**Welcome to E-XFL.COM**

## What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

## Applications of "Embedded - Microcontrollers"

| Details | |
|---|---|
| Product Status | Active |
| Core Processor | ARM® Cortex®-M7 |
| Core Size | 32-Bit Single-Core |
| Speed | 300MHz |
| Connectivity | EBI/EMI, I²C, IrDA, LINbus, MMC/SD/SDIO, QSPI, SPI, UART/USART, USB |
| Peripherals | Brown-out Detect/Reset, DMA, I²S, POR, PWM, WDT |
| Number of I/O | 114 |
| Program Memory Size | 1MB (1M x 8) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 384K x 8 |
| Voltage - Supply (Vcc/Vdd) | 1.08V ~ 3.6V |
| Data Converters | A/D 24x12b; D/A 2x12b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 105°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 144-LQFP |
| Supplier Device Package | 144-LQFP (20x20) |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/atsams70q20b-an |

| Signal Name | Function | Type | Active Level | Voltage Reference | Comments |
|---|---|---|---|---|---|
| PCK0–PCK2 | Programmable Clock Output | Output | – | | – |
| Real Time Clock | | | | | |
| RTCOUT0 | Programmable RTC Waveform Output | Output | – | VDDIO | – |
| RTCOUT1 | Programmable RTC Waveform Output | Output | – | | – |
| Serial Wire Debug/JTAG Boundary Scan | | | | | |
| SWCLK/TCK | Serial Wire Clock / Test Clock (Boundary scan mode only) | Input | – | VDDIO | – |
| TDI | Test Data In (Boundary scan mode only) | Input | – | | – |
| TDO/TRACESWO | Test Data Out (Boundary scan mode only) | Output | – | | – |
| SWDIO/TMS | Serial Wire Input/ Output / Test Mode Select (Boundary scan mode only) | I/O / Input | – | | – |
| JTAGSEL | JTAG Selection | Input | High | | – |
| Trace Debug Port | | | | | |
| TRACECLK | Trace Clock | Output | – | VDDIO | PCK3 is used for ETM |
| TRACED0–TRACED3 | Trace Data | Output | – | | – |
| Flash Memory | | | | | |
| ERASE | Flash and NVM Configuration Bits Erase Command | Input | High | VDDIO | – |
| Reset/Test | | | | | |
| NRST | Synchronous Microcontroller Reset | I/O | Low | VDDIO | – |
| TST | Test Select | Input | – | | – |
| Universal Asynchronous Receiver Transceiver - UART(x=[0:4]) | | | | | |

### 19.4.3 Bus Matrix Priority Registers A For Slaves

**Name:** MATRIX_PRASx
**Offset:** 0x80 + x*0x08 [x=0..8]
**Reset:** 0x00000222
**Property:** Read/Write

This register can only be written if the WPE bit is cleared in the Write Protection Mode Register.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | M7PR[1:0] | | | | M6PR[1:0] | |
| Access | | | R/W | R/W | | | R/W | R/W |
| Reset | | | 0 | 0 | | | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | M5PR[1:0] | | | | M4PR[1:0] | |
| Access | | | R/W | R/W | | | R/W | R/W |
| Reset | | | 0 | 0 | | | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | M3PR[1:0] | | | | M2PR[1:0] | |
| Access | | | R/W | R/W | | | R/W | R/W |
| Reset | | | 0 | 0 | | | 1 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | M1PR[1:0] | | | | M0PR[1:0] | |
| Access | | | R/W | R/W | | | R/W | R/W |
| Reset | | | 1 | 0 | | | 1 | 0 |

**Bits 0:1, 4:5, 8:9, 12:13, 16:17, 20:21, 24:25, 28:29 – MxPR** Master x Priority
Fixed priority of Master x for accessing the selected slave. The higher the number, the higher the priority.

All the masters programmed with the same MxPR value for the slave make up a priority pool.

Round-robin arbitration is used in the lowest (MxPR = 0) and highest (MxPR = 3) priority pools.

Fixed priority is used in intermediate priority pools (MxPR = 1) and (MxPR = 2).

