset: 0x34)

| Bit | Name | Type | Default Value | Description | |
|---|---|---|---|---|---|
| 7:4 | Reserved | - | - | - | |
| 3:2 | TX_TRIG | RW | 2'h0 | | TX FIFO Trigger Level |


Copyright © Bridgetek Pte Ltd

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | Bits | Standard FIFO Mode (16W)* | Extended FIFO Mode (64W)* |
| | | | | 00 | 01 | 01 |
| | | | | 01 | 04 | 16 |
| | | | | 10 | 08 | 32 |
| | | | | 11 | 14 | 56 |
| | | | | * Depends on SPIM_FIFO_CNTL[5] value | | |
| 1 | TXTRIG_EN | RW | 1'b0 | 1 : Enable TX TRIGGER LEVEL | | |
| 0 | RXTRIG_EN | RW | 1'b1 | 1 : Enable RX TRIGGER LEVEL | | |

**Table 11-12 - SPIM_FIFO_CNTL_2 – FIFO Control 2 Register**

### 11.2.12 SPIM_TX_FIFO_COUNT – SPI Master TX FIFO Count Register (address offset: 0x38)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | Reserved | - | - | - |
| 6:0 | DATA | RO | 7'h10 | The number of bytes available in the TX FIFO.<br><br>Note that the counter can only count up to 63, then rolls back to 0. If it is 0 and the SPI_FLAG bit is 1, it means there are 64 bytes in the FIFO. |

**Table 11-13 - SPIM_TX_FIFO_COUNT – SPI Master TX FIFO Count Register**

### 11.2.13 SPIM_BAUD – SPI Master Baud Register (address offset: 0x3C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | BAUD_FACTOR | RO | 8'h04 | When BAUD REG EN is set, SCK frequency is determined by BAUD FACTOR ( divided 2 to divided by 256).<br><br>When the division factor is odd, the default state of SCK specified by CPOL will be 1 CLK period longer than the other state. |

**Table 11-14 - SPIM_BAUD – SPI Master Baud Register**


Copyright © Bridgetek Pte Ltd

# 12 SPI Slave

There is a SPI slave in the device. Listed below are the key features of the SPI slave:

- Full duplex synchronous serial data transfer
- Two modes of operations: SPI mode and FIFO mode
- FIFO size of 64 bytes
- System error detection
- Interrupt generation
- Four transfer formats supported

## 12.1 Register Summary

Listed below are the registers with their offset from the base addresses (0x10480). All registers can only be accessed via Double-Word (32-bit) mode. However, the least significant byte of the FIFO can be accessed via Byte (8-bit) mode. This facilitates fast byte-oriented operations.

| Address Offset | Register | Default Value | Reference |
|---|---|---|---|
| | | | |
| **0x00** | SPIS_CNTL – Control Register | 0x04 | Section 12.2.1 |
| **0x04** | SPIS_STATUS – Status Register | 0x0C | Section 12.2.2 |
| **0x08** | SPIS_DATA – Receiver and Transmitter Data Registers | 0x00 | Section 12.2.3 |
| **0x0C** | SPIS_SLV_SEL_CNTL – Slave Select Control Register | - | Section 12.2.4 |
| **0x10** | SPIS_FIFO_CNTL – FIFO Control Register | 0x00 | Section 12.2.5 |
| **0x14** | SPIS_TNSFR_FRMT_CNTL – Transfer Format Control Register | 0x00 | Section 12.2.6 |
| **0x18** | SPIS_ALT_DATA – Alternative SPI Slave Data Register | 0x00 | Section 12.2.7 |
| **0x1C** | SPIS_RX_FIFO_COUNT – SPI Slave RX FIFO Count Register | 0x00 | Section 12.2.8 |
| **0x20** | SPIS_CNTL_2 – Control 2 Register | 0x01 | Section 12.2.9 |
| **0x30** | SPIS_STATUS_2 – Status 2 Register | 0x45 | Section 12.2.10 |
| **0x34** | SPIS_FIFO_CNTL_2 – FIFO Control 2 Register | 0x01 | Section 12.2.11 |
| **0x38** | SPIS_TX_FIFO_COUNT – SPI Slave TX FIFO Count Register | 0x10 | Section 12.2.12 |
| **0x3C** | SPIS_BAUD – SPI Slave Baud Register | 0x04 | Section 12.2.13 |

**Table 12-1 - Overview of SPI Slave Registers**


Copyright © Bridgetek Pte Ltd

## 12.2 Register Details

### 12.2.1 SPIS_CNTL – Control Register (address offset: 0x00)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | SP_IE | RW | 1'b0 | 1: To enable SPI Slave interrupt |
| 6 | SP_E | RW | 1'b0 | SPI System enable; 1 to enable. |
| 5 | SP_R2 | RW | 1'b0 | See table at SP_R1 and SP_R0 |
| 4 | Reserved | - | - | - |
| 3 | CLK_POL | RW | 1'b0 | Clock polarity select<br>0: High level; SCK idles Low<br>1: Low level; SCK idles high |
| 2 | CLK_PHA | RW | 1'b1 | Clock phase<br>0: Shift Data Out on Falling edge; capture Data In on Rising edge<br>1: Shift Data Out on Rising edge; capture Data In on Falling edge |
| 1:0 | SP_R[1:0] | RW | 1'b0 | Set SP_R2/1/0 value to 0 if the SPI master is operating at high speed. Otherwise, set it to a non-zero value. |

**Table 12-2 - SPIS_CNTL – Control Register**

### 12.2.2 SPIS_STATUS – Status Register (address offset: 0x04)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | SPI_FLAG | RW | 1'b0 | Interrupt request; this flag is automatically set to one at the end of an SPI transfer |
| 6 | WR_COL | RW | 1'b0 | Write collision error status flag. The flag is automatically set if the SPDR is written when the TX register is full (in FIFO Mode when the TX FIFO is full) |
| 5 | SPI_BIS | RW | 1'b0 | Indicates end of transmission from the SPIS_ALT_DATA register. This flag can generate an interrupt if enabled by SPIS_TNSFR_FRMT_CNTL[6]=1 |
| 4 | Reserved | - | - | - |
| 3 | THRE | RW | 1'b1 | SPI in IDLE state with TX FIFO or THR register empty<br>0: Transmission is in progress |
| 2 | TX_EMPTY | RW | 1'b1 | Transmitter Empty<br>0: TX FIFO contains at least one byte.<br>1: TX FIFO is empty |
| 1 | RX_FIFOFULL | RW | 1'b0 | Receiver FIFO Full |
| 0 | SSC_EN | RW | 1'b0 | Slave Select Control Enable<br>1: auto SS assertions enabled<br>0: auto SS assertions disabled – SS always shows contents of Slave Select Control Register |

**Table 12-3 - SPIS_STATUS – Status Register**


Copyright © Bridgetek Pte Ltd

### 12.2.3 SPIS_DATA – Receiver and Transmitter Data Registers (address offset: 0x08)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | SPDR_RX | RO | 8'h00 | Data from last Receive operation |
| 7:0 | SPDR_TX | WO | 8'h00 | Data for next Transmit operation |

**Table 12-4 - SPIS_DATA – Receiver and Transmitter Data Registers**

### 12.2.4 SPIS_SLV_SEL_CNTL – Slave Select Control Register (address offset: 0x0C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | Reserved | - | - | - |

**Table 12-5 - SPIS_SLV_SEL_CNTL – Slave Select Control Register**

### 12.2.5 SPIS_FIFO_CNTL – FIFO Control Register (address offset: 0x10)

<table>
<tr><th>Bit</th><th>Name</th><th>Type</th><th>Default Value</th><th colspan="3">Description</th></tr>
<tr><td rowspan="7">7:6</td><td rowspan="7">RCVR_TRIG</td><td rowspan="7">RW</td><td rowspan="7">2'h0</td><td></td><td colspan="2">RCVR FIFO Trigger Level</td></tr>
<tr><td>Bits</td><td>Standard FIFO Mode (16W)*</td><td>Extended FIFO Mode (64W)*</td></tr>
<tr><td>00</td><td>01</td><td>01</td></tr>
<tr><td>01</td><td>04</td><td>16</td></tr>
<tr><td>10</td><td>08</td><td>32</td></tr>
<tr><td>11</td><td>14</td><td>56</td></tr>
<tr><td colspan="3">* Depends on SPIS_FIFO_CNTL[5] value</td></tr>
<tr><td>5</td><td>64_BYTE</td><td>RW</td><td>1'b0</td><td colspan="3">1: 64 Byte deep FIFOs enabled</td></tr>
<tr><td>4</td><td>TIMEOUT</td><td>RW</td><td>1'b0</td><td colspan="3">1: Enable Timeout interrupt</td></tr>
<tr><td>3</td><td>Reserved</td><td>-</td><td>-</td><td colspan="3">-</td></tr>
<tr><td>2</td><td>TX_RST</td><td>RW</td><td>1'b0</td><td colspan="3">Write 1 to TX FIFO and its logic; The shift register is not affected; This bit will clear itself</td></tr>
<tr><td>1</td><td>RCVR_RST</td><td>RW</td><td>1'b0</td><td colspan="3">Write 1 to RX FIFO and its logic; The shift register is not affected; This bit will clear itself</td></tr>
<tr><td>0</td><td>FIFO_EN</td><td>RW</td><td>1'b0</td><td colspan="3">RX and TX FIFO's enable</td></tr>
</table>

**Table 12-6 - SPIS_FIFO_CNTL – FIFO Control Register**


Copyright © Bridgetek Pte Ltd

### 12.2.6 SPIS_TNSFR_FRMT_CNTL – Transfer Format Control Register (address offset: 0x14)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | FIFO_EXT | RW | 1'b0 | 1: Enable FIFO extension and allow 16 bits data transfer to / from FIFO |
| 6 | BISINT_EN | RW | 1'b0 | 1: Enable interrupt generation after transfer is complete from the SPIS_ALT_DATA register |
| 5 | MULTI_REC | RW | 1'b0 | 1: Allow continuous reception of data without the necessity of loading the TX FIFO |
| 4 | Reserved | - | - | - |
| 3 | TX_IEN | RW | 1'b0 | 1: Transmitter FIFO Empty interrupt enabled |
| 2:0 | Reserved | - | - | - |

**Table 12-7 - SPIS_TNSFR_FRMT_CNTL – Transfer Format Control Register**

### 12.2.7 SPIS_ALT_DATA – Alternative SPI Slave Data Register (address offset: 0x18)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | RW | 8'h00 | Alternative SPI Mode Data register. Data transmitted through this register are done as a single channel SPI only, regardless of QUAD/DUAL mode setting |

**Table 12-8 - SPIS_ALT_DATA – Alternative SPI Slave Data Register**

### 12.2.8 SPIS_RX_FIFO_COUNT – SPI Slave RX FIFO Count Register (address offset: 0x1C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | RW | 8'h00 | The number of bytes available in the RX FIFO. Note that the counter can only count up to 63, then rolls back to 0. If it is 0 and SPI_FLAG bit is 1, it means there are 64 bytes in the FIFO |

**Table 12-9 - SPIS_RX_FIFO_COUNT – SPI Slave RX FIFO Count Register**

### 12.2.9 SPIS_CNTL_2 – Control 2 Register (address offset: 0x20)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | BAUD_REG_EN | RW | 1'b0 | 1 : Use BAUD Register to generate SCK |
| 6:4 | Reserved | - | - | - |
| 3 | CLSBF | RW | 1'b0 | 1 : LSB (0) of Command is sent/received first; this bit affects only TX operations. |
| 2 | LSBF | RW | 1'b0 | 1 : LSB (0) of Data is sent/received first; this bit affects both TX and RX operations. |


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 1 | Reserved | - | - | - |
| 0 | RX_EMPTY | RO | 1'b1 | 1 : RX data buffer or RX FIFO are empty |

**Table 12-10 - SPIS_CNTL_2 – Control 2 Register**

### 12.2.10 SPIS_STATUS_2 – Status 2 Register (address offset: 0x30)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | Reserved | - | - | - |
| 6 | TEMT | RO | 1'b1 | 1 : Indicates TX FIFO or THR register are empty |
| 5 | TXSPIF | RO | 1'b0 | 1 : Indicates that the vacant entries in TX FIFO are more than the TX FIFO Trigger level |
| 4 | RX_FULLOVERRIDDEN | RO | 1'b0 | 1 : RX FIFO overflows and data has been overridden. Clear this bit by reading SPDR |
| 3 | RX_FIFOFULL | RO | 1'b0 | 1 : RX FIFO is full |
| 2 | RX_EMPTY | RO | 1'b1 | 1 : RX FIFO is empty |
| 1 | TX_FIFOFULL | RO | 1'b0 | 1 : TX FIFO is full |
| 0 | TX_FIFOEMPTY | RO | 1'b1 | 1 : TX FIFO is empty |

**Table 12-11 - SPIS_STATUS_2 – Status 2 Register**


Copyright © Bridgetek Pte Ltd

### 12.2.11 SPIS_FIFO_CNTL_2 – FIFO Control 2 Register (address offset: 0x34)

| Bit | Name | Type | Default Value | Description | | |
|---|---|---|---|---|---|---|
| 7:4 | Reserved | - | - | - | | |
| 3:2 | TX_TRIG | RW | 2'h0 | | TX FIFO Trigger Level | |
| | | | | Bits | Standard FIFO Mode (16W)* | Extended FIFO Mode (64W)* |
| | | | | 00 | 01 | 01 |
| | | | | 01 | 04 | 16 |
| | | | | 10 | 08 | 32 |
| | | | | 11 | 14 | 56 |
| | | | | * Depends on SPIS_FIFO_CNTL[5] value | | |
| 1 | TXTRIG_EN | RW | 1'b0 | 1 : Enable TX TRIGGER LEVEL | | |
| 0 | RXTRIG_EN | RW | 1'b1 | 1 : Enable RX TRIGGER LEVEL | | |

**Table 12-12 - SPIS_FIFO_CNTL_2 – FIFO Control 2 Register**

### 12.2.12 SPIS_TX_FIFO_COUNT – SPI Slave TX FIFO Count Register (address offset: 0x38)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | Reserved | - | - | - |
| 6:0 | DATA | RO | 7'h10 | The number of bytes available in the TX FIFO.<br><br>Note that the counter can only count up to 63, then rolls back to 0. If it is 0 and the SPI_FLAG bit is 1, it means there are 64 bytes in the FIFO |

**Table 12-13 - SPIS_TX_FIFO_COUNT – SPI Slave TX FIFO Count Register**

### 12.2.13 SPIS_BAUD – SPI Slave Baud Register (address offset: 0x3C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | BAUD_FACTOR | RO | 8'h04 | When BAUD REG EN is set, SCK frequency is determined by BAUD FACTOR ( divided 2 to divided by 256).<br><br>When the division factor is odd, the default state of SCK specified by CPOL will be 1 CLK period longer than the other state |

**Table 12-14 - SPIS_BAUD – SPI Slave Baud Register**


Copyright © Bridgetek Pte Ltd

# 13 I²C Master

The I²C Master conforms to v2.1 and v3.0 of the I²C specification.

