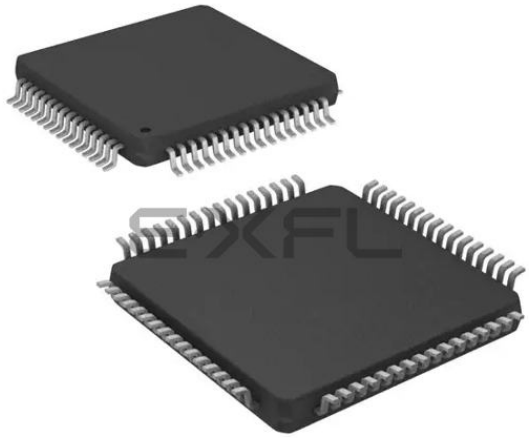


**Welcome to E-XFL.COM**

## 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®-M4F |
| Core Size | 32-Bit Single-Core |
| Speed | 120MHz |
| Connectivity | EBI/EMI, I²C, IrDA, LINbus, MMC/SD, QSPI, SPI, UART/USART, USB |
| Peripherals | Brown-out Detect/Reset, DMA, I²S, POR, PWM |
| Number of I/O | 51 |
| Program Memory Size | 256KB (256K x 8) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 128K x 8 |
| Voltage - Supply (Vcc/Vdd) | 1.71V ~ 3.63V |
| Data Converters | A/D 24x12b; D/A 2x12b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 64-TQFP |
| Supplier Device Package | 64-TQFP (10x10) |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/atsamd51j18a-au |

### 12.13.5 Length

**Name:** LENGTH
**Offset:** 0x0008
**Reset:** 0x00000000
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | LENGTH[29:22] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | LENGTH[21:14] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | LENGTH[13:6] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

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

**Bits 31:2 – LENGTH[29:0]** Length
Length in words needed for memory operations.

### 24.9.2 GMAC Network Configuration Register

**Name:** NCFGR
**Offset:** 0x004
**Reset:** 0x00080000
**Property:** R/W

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | IRXER | RXBP | IPGSEN | | IRXFCS | EFRHD | RXCOEN |
| Access | | R/W | R/W | R/W | | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | DCPF | | | CLK[2:0] | | | RFCS | LFERD |
| Access | R/W | | | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | | | 0 | 1 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | RXBUFO[1:0] | | PEN | RTY | | | | MAXFS |
| Access | R/W | R/W | R/W | R/W | | | | R/W |
| Reset | 0 | 0 | 0 | 0 | | | | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | UNIHEN | MTIHEN | NBC | CAF | JFRAME | DNVLAN | FD | SPD |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 30 – IRXER** Ignore IPG GRXER
When this bit is written to '1', the Receive Error signal (GRXER) has no effect on the GMAC operation when Receive Data Valid signal (GRXDV) is low.

**Bit 29 – RXBP** Receive Bad Preamble
When written to '1', frames with non-standard preamble are not rejected.

**Bit 28 – IPGSEN** IP Stretch Enable
Writing a '1' to this bit allows the transmit IPG to increase above 96 bit times, depending on the previous frame length using the IPG Stretch Register.

**Bit 26 – IRXFCS** Ignore RX FCS
For normal operation this bit must be written to zero.

When this bit is written to '1', frames with FCS/CRC errors will not be rejected. FCS error statistics will still be collected for frames with bad FCS, and FCS status will be recorded in the DMA descriptor of the frame.

**Bit 25 – EFRHD** Enable Frames Received in half-duplex
Writing a '1' to this bit enables frames to be received in half-duplex mode while transmitting.

### 24.9.39 GMAC Octets Transmitted High Register

**Name:** OTHI
**Offset:** 0x104
**Reset:** 0x00000000
**Property:** Read-Only

When reading the Octets Transmitted and Octets Received Registers, bits [31:0] should be read prior to bits [47:32] to ensure reliable operation.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | TXO[15:8] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

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

**Bits 15:0 – TXO[15:0]** Transmitted Octets
Transmitted octets in valid frames of any type without errors, bits [47:32]. This counter is 48-bits, and is read through two registers. This count does not include octets from automatically generated pause frames.

### 24.9.62 GMAC Multicast Frames Received Register

**Name:** MFR
**Offset:** 0x160
**Reset:** 0x00000000
**Property:** Read-only

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

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

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

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

**Bits 31:0 – MFRX[31:0]** Multicast Frames Received without Error
This register counts the number of multicast frames successfully received without error, excluding pause frames, and is only incremented if the frame is successfully filtered and copied to memory.

### 25.6.10 Security Bit

The security bit allows the entire chip to be locked from external access for code security.

**Related Links**

12. DSU - Device Service Unit

#### 25.6.10.1 Security Bit Set Procedure

1. Issue the Set Security Bit command (SSB)
   This command changes the NVM security bits. The device shadow registers are not changed at that point. If a debugger was connected, it will still have access to the device after issuing this command (DSU.STATUSB.PROT will still read '0').
2. Check NVMNCTRL.INTFLAG.PROGE and NVMNCTRL.INTFLAG.DONE.
3. Reset the NVMCTRL peripheral or the device.

To reflect the NVM security bits' state correctly, the NVMCTRL needs to replay the start-up procedure. This is done by issuing a SWRST command or by resetting the device.

**Related Links**

12. DSU - Device Service Unit

#### 25.6.10.2 Security Bit Clear Procedure

The only way to clear the security bit is through a debugger Chip Erase command. The NVM security bit is cleared after all internal volatile and NVM have been cleared. The device protection status is updated at the end of the command meaning that no reset is necessary.

**Related Links**

12. DSU - Device Service Unit

### 25.6.11 Line Cache

NVM reads 128-bit at a time. AHB0 and AHB1 interfaces implement each a 128-bit cache line.This reduces the device power consumption when reading continuous data and improves system performance when wait states are required. Line cache are enabled by default and can be individually disabled per AHB interface by writing a one in the CACHEDIS[0] or CACHEDIS[1] bit in the CTRLA register (CTRLA.CACHEDIS[1:0]). Refer to CTRLA register description for more details. Commands affecting NVM content automatically invalidate cache lines.

### 25.6.12 Error Correction Code (ECC)

Error Correcting Code (ECC) is implemented to detect and correct errors that may arise in the NVM array. ECC is by default enabled and cannot be disabled by the user.

# 26. ICM - Integrity Check Monitor

## 26.1 Overview

The Integrity Check Monitor (ICM) is a DMA controller that performs hash calculation over multiple memory regions through the use of transfer descriptors located in memory (ICM Descriptor Area). The Hash function is based on the Secure Hash Algorithm (SHA). The ICM controller integrates two modes of operation. The first one is used to hash a list of memory regions and save the digests to memory (ICM Hash Area). The second operation mode is an active monitoring of the memory. In that mode, the hash function is evaluated and compared to the digest located at a predefined memory address (ICM Hash Area). If a mismatch occurs, an interrupt is raised.

## 26.2 Features

- DMA AHB master interface
- Supports monitoring of up to four non-contiguous memory regions
- Supports block gathering through the use of linked list
- Supports Secure Hash Algorithm (SHA1, SHA224, SHA256)
- Compliant with FIPS Publication 180-2
- Configurable processing period:
  - When SHA1 algorithm is processed, the run-time period is either 85 or 209 clock cycles.
  - When SHA256 or SHA224 algorithm is processed, the run-time period is either 72 or 194 clock cycles.
- Programmable bus