See “Arbitration Priority Scheme” for details.

| Chip Name | CHIPID_CIDR (see Notes 1 and 2) | CHIPID_EXID |
|---|---|---|
| SAMS70N21 | 0xA112_0E0x | 0x00000001 |
| SAMS70N20 | 0xA112_0C0x | 0x00000001 |
| SAMS70N19 | 0xA11D_0A0x | 0x00000001 |
| SAMS70J21 | 0xA112_0E0x | 0x00000000 |
| SAMS70J20 | 0xA112_0C0x | 0x00000000 |
| SAMS70J19 | 0xA11D_0A0x | 0x00000000 |
| SAMV71Q21 | 0xA122_0E0x | 0x00000002 |
| SAMV71Q20 | 0xA122_0C0x | 0x00000002 |
| SAMV71Q19 | 0xA12D_0A0x | 0x00000002 |
| SAMV71N21 | 0xA122_0E0x | 0x00000001 |
| SAMV71N20 | 0xA122_0C0x | 0x00000001 |
| SAMV71N19 | 0xA12D_0A0x | 0x00000001 |
| SAMV71J21 | 0xA122_0E0x | 0x00000000 |
| SAMV71J20 | 0xA122_0C0x | 0x00000000 |
| SAMV71J19 | 0xA12D_0A0x | 0x00000000 |
| SAMV70Q20 | 0xA132_0C0x | 0x00000002 |
| SAMV70Q19 | 0xA13D_0A0x | 0x00000002 |
| SAMV70N20 | 0xA132_0C0x | 0x00000001 |
| SAMV70N19 | 0xA13D_0A0x | 0x00000001 |
| SAMV70J20 | 0xA132_0C0x | 0x00000000 |
| SAMV70J19 | 0xA13D_0A0x | 0x00000000 |

1. x = 0 for MRL A devices.
2. x = 1 for MRL B devices.

3. When Flash programming is completed, the bit EEFC_FSR.FRDY rises. If an interrupt has been enabled by setting the bit EEFC_FMR.FRDY, the interrupt line of the EEFC is activated.

Three errors can be detected in EEFC_FSR after a programming sequence:

- Command Error: A bad keyword has been written in EEFC_FCR.
- Lock Error: The page to be programmed belongs to a locked region. A command must be run previously to unlock the corresponding region.
- Flash Error: When programming is completed, the WriteVerify test of the Flash memory has failed.

Only one page can be programmed at a time. It is possible to program all the bits of a page (full page programming) or only some of the bits of the page (partial page programming).

Depending on the number of bits to be programmed within the page, the EEFC adapts the write operations required to program the Flash.

When a ‘Write Page’ (WP) command is issued, the EEFC starts the programming sequence and all the bits written at 0 in the latch buffer are cleared in the Flash memory array.

During programming, i.e., until EEFC_FSR.FDRY rises, access to the Flash is not allowed.

#### 22.4.3.2.1 Full Page Programming

To program a full page, all the bits of the page must be erased before writing the latch buffer and issuing the WP command. The latch buffer must be written in ascending order, starting from the first address of the page. See Figure 22-8.

#### 22.4.3.2.2 Partial Page Programming

To program only part of a page using the WP command, the following constraints must be respected:

- Data to be programmed must be contained in integer multiples of 128-bit address-aligned words.
- 128-bit words can be programmed only if all the corresponding bits in the Flash array are erased (at logical value ‘1’).

See 22.4.3.2.4 Programming Bytes.

#### 22.4.3.2.3 Optimized Partial Page Programming

The EEFC automatically detects the number of 128-bit words to be programmed. If only one 128-bit aligned word is to be programmed in the Flash array, the process is optimized to reduce the time needed for programming.

If several 128-bit words are to be programmed, a standard page programming operation is performed.

See Figure 22-10.