Listed below are the key features supported by the I²C master:

- Supports Standard Speed Mode (up to 100kb/s)
- Supports Fast mode (up to 400kb/s)
- Supports Fast-plus mode (up to 1Mb/s)
- Supports High-speed mode (up to 3.4Mb/s)
- Performs arbitration and clock synchronisation
- Supports multi-master systems
- Supports both 7-bit and 10-bit address modes
- Supports interrupt generation
- Supports FIFO mode

## 13.1 Register Summary

Listed below are the registers with their offset from the base address (0x10500). All registers can only be accessed via Byte (8-bit) mode.

| Address Offset | Register | Default Value | Reference |
|---|---|---|---|
| | | | |
| **0x00** | I2CM_SLV_ADDR - Slave Address Register | 0x00 | Section 18.2.1 |
| **0x01** | I2CM_CNTL - Control Register | 0x00 | Section 18.2.2 |
| **0x01** | I2CM_STATUS - Status Register | 0x20 | Section 18.2.3 |
| **0x02** | I2CM_DATA - Receive / Transmit Data Register | 0x00 | Section 18.2.4 |
| **0x03** | I2CM_TIME_PERIOD - Timer Period Register | 0x01 | Section 18.2.5 |
| **0x03** | I2CM_HS_TIME_PERIOD - High Speed Timer Period Register | 0x01 | Section 18.2.6 |
| **0x04** | I2CM_FIFO_LEN - FIFO Mode Byte Length | 0x00 | Section 18.2.7 |
| **0x05** | I2CM_FIFO_INT_ENABLE - FIFO Mode Interrupt Enable | 0x00 | Section 18.2.8 |
| **0x06** | I2CM_FIFO_INT_PEND - FIFO Mode Interrupt Pending | 0x00 | Section 18.2.9 |
| **0x07** | I2CM_FIFO_DATA - FIFO Data Register | 0x00 | Section 18.2.10 |
| **0x08** | I2CM_TRIG - Trigger Register | 0x00 | Section 18.2.11 |

**Table 13-1 - Overview of I2C Master Registers**


Copyright © Bridgetek Pte Ltd

## 13.2 Register Details

### 13.2.1 I2CM_SLV_ADDR – Slave Address Register (address offset: 0x00)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:1 | SLV_ADDR | RW | 7'h00 | This is 7-bit address bits |
| 0 | RX_OP | RW | 1'b0 | 0: next operation is a transmission<br>1: next operation is a reception |

**Table 13-2 - I2CM_SLV_ADDR – Slave Address Register**

### 13.2.2 I2CM_CNTL – Control Register (address offset: 0x01)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | I2C_RST | WO | 1'b0 | Resets the whole $I^2C$ Master controller. |
| 6 | SLV_RST | WO | 1'b0 | If set together with the RUN bit, the master will generate 9 $I^2C$ clocks without generating the START condition to recover a blocking Slave device to a known state. A STOP condition will be generated.<br>This bit will be automatically cleared. |
| 5 | ADDR | WO | 1'b0 | Setting this together with the RUN bit will cause the generation of a START condition and transmission of a Slave address. |
| 4 | HS | WO | 1'b0 | Setting this together with the RUN bit switches the Bus controller into high-speed mode. |
| 3 | ACK | WO | 1'b0 | This bit should normally be set when the Master is in the receiver mode to generate ACK. It must be cleared when the master requires no further data from the Slave transmitter. |
| 2 | STOP | WO | 1'b0 | Setting this will cause the STOP condition to be generated. |
| 1 | START | WO | 1'b0 | Setting this will cause the START or Repeated START condition to be generated. |
| 0 | RUN | WO | 1'b0 | Setting this will cause the Bus controller to be active. |

**Table 13-3 - I2CM_CNTL – Control Register**

### 13.2.3 I2CM_STATUS – Status Register (address offset: 0x01)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | Reserved | - | - | - |
| 6 | BUS_BUSY | RO | 1'b0 | 1: indicates the Bus is Busy, and access is not possible. It's reset by START/STOP conditions. |
| 5 | I2C_IDLE | RO | 1'b1 | 1: indicates the Bus controller is in the IDLE state. |
| 4 | ARB_LOST | RO | 1'b0 | 1: indicates that during the last operation the Bus controller lost the arbitration |


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 3 | DATA_ACK | RO | 1'b0 | 1: indicates that during the last transmit operation data wasn't acknowledged. |
| 2 | ADDR_ACK | RO | 1'b0 | 1: indicates that during the last operation the slave address wasn't acknowledged. |
| 1 | I2C_ERR | RO | 1'b0 | 1: indicates an error occurred during the last operation – ARB_LOST, DATA_ACK or ADDR_ACK |
| 0 | I2C_BUSY | RO | 1'b0 | 1: indicates that the Bus controller is receiving / transmitting data on the bus; other status bits of the Status register are not valid. |

**Table 13-4 - I2CM_STATUS – Status Register**

### 13.2.4 I2CM_DATA – Receive / Transmit Data Register (address offset: 0x02)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | RW | 8'h00 | When read, this is the data received in the last transaction.<br>When written, this is the data to be transmitted in the next transaction. |

**Table 13-5 - I2CM_DATA – Receive / Transmit Data Register**

### 13.2.5 I2CM_TIME_PERIOD – Timer Period Register (address offset: 0x03)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | TIME_ENB | RW | 1'b0 | Cleared to use this register. |
| 6:0 | SCL_LP | RW | 7'h01 | Frequency scaler used in STANDARD_FAST and FAST_PLUS modes. The value is appended with a 1 at LSB to make it 8-bit.<br>SCL_PERIOD = (SCL_LP[6] * 128 + SCL_LP[5] * 64 + SCL_LP[4] * 32 + SCL_LP[3] * 16 + SCL_LP[2] * 8 + SCL_LP[1] * 4 + SCL_LP[0] * 2 + 1) * CLK_PERIOD |

**Table 13-6 - I2CM_TIME_PERIOD – Timer Period Register**

### 13.2.6 I2CM_HS_TIME_PERIOD – High Speed Timer Period Register (address offset: 0x03)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | TIME_ENB | WO | 1'b0 | Set to use this register. |
| 6 | FAST | WO | 1'b0 | Set to indicate to the Bus controller to use FAST generic timing parameters. |
| 5 | Reserved | - | - | - |
| 4:0 | SCL_HP | WO | 1'b1 | Frequency scalar used in FAST mode. The value is appended with a 1 at the LSB, and prepended with 2 0's to make it 8-bit.<br>SCL_PERIOD = (SCL_HP[4] * 32 + SCL_HP[3] * 16 + SCL_HP[2] * 8 + SCL_HP[1] * 4 + SCL_HP[0] * 2 + 1) |


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| | | | | * CLK_PERIOD |

**Table 13-7 - I2CM_HS_TIME_PERIOD – High Speed Timer Period Register**

### 13.2.7 I2CM_FIFO_LEN – FIFO Mode Byte Length (address offset: 0x04)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | FIFO_BL | RW | 8'h00 | Number of bytes (FIFO_BL + 1) to transmit / receive when FIFO mode is enabled |

**Table 13-8 - I2CM_FIFO_LEN – FIFO Mode Byte Length**

### 13.2.8 I2CM_FIFO_INT_ENABLE – FIFO Mode Interrupt Enable (address offset: 0x05)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | DONE | RW | 1'b0 | FIFO_BL operation completed Interrupt |
| 6 | I2C_INT | RW | 1'b0 | I2C Interrupt status |
| 5 | RX_FULL | RW | 1'b0 | RX FIFO Full interrupt enable |
| 4 | RX_HALF | RW | 1'b0 | RX FIFO Half Full interrupt enable |
| 3 | RX_EMPTY | RW | 1'b0 | RX FIFO Empty interrupt enable |
| 2 | TX_FULL | RW | 1'b0 | TX FIFO Full interrupt enable |
| 1 | TX_HALF | RW | 1'b0 | TX FIFO Half Full interrupt enable |
| 0 | TX_EMPTY | RW | 1'b0 | TX FIFO Empty interrupt enable |

**Table 13-9 - I2CM_FIFO_INT_ENABLE – FIFO Mode Interrupt Enable**

### 13.2.9 I2CM_FIFO_INT_PEND – FIFO Mode Interrupt Pending (address offset: 0x06)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | DONE | RW1C | 1'b0 | FIFO_BL operation complete interrupt pending |
| 6 | I2C_INT | RW1C | 1'b0 | I2C Interrupt pending |
| 5 | RX_FULL | RW1C | 1'b0 | RX FIFO Full interrupt pending |
| 4 | RX_HALF | RW1C | 1'b0 | RX FIFO Half Full interrupt pending |


Copyright © Bridgetek Pte Ltd

| | | | | |
|---|---|---|---|---|
| 3 | RX_EMPTY | RW1C | 1'b0 | RX FIFO Empty interrupt pending |
| 2 | TX_FULL | RW1C | 1'b0 | TX FIFO Full interrupt pending |
| 1 | TX_HALF | RW1C | 1'b0 | TX FIFO Half Full interrupt pending |
| 0 | TX_EMPTY | RW1C | 1'b0 | TX FIFO Empty interrupt pending |

**Table 13-10 - I2CM_FIFO_INT_PEND – FIFO Mode Interrupt Pending**

### 13.2.10 I2CM_FIFO_DATA - FIFO Data Register (address offset: 0x07)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | FIFO_DATA | RW | 8'h00 | FIFO Data Read from RX Buffer |

**Table 13-11 - I2CM_FIFO_DATA - FIFO Data Register**

### 13.2.11 I2CM_TRIG - Trigger Register (address offset: 0x08)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | RX_OP | RW | 1'b0 | A write to this register triggers the FIFO mode operation. Set this bit to 1 for RX, and 0 for TX FIFO operations.<br>The operation will end when FIFO_BL expires. |
| 6:0 | Reserved | - | - | - |

**Table 13-12 - I2CM_TRIG - Trigger Register**


Copyright © Bridgetek Pte Ltd

# 14 I²C Slave

The I²C Slave conforms to v2.1 and v3.0 of the I²C specification.

Listed below are the key features supported by the I²C slave:

- Supports Standard Speed Mode (up to 100kb/s)
- Supports Fast mode (up to 400kb/s)
- Supports Fast-plus mode (up to 1Mb/s)
- Supports High-speed mode (up to 3.4Mb/s)
- Performs arbitration and clock synchronisation
- Supports interrupt generation
- Supports FIFO mode

## 14.1 Register Summary

Listed below are the registers with their offset from the base address (0x10510). All registers can only be accessed via Byte (8-bit) mode.

| Address Offset | Register | Default Value | Reference |
|---|---|---|---|
| | | | |
| **0x00** | I2CS_OWN_ADDR - Own Address Register | 0x00 | Section 14.2.1 |
| **0x01** | I2CS_CNTL - Control Register | 0x00 | Section 14.2.2 |
| **0x01** | I2CS_STATUS - Status Register | 0x00 | Section 14.2.3 |
| **0x02** | I2CS_DATA - Receive / Transmit Data Register | 0x00 | Section 14.2.4 |
| **0x04** | I2CS_FIFO_LEN - FIFO Mode Byte Length | 0x00 | Section 14.2.5 |
| **0x05** | I2CS_FIFO_INT_ENABLE - FIFO Mode Interrupt Enable | 0x00 | Section 14.2.6 |
| **0x06** | I2CS_FIFO_INT_PEND - FIFO Mode Interrupt Pending | 0x00 | Section 14.2.7 |
| **0x07** | I2CS_FIFO_DATA - FIFO Data Register | 0x00 | Section 14.2.8 |
| **0x08** | I2CS_TRIG - Trigger Register | 0x00 | Section 14.2.9 |

**Table 14-1 - Overview of I2C Master Registers**

## 14.2 Register Details

### 14.2.1 I2CS_OWN_ADDR – Own Address Register (address offset: 0x00)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | Reserved | - | - | - |


Copyright © Bridgetek Pte Ltd

| | | | | |
|---|---|---|---|---|
| 6:0 | OWN_ADDR | RW | 7'h00 | This is the seven address bits of the Slave controller. |

**Table 14-2 - I2CS_OWN_ADDR – Own Address Register**

### 14.2.2 I2CS_CNTL – Control Register (address offset: 0x01)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | I2C_RST | WO | 1'b0 | Setting this bit will reset the whole Slave controller. |
| 6 | DEV_ACTV | RW | 1'b0 | Device Active<br>1: enables the Slave controller operations<br>0: disables the Slave controller operations<br>Writing a 1 sets DEV_ACTV to 1 immediately while writing 0 will not be effective immediately if there is any on-going transmission. It's suggested that this bit is polled if a 0 is written. |
| 5:4 | Reserved | - | - | - |
| 3 | REC_FIN_CLR | WO | 1'b0 | Writing 1 to this bit clears REC_FIN bit from the Status register. |
| 2 | SEND_FIN_CLR | WO | 1'b0 | Writing 1 to this bit clears SEND_FIN bit from the Status register. |
| 1:0 | Reserved | - | - | - |

**Table 14-3 - I2CS_CNTL – Control Register**

### 14.2.3 I2CS_STATUS – Status Register (address offset: 0x01)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | Reserved | - | - | - |
| 6 | DEV_ACTV | RW | 1'b0 | Device Active<br>1: enables the Slave controller operations<br>0: disables the Slave controller operations<br>Writing a 1 set DEV_ACTV to 1 immediately while writing 0 will not be effective immediately if there is any on-going transmission. It's suggested that this bit is polled if a 0 is written. |
| 5 | Reserved | - | - | - |
| 4 | BUS_ACTV | RO | 1'b0 | 1: indicates that there is transmission: send, receive or own address detection in progress |
| 3 | REC_FIN | RO | 1'b0 | 1: indicates that the Master has ended the transmit operation. It means no more RX_REQ will be set during this single or bursts receive operation. It is cleared by writing 1 to REC_FINCLR bit in the Control register. |
| 2 | SEND_FIN | RO | 1'b0 | 1: indicates that the Master has ended the receive operation. It means no more TX_REQ will be set during this single or burst send operation. It is cleared by writing 1 to SEND_FINCLR bit in the Control register. |
| 1 | TX_REQ | RO | 1'b0 | 1: indicates the Slave controller is addressed as transmitter and requires data from the host device. |


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 0 | RX_REQ | RO | 1'b0 | 1: indicates the Slave controller has received data from the Master. It is automatically cleared by reading of I2CS_DATA. |

**Table 14-4 - I2CS_STATUS – Status Register**

### 14.2.4 I2CS_DATA – Receive / Transmit Data Register (address offset: 0x02)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | DATA | RW | 8'h00 | When read, this is the data received in the last transaction.<br>When written, this is the data to be transmitted in the next transaction. |

**Table 14-5 - I2CS_DATA – Receive / Transmit Data Register**


Copyright © Bridgetek Pte Ltd

# 15 RTC

This is a Real Time Clock (RTC) running off a dedicated 32.768 kHz oscillator. It is powered by a 1.2V input (VBAT). If power is lost to VBAT, all RTC register values will reset to their default values. To maintain continuous power to VBAT refer to BRT_AN_004_FT930_RTC External Power Switch Circuit. This application note shows an example battery backup change-over circuit.