#### 22.4.3.2.4 Programming Bytes

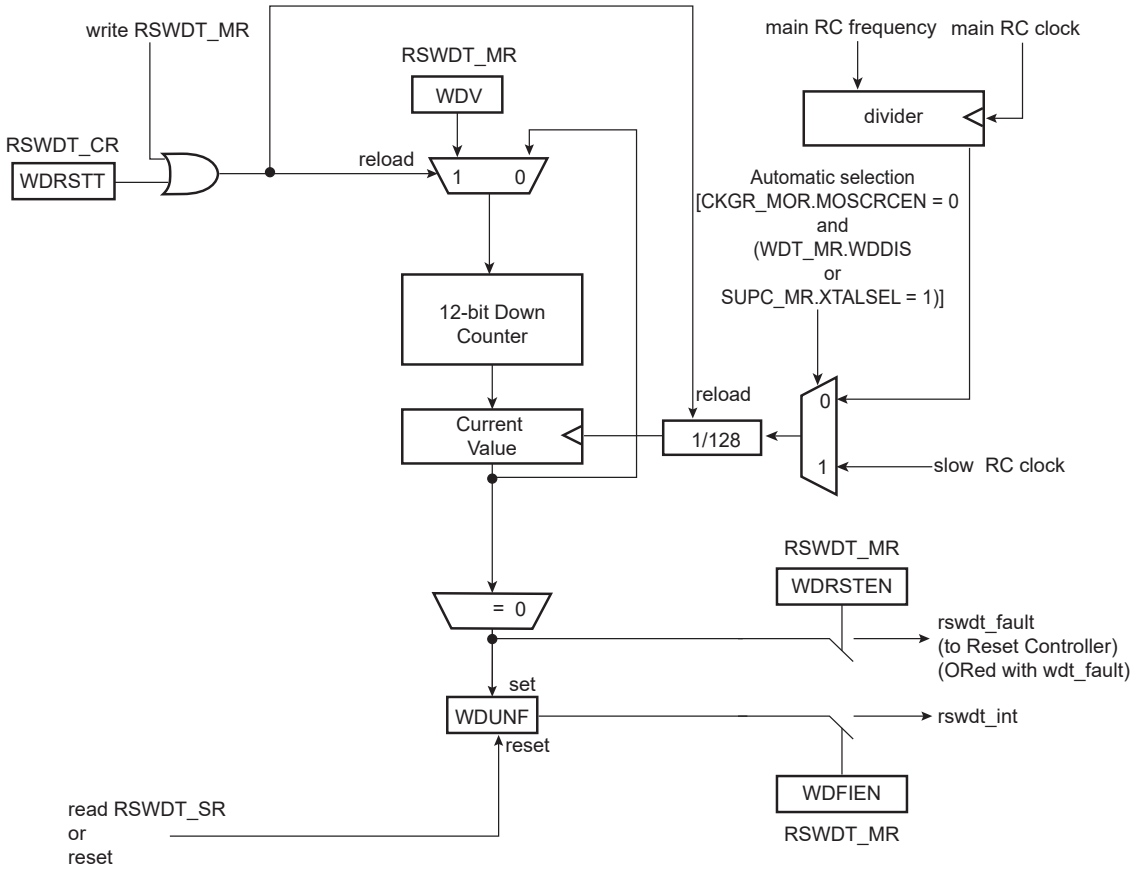
Individual bytes can be programmed using the Partial Page Programming mode.

In this case, an area of 128 bits must be reserved for each byte.

Refer to Figure 22-11

## 25.3 Block Diagram

**Figure 25-1. Reinforced Safety Watchdog Timer Block Diagram**




## 25.4 Functional Description

The RSWDT is supplied by VDDCORE. The RSWDT is initialized with default values on processor reset or on a power-on sequence and is disabled (its default mode) under such conditions.

The RSWDT must not be enabled if the WDT is disabled.

The Main RC oscillator divided clock is selected if the Main RC oscillator is already enabled by the application (CKGR_MOR.MOSCRCEN = 1) or if the WDT is driven by the Slow RC oscillator.

The RSWDT is built around a 12-bit down counter, which is loaded with a slow clock value other than that of the slow clock in the WDT, defined in the WDV (Watchdog Counter Value) field of the Mode Register (RSWDT_MR). The RSWDT uses the slow clock divided by 128 to establish the maximum watchdog period to be 16 seconds (with a typical slow clock of 32.768 kHz).

After a processor reset, the value of RSWDT_MR.WDV is 0xFFF, corresponding to the maximum value of the counter with the external reset generation enabled (RSWDT_MR.WDRSTEN = 1 after a backup reset). This means that a default watchdog is running at reset, i.e., at power-up.

If the watchdog is restarted by writing into the Control Register (RSWDT_CR), the RSWDT_MR must not be programmed during a period of time of three slow clock periods following the RSWDT_CR write access. Programming a new value in the RSWDT_MR automatically initiates a restart instruction.

### 26.4.5.3 RSTC Mode Register

**Name:** RSTC_MR
**Offset:** 0x08
**Reset:** 0x00000001
**Property:** Read/Write

This register can only be written if the WPEN bit is cleared in the System Controller Write Protection Mode Register (SYSC_WPMR).

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | KEY[7:0] | | | | | | | |
| Access | W | W | W | W | W | W | W | W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | ERSTL[3:0] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | URSTIEN | | | | URSTEN |
| Access | | | | R/W | | | | R/W |
| Reset | | | | 0 | | | | 1 |

**Bits 31:24 – KEY[7:0]** Write Access Password

| Value | Name | Description |
|---|---|---|
| 0xA5 | PASSWD | Writing any other value in this field aborts the write operation. Always reads as 0. |

**Bits 11:8 – ERSTL[3:0]** External Reset Length
This field defines the external reset length. The external reset is asserted during a time of $2^{(ERSTL+1)}$ SLCK cycles. This allows assertion duration to be programmed between 60 µs and 2 seconds. Note that synchronization cycles must also be considered when calculating the actual reset length as previously described.

**Bit 4 – URSTIEN** User Reset Interrupt Enable

| Value | Description |
|---|---|
| 0 | RSTC_SR.USRTS at ‘1’ has no effect on the RSTC interrupt line. |
| 1 | RSTC_SR.USRTS at ‘1’ asserts the RSTC interrupt line if URSTEN = 0. |

**Bit 0 – URSTEN** User Reset Enable

| Peripheral Name | Transfer Type | HW Interface Number (XDMAC_CC.PERID) |
|---|---|---|
| I2SC0 | Receive Left | 45 |
| I2SC1 | Transmit Left | 46 |
| I2SC1 | Receive Left | 47 |
| I2SC0 | Transmit Right | 48 |
| I2SC0 | Receive Right | 49 |
| I2SC1 | Transmit Right | 50 |
| I2SC1 | Receive Right | 51 |

## 36.5 Functional Description

### 36.5.1 Basic Definitions

**Source Peripheral**: Slave device, memory mapped on the interconnection network, from where the XDMAC reads data. The source peripheral teams up with a destination peripheral to form a channel. A data read operation is scheduled when the peripheral transfer request is asserted.

**Destination Peripheral**: Slave device, memory mapped on the interconnection network, to which the XDMAC writes. A write data operation is scheduled when the peripheral transfer request is asserted.

**Channel**: The data movement between source and destination creates a logical channel.

**Transfer Type**: The transfer is hardware-synchronized when it is paced by the peripheral hardware request, otherwise the transfer is self-triggered (memory to memory transfer).

### 36.5.2 Transfer Hierarchy Diagram

**XDMAC Master Transfer**: The Master Transfer is composed of a linked list of blocks. The channel address, control and configuration registers can be modified at the inter block boundary. The descriptor structure modifies the channel registers conditionally. Interrupts can be generated on a per block basis or when the end of linked list event occurs.

**XDMAC Block**: An XDMAC block is composed of a programmable number of microblocks. The channel configuration registers remain unchanged at the inter microblock boundary. The source and destination addresses are conditionally updated with a programmable signed number.

**XDMAC Microblock**: The microblock is composed of a programmable number of data. The channel configuration registers remain unchanged at the data boundary. The data address may be fixed (a FIFO location, a peripheral transmit or receive register), incrementing (a memory-mapped area) by a programmable signed number.

**XDMAC Burst and Incomplete Burst**: In order to improve the overall performance when accessing dynamic external memory, burst access is mandatory. Each data of the microblock is considered as a part of a memory burst. The programmable burst value indicates the largest memory burst allowed on a per channel basis. When the microblock length is not an integral multiple of the burst size, an incomplete burst is performed to read or write the last trailing bytes.

**XDMAC Chunk and Incomplete Chunk**: When a peripheral synchronized transfer is activated, the microblock splits into a number of data chunks. The chunk size is programmable. The larger the chunk is,

| Offset | Name | Bit Pos. | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x01AC | GMAC_TCE | 7:0 | TCKER[7:0] | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x01B0 | GMAC_UCE | 7:0 | UCKER[7:0] | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x01B4 ... 0x01BB | Reserved | | | | | | | | | |
| 0x01BC | GMAC_TISUBN | 7:0 | LSBTIR[7:0] | | | | | | | |
| | | 15:8 | LSBTIR[15:8] | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x01C0 | GMAC_TSH | 7:0 | TCS[7:0] | | | | | | | |
| | | 15:8 | TCS[15:8] | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x01C4 ... 0x01CF | Reserved | | | | | | | | | |
| 0x01D0 | GMAC_TSL | 7:0 | TCS[7:0] | | | | | | | |
| | | 15:8 | TCS[15:8] | | | | | | | |
| | | 23:16 | TCS[23:16] | | | | | | | |
| | | 31:24 | TCS[31:24] | | | | | | | |
| 0x01D4 | GMAC_TN | 7:0 | TNS[7:0] | | | | | | | |
| | | 15:8 | TNS[15:8] | | | | | | | |
| | | 23:16 | TNS[23:16] | | | | | | | |
| | | 31:24 | | | TNS[29:24] | | | | | |
| 0x01D8 | GMAC_TA | 7:0 | ITDT[7:0] | | | | | | | |
| | | 15:8 | ITDT[15:8] | | | | | | | |
| | | 23:16 | ITDT[23:16] | | | | | | | |
| | | 31:24 | ADJ | | ITDT[29:24] | | | | | |
| 0x01DC | GMAC_TI | 7:0 | CNS[7:0] | | | | | | | |
| | | 15:8 | ACNS[7:0] | | | | | | | |
| | | 23:16 | NIT[7:0] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x01E0 | GMAC_EFTSL | 7:0 | RUD[7:0] | | | | | | | |
| | | 15:8 | RUD[15:8] | | | | | | | |
| | | 23:16 | RUD[23:16] | | | | | | | |
| | | 31:24 | RUD[31:24] | | | | | | | |
| 0x01E4 | GMAC_EFTN | 7:0 | RUD[7:0] | | | | | | | |
| | | 15:8 | RUD[15:8] | | | | | | | |
| | | 23:16 | RUD[23:16] | | | | | | | |
| | | 31:24 | | | RUD[29:24] | | | | | |