## 15.1 Register Summary

Listed below are the registers with their offset from the base address (0x10400). All registers can be accessed via Double-Word (32-bit) mode.

| Address Offset | Register | Default Value | Reference |
|---|---|---|---|
| | | | |
| **0x00** | RTC_TIME – Contains second, minute and hour | 0x0000 0000 | Section 16.2.1 |
| **0x04** | RTC_DATE – Contains day, date, month, and year | 0x0001 0101 | Section 16.2.2 |
| **0x08** | RTC_AALARM1 – Time-of-Day Auto Alarm 1 | 0x0000 0000 | Section 16.2.3 |
| **0x0C** | RTC_AALARM2 – Time-of-Day Auto Alarm 2 | 0x0000 0000 | Section 16.2.4 |
| **0x10** | RTC_CTRL – Control Register | 0x0000 0000 | Section 16.2.5 |
| **0x14** | RTC_STAT – Status Register | 0x0000 0000 | Section 16.2.6 |
| **0x18** | RTC_TRIM_VALUE – Trimming Register | 0x03e8 7FFF | Section 16.2.7 |
| **0x1C** | RTC_SLE_TIME – Sleep Time Register | 0x0000 0000 | Section 16.2.8 |
| **0x20** | RTC_SLE_DATE – Sleep Date Register | 0x0001 0101 | Section 16.2.9 |
| **0x24** | RTC_COMP_PER – Compare Period Register | 0x0000 0000 | Section 16.2.10 |
| **0x28** | RTC_VREF_TRIM – LVREF Trimming Value | 0x0000 0003 | Section 16.2.11 |

**Table 15-1 - Overview of RTC Registers**

## 15.2 Register Details

### 15.2.1 RTC_TIME – Time Register (address offset: 0x00)

<table>
<tr><th>Byte\Bit</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th><th>Function</th><th>Range</th></tr>
<tr><td>0</td><td>0</td><td colspan="3">10 Seconds</td><td colspan="4">Seconds</td><td>Seconds</td><td>00 – 59</td></tr>
<tr><td>1</td><td>0</td><td colspan="3">10 Minutes</td><td colspan="4">Minutes</td><td>Minutes</td><td>00 – 59</td></tr>
<tr><td rowspan="2">2</td><td rowspan="2">0</td><td rowspan="2">1:12<br>0:24</td><td>0: AM<br>1: PM</td><td rowspan="2">10 Hours</td><td colspan="4" rowspan="2">Hours</td><td rowspan="2">Hours</td><td>01 – 12<br>+AM/PM</td></tr>
<tr><td>20 Hours</td><td>00 – 23</td></tr>
<tr><td>3</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>Unused</td><td></td></tr>
</table>

**Table 15-2 - RTC_TIME - Time Register**

- Address : 0x00
- Type : RW
- Reset Value : 0x0000 0000
- Description : Contains the Second, Minute, Hour counters.


Copyright © Bridgetek Pte Ltd

### 15.2.2 RTC_DATE – Date Register (address offset: 0x04)

| Byte\Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Function | Range |
|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 0 | 0 | 0 | 0 | 0 | Day | | | Day | 1 – 7 |
| **1** | 0 | 0 | 10 Date | | Date | | | | Date | 01 – 31 |
| **2** | Century | 0 | 0 | 10 Month | Month | | | | Month | 01 – 12 + Century |
| **3** | 10 Year | | | | Year | | | | Year | 00 – 99 |

**Table 15-3 - RTC_DATE - Date Register**

- Address : 0x04
- Type : RW
- Reset Value : 0x0001 0101
- Description : Contains Day, Date, Month, Year counters.

### 15.2.3 RTC_AALARM1 – Auto Alarm 1 Register (address offset: 0x08)

| Byte\Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Function | Range |
|---|---|---|---|---|---|---|---|---|---|---|
| **0** | A1M1 | 10 Seconds | | | Seconds | | | | Seconds | 00 – 59 |
| **1** | A1M2 | 10 Minutes | | | Minutes | | | | Minutes | 00 – 59 |
| **2** | A1M3 | 1:12<br>0:24 | 0:AM<br>1:PM<br>20 Hours | 10 Hours | Hours | | | | Hours | 01 – 12 +AM.PM<br>00 – 23 |
| **3** | A1M4 | 0:Date<br>1: Day | 10 Date | | Day/Date | | | | Date<br>Day | 01-07<br>01-31 |

**Table 15-4 - RTC_AALARM1 – Auto Alarm 1 Register**

- Address : 0x08
- Type : RW
- Reset Value : 0x0000 0000
- Description : When counters match the configured values of this register, AALARM1 will be asserted.

| DY/!DT (Byte 3 Bit 6) | AALARM1 Register Mask (bit 7) | | | | Alarm Rate |
|---|---|---|---|---|---|
| | A1M4 | A1M3 | A1M2 | A1M1 | |
| X | 1 | 1 | 1 | 1 | Alarm once per second |
| X | 1 | 1 | 1 | 0 | Alarm when seconds match |
| X | 1 | 1 | 0 | 0 | Alarm when minutes and seconds match |
| X | 1 | 0 | 0 | 0 | Alarm when hours, minutes and seconds match |
| 0 | 0 | 0 | 0 | 0 | Alarm when date, hours, minutes and second match |
| 1 | 0 | 0 | 0 | 0 | Alarm when day, hours, minutes and second match |

**Table 15-5 - Alarm 1 Mask Bits**

Note: Configurations not listed in this table result in illogical operation.


Copyright © Bridgetek Pte Ltd

### 15.2.4 RTC_AALARM2 – Auto Alarm 2 Register (address offset: 0x0C)

| Byte\Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Function | Range |
|---|---|---|---|---|---|---|---|---|---|---|
| **0** | A2M1 | 10 Seconds | | | Seconds | | | | Seconds | 00 – 59 |
| **1** | A2M2 | 10 Minutes | | | Minutes | | | | Minutes | 00 – 59 |
| **2** | A2M3 | 1:12<br>0:24 | 0:AM<br>1:PM<br>20 Hours | 10 Hours | Hours | | | | Hours | 01 – 12<br>+AM.PM<br><br>00 – 23 |
| **3** | A2M4 | 0:Date<br>1: Day | 10 Date | | Day/Date | | | | Date<br>Day | 01-07<br>01-31 |

**Table 15-6 - RTC_AALARM2 – Auto Alarm 2 Register**

- Address : 0x0C
- Type : RW
- Reset Value : 0x0000 0000
- Description : When counters match the configured values of this register, AALARM1 will be asserted.

| DY/!DT (Byte 3 Bit 6) | AALARM1 Register Mask (bit 7) | | | | Alarm Rate |
|---|---|---|---|---|---|
| | A2M4 | A2M3 | A2M2 | A2M1 | |
| X | 1 | 1 | 1 | 1 | Alarm once per second |
| X | 1 | 1 | 1 | 0 | Alarm when seconds match |
| X | 1 | 1 | 0 | 0 | Alarm when minutes and seconds match |
| X | 1 | 0 | 0 | 0 | Alarm when hours, minutes and seconds match |
| 0 | 0 | 0 | 0 | 0 | Alarm when date, hours, minutes and second match |
| 1 | 0 | 0 | 0 | 0 | Alarm when day, hours, minutes and second match |

**Table 15-7 - Alarm 2 Mask Bits**

Note: Configurations not listed in this table result in illogical operation.

### 15.2.5 RTC_CTRL – Control Register (address offset: 0x10)

| Byte\Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **0** | RFEN | AUCAL | IOSIE<br>CATIM[1] | -<br>CATIM[0] | A2IE | A1IE | INTEN | RTCEN |
| **1** | Unused | | | | | | WRST/BS | MARFE |
| **2,3** | Unused | | | | | | | |

**Table 15-8 - RTC_CTRL – Control Register**

- Address : 0x10
- Type : RW
- Reset Value : 0x0000 0000
- Description : This register controls the operation of RTC when system power is turned on. In sleep mode, its value is still kept in the Alive module and will be recovered to the Wrapper when the system wakes up. It includes following bits:


Copyright © Bridgetek Pte Ltd

- **RTC_CTRL[0] – RTCEN:** RTC Enable bit. When this bit is disabled, INT and ALARMs will be gated to 0, and the RTC counters will be held.
- **RTC_CTRL[1] – INTEN:** Interrupt Enable bit. When this bit is disabled, no interrupt occurs.
- **RTC_CTRL[2] – A1IE:** Alarm 1 Interrupt Enable. ALARM 1 will cause an interrupt if this bit is enabled.
- **RTC_CTRL[3] – A2IE:** Alarm 2 Interrupt Enable. ALARM 2 will cause an interrupt if this bit is enabled.
- **RTC_CTRL[4] – CATIM[0]:** Calibration Time bit 0
- **RTC_CTRL[5] - IOSIE/CATIM[1]:** Internal Oscillator Stop Interrupt Enable. Auto Detection Oscillator Stop will cause an interrupt if this bit is enabled / Calibration Time bit 1
- **RTC_CTRL[6] – AUCAL:** Write 1 to do auto calibration. The calibration time depends on CATIM[1:0]:
  - 00 : Calibration in 1 second
  - 01 : Calibration in 1 minute
  - 10 : Calibration in 10 minutes
  - 11 : Calibration in 16.67 minutes (1000 seconds)
- **RTC_CTRL[7] – AURFE:** Auto Refresh Enable. When this bit is turned on, time and date counters from Alive will be updated to registers every second.
- **RTC_CTRL[8] – MARFE:** Manual Refresh Enable. Write 1 to this bit to update time and date counters from Alive to Wrapper once. This bit will be cleared automatically after updating. Poll to this bit to get the status of Manual Refresh function. 1: updating, 0: done
- **RTC_CTRL[9] – WRST/BS :**
  - Write 1 to this bit to start writing target registers.
  - Read this bit to check the status. 1: Busy, 0: Ready
  - This bit is auto cleared.

### 15.2.6 RTC_STAT - Status Register (address offset: 0x14)

| Byte\Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **0** | Unused | CALINT | IOSINT | | A2INT | A1INT | Unused | Unused |
| **1 ,2, 3** | Unused | | | | | | | |

**Table 15-9 - RTC_STAT - Status Register**

- Address : 0x14
- Type : RW1C, RO
- Reset Value : 0x0000 0000
- Description : This register contains the status and the interrupt events that cause the interrupts. These events are enabled by respective bits of Control Registers. The user must write 1 to clear these status bits.
  - **Stat[2] – A1INT**: Alarm 1 Interrupt. Set when timers match the value of AAlarm1 Register, an interrupt will be asserted if RtcCtrl[2] – A1IE is set.
  - **Stat[3] – A2INT**: Alarm 2 Interrupt. Set when timers match the value of AAlarm2 Register, an interrupt will be asserted if RtcCtrl[3] – A2IE is set.
  - **Stat[5] – IOSINT**: Internal Oscillator Stop Interrupt. Set when detecting two continuous values are the same after a defined period of time. If RtcCtrl[5] – IOSIE is enabled, an interrupt will be asserted.
  - **Stat[6] – CALINT**: Auto Calibration Interrupt. Set after calibration done.
  - **Stat[7] – BUSY**: Data is transferring, user can't write any data to RTC. This bit is read only.


Copyright © Bridgetek Pte Ltd

### 15.2.7 RTC_TRIM – Trimming Register (address offset: 0x18)

| Byte \Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **1, 0** | TrimVal[7:0] | | | | | | | |
| | TrimVal[15:8] | | | | | | | |
| **2, 3** | TrimTim[7:0] | | | | | | | |
| | TrimVal[15:8] | | | | | | | |

**Table 15-10 - RTC_TRIM – Trimming Value Register**

- Address : 0x18
- Type : RW
- Reset Value : 0x03E8 7FFF
- Description : This register contains trimming value and trimming period. After calibrating, the new values will be loaded automatically. The user also can write values to this register.

### 15.2.8 RTC_SLE_TIME – Sleep Time Register (address offset: 0x1C)

| Byte\Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Function | Range |
|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 0 | 10 Seconds | | | Seconds | | | | Seconds | 00 – 59 |
| **1** | 0 | 10 Minutes | | | Minutes | | | | Minutes | 00 – 59 |
| **2** | 0 | 1:12<br>0:24 | 0: AM<br>1: PM<br>20 Hours | 10<br>Hours | Hours | | | | Hours | 01 – 12<br>+ AM.PM<br>00 – 23 |
| **3** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Unused | |

**Table 15-11 - RTC_SLE_TIME – Sleep Time Register**

- Address : 0x1C
- Type : RW
- Reset Value : 0x0000 0000
- Description : Contains the Second, Minute, Hour values when system power is turned off.


Copyright © Bridgetek Pte Ltd

### 15.2.9 RTC_SLE_DATE – Sleep Date Register (address offset: 0x20)

| Byte\Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Function | Range |
|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 0 | 0 | 0 | 0 | 0 | Day | | | Day | 1 – 7 |
| **1** | 0 | 0 | 10 Date | | Date | | | | Date | 01 – 31 |
| **2** | Century | 0 | 0 | 10 Month | Month | | | | Month | 01 – 12 + Century |
| **3** | 10 Year | | | | Year | | | | Year | 00 – 99 |

**Table 15-12 - RTC_SLE_DATE - Sleep Date Register**

- Address : 0x20
- Type : RW
- Reset Value : 0x0001 0101
- Description : Contains Day, Date, Month, Year counters when system power is turned off.

### 15.2.10 RTC_COMP_PER – Compare Period Register (address offset: 0x24)

| Byte\Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Function |
|---|---|---|---|---|---|---|---|---|---|
| **0** | Unused | | | | | 0 | 0 | 0 | Auto detection is turned off |
| | | | | | | 0 | 0 | 1 | Auto detect per 100,000 * pclk_cycle_time |
| | | | | | | 0 | 1 | 0 | Auto detect per 1,000,000 * pclk_cycle_time |
| | | | | | | 0 | 1 | 1 | Auto detect per 10,000,000 * pclk_cycle_time |
| | | | | | | 1 | 0 | 0 | Auto detect per 100,000,000 * pclk_cycle_time |
| **1, 2, 3** | Reserved | | | | | | | | |

**Table 15-13 – RTC_COMP_PER - Compare Period Register**

- Address : 0x24
- Type : RW
- Reset Value : 0x0000 0000
- Description : This register contains the Oscillator Stop compare period which is automatically detected. Set the compare period within the range of 1ms ~ 2s based on the PCLK cycle time.

### 15.2.11 RTC_VREF_TRIM – LVREF Trimming Value (address offset: 0x28)

- Address : 0x28
- Type : RW
- Reset Value : 0x0000 0003
- Description : Allow the reference voltage to be accurately trimmed. Only 3 LSB bits are used.