### 38.8.73 GMAC Jabbers Received Register

**Name:** GMAC_JR
**Offset:** 0x18C
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | JRX[9:8] | |
| Access | | | | | | | R | R |
| Reset | | | | | | | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | JRX[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 9:0 – JRX[9:0]** Jabbers Received
This bit field counts the number of frames received exceeding 1518 Bytes in length (1536 Bytes if GMAC_NCFGR.MAXFS is written to '1') and have either a CRC error, an alignment error or a receive symbol error.

### 38.8.103 GMAC Receive Buffer Queue Base Address Register Priority Queue x

**Name:** GMAC_RBQBAPQx
**Offset:** 0x0480 + x*0x04 [x=0..4]
**Reset:** 0x00000000
**Property:** Read/Write

These registers hold the start address of the receive buffer queues (receive buffers descriptor lists) for the additional queues used when priority queues are employed.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | RXBQBA[29:22] | | | | | | | |
| Access | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | RXBQBA[21:14] | | | | | | | |
| Access | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | RXBQBA[13:6] | | | | | | | |
| Access | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | RXBQBA[5:0] | | | | | | | |
| Access | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | | |

**Bits 31:2 – RXBQBA[29:0]** Receive Buffer Queue Base Address
Holds the address of the start of the receive queue.

**Bit 9 – RESET** Send USB Reset
This bit is cleared when the USB Reset has been sent.

It may be useful to write a zero to this bit when a device disconnection is detected (USBHS_HSTISR.DDISCI = 1) whereas a USB Reset is being sent.

| Value | Description |
|---|---|
| 0 | No effect. |
| 1 | Generates a USB Reset on the USB bus. |

**Bit 8 – SOFE** Start of Frame Generation Enable
This bit is set when a USB reset is requested or an upstream resume interrupt is detected (USBHS_HSTISR.TXRSMI).

| Value | Description |
|---|---|
| 0 | Disables the SOF generation and leaves the USB bus in idle state. |
| 1 | Generates SOF on the USB bus in Full- or High-speed mode and sends “keep alive” signals in Low-speed mode. |

### 39.6.58 Host Pipe x Disable Register (Interrupt Pipes)

**Name:** USBHS_HSTPIPIDRx (INTPIPES)
**Offset:** 0x0620 + x*0x04 [x=0..9]
**Reset:** 0
**Property:** Read/Write

This register view is relevant only if PTYPE = 0x3 in "Host Pipe x Configuration Register".

For additional information, see "Host Pipe x Mask Register (Interrupt Pipes)".

This register always reads as zero.

The following configuration values are valid for all listed bit names of this register:

0: No effect.

1: Clears the corresponding bit in USBHS_HSTPIPIMRx.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | PFREEZEC | PDISHDMAC |
| Access | | | | | | | | |
| Reset | | | | | | | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | FIFOCONC | | NBUSYBKEC | | | | |
| Access | | | | | | | | |
| Reset | | 0 | | 0 | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SHORTPACKETIEC | RXSTALLDEC | OVERFIEC | NAKEDEC | PERREC | UNDERFIEC | TXOUTEC | RXINEC |
| Access | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 17 – PFREEZEC** Pipe Freeze Disable

**Bit 16 – PDISHDMAC** Pipe Interrupts Disable HDMA Request Disable

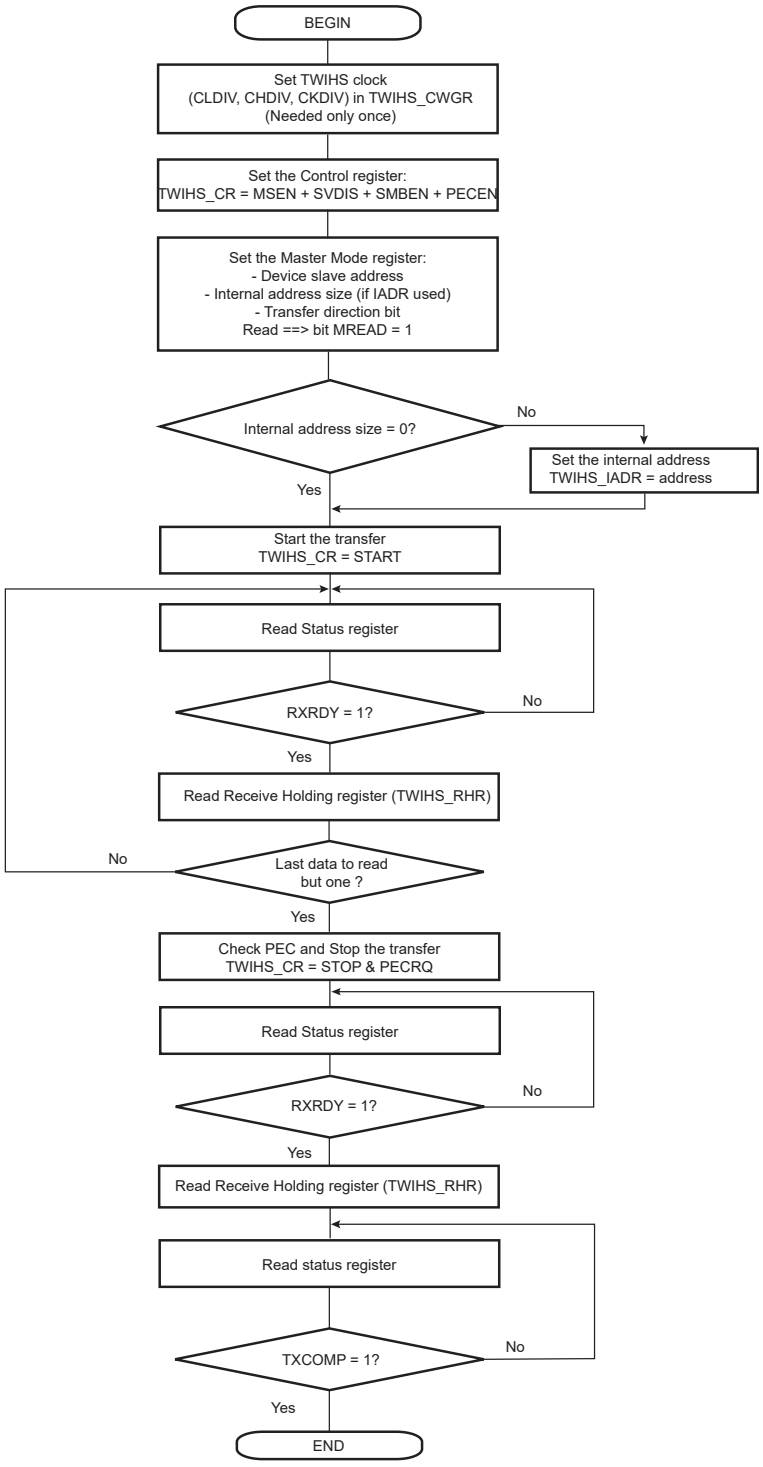
**Bit 14 – FIFOCONC** FIFO Control Disable

**Bit 12 – NBUSYBKEC** Number of Busy Banks Disable

**Bit 7 – SHORTPACKETIEC** Short Packet Interrupt Disable

**Bit 6 – RXSTALLDEC** Received STALLed Interrupt Disable

**Figure 43-24. TWIHS Read Operation with Multiple Data Bytes with or without Internal Address with PEC**



#### 46.6.10.8.2 Beta1 Tx/Rx

Beta1 is the period immediately following the end of a packet cycle (see Figure 46-59). A node attempting to transmit monitors the state of the channel, and if it detects no transmission during the Beta1 period, it determines the channel to be idle.

The Beta1 value is different depending on the previous packet type (received packet or transmitted packet).

Beta1Rx and Beta1Tx length can be configured respectively through the USART LON Beta1 Rx register (US_LONB1RX) and the USART LON Beta1 Tx register (US_LONB1TX). Note that a length of '0' is not allowed.