Copyright © Bridgetek Pte Ltd

### 15.2.12 I2CS_FIFO_DATA - FIFO Data Register (address offset: 0x07)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | FIFO_DATA | RO | 8'h00 | FIFO Data Read from the RX Buffer |

**Table 15-14 - I2CS_FIFO_DATA - FIFO Data Register**

### 15.2.13 I2CS_TRIG - Trigger Register (address offset: 0x08)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | RX_OP | RW | 1'b0 | A write to this register triggers the FIFO mode operation. Set this bit to 1 for RX, and 0 for TX FIFO operations.<br>The operation will end when FIFO_BL expires. |
| 6:0 | Reserved | - | - | - |

**Table 15-15 - I2CS_TRIG - Trigger Register**

### 15.2.14 I2CS_FIFO_LEN – FIFO Mode Byte Length (address offset: 0x04)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | FIFO_BL | RW | 8'h00 | Number of bytes (FIFO_BL + 1) to transmit / receive when FIFO mode is enabled |

**Table 15-16 - I2CS_FIFO_LEN – FIFO Mode Byte Length**

### 15.2.15 I2CS_FIFO_INT_ENABLE – FIFO Mode Interrupt Enable (address offset: 0x05)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | DONE | RW | 1'b0 | FIFO_BL operation completed Interrupt |
| 6 | I2C_INT | RW | 1'b0 | I2C Interrupt |
| 5 | RX_FULL | RW | 1'b0 | RX FIFO Full interrupt enable |
| 4 | RX_HALF | RW | 1'b0 | RX FIFO Half Full interrupt enable |
| 3 | RX_EMPTY | RW | 1'b0 | RX FIFO Empty interrupt enable |
| 2 | TX_FULL | RW | 1'b0 | TX FIFO Full interrupt enable |
| 1 | TX_HALF | RW | 1'b0 | TX FIFO Half Full interrupt enable |
| 0 | TX_EMPTY | RW | 1'b0 | TX FIFO Empty interrupt enable |

**Table 15-17 - I2CS_FIFO_INT_ENABLE – FIFO Mode Interrupt Enable**


Copyright © Bridgetek Pte Ltd

### 15.2.16 I2CS_FIFO_INT_PEND – FIFO Mode Interrupt Pending (address offset: 0x06)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | DONE | RW1C | 1'b0 | FIFO_BL operation complete interrupt pending |
| 6 | I2C_INT | RW1C | 1'b0 | I2C Interrupt pending |
| 5 | RX_FULL | RW1C | 1'b0 | RX FIFO Full interrupt pending |
| 4 | RX_HALF | RW1C | 1'b0 | RX FIFO Half Full interrupt pending |
| 3 | RX_EMPTY | RW1C | 1'b0 | RX FIFO Empty interrupt pending |
| 2 | TX_FULL | RW1C | 1'b0 | TX FIFO Full interrupt pending |
| 1 | TX_HALF | RW1C | 1'b0 | TX FIFO Half Full interrupt pending |
| 0 | TX_EMPTY | RW1C | 1'b0 | TX FIFO Empty interrupt pending |

**Table 15-18 - I2CS_FIFO_INT_PEND – FIFO Mode Interrupt Pending**


Copyright © Bridgetek Pte Ltd

# 16 PWM

The device supports 8 separate independent PWM channels. All channels share an 8-bit prescaler to scale the system clock frequency to the desired channels.

Each channel has its own 16-bit comparator value. This is the value that would be matched to a preset 16-bit counter. When a channel's 16-bit comparator value matches that of the 16-bit counter, the corresponding PWM channel output will toggle. This 16-bit comparator value will continue to count until it reaches its preset value, and the counter will just roll over.

A special feature allows the 8 channels each to also toggle its own output based on the comparison results of other channels. Hence each channel potentially can have up to 8 toggle edges.

The PWM can be generated as a multi-shot or continuously. When defined as a multi-shot, an interrupt may be generated at the end of the PWM production.

Channels 0 and 1 can double as a stereo 11 kHz or 22 kHz PWM audio channel. Once it's set up, the 16-bit or 8-bit PWM audio data can be downloaded to the PWM's local FIFO which can hold up to 64 stereo or 128 mono audio data. The data will be played back based on the prescaler and 16-bit counter and the data will be automatically scaled to fit the playback period if necessary.

The FIFO can generate a number of interrupts for FIFO management. They are the FIFO full, empty, half-empty, overflow and underflow. Each of these interrupts can be individually masked if required.

## 16.1 Register Summary

Listed below are the registers with their offset from the base address (0x105B0). All registers can only be accessed via Byte (8-bit) mode but the FIFO can only be accessed via Word (16-bit) mode.

| Address Offset | Register | Default Value | Reference |
|---|---|---|---|
| | | | |
| **0x00** | PWM_CTRL0 - PCM Control Register | 0x00 | Section 16.2.1 |
| **0x01** | PWM_CTRL1 - PWM Control Register | 0x00 | Section 16.2.2 |
| **0x02** | PWM_PRESCALER - PWM Prescaler Register | 0x00 | Section 16.2.3 |
| **0x03** | PWM_CNTL - PWM Counter Register (LSB) | 0x00 | Section 16.2.4 |
| **0x04** | PWM_CNTH - PWM Counter Register (MSB) | 0x00 | Section 16.2.5 |
| **0x05** | PWM_CMP0L - Comparator 0 Value Register (LSB) | 0x00 | Section 16.2.6 |
| **0x06** | PWM_CMP0H - Comparator 0 Value Register (MSB) | 0x00 | Section 16.2.7 |
| **0x07** | PWM_CMP1L - Comparator 1Value Register (LSB) | 0x00 | Section 16.2.8 |
| **0x08** | PWM_CMP1H - Comparator 1 Value Register (MSB) | 0x00 | Section 16.2.9 |
| **0x09** | PWM_CMP2L - Comparator 2 Value Register (LSB) | 0x00 | Section 16.2.10 |
| **0x0A** | PWM_CMP2H - Comparator 2 Value Register (MSB) | 0x00 | Section 16.2.11 |
| **0x0B** | PWM_CMP3L - Comparator 3 Value Register (LSB) | 0x00 | Section 16.2.12 |
| **0x0C** | PWM_CMP3H - Comparator 3 Value Register (MSB) | 0x00 | Section 16.2.13 |
| **0x0D** | PWM_CMP4L - Comparator 4 Value Register (LSB) | 0x00 | Section 16.2.14 |
| **0x0E** | PWM_CMP4H - Comparator 4 Value Register (MSB) | 0x00 | Section 16.2.15 |
| **0x0F** | PWM_CMP5L - Comparator 5 Value Register (LSB) | 0x00 | Section 16.2.16 |


Copyright © Bridgetek Pte Ltd

| | | | |
|---|---|---|---|
| **0x10** | PWM_CMP5H - Comparator 5 Value Register (MSB) | 0x00 | Section 16.2.17 |
| **0x11** | PWM_CMP6L - Comparator 6 Value Register (LSB) | 0x00 | Section 16.2.18 |
| **0x12** | PWM_CMP6H - Comparator 6 Value Register (MSB) | 0x00 | Section 16.2.19 |
| **0x13** | PWM_CMP7L - Comparator 7 Value Register (LSB) | 0x00 | Section 16.2.20 |
| **0x14** | PWM_CMP7H - Comparator 7 Value Register (MSB) | 0x00 | Section 16.2.21 |
| **0x15** | PWM_TOGGLE0 - Channel 0 OUT Toggle Comparator Mask Register | 0x00 | Section 16.2.22 |
| **0x16** | PWM_TOGGLE1 - Channel 1 OUT Toggle Comparator Mask Register | 0x00 | Section 16.2.23 |
| **0x17** | PWM_TOGGLE2 - Channel 2 OUT Toggle Comparator Mask Register | 0x00 | Section 16.2.24 |
| **0x18** | PWM_TOGGLE3 - Channel 3 OUT Toggle Comparator Mask Register | 0x00 | Section 16.2.25 |
| **0x19** | PWM_TOGGLE4 - Channel 4 OUT Toggle Comparator Mask Register | 0x00 | Section 16.2.26 |
| **0x1A** | PWM_TOGGLE5 - Channel 5 OUT Toggle Comparator Mask Register | 0x00 | Section 16.2.27 |
| **0x1B** | PWM_TOGGLE6 - Channel 6 OUT Toggle Comparator Mask Register | 0x00 | Section 16.2.28 |
| **0x1C** | PWM_TOGGLE7 - Channel 7 OUT Toggle Comparator Mask Register | 0x00 | Section 16.2.29 |
| **0x1D** | PWM_OUT_CLR_EN - PWM OUT Clear Enable Register | 0x00 | Section 16.2.30 |
| **0x1E** | PWM_CTRL_BL_CMP8 - Control Block CMP8 Value Register | 0x00 | Section 16.2.31 |
| **0x1F** | PWM_INIT - PWM Initialization Register | 0x00 | Section 16.2.32 |
| **0x20** | PWM_INTMASK - PWM Interrupt Mask Register | 0x00 | Section 16.2.33 |
| **0x21** | PWM_INTSTATUS - PWM Interrupt Status Register | 0x20 | Section 16.2.34 |
| **0x22** | PWM_SAMPLE_FREQ_H - PWM Data Sampling Frequency High Byte Register | 0x56 | Section 16.2.35 |
| **0x23** | PWM_SAMPLE_FREQ_L - PWM Data Sampling Frequency Low Byte Register | 0x22 | Section 16.2.36 |
| **0x24** | PCM_VOLUME - PCM Volume Register | 0x00 | Section 16.2.37 |
| **0x3C** | PWM_BUFFER - PCM Buffer Register | 0xXXXX | Section 16.2.38 |

**Table 16-1 - Overview of PWM Registers**

## 16.2 Register Details

### 16.2.1 PWM_CTRL0 - PCM Control Register (address offset: 0x00)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | PWM_CH01_AUTO | RW | 1'b0 | Set to 1 to use channels 0 & 1 for audio playback via the internal FIFO. |
| 6 | PWM_CH01_MONO | RW | 1'b0 | Set to 1 for mono audio. Both channels will playback the same data. |
| 5 | PWM_CH01_8BIT | RW | 1'b0 | Set to 1 for 8-bit audio data. |


Copyright © Bridgetek Pte Ltd

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 4 | PWM_CH01_SCALE | RW | 1'b0 | Set to 1 if automatic scaling of data is required. |
| 3 | PWM_CH01_FILTER | RW | 1'b0 | Set to 1 if PCM filter is required. This is valid only with PWM_CH01_AUTO and PCM_EN set. |
| 2 | PCM_EN | RW | 1'b0 | Set to 1 if channel 0 and channel 1 are used for PCM playback. |
| 1 | PWM_DEV_EN | RW | 1'b0 | Set to 1 to enable PWM. This bit is not really used. |
| 0 | PWM_SOFT_RESET | RW | 1'b0 | Set to 1 to reset PWM. |

**Table 16-2 - PWM_CTRL0 - PCM Control Register**

### 16.2.2 PWM_CTRL1 - PWM Control Register (address offset: 0x01)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | Reserved | - | - | - |
| 6 | PCM_BIYTEREVERSE | RW | 1'b0 | 0: Data is treated as in big endian format<br>For 8-bit, this has no effect.<br>1: Data is treated as in little endian format<br>For 8-bit, this has no effect.<br>For 16-bit, the lower and upper bytes are swapped. |
| 5 | PWM_INT | RW | 1'b0 | PWM Interrupt |
| 4 | PWM_INT_MASK | RW | 1'b0 | Interrupt mask bit for PWM_INT |
| 2:1 | PWM_TRIGGER_EN | RW | 2'h0 | PWM trigger enable:<br>00 Disabled<br>01 Positive Edge<br>10 Negative Edge<br>11 Any Edge |
| 0 | PWM_EN | RW | 1'b0 | Set to 1 to enable PWM. |

**Table 16-3 - PWM_CTRL1 - PWM Control Register**

### 16.2.3 PWM_PRESCALER - PWM Prescaler Register (address offset: 0x02)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | PRESCALER | RW | 8'h00 | 8-bit Prescaler value. |

**Table 16-4 - PWM_PRESCALER - PWM Prescaler Register**


Copyright © Bridgetek Pte Ltd

### 16.2.4 PWM_CNTL - PWM Counter Register (LSB) (address offset: 0x03)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | CNT16_LSB | RW | 8'h00 | 16-bit counter LSB. |

**Table 16-5 - PWM_CNTL - PWM Counter Register (LSB)**

### 16.2.5 PWM_CNTH - PWM Counter Register (MSB) (address offset: 0x04)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | CNT16_MSB | RW | 8'h00 | 16-bit counter MSB. |

**Table 16-6 - PWM_CNTH - PWM Counter Register (MSB)**

### 16.2.6 PWM_CMP0L - Comparator 0 Value Register (LSB) (address offset: 0x05)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | CMP16_0_LSB | RW | 8'h00 | LSB of comparator 0 16-bit value. |

**Table 16-7 - PWM_CMP0L - Comparator 0 Value Register (LSB)**

### 16.2.7 PWM_CMP0H - Comparator 0 Value Register (MSB) (address offset: 0x06)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | CMP16_0_MSB | RW | 8'h00 | MSB of comparator 0 16-bit value. |

**Table 16-8 - PWM_CMP0H - Comparator 0 Value Register (MSB)**

### 16.2.8 PWM_CMP1L - Comparator 1 Value Register (LSB) (address offset: 0x07)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | CMP16_1_LSB | RW | 8'h00 | LSB of comparator 1 16-bit value. |

**Table 16-9 - PWM_CMP1L - Comparator 1 Value Register (LSB)**

### 16.2.9 PWM_CMP1H - Comparator 1 Value Register (MSB) (address offset: 0x08)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | CMP16_1_MSB | RW | 8'h00 | MSB of comparator 1 16-bit value. |

**Table 16-10 - PWM_CMP1H - Comparator 1 Value Register (MSB)**


Copyright © Bridgetek Pte Ltd

### 16.2.10 PWM_CMP2L - Comparator 2 Value Register (LSB) (address offset: 0x09)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | CMP16_2_LSB | RW | 8'h00 | LSB of comparator 2 16-bit value. |

**Table 16-11 - PWM_CMP2L - Comparator 2 Value Register (LSB)**

### 16.2.11 PWM_CMP2H - Comparator 2 Value Register (MSB) (address offset: 0x0A)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | CMP16_2_MSB | RW | 8'h00 | MSB of comparator 2 16-bit value. |

**Table 16-12 - PWM_CMP2H - Comparator 2 Value Register (MSB)**

### 16.2.12 PWM_CMP3L - Comparator 3 Value Register (LSB) (address offset: 0x0B)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | CMP16_3_LSB | RW | 8'h00 | LSB of comparator 3 16-bit value. |

**Table 16-13 - PWM_CMP3L - Comparator 3 Value Register (LSB)**

### 16.2.13 PWM_CMP3H - Comparator 3 Value Register (MSB) (address offset: 0x0C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | CMP16_3_MSB | RW | 8'h00 | MSB of comparator 3 16-bit value. |

**Table 16-14 - PWM_CMP3H - Comparator 3 Value Register (MSB)**

### 16.2.14 PWM_CMP4L - Comparator 4 Value Register (LSB) (address offset: 0x0D)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | CMP16_4_LSB | RW | 8'h00 | LSB of comparator 4 16-bit value. |

**Table 16-15 - PWM_CMP4L - Comparator 4 Value Register (LSB)**

### 16.2.15 PWM_CMP4H - Comparator 4 Value Register (MSB) (address offset: 0x0E)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | CMP16_4_MSB | RW | 8'h00 | MSB of comparator 4 16-bit value. |

**Table 16-16 - PWM_CMP4H - Comparator 4 Value Register (MSB)**


Copyright © Bridgetek Pte Ltd

### 16.2.16 PWM_CMP5L - Comparator 5 Value Register (LSB) (address offset: 0x0F)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | CMP16_5_LSB | RW | 8'h00 | LSB of comparator 5 16-bit value. |

**Table 16-17 - PWM_CMP5L - Comparator 5 Value Register (LSB)**

### 16.2.17 PWM_CMP5H - Comparator 5 Value Register (MSB) (address offset: 0x10)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | CMP16_5_MSB | RW | 8'h00 | MSB of comparator 5 16-bit value. |