#### 46.6.10.8.3 Pcycle Timer

The packet cycle timer is reset to its initial value whenever the backlog is changed. It is started (begins counting down at its current value) whenever the MAC layer becomes idle. An idle MAC layer is defined as:

- Not receiving
- Not transmitting
- Not waiting to transmit
- Not timing Beta1
- Not waiting for priority slots, and not waiting for the first Wbase randomizing window to complete

On transition from idle to either transmit or receive, the packet cycle timer is halted.

The pcycle timer value can be configured in US_TTGR. Note that '0' value is not allowed.

#### 46.6.10.8.4 Wbase

The wbase timer represents the base windows size. Its duration, derived from Beta2, equals 16 Beta2 slots.

#### 46.6.10.8.5 Priority Slots

On a channel by channel basis, the protocol supports optional priority. Priority slots, if any, follow immediately after the Beta1 period that follows the transmission of a packet (see Figure 46-59). The number of priority slots per channel ranges from 0 to 127.

The number of priority slots in the LON network configuration is defined through the PSNB field of the USART LON Priority register (US_LONPRIO). And the priority slot affected to the LON node, if any, is defined through US_LONPRIO.NPS.

#### 46.6.10.8.6 Indeterminate Time

See "comm_type".

Like Beta1, the IDT value is different depending on what was the previous frame (transmitted or received frame).

IDTRx and IDTTx can be configured respectively through the USART LON IDT Rx register (US_LONIDTRX) and the USART LON IDT Tx register (US_LONIDTTX).

#### 46.6.10.8.7 End of Frame Condition

The USART configured in LON mode terminates the frame with a 3 $t_{bit}$ long Manchester code violation. After sending the last CRC bit, it maintains the data transitionless during three bit periods.

While receiving data the USART configured in LON mode will detect an end of frame condition after a $t_{eof}$ transitionless Manchester code violation. US_LONMR.EOFS can configure $t_{eof}$.

### 46.7.14 USART Interrupt Mask Register (SPI_MODE)

**Name:** US_IMR (SPI_MODE)
**Offset:** 0x0010
**Reset:** 0x0
**Property:** Read-only

This configuration is relevant only if USART_MODE = 0xE or 0xF in the USART Mode Register.

The following configuration values are valid for all listed bit names of this register:

0: The corresponding interrupt is not enabled.

1: The corresponding interrupt is enabled.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | 0 | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | 0 | 0 | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | 0 | | | | 0 | 0 |

**Bit 19 – NSSE** NSS Line (Driving CTS Pin) Rising or Falling Edge Event Interrupt Mask

**Bit 10 – UNRE** SPI Underrun Error Interrupt Mask

**Bit 9 – TXEMPTY** TXEMPTY Interrupt Mask

**Bit 5 – OVRE** Overrun Error Interrupt Mask

**Bit 1 – TXRDY** TXRDY Interrupt Mask

**Bit 0 – RXRDY** RXRDY Interrupt Mask

### 46.7.19 USART Channel Status Register (LIN_MODE)

**Name:** US_CSR (LIN_MODE)
**Offset:** 0x0014
**Reset:** 0x0
**Property:** Read-only

This configuration is relevant only if USART_MODE = 0xA or 0xB in the USART Mode Register.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | LINHTE | LINSTE | LINSNRE | LINCE | LINIPE | LINISFE | LINBE | |
| Access | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | LINBLS | | | | | | | |
| Access | | | | | | | | |
| Reset | 0 | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | LINTC | LINID | LINBK | | | | TXEMPTY | TIMEOUT |
| Access | | | | | | | | |
| Reset | 0 | 0 | 0 | | | | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PARE | FRAME | OVRE | | | | TXRDY | RXRDY |
| Access | | | | | | | | |
| Reset | 0 | 0 | 0 | | | | 0 | 0 |

**Bit 31 – LINHTE** LIN Header Timeout Error (cleared by writing a one to bit US_CR.RSTSTA)

| Value | Description |
|---|---|
| 0 | No LIN header timeout error has been detected since the last RSTSTA. |
| 1 | A LIN header timeout error has been detected since the last RSTSTA. |

**Bit 30 – LINSTE** LIN Synch Tolerance Error (cleared by writing a one to bit US_CR.RSTSTA)

| Value | Description |
|---|---|
| 0 | No LIN synch tolerance error has been detected since the last RSTSTA. |
| 1 | A LIN synch tolerance error has been detected since the last RSTSTA. |

**Bit 29 – LINSNRE** LIN Slave Not Responding Error (cleared by writing a one to bit US_CR.RSTSTA)

| Value | Description |
|---|---|
| 0 | No LIN slave not responding error has been detected since the last RSTSTA. |
| 1 | A LIN slave not responding error has been detected since the last RSTSTA. |