**Table 16-18 - PWM_CMP5H - Comparator 5 Value Register (MSB)**

### 16.2.18 PWM_CMP6L - Comparator 6 Value Register (LSB) (address offset: 0x11)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | CMP16_6_LSB | RW | 8'h00 | LSB of comparator 6 16-bit value. |

**Table 16-19 - PWM_CMP6L - Comparator 6 Value Register (LSB)**

### 16.2.19 PWM_CMP6H - Comparator 6 Value Register (MSB) (address offset: 0x12)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | CMP16_6_MSB | RW | 8'h00 | MSB of comparator 6 16-bit value. |

**Table 16-20 - PWM_CMP6H - Comparator 6 Value Register (MSB)**

### 16.2.20 PWM_CMP7L - Comparator 7 Value Register (LSB) (address offset: 0x13)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | CMP16_7_LSB | RW | 8'h00 | LSB of comparator 7 16-bit value. |

**Table 16-21 - PWM_CMP7L - Comparator 7 Value Register (LSB)**


Copyright © Bridgetek Pte Ltd

### 16.2.21 PWM_CMP7H - Comparator 7 Value Register (MSB) (address offset: 0x14)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | CMP16_7_MSB | RW | 8'h00 | MSB of comparator 7 16-bit value. |

**Table 16-22 - PWM_CMP7H - Comparator 7 Value Register (MSB)**

### 16.2.22 PWM_TOGGLE0 - Channel 0 OUT Toggle Comparator Mask Register (address offset: 0x15)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | TOGGLE_EN_0 | RW | 8'h00 | Channel 0 PWM OUT toggle comparator mask (each bit corresponds to 1 comparator) |

**Table 16-23 - PWM_TOGGLE0 - Channel 0 OUT Toggle Comparator Mask**

### 16.2.23 PWM_TOGGLE1 - Channel 1 OUT Toggle Comparator Mask Register (address offset: 0x16)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | TOGGLE_EN_1 | RW | 8'h00 | Channel 1 PWM OUT toggle comparator mask (each bit corresponds to 1 comparator) |

**Table 16-24 - PWM_TOGGLE1 - Channel 1 OUT Toggle Comparator Mask**

### 16.2.24 PWM_TOGGLE2 - Channel 2 OUT Toggle Comparator Mask Register (address offset: 0x17)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | TOGGLE_EN_2 | RW | 8'h00 | Channel 2 PWM OUT toggle comparator mask (each bit corresponds to 1 comparator) |

**Table 16-25 - PWM_TOGGLE2 - Channel 2 OUT Toggle Comparator Mask Register**

### 16.2.25 PWM_TOGGLE3 - Channel 3 OUT Toggle Comparator Mask Register (address offset: 0x18)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | TOGGLE_EN_3 | RW | 8'h00 | Channel 3 PWM OUT toggle comparator mask (each bit corresponds to 1 comparator) |

**Table 16-26 - PWM_TOGGLE3 - Channel 3 OUT Toggle Comparator Mask Register**

### 16.2.26 PWM_TOGGLE4 - Channel 4 OUT Toggle Comparator Mask Register (address offset: 0x19)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | TOGGLE_EN_4 | RW | 8'h00 | Channel 4 PWM OUT toggle comparator mask (each bit corresponds to 1 comparator) |

**Table 16-27 - PWM_TOGGLE4 - Channel 4 OUT Toggle Comparator Mask Register**


Copyright © Bridgetek Pte Ltd

### 16.2.27 PWM_TOGGLE5 - Channel 5 OUT Toggle Comparator Mask Register (address offset: 0x1A)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | TOGGLE_EN_5 | RW | 8'h00 | Channel 5 PWM OUT toggle comparator mask (each bit corresponds to 1 comparator) |

**Table 16-28 - PWM_TOGGLE5 - Channel 5 OUT Toggle Comparator Mask Register**

### 16.2.28 PWM_TOGGLE6 - Channel 6 OUT Toggle Comparator Mask Register (address offset: 0x1B)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | TOGGLE_EN_6 | RW | 8'h00 | Channel 6 PWM OUT toggle comparator mask (each bit corresponds to 1 comparator) |

**Table 16-29 - PWM_TOGGLE6 - Channel 6 OUT Toggle Comparator Mask Register**

### 16.2.29 PWM_TOGGLE7 - Channel 7 OUT Toggle Comparator Mask Register (address offset: 0x1C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | TOGGLE_EN_7 | RW | 8'h00 | Channel 7 PWM OUT toggle comparator mask (each bit corresponds to 1 comparator) |

**Table 16-30 - PWM_TOGGLE7 - Channel 7 OUT Toggle Comparator Mask Register**

### 16.2.30 PWM_OUT_CLR_EN - PWM OUT Clear Enable Register (address offset: 0x1D)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | CLR_EN | RW | 8'h00 | PWM Out clear enable |

**Table 16-31 - PWM_OUT_CLR_EN - PWM OUT Clear Enable Register**

### 16.2.31 PWM_CTRL_BL_CMP8 - Control Block CMP8 Value Register (address offset: 0x1E)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | CTRL_BL_CMP8 | RW | 8'h00 | Control block CMP8 value.<br>0: continuous<br>1: one-shot, 2 - 255 |

**Table 16-32 - PWM_CTRL_BL_CMP8 - Control Block CMP8 Value Register**

### 16.2.32 PWM_INIT - PWM Initialization Register (address offset: 0x1F)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | INIT | RW | 8'h00 | PWM Initialisation register |

**Table 16-33 - PWM_INIT - PWM Initialization Register**


Copyright © Bridgetek Pte Ltd

### 16.2.33 PWM_INTMASK - PWM Interrupt Mask Register (address offset: 0x20)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:6 | Reserved | - | - | - |
| 5 | FIFO_EMPTY_MASK | RW | 1'b0 | FIFO empty interrupt mask |
| 4 | FIFO_FULL_MASK | RW | 1'b0 | FIFO full interrupt mask |
| 3 | FIFO_HALF_MASK | RW | 1'b0 | FIFO half full interrupt mask |
| 2 | FIFO_OV_MASK | RW | 1'b0 | FIFO overflow interrupt mask |
| 1 | FIFO_UNDER_MASK | RW | 1'b0 | FIFO underflow interrupt mask |
| 0 | PWM_INT_MASK | RW | 1'b0 | PWM_INT interrupt mask (same as bit[4] of PWM_CTRL) |

**Table 16-34 - PWM_INTMASK - PWM Interrupt Mask Register**

### 16.2.34 PWM_INTSTATUS - PWM Interrupt Status Register (address offset: 0x21)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:6 | Reserved | - | - | - |
| 5 | FIFO_EMPTY_INT | RW1C | 1'b1 | FIFO empty interrupt; write 1 to clear. |
| 4 | FIFO_FULL_INT | RW1C | 1'b0 | FIFO full interrupt; write 1 to clear. |
| 3 | FIFO_HALF_INT | RW1C | 1'b0 | FIFO half full interrupt; write 1 to clear. |
| 2 | FIFO_OV_INT | RW1C | 1'b0 | FIFO overflow interrupt; write 1 to clear. |
| 1 | FIFO_UNDER_INT | RW1C | 1'b0 | FIFO underflow interrupt; write 1 to clear. |
| 0 | PWM_INT | RW1C | 1'b0 | PWM interrupt; write 1 to clear. (Same as bit[5] of PWM_CTRL1) |

**Table 16-35 - PWM_INTSTATUS - PWM Interrupt Status Register**

### 16.2.35 PWM_SAMPLE_FREQ_H - PWM Data Sampling Frequency High Byte Register (address offset: 0x22)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | PWM_SAMPLE_FREQ_H | RW | 8'h56 | PWM Data Sampling Frequency High Byte |

**Table 16-36 - PWM_SAMPLE_FREQ_H - PWM Data Sampling Frequency High Byte Register**


Copyright © Bridgetek Pte Ltd

### 16.2.36 PWM_SAMPLE_FREQ_L - PWM Data Sampling Frequency Low Byte Register (address offset: 0x23)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | PWM_SAMPLE_FREQ_L | RW | 8'h22 | PWM Data Sampling Frequency Low Byte |

**Table 16-37 - PWM_SAMPLE_FREQ_L - PWM Data Sampling Frequency Low Byte Register**

### 16.2.37 PCM_VOLUME - PCM Volume Register (address offset: 0x24)

| Bit | Name | Type | Default Value | Description | |
|---|---|---|---|---|---|
| 7:5 | Reserved | - | - | - | |
| 4:0 | Volume Control | RW | 5'h00 | 0x00 | Mute |
| | | | | 0x01 | ~6.25% |
| | | | | 0x02 | ~12.5% |
| | | | | 0x03 | ~19% |
| | | | | 0x04 | ~25% |
| | | | | 0x05 | ~31% |
| | | | | 0x06 | ~37% |
| | | | | 0x07 | ~44% |
| | | | | 0x08 | ~50% |
| | | | | 0x09 | ~56% |
| | | | | 0x0A | ~63% |
| | | | | 0x0B | ~69% |
| | | | | 0x0C | ~75% |
| | | | | 0x0D | ~81% |
| | | | | 0x0E | ~88% |
| | | | | 0x0F | ~94% |
| | | | | 0x10 | ~100% |
| | | | | 0x11-0x1F | Illegal; forced to 0x10 |

**Table 16-38 - PCM_VOLUME - PCM Volume Register**

### 16.2.38 PWM_BUFFER - PCM Buffer Register (address offset: 0x3C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 15:0 | Buffer Data | WO | - | The entry point to the FIFO. It must be written in 16-bit. For 8-bit data, the upper 8-bit will be ignored. |

**Table 16-39 - PWM_BUFFER - PCM Buffer Register**


Copyright © Bridgetek Pte Ltd

# 17 Flash Controller

There are two ways to access the memory control unit. One is via the CPU I/O interface and the other via the one-wire debugger interface. The CPU I/O interface is described here.

From the CPU I/O interface, memory transfer can occur between the Flash and the Program Memory, and between the Flash and the Data Memory. A number of the serial Flash commands are also supported, and the CPU may issue these commands to the Flash.

As this is a shared resource between the debugger interface and CPU I/O interface, the interface must acquire the resource first before performing any of the activities. This is done by reading a test-and-set semaphore. If the interface reads a 0 from the semaphore it can safely assume it has acquired the resource. Any further read to this semaphore will return a 1 until it is released by the interface that acquired the resource.

If both the debugger and CPU attempt to acquire the resource while it is free at exactly the same time, priority is given to the debugger and the CPU interface will read a 1 instead.

Until the semaphore is acquired, the semaphore is the only register any of the interfaces can read. Once the semaphore is acquired by the interface, the interface will have full access to all the registers in the control unit.

If an interface that does not have the resource writes to the control unit registers, they will be ignored while a read from any of the control unit registers will always return 0.

## 17.1 Register Summary

Listed below are the registers with their offset from the base address (0x10800). All registers can only be accessed via Byte (8-bit) mode.

| Address Offset | Register | Default Value | Reference |
|---|---|---|---|
| | | | |
| **0x00** | RSADDR0 – Memory Start Address Register (LSB) | 0x00 | Section 23.2.1 |
| **0x01** | RSADDR1 – Memory Start Address Register (Byte 1) | 0x00 | Section 23.2.2 |
| **0x02** | RSADDR2 – Memory Start Address Register (MSB) | 0x00 | Section 23.2.3 |
| **0x03** | FSADDR0 – Flash Start Address Register (LSB) | 0x00 | Section 23.2.4 |
| **0x04** | FSADDR1 – Flash Start Address Register (Byte 1) | 0x00 | Section 23.2.5 |
| **0x05** | FSADDR2 – Flash Start Address Register (MSB) | 0x00 | Section 23.2.6 |
| **0x06** | BLENGTH0 – Data Byte Length Register (LSB) | 0x00 | Section 23.2.7 |
| **0x07** | BLENGTH1 – Data Byte Length Register (Byte 1) | 0x00 | Section 23.2.8 |
| **0x08** | BLENGTH2 – Data Byte Length Register (MSB) | 0x00 | Section 23.2.9 |
| **0x09** | COMMAND – Command Register | 0x00 | Section 23.2.1 |
| **0x0A** | Reserved | - | |
| **0x0B** | SEMAPHORE – Semaphore Register | 0x00 | Section 23.2.11 |
| **0x0C** | CONFIG – Configuration Register | 0x00 | Section 23.2.12 |
| **0x0D** | STATUS – Status Register | 0x00 | Section 23.2.13 |
| **0x0E** | CRCL – Flash Content CRC Register (LSB) | 0xXX | Section 23.2.14 |


Copyright © Bridgetek Pte Ltd

| | | | |
|---|---|---|---|
| **0x0F** | CRCH – Flash Content CRC Register (MSB) | 0xXX | Section 23.2.15 |
| **0x7C** | CHIPID0 – Chip ID Register (LSB) (only via one-wire debugger) | 0xXX | Section 23.2.16 |
| **0x7D** | CHIPID1 – Chip ID Register (Byte 1) (only via one-wire debugger) | 0xXX | Section 23.2.17 |
| **0x7E** | CHIPID2 – Chip ID Register (Byte 2) (only via one-wire debugger) | 0xXX | Section 23.2.18 |
| **0x7F** | CHIPID3 – Chip ID Register (MSB) (only via one-wire debugger) | 0xXX | Section 23.2.19 |
| **0x80** | DRWDATA – Data Register | 0x00 | Section 23.2.20 |

**Table 17-1 - Overview of Data Capture Interface Registers**

## 17.2 Register Details

### 17.2.1 RSADDR0 – Memory Start Address Register (LSB) (address offset: 0x00)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | - | WO | 0x00 | LSB of the start address of the memory location to perform read/write for either program or data memory |

**Table 17-2 - RSADDR1 – Memory Start Address Register (LSB)**

### 17.2.2 RSADDR1 – Memory Start Address Register (Byte 1) (address offset: 0x01)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | - | WO | 0x00 | Byte 1 of the start address of the memory location to perform read/write for either program or data memory |

**Table 17-3 - RSADDR1 – Memory Start Address Register (Byte 1)**

### 17.2.3 RSADDR2 – Memory Start Address Register (MSB) (address offset: 0x02)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | - | WO | 0x00 | MSB of the start address of the memory location to perform read/write for either program or data memory |

**Table 17-4 - RSADDR2 – Memory Start Address Register (MSB)**

***Note:*** *The memory start address must always be aligned to a 4-byte boundary in both read/write cases. The address is treated as a Double-Word address (e.g., 01 is byte address 4).*


Copyright © Bridgetek Pte Ltd

### 17.2.4 FSADDR0 – Flash Start Address Register (LSB) (address offset: 0x03)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | - | WO | 0x00 | LSB of the start address of the flash location to perform read/write |

**Table 17-5 - FSADDR0 – Flash Start Address Register (LSB)**

### 17.2.5 FSADDR1 – Flash Start Address Register (Byte 1) (address offset: 0x04)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | - | WO | 0x00 | Byte 1 of the start address of the flash location to perform read/write |

**Table 17-6 - FSADDR1 – Flash Start Address Register (Byte 1)**

### 17.2.6 FSADDR2 – Flash Start Address Register (MSB) (address offset: 0x05)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | - | WO | 0x00 | MSB of the start address of the flash location to perform read/write |

**Table 17-7 - FSADDR2 – Flash Start Address Register (MSB)**

***Note:*** *The flash start address must always be aligned to 256-byte boundary in the write case, and can be any value in the read case. The address is treated as a byte address (e.g., 01 is byte address 1).*

### 17.2.7 BLENGTH0 – Data Byte Length Register (LSB) (address offset: 0x06)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | - | WO | 0x00 | LSB of the Byte Length to transfer |

**Table 17-8 - BLENGTH0 – Data Byte Length Register (LSB)**

### 17.2.8 BLENGTH1 – Data Byte Length Register (Byte 1) (address offset: 0x07)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | - | WO | 0x00 | Byte 1 of the Byte Length to transfer |

**Table 17-9 - BLENGTH1 – Data Byte Length Register (Byte 1)**

### 17.2.9 BLENGTH2 – Data Byte Length Register (MSB) (address offset: 0x08)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | - | WO | 0x00 | MSB of the Byte Length to transfer |

**Table 17-10 - BLENGTH2 – Data Byte Length Register (MSB)**


Copyright © Bridgetek Pte Ltd

***Note 1:*** *When the flash is the destination (write case), the byte length must be a multiple of 256 bytes (1 page of flash entry). There is no such restriction on byte length if the flash is the source (read case).*

***Note 2:*** *The registers must be set to (byte length - 1). For example, if the byte length is 256, then BLENGTH0 is set to 255 and BLENGTH1 and BLENGTH2 are set to 0.*

### 17.2.10 COMMAND – Command Register (address offset: 0x09)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | - | WO | 0x00 | Command to perform. See section 23.3. |

**Table 17-11 - COMMAND – Command Register**

### 17.2.11 SEMAPHORE – Semaphore Register (address offset: 0x0B)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | - | RW | 0x00 | Semaphore value<br>If a 0 is read, then the control unit's resource is allocated to whichever that does the test. This register is then automatically set to 1.<br>If a 1 is read, then the resource is being used and not free.<br>This semaphore can be released by only the interface which acquires the semaphore with writing a 1 to this register.<br>All other registers in this control unit are not readable or writeable until this semaphore is acquired.<br>If both the CPU and Debugger test this semaphore register at the same time when it is free, priority is given to the Debugger and the CPU will read a 1 instead. |

**Table 17-12 - SEMAPHORE – Semaphore Register**

### 17.2.12 CONFIG – Configuration Register (address offset: 0x0C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:2 | Reserved | - | - | Reserved bits |
| 1:0 | SPI_CLK | WO | 2'b0 | The serial SPI clock speed to the serial flash<br>0x0: Flash SPI clock speed = 1/2 System clock speed<br>0x1: Flash SPI clock speed = 1/3 System clock speed<br>0x2: Flash SPI clock speed = 1/4 System clock speed<br>0x3: Flash SPI clock speed = 1/5 System clock speed |

**Table 17-13 - CONFIG – Configuration Register**


Copyright © Bridgetek Pte Ltd

### 17.2.13 STATUS – Status Register (address offset: 0x0D)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7 | Reserved | - | - | - |
| 6 | Control Busy | RO | 1'b0 | 1: The control unit is busy. This means no other command should be issued.<br>When this bit is set, bits 7, 3-0 may not be valid until this bit is cleared. |
| 5 | Data Read Ready | RO | 1'b0 | 1: Data for read is available at data read port |
| 4 | Data Write Ready | RO | 1'b0 | 1: Data for write can be written to data write port |
| 3 | Reserved | - | - | - |
| 2 | Reserved | - | - | - |
| 1 | Reserved | - | - | - |
| 0 | Write in Progress | RO | 1'b0 | 1: Flash Write Operations in Progress |

**Table 17-14 - STATUS – Status Register**

### 17.2.14 CRCL – Flash Content CRC Register (LSB) (address offset: 0x0E)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | - | RO | 0xXX | LSB of the CRC16 of the flash content. The CRC is calculated upon reset when flash content is transferred to the programme memory, and the CRC is based on the polynomial X^16+X^15+X^2+1 |

**Table 17-15 - CRCL – Flash Content CRC Register (LSB)**

### 17.2.15 CRCH – Flash Content CRC Register (MSB) (address offset: 0x0F)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | - | RO | 0xXX | MSB of the CRC16 of the flash content. The CRC is calculated upon reset when flash content is transferred to the programme memory, and the CRC is based on the polynomial X^16+X^15+X^2+1 |

**Table 17-16 - CRCH – Flash Content CRC Register (MSB)**

### 17.2.16 CHIPID0 – Chip ID Register (LSB) (address offset: 0x7C)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | - | RO | 0xXX | LSB of the 32-bit chip ID, only accessible via the one-wire debugger. The same ID is available to the CPU in another register address |

**Table 17-17 - CHIPID0 – Chip ID Register (LSB)**

### 17.2.17 CHIPID1 – Chip ID Register (Byte 1) (address offset: 0x7D)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | - | RO | 0xXX | Byte 1 of the 32-bit chip ID, only accessible via the one-wire debugger. The same ID is available to the CPU in another register address |

**Table 17-18 - CHIPID1 – Chip ID Register (Byte 1)**


Copyright © Bridgetek Pte Ltd

### 17.2.18 CHIPID2 – Chip ID Register (Byte 2) (address offset: 0x7E)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | - | RO | 0xXX | Byte 2 of the 32-bit chip ID, only accessible via the one-wire debugger. The same ID is available to the CPU in another register address |

**Table 17-19 - CHIPID2 – Chip ID Register (Byte 2)**

### 17.2.19 CHIPID3 – Chip ID Register (MSB) (address offset: 0x7F)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | - | RO | 0xXX | MSB of the 32-bit chip ID, only accessible via the one-wire debugger. The same ID is available to the CPU in another register address |

**Table 17-20 - CHIPID3 – Chip ID Register (MSB)**

### 17.2.20 DRWDATA – Data Register (address offset: 0x80)

| Bit | Name | Type | Default Value | Description |
|---|---|---|---|---|
| 7:0 | - | RW | 0x00 | This is the data read or write port used to transfer data in and out of this control unit. Up to 64 addresses may be used.<br>When bit 4 of STATUS is not set, writing to this port will have no effect.<br>When bit 5 of STATUS is not set, reading this port will return 0. |

**Table 17-21 - DRWDATA – Data Register**

## 17.3 Flash Controller Commands

The following commands are supported by the control unit. The command is initiated once the command register is updated. No further command can be entered until the current one is completed. If two successive commands, regardless of what they are, are entered while one is on-going, the current operation will be aborted. The two commands will not be executed either.

The commands can be divided into 2 groups. The first group consists of commands that are supported directly by the serial flash. The second group consists of commands that initiate data transfer between the debugger/CPU interface and flash, debugger/CPU interface and program/data memory, or flash and program/data memory.

The following table lists the first group of commands. If executing any of these commands that require data input, write the required data to the DRWDATA register.

| Command | Code | Description |
|---|---|---|
| CMDWREN | 0x06 | The Write Enable (WREN) instruction is for setting the Write Enable Latch (WEL) bit. Those instructions such as PP, SE,<br>BE, CE, and WRSR, which are intended to change the device content, should be set every time after the WREN instruction sets the WEL bit. |
| CMDWRDI | 0x04 | The Write Disable (WRDI) instruction is for resetting the |


Copyright © Bridgetek Pte Ltd

| Command | Code | Description |
|---|---|---|
| | | Write Enable Latch (WEL) bit.<br>The WEL bit is reset by the following situations:<br>- Power-up<br>- Write Disable (WRDI) instruction completion<br>- Write Status Register (WRSR) instruction completion<br>- Page Program (PP) instruction completion<br>- Sector Erase (SE) instruction completion<br>- Block Erase (BE) instruction completion<br>- Chip Erase (CE) instruction completion |
| CMDWRSR | 0x01 | The WRSR instruction is for changing the values of Status Register Bits. Before sending WRSR instruction, the Write Enable (WREN) instruction must be executed first. The WRSR instruction can change the value of Block Protect (BP1 - bit3, BP0 - bit2) bits to define the protected area of memory (as shown in table 2). The WRSR can also set or reset the Status Register Write Disable (SRWD - bit 7) bit in accordance with Write Protection (WP#) pin signal. The WRSR instruction cannot be executed once the Hardware Protected Mode (HPM) is entered.<br>This instruction has no effect on bits 6, 5, 1 and 0. |
| CMDRDID | 0x9F | The RDID instruction is for reading the manufacturer ID of 1-byte and followed by Device ID of 2-byte. The MXIC Manufacturer ID and Device ID are listed as table of "ID Definitions" of the MXIC specification.<br>While Program/Erase operation is in progress, it will not decode the RDID instruction, so there's no effect on the cycle of program/erase operation which is currently in progress. |
| CMDRDSR | 0x05 | The RDSR instruction is for reading Status Register Bits. The Read Status Register can be read at any time (even in program/erase/write status register condition) and continuously. It is recommended to check the Write in Progress (WIP) bit before sending a new instruction when a program, erase, or write status register operation is in progress.<br>The status data can be read back from the STATUS register. The bits 7, 3-0 of STATUS reflect the content from this flash command. |
| CMDRDSFDP | 0x5A | The Serial Flash Discoverable Parameter (SFDP) standard provides a consistent method of describing the functional and feature capabilities of serial flash devices in a standard set of internal parameter tables.<br>Refer to MXIC specification for the parameter details. |
| CMDSE | 0x20 | Sector erase; Sector address can be set at FSADDRx. |
| CMDBE1 / CMDBE2 | 0x52 / 0xD8 | Block erase; Block address can be set at FSADDRx. |
| CMDCE1 / CMDCE2 | 0x60 / 0xC7 | Chip erase. |


Copyright © Bridgetek Pte Ltd

| Command | Code | Description |
|---|---|---|
| CMDDP | 0xB9 | The Deep Power-down (DP) instruction is for setting the device to minimizing the power consumption. During the Deep Power-down mode, the device is not active and all Write/Program/Erase instruction are ignored. |
| CMDRDP | 0xAB | The RDP instruction is for releasing from Deep Power Down Mode. |

**Table 17-22 - Flash Controller Command Group 1**

The following table lists the commands that initiate data transfer between the Debugger/CPU interface and flash/program/data memory, or between flash and program/data. Each command allows the option to reset the CPU, or reboot the system, or allow the system to continue.

| Command | Code | Description |
|---|---|---|
| CMDDBG2P1 | 0xE0 | Initiates data transfer from Debugger Interface to Program Memory.<br>The start address of the Program Memory destination will be dictated by RSADDRx which must be 32-bit aligned.<br>The number of bytes to transfer will be dictated by BLENGTHx.<br>Data to be transferred will be in DRWDATA. |
| CMDDBG2P2 | 0xE1 | Similar to CMDDBG2P1 except at the end of the transfer, a CPU reset will be performed. |
| CMDDBG2P3 | 0xE2 | Similar to CMDDBG2P1 except at the end of the transfer, a system reboot will be performed. |
| CMDDBG2D1 | 0xE4 | Similar to CMDDBG2P1 except the destination is the Data Memory. |
| CMDDBG2D2 | 0xE5 | Similar to CMDDBG2P2 except the destination is the Data Memory. |
| CMDDBG2D3 | 0xE6 | Similar to CMDDBG2P3 except the destination is the Data Memory. |
| CMDDBG2F1 | 0xE8 | Initiates data transfer from Debugger Interface to Flash Memory.<br>The start address of the Flash Memory destination will be dictated by FSADDRx which must be 256-byte aligned.<br>The number of bytes to transfer will be dictated by BLENGTHx and must be multiples of 256 bytes.<br>Data to be transferred will be in DRWDATA. |
| CMDDBG2F2 | 0xE9 | Similar to CMDDBG2F1 except at the end of the transfer, a CPU reset will be performed. |
| CMDDBG2F3 | 0xEA | Similar to CMDDBG2F1 except at the end of the transfer, a system reboot will be performed. |
| CMDP2F1 | 0xF0 | Initiates data transfer from Program Memory to Flash Memory.<br>The start address of the Program Memory destination will be dictated by RSADDRx which must be 32-bit aligned.<br>The start address of the Flash Memory destination will be dictated by FSADDRx which must be 256-byte aligned.<br>The number of bytes to transfer will be dictated by BLENGTHx and must be multiples of 256 bytes. |


Copyright © Bridgetek Pte Ltd

| Command | Code | Description |
|---|---|---|
| CMDP2F2 | 0xF1 | Similar to CMDP2F1 except at the end of the transfer, a CPU reset will be performed. |
| CMDP2F3 | 0xF2 | Similar to CMDP2F1 except at the end of the transfer, a system reboot will be performed. |
| CMDF2P1 | 0xF4 | Initiates data transfer from Flash Memory to Program Memory.<br>The start address of the Program Memory destination will be dictated by RSADDRx which must be 32-bit aligned.<br>The start address of the Flash Memory destination will be dictated by FSADDRx which must be 256-byte aligned.<br>The number of bytes to transfer will be dictated by BLENGTHx. |
| CMDF2P2 | 0xF5 | Similar to CMDF2P1 except at the end of the transfer, a CPU reset will be performed. |
| CMDF2P3 | 0xF6 | Similar to CMDF2P1 except at the end of the transfer, a system reboot will be performed. |
| CMDD2F1 | 0xF8 | Initiates data transfer from Data Memory to Flash Memory.<br>The start address of the Data Memory destination will be dictated by RSADDRx which must be 32-bit aligned.<br>The start address of the Flash Memory destination will be dictated by FSADDRx which must be 256-byte aligned.<br>The number of bytes to transfer will be dictated by BLENGTHx and must be multiples of 256 bytes. |
| CMDD2F2 | 0xF9 | Similar to CMDD2F1 except at the end of the transfer, a CPU reset will be performed. |
| CMDD2F3 | 0xFA | Similar to CMDD2F1 except at the end of the transfer, a system reboot will be performed. |
| CMDF2D1 | 0xFC | Initiates data transfer from Flash Memory to Data Memory.<br>The start address of the Data Memory destination will be dictated by RSADDRx which must be 32-bit aligned.<br>The start address of the Flash Memory destination will be dictated by FSADDRx which must be 256-byte aligned.<br>The number of bytes to transfer will be dictated by BLENGTHx. |
| CMDF2D2 | 0xFD | Similar to CMDF2D1 except at the end of the transfer, a CPU reset will be performed. |
| CMDF2D3 | 0xFE | Similar to CMDF2D1 except at the end of the transfer, a system reboot will be performed. |
| CMDHALT | 0xFF | This command forces halt immediately to the CPU. The use should be avoided as this cannot be undone. Only a system reset will remove the halt effect. |

**Table 17-23 - Flash Controller Command Group 2**


Copyright © Bridgetek Pte Ltd

# 18 Contact Information

Refer to https://brtchip.com/contact-us/ for contact information.

**Distributor and Sales Representatives**

Please visit the Sales Network page for the contact details of our distributor(s) and sales representative(s) in your country.

System and equipment manufacturers and designers are responsible to ensure that their systems, and any Bridgetek Pte Ltd (BRTChip) devices incorporated in their systems, meet all applicable safety, regulatory and system-level performance requirements. All application-related information in this document (including application descriptions, suggested Bridgetek devices and other materials) is provided for reference only. While Bridgetek has taken care to assure it is accurate, this information is subject to customer confirmation, and Bridgetek disclaims all liability for system designs and for any applications assistance provided by Bridgetek. Use of Bridgetek devices in life support and/or safety applications is entirely at the user's risk, and the user agrees to defend, indemnify and hold harmless Bridgetek from any and all damages, claims, suits or expense resulting from such use. This document is subject to change without notice. No freedom to use patents or other intellectual property rights is implied by the publication of this document. Neither the whole nor any part of the information contained in, or the product described in this document, may be adapted or reproduced in any material or electronic form without the prior written consent of the copyright holder. Bridgetek Pte Ltd, 1 Tai Seng Avenue, Tower A, #03-05, Singapore 536464. Singapore Registered Company Number: 201542387H.


Copyright © Bridgetek Pte Ltd

# Appendix A – References

## Document References

Datasheet DS_FT930/1/2/3

FT9XX toolchain

## Acronyms and Abbreviations

| Terms | Description |
|---|---|
| ADC | Analogue to Digital Converter |
| ATX | Analog Transceiver (USB PHY) |
| BCD | Battery Charger Detection |
| CPRM | Content Protection for Recordable Media |
| CPU | Central Processing unit |
| CRC | Cyclic Redundancy Check |
| DAC | Digital to Analogue Converter |
| FIFO | First In First Out |
| GPIO | General Purpose Input Output |
| $I^2C$ | Inter-Integrated Circuit |
| IP | Intellectual Property |
| LSB | Least Significant Byte |
| MMC | Multi Media Card |
| MSB | Most Significant Byte |
| SPI | Serial Peripheral Interface |
| PHY | Physical Layer |
| PWM | Pulse Width Modulation |
| RAM | Random Access Memory |
| UART | Universal Asynchronous Receiver Transmitter |
| $V_{DM_SRC}$ | D- Source Voltage |
| $V_{DP_SRC}$ | D+ Source Voltage |


Copyright © Bridgetek Pte Limited

| Terms | Description |
|---|---|
| USB | Universal Serial Bus |
| USB-IF | USB Implementers Forum |


Copyright © Bridgetek Pte Limited

# Appendix B – List of Tables & Figures

## List of Tables

Table 2-1 - FT93x Program Memory Organization ..... 14
Table 3-1 - Peripheral Availability on FT93x Series Models ..... 16
Table 3-2 - Register Map for FT93x Series ..... 16
Table 4.1 - Notations used in Register Description ..... 17
Table 5-1 - Overview of General System Registers ..... 20
Table 5-2 - CHIPID - Chip ID Register ..... 21
Table 5-3 - FT93x Series Chip ID Configuration ..... 21
Table 5-4 - CHIPCFG - Chip Configuration Register ..... 21
Table 5-5 - CLKCFG - Clock Configuration Register ..... 22
Table 5-6 - PMCFG - Power Management Configuration Register ..... 23
Table 5-7 - Reserved ..... 24
Table 5-8 - Reserved ..... 24
Table 5-9 - MSC0CFG - Miscellaneous Configuration Register ..... 25
Table 5-10 - Pin Configuration Register Description ..... 26
Table 5-11 - Pin 00 – 03 Register ..... 26
Table 5-12 - Pin 04 – 07 Register ..... 26
Table 5-13 - Pin 08 – 11 Register ..... 26
Table 5-14 - Pin 12 – 15 Register ..... 27
Table 5-15 - Pin 16 – 19 Register ..... 27
Table 5-16 - Pin 20 – 23 Register ..... 27
Table 5-17 - Pin 24 – 27 Register ..... 27
Table 5-18 - Pin 28 – 31 Register ..... 27
Table 5-19 - Pin 32 – 35 Register ..... 27
Table 5-20 - Pin 36 – 39 Register ..... 28
Table 5-21 - GPIO Configuration Register Description ..... 28
Table 5-22 - GPIO 00 – 07 Configuration Register ..... 29
Table 5-23 - GPIO 08 – 15 Configuration Register ..... 29
Table 5-24 - GPIO 16 – 23 Configuration Register ..... 29
Table 5-25 - GPIO 24 – 31 Configuration Register ..... 29
Table 5-26 - GPIO 32 – 39 Configuration Register ..... 30
Table 5-27 - GPIO 00 – 31 Value Register ..... 30
Table 5-28 - GPIO 32 – 39 Value Register ..... 30
Table 5-29 - GPIO 00 – 31 Interrupt Enable Register ..... 30
Table 5-30 - GPIO 32 – 39 Interrupt Enable Register ..... 30
Table 5-31 - GPIO 00 – 31 Interrupt Pending Register ..... 31


Copyright © Bridgetek Pte Limited

Table 5-32 - GPIO 32 – 39 Interrupt Pending Register ..... 31
Table 5-33 - DAC_ADC_CONF - ADC/DAC Configuration/Status Register ..... 32
Table 5-34 - DAC_ADC_CNT - ADC/DAC Count Register ..... 32
Table 5-35 - DAC_ADC_DATA - ADC/DAC Data Register ..... 33
Table 5-36 - GLOBAL_RESET_STATUS - Global Reset and Status Register ..... 33
Table 6-1 - Interrupt Assignment Table ..... 34
Table 6-2 - Overview of Interrupt Control Registers ..... 35
Table 6-3 - IRQ00-03 Assignment Register ..... 35
Table 6-4 - IRQ04-07 Assignment Register ..... 35
Table 6-5 - IRQ08-11 Assignment Register ..... 36
Table 6-6 - IRQ12-15 Assignment Register ..... 36
Table 6-7 - IRQ16-19 Assignment Register ..... 36
Table 6-8 - IRQ20-23 Assignment Register ..... 37
Table 6-9 - IRQ24-27 Assignment Register ..... 37
Table 6-10 - IRQ28-31 Assignment Register ..... 37
Table 6-11 - IRQ Control Register ..... 38
Table 7-1 - Overview of USB Device Registers ..... 41
Table 7-2 - DC_ADDRESS_ENABLE – Address Register ..... 41
Table 7-3 - DC_MODE – Mode Register ..... 41
Table 7-4 - DC_INT_ENABLE – Interrupt Enable Register ..... 41
Table 7-5 - DC_EP_INT_ENABLE – Endpoints Interrupt Enable Register ..... 42
Table 7-6 - DC_EP0_CONTROL – Endpoint 0 Control Register ..... 42
Table 7-7 - DC_EP0_STATUS – Endpoint 0 Status Register ..... 43
Table 7-8 - DC_EP0_BUFFER_LENGTH – Endpoint 0 Buffer Length Register ..... 43
Table 7-9 - DC_EP0_BUFFER – Endpoint 0 Buffer Register ..... 43
Table 7-10 - DC_EP(x)_CONTROL – Endpoint Control Registers ..... 44
Table 7-11 - DC_EP(x)_STATUS – Endpoint Status Registers ..... 45
Table 7-12 - DC_EP(x)_BUFFER_LENGTH_LSB – Endpoint Buffer Length LSB Registers ..... 45
Table 7-13 - DC_EP(x)_BUFFER_LENGTH_MSB – Endpoint Buffer Length MSB Registers ..... 46
Table 7-14 - DC_EP(x)_BUFFER – Endpoint Buffer Registers ..... 46
Table 7-15 - DC_INT_STATUS – Interrupt Status Register ..... 46
Table 7-16 - DC_EP_INT_STATUS – Endpoints Interrupt Status Register ..... 47
Table 7-17 - DC_FRAME_NUMBER_LSB – Frame Number LSB Register ..... 47
Table 7-18 - DC_FRAME_NUMBER_MSB – Frame Number MSB Register ..... 47
Table 8-1 - Overview of SD Host Registers ..... 50
Table 8-2 - 11.2.1 SDH_AUTO_CMD23_ARG2 – Auto CMD23 Argument 2 Register ..... 50
Table 8-3 - 11.2.2 SDH_BLK_SIZE – Block Size Register ..... 51
Table 8-4 - 11.2.3 SDH_BLK_COUNT – Block Count Register ..... 51
Table 8-5 - 11.2.4 SDH_ARG_1 – Argument 1 Register ..... 51


Copyright © Bridgetek Pte Limited

Table 8-6 - 11.2.5 SDH_TNSFER_MODE – Transfer Mode Register ..... 52
Table 8-7 - 11.2.6 SDH_CMD – Command Register ..... 53
Table 8-8 - 11.2.7 SDH_RESPONSE – Response Register ..... 53
Table 8-9 - 11.2.8 SDH_BUF_DATA – Buffer Data Port Register ..... 53
Table 8-10 - 11.2.9 SDH_PRESENT_STATE – Present State Register ..... 55
Table 8-11 - SDH_HST_CNTL_1 – Host Control 1 Register ..... 55
Table 8-12 - SDH_PWR_CNTL – Power Control Register ..... 56
Table 8-13 - SDH_BLK_GAP_CNTL – Block Gap Control Register ..... 56
Table 8-14 - SDH_CLK_CNTL – Clock Control Register ..... 56
Table 8-15 - SDH_TIMEOUT_CNTL – Timeout Control Register ..... 57
Table 8-16 - SDH_SW_RST – Software Reset Register ..... 57
Table 8-17 - SDH_NRML_INT_STATUS – Normal Interrupt Status Register ..... 58
Table 8-18 - SDH_ERR_INT_STATUS – Error Interrupt Status Register ..... 58
Table 8-19 - SDH_NRML_INT_ENABLE – Normal Interrupt Status Enable Register ..... 59
Table 8-20 - SDH_ERR_INT_ENABLE – Error Interrupt Status Enable Register ..... 59
Table 8-21 - SDH_NRML_INT_SGNL_ENABLE – Normal Interrupt Signal Enable Register ..... 60
Table 8-22 - SDH_ERR_INT_SGNL_ENABLE – Error Interrupt Signal Enable Register ..... 61
Table 8-23 - SDH_AUTO_CMD12_ERR_STATUS – Auto CMD12 Error Status Register ..... 61
Table 8-24 - SDH_HOST_CNTL_2 – Host Control 2 Register ..... 61
Table 8-25 - SDH_CAP_1 – Capabilities Register 1 ..... 62
Table 8-26 - SDH_CAP_2 – Capabilities Register 2 ..... 63
Table 8-27 - SDH_RSRV_1 – Reserved 1 Register ..... 63
Table 8-28 - SDH_RSRV_2 – Reserved 2 Register ..... 63
Table 8-29 - SDH_FORCE_EVT_CMD_ERR_STATUS – Force Event Register for Auto CMD Error Status ..... 64
Table 8-30 - SDH_FORCE_EVT_ERR_INT_STATUS – Force Event for Error Interrupt Status Register ..... 64
Table 8-31 - SDH_RSRV_3 – Reserved 3 Register ..... 64
Table 8-32 - SDH_RSRV_4 – Reserved 4 Register ..... 65
Table 8-33 - SDH_PRST_INIT – Preset value for initialization ..... 65
Table 8-34 - SDH_PRST_DFLT_SPD – Preset value for default speed ..... 65
Table 8-35 - SDH_PRST_HIGH_SPD – Preset value for the high speed ..... 66
Table 8-36 - SDH_PRST_SDR12 – Preset value for SDR12 ..... 66
Table 8-37 - SDH_PRST_SDR25 – Preset value for SDR25 ..... 67
Table 8-38 - SDH_PRST_SDR50 – Preset value for SDR50 ..... 67
Table 8-39 - SDH_PRST_SDR104 – Preset value for SDR104 ..... 67
Table 8-40 - SDH_PRST_DDR50 – Preset value for DDR50 ..... 68
Table 8-41 - SDH_RSRV_5 – Reserved 5 Register ..... 68
Table 8-42 - SDH_HC_VER – Host Controller Version Register ..... 68
Table 8-43 - SDH_VNDR_0 – Vendor-defined 0 Register ..... 69


Copyright © Bridgetek Pte Limited

Table 8-44 - SDH_VNDR_1 – Vendor-defined 1 Register ..... 69
Table 8-45 - SDH_VNDR_2 – Vendor-defined 2 Register ..... 70
Table 8-46 - SDH_VNDR_3 – Vendor-defined 3 Register ..... 70
Table 8-47 - SDH_VNDR_4 – Vendor-defined 4 Register ..... 70
Table 8-48 - SDH_VNDR_5 – Vendor-defined 5 Register ..... 70
Table 8-49 - SDH_VNDR_6 – Vendor-defined 6 Register ..... 70
Table 8-50 - SDH_VNDR_7 – Vendor-defined 7 Register ..... 70
Table 8-51 - SDH_VNDR_8 – Vendor-defined 8 Register ..... 71
Table 8-52 - SDH_VNDR_9 – Vendor-defined 9 Register ..... 71
Table 8-53 - SDH_RSRV_6 – Reserved 6 Register ..... 71
Table 8-54 - SDH_HW_ATTR – Hardware Attributes Register ..... 71
Table 8-55 - SDH_CPR_MOD_CNTL – Cipher Mode Control Register ..... 72
Table 8-56 - SDH_CPR_MOD_STATUS – Cipher Mode Status Register ..... 72
Table 8-57 - SDH_CPR_MOD_STATUS_EN – Cipher Mode Status Enable Register ..... 73
Table 8-58 - SDH_CPR_MOD_SIG_EN – Cipher Mode Signal Enable Register ..... 73
Table 8-59 - SDH_IN_DATA_LSB – Input Data LSB Register ..... 73
Table 8-60 - SDH_IN_DATA_MSB – Input Data MSB Register ..... 73
Table 8-61 - SDH_IN_KEY_LSB – Input Key LSB Register ..... 73
Table 8-62 - SDH_IN_KEY_MSB – Input Key MSB Register ..... 74
Table 8-63 - SDH_OUT_DATA_LSB – Output Data LSB Register ..... 74
Table 8-64 - SDH_OUT_DATA_MSB – Output Data MSB Register ..... 74
Table 8-65 - SDH_SCRT_CONS_DATA – Secret Constant Table Data Port ..... 74
Table 9-1 - Overview of UART Registers ..... 77
Table 9-2 - UART mode selection ..... 78
Table 9-3 - UART_RBR - Receiver Buffer Register ..... 79
Table 9-4 - UART_THR - Transmitter Holding Register ..... 80
Table 9-5 - UART_DIV_LSB - Divisor LSB Register ..... 80
Table 9-6 - UART_DIV_MSB - Divisor MSB Register ..... 80
Table 9-7 - UART_INT_ENABLE - Interrupt Enable Register ..... 80
Table 9-8 - UART_INT_STATUS - Interrupt Status Register ..... 81
Table 9-9 - Interrupt Status Register Software Handling ..... 81
Table 9-10 - UART_FCR - FIFO Control Register – 550 mode ..... 82
Table 9-11 - UART_RCVR - FIFO Trigger Level – 550 mode ..... 82
Table 9-12 - UART_FCR - FIFO Control Register – 650 mode ..... 83
Table 9-13 - UART_RCVR - FIFO Trigger Level – 650 mode ..... 83
Table 9-14 - XMIT FIFO Trigger Level ..... 83
Table 9-15 - UART_LCR - Line Control Register ..... 84
Table 9-16 - UART_MCR - Modem Control Register ..... 85
Table 9-17 - UART_LSR - Line Status Register ..... 86


Copyright © Bridgetek Pte Limited

Table 9-18 - UART_MSR - Modem Status Register ..... 87
Table 9-19 - UART_SPR - SPR Register ..... 87
Table 9-20 - UART_EFR - Enhanced Feature Register ..... 88
Table 9-21 - UART_XON1 - XON1 Register ..... 88
Table 9-22 - UART_XON2 - XON2 Register ..... 88
Table 9-23 - UART_XOFF1 - XOFF1 Register ..... 88
Table 9-24 - UART_XOFF2 - XOFF2 Register ..... 89
Table 9-25 - UART_ASR - Additional Status Register ..... 89
Table 9-26 - UART_RFL - Receiver FIFO Level Register ..... 89
Table 9-27 - UART_TFL - Transmitter FIFO Level Register ..... 90
Table 9-28 - UART_ICR - ICR Register ..... 90
Table 9-29 - UART_ACR- Additional Control Register ..... 92
Table 9-30 - UART_CPR - Clock Prescaler Register ..... 92
Table 9-31 - UART_TCR - Time Clock Register ..... 93
Table 9-32 - UART_CKS Clock Select Register ..... 93
Table 9-33 - UART_TTL - Transmitter Trigger Level Register ..... 94
Table 9-34 - UART_RTL - Receiver Trigger Level Register ..... 94
Table 9-35 - UART_FCL - Flow Control Level LSB Register ..... 94
Table 9-36 - UART_FCH - Flow Control Level Register MSB ..... 94
Table 9-37 - UART_ID1 - Identification 1 Register ..... 95
Table 9-38 - UART_ID2 - Identification 2 Register ..... 95
Table 9-39 - UART_ID3 - Identification 3 Register ..... 95
Table 9-40 - UART_REV - Revision Register ..... 95
Table 9-41 - UART_CSR - Channel Software Reset Register ..... 95
Table 9-42 - UART_NMR - Nine Bit Mode Register ..... 96
Table 9-43 - UART_MDM - Modem Disable Mask Register ..... 96
Table 9-44 - UART_RFC - Readable FCR Register ..... 97
Table 9-45 - UART_GDS - Good Data Status Register ..... 97
Table 9-46 - UART_RSRV_1 - Reserved 1 Register ..... 97
Table 9-47 - UART_PIDX - Port Index Register ..... 97
Table 9-48 - UART_CKA - Clock Alteration Register ..... 97
Table 10-1 - Timers/Watchdog Operation ..... 98
Table 10-2 - Overview of Timers/Watchdog Registers ..... 99
Table 10-3 - TIMER_CONTROL_0 - Timers Control Register 0 ..... 99
Table 10-4 - TIMER_CONTROL_1 - Timers Control Register 1 ..... 99
Table 10-5 - TIMER_CONTROL_2 - Timers Control Register 2 ..... 100
Table 10-6 - TIMER_CONTROL_3 - Timers Control Register 3 ..... 100
Table 10-7 - TIMER_CONTROL_4 - Timers Control Register 4 ..... 100
Table 10-8 - TIMER_INT - Timers Interrupt Register ..... 100


Copyright © Bridgetek Pte Limited

Table 10-9 - TIMER_SELECT - Timers A..D Select Register ..... 101
Table 10-10 - TIMER_WDG - Watchdog Start Value ..... 101
Table 10-11 - TIMER_WRITE_LS - Timer A..D Start Value 7:0 ..... 101
Table 10-12 - TIMER_WRITE_MS - Timer A..D Start Value 15:8 ..... 101
Table 10-13 - TIMER_PRESC_LS - Prescaler Start Value 7:0 ..... 101
Table 10-14 - TIMER_PRESC_MS - Prescaler Start Value 15:8 ..... 101
Table 10-15 - TIMER_READ_LS - Timer A..D Current Value 7:0 ..... 102
Table 10-16 - TIMER_READ_MS - Timer A..D Current Value 15:8 ..... 102
Table 11-1 - Overview of SPI Master Registers ..... 103
Table 11-2 - SPIM_CNTL – Control Register ..... 104
Table 11-3 - SPIM_STATUS – Status Register ..... 105
Table 11-4 - SPIM_DATA – Receiver and Transmitter Data Registers ..... 105
Table 11-5 - SPIM_SLV_SEL_CNTL – Slave Select Control Register ..... 105
Table 11-6 - SPIM_FIFO_CNTL – FIFO Control Register ..... 106
Table 11-7 - SPIM_TNSFR_FRMT_CNTL – Transfer Format Control Register ..... 106
Table 11-8 - SPIM_ALT_DATA – Alternative SPI Master Data Register ..... 106
Table 11-9 - SPIM_RX_FIFO_COUNT – SPI Master RX FIFO Count Register ..... 106
Table 11-10 - SPIM_CNTL_2 – Control 2 Register ..... 107
Table 11-11 - SPIM_STATUS_2 – Status 2 Register ..... 107
Table 11-12 - SPIM_FIFO_CNTL_2 – FIFO Control 2 Register ..... 108
Table 11-13 - SPIM_TX_FIFO_COUNT – SPI Master TX FIFO Count Register ..... 108
Table 11-14 - SPIM_BAUD – SPI Master Baud Register ..... 108
Table 12-1 - Overview of SPI Slave Registers ..... 109
Table 12-2 - SPIS_CNTL – Control Register ..... 110
Table 12-3 - SPIS_STATUS – Status Register ..... 110
Table 12-4 - SPIS_DATA – Receiver and Transmitter Data Registers ..... 111
Table 12-5 - SPIS_SLV_SEL_CNTL – Slave Select Control Register ..... 111
Table 12-6 - SPIS_FIFO_CNTL – FIFO Control Register ..... 111
Table 12-7 - SPIS_TNSFR_FRMT_CNTL – Transfer Format Control Register ..... 112
Table 12-8 - SPIS_ALT_DATA – Alternative SPI Slave Data Register ..... 112
Table 12-9 - SPIS_RX_FIFO_COUNT – SPI Slave RX FIFO Count Register ..... 112
Table 12-10 - SPIS_CNTL_2 – Control 2 Register ..... 113
Table 12-11 - SPIS_STATUS_2 – Status 2 Register ..... 113
Table 12-12 - SPIS_FIFO_CNTL_2 – FIFO Control 2 Register ..... 114
Table 12-13 - SPIS_TX_FIFO_COUNT – SPI Slave TX FIFO Count Register ..... 114
Table 12-14 - SPIS_BAUD – SPI Slave Baud Register ..... 114
Table 13-1 - Overview of I2C Master Registers ..... 115
Table 13-2 - I2CM_SLV_ADDR – Slave Address Register ..... 116
Table 13-3 - I2CM_CNTL – Control Register ..... 116


Copyright © Bridgetek Pte Limited

Table 13-4 - I2CM_STATUS – Status Register....117
Table 13-5 - I2CM_DATA – Receive / Transmit Data Register....117
Table 13-6 - I2CM_TIME_PERIOD – Timer Period Register....117
Table 13-7 - I2CM_HS_TIME_PERIOD – High Speed Timer Period Register....118
Table 13-8 - I2CM_FIFO_LEN – FIFO Mode Byte Length....118
Table 13-9 - I2CM_FIFO_INT_ENABLE – FIFO Mode Interrupt Enable....118
Table 13-10 - I2CM_FIFO_INT_PEND – FIFO Mode Interrupt Pending....119
Table 13-11 - I2CM_FIFO_DATA - FIFO Data Register....119
Table 13-12 - I2CM_TRIG - Trigger Register....119
Table 14-1 - Overview of I2C Master Registers....120
Table 14-2 - I2CS_OWN_ADDR – Own Address Register....121
Table 14-3 - I2CS_CNTL – Control Register....121
Table 14-4 - I2CS_STATUS – Status Register....122
Table 14-5 - I2CS_DATA – Receive / Transmit Data Register....122
Table 15-1 - Overview of RTC Registers....123
Table 15-2 - RTC_TIME - Time Register....123
Table 15-3 - RTC_DATE - Date Register....124
Table 15-4 - RTC_AALARM1 – Auto Alarm 1 Register....124
Table 15-5 - Alarm 1 Mask Bits....124
Table 15-6 - RTC_AALARM2 – Auto Alarm 2 Register....125
Table 15-7 - Alarm 2 Mask Bits....125
Table 15-8 - RTC_CTRL – Control Register....125
Table 15-9 - RTC_STAT - Status Register....126
Table 15-10 - RTC_TRIM – Trimming Value Register....127
Table 15-11 - RTC_SLE_TIME – Sleep Time Register....127
Table 15-12 - RTC_SLE_DATE - Sleep Date Register....128
Table 15-13 – RTC_COMP_PER - Compare Period Register....128
Table 15-14 - I2CS_FIFO_DATA - FIFO Data Register....129
Table 15-15 - I2CS_TRIG - Trigger Register....129
Table 15-16 - I2CS_FIFO_LEN – FIFO Mode Byte Length....129
Table 15-17 - I2CS_FIFO_INT_ENABLE – FIFO Mode Interrupt Enable....129
Table 15-18 - I2CS_FIFO_INT_PEND – FIFO Mode Interrupt Pending....130
Table 16-1 - Overview of PWM Registers....132
Table 16-2 - PWM_CTRL0 - PCM Control Register....133
Table 16-3 - PWM_CTRL1 - PWM Control Register....133
Table 16-4 - PWM_PRESCALER - PWM Prescaler Register....133
Table 16-5 - PWM_CNTL - PWM Counter Register (LSB)....134
Table 16-6 - PWM_CNTH - PWM Counter Register (MSB)....134
Table 16-7 - PWM_CMP0L - Comparator 0 Value Register (LSB)....134


Copyright © Bridgetek Pte Limited

Table 16-8 - PWM_CMP0H - Comparator 0 Value Register (MSB) .....134
Table 16-9 - PWM_CMP1L - Comparator 1 Value Register (LSB).....134
Table 16-10 - PWM_CMP1H - Comparator 1 Value Register (MSB).....134
Table 16-11 - PWM_CMP2L - Comparator 2 Value Register (LSB) .....135
Table 16-12 - PWM_CMP2H - Comparator 2 Value Register (MSB).....135
Table 16-13 - PWM_CMP3L - Comparator 3 Value Register (LSB) .....135
Table 16-14 - PWM_CMP3H - Comparator 3 Value Register (MSB).....135
Table 16-15 - PWM_CMP4L - Comparator 4 Value Register (LSB) .....135
Table 16-16 - PWM_CMP4H - Comparator 4 Value Register (MSB).....135
Table 16-17 - PWM_CMP5L - Comparator 5 Value Register (LSB) .....136
Table 16-18 - PWM_CMP5H - Comparator 5 Value Register (MSB).....136
Table 16-19 - PWM_CMP6L - Comparator 6 Value Register (LSB) .....136
Table 16-20 - PWM_CMP6H - Comparator 6 Value Register (MSB).....136
Table 16-21 - PWM_CMP7L - Comparator 7 Value Register (LSB) .....136
Table 16-22 - PWM_CMP7H - Comparator 7 Value Register (MSB).....137
Table 16-23 - PWM_TOGGLE0 - Channel 0 OUT Toggle Comparator Mask .....137
Table 16-24 - PWM_TOGGLE1 - Channel 1 OUT Toggle Comparator Mask .....137
Table 16-25 - PWM_TOGGLE2 - Channel 2 OUT Toggle Comparator Mask Register .....137
Table 16-26 - PWM_TOGGLE3 - Channel 3 OUT Toggle Comparator Mask Register .....137
Table 16-27 - PWM_TOGGLE4 - Channel 4 OUT Toggle Comparator Mask Register .....137
Table 16-28 - PWM_TOGGLE5 - Channel 5 OUT Toggle Comparator Mask Register .....138
Table 16-29 - PWM_TOGGLE6 - Channel 6 OUT Toggle Comparator Mask Register .....138
Table 16-30 - PWM_TOGGLE7 - Channel 7 OUT Toggle Comparator Mask Register .....138
Table 16-31 - PWM_OUT_CLR_EN - PWM OUT Clear Enable Register .....138
Table 16-32 - PWM_CTRL_BL_CMP8 - Control Block CMP8 Value Register .....138
Table 16-33 - PWM_INIT - PWM Initialization Register .....138
Table 16-34 - PWM_INTMASK - PWM Interrupt Mask Register.....139
Table 16-35 - PWM_INTSTATUS - PWM Interrupt Status Register.....139
Table 16-36 - PWM_SAMPLE_FREQ_H - PWM Data Sampling Frequency High Byte Register.....139
Table 16-37 - PWM_SAMPLE_FREQ_L - PWM Data Sampling Frequency Low Byte Register .....140
Table 16-38 - PCM_VOLUME - PCM Volume Register.....140
Table 16-39 - PWM_BUFFER - PCM Buffer Register.....140
Table 17-1 - Overview of Data Capture Interface Registers .....142
Table 17-2 - RSADDR1 – Memory Start Address Register (LSB) .....142
Table 17-3 - RSADDR1 – Memory Start Address Register (Byte 1) .....142
Table 17-4 - RSADDR2 – Memory Start Address Register (MSB) .....142
Table 17-5 - FSADDR0 – Flash Start Address Register (LSB) .....143
Table 17-6 - FSADDR1 – Flash Start Address Register (Byte 1).....143
Table 17-7 - FSADDR2 – Flash Start Address Register (MSB).....143


Copyright © Bridgetek Pte Limited

Table 17-8 - BLENGTH0 – Data Byte Length Register (LSB) ..... 143
Table 17-9 - BLENGTH1 – Data Byte Length Register (Byte 1) ..... 143
Table 17-10 - BLENGTH2 – Data Byte Length Register (MSB) ..... 143
Table 17-11 - COMMAND – Command Register ..... 144
Table 17-12 - SEMAPHORE – Semaphore Register ..... 144
Table 17-13 - CONFIG – Configuration Register ..... 144
Table 17-14 - STATUS – Status Register ..... 145
Table 17-15 - CRCL – Flash Content CRC Register (LSB) ..... 145
Table 17-16 - CRCH – Flash Content CRC Register (MSB) ..... 145
Table 17-17 - CHIPID0 – Chip ID Register (LSB) ..... 145
Table 17-18 - CHIPID1 – Chip ID Register (Byte 1) ..... 145
Table 17-19 - CHIPID2 – Chip ID Register (Byte 2) ..... 146
Table 17-20 - CHIPID3 – Chip ID Register (MSB) ..... 146
Table 17-21 - DRWDATA – Data Register ..... 146
Table 17-22 - Flash Controller Command Group 1 ..... 148
Table 17-23 - Flash Controller Command Group 2 ..... 149

## List of Figures

Figure 2.1 - FT93x System Architecture ..... 13
Figure 2.2 - FT93x Boot Control ..... 15
Figure 2.3 - FT93x Debugging Support ..... 15
Figure 9.1 - ICR registers write access ..... 90
Figure 9.2 - ICR registers read access ..... 91


Copyright © Bridgetek Pte Limited

# Appendix C – Revision History

Document Title: BRT_AN_010 FT93x_User_Manual
Document Reference No.: BRT_000053
Clearance No.: BRT#067
Product Page: https://brtchip.com/ft9xx-toolchain/
Document Feedback: Send Feedback

| Revision | Changes | Date |
|---|---|---|
| 1.0 | Initial release | 02-03-2017 |
| 1.1 | Updated Release | 09-03-2017 |
| 1.2 | Updated as per Ver.2.7.0 Release;<br>Updated company address | 29-08-2023 |
| 1.3 | Updated for v2.7.6 release | 16-10-2023 |


Copyright © Bridgetek Pte Limited