**Bit 28 – LINCE** LIN Checksum Error (cleared by writing a one to bit US_CR.RSTSTA)

| Value | Description |
|---|---|
| 0 | Detect collisions after CRC has been sent but prior end of transmission in LON comm_type = 1 mode. |
| 1 | Ignore collisions after CRC has been sent but prior end of transmission in LON comm_type = 1 mode. |

**Bit 2 – TCOL** Terminate Frame upon Collision Notification

| Value | Description |
|---|---|
| 0 | Do not terminate the frame in LON comm_type = 1 mode upon collision detection. |
| 1 | Terminate the frame in LON comm_type = 1 mode upon collision detection if possible. |

**Bit 1 – COLDET** LON Collision Detection Feature

| Value | Description |
|---|---|
| 0 | LON collision detection feature disabled. |
| 1 | LON collision detection feature enabled. |

**Bit 0 – COMMT** LON comm_type Parameter Value

| Value | Description |
|---|---|
| 0 | LON comm_type = 1 mode. |
| 1 | LON comm_type = 2 mode. |

### 51.7.42 PWM Channel Duty Cycle Update Register

**Name:** PWM_CDTYUPDx
**Offset:** 0x0208 + x*0x20 [x=0..3]
**Reset:** –
**Property:** Write-only

This register acts as a double buffer for the CDTY value. This prevents an unexpected waveform when modifying the waveform duty-cycle.

Only the first 16 bits (channel counter size) are significant.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | CDTYUPD[23:16] | | | | | | | |
| Access | W | W | W | W | W | W | W | W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | CDTYUPD[15:8] | | | | | | | |
| Access | W | W | W | W | W | W | W | W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CDTYUPD[7:0] | | | | | | | |
| Access | W | W | W | W | W | W | W | W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | – |

**Bits 23:0 – CDTYUPD[23:0]** Channel Duty-Cycle Update
Defines the waveform duty-cycle. This value must be defined between 0 and CPRD (PWM_CPRDx).

### 54.7.5 ACC Interrupt Mask Register

**Name:** ACC_IMR
**Offset:** 0x2C
**Reset:** 0x00000000
**Property:** Read-only

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | CE |
| Access | | | | | | | | R |
| Reset | | | | | | | | 0 |

**Bit 0 – CE** Comparison Edge

| Value | Description |
|---|---|
| 0 | The interrupt is disabled. |
| 1 | The interrupt is enabled. |

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $t_S$ | VTEMP Settling Time | When $V_{TEMP}$ is sampled by the AFEC, the required track-and-hold time to ensure 1°C accurate settling | – | – | 1 | µs |
| – | Temperature Accuracy | After offset calibration over $T_A$ range [-40°C : +105°C] | -10 | – | 10 | °C |
| $t_{START}$ | Startup Time | – | – | – | 30 | µs |
| $I_{VDDIN}$ | Current Consumption | – | – | 130 | 270 | µA |

**Note:** AFE Gain Error and Offset error considered calibrated. This calibration at ambient temperature is not a feature of the product and is performed by the user's application.

## 58.11 12-bit DAC Characteristics

**Table 58-43. Analog Power Supply Characteristics**

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $I_{VDDIN}$ | Current Consumption | Sleep mode (Clock OFF) | – | 10 | – | µA |
| | | Normal mode with one output on,<br>DACC_ACR.IBCTLCHx =3 (see **Note 1**)<br>FS = 1 MSps, no $R_{LOAD}$, $V_{DDIN}$ = 3.3V | – | 200 | 800 | |
| | | Normal mode with one output on,<br>DACC_ACR.IBCTLCHx =1 (see **Note 1**)<br>FS = 500 KSps, no $R_{LOAD}$, $V_{DDIN}$ = 3.3V | – | 100 | 400 | |
| | | Bypass mode (output buffer off) with one output on,<br>DACC_ACR.IBCTLCHx =0 (see **Note 1**)<br>FS = 500 KSps, no $R_{LOAD}$, $V_{DDIN}$ = 3.3V | – | 10 | 30 | |
| PSRR | Power Supply RejectionRatio ($V_{DDIN}$) | $V_{DDIN}$ ±10 mV<br>Up to 10 kHz | – | 70 | – | dB |

**Note:**

1. The maximum conversion rate versus the configuration of DACC_ACR.IBCTL is shown in the following table.

**Table 58-44. Maximum Conversion Rate vs. Configuration of DACC_ACR.IBCTL**

| DACC_ACR.IBCTLCHx | Maximum Conversion Rate |
|---|---|
| 0 | Bypass |
| 1 | 500 ks/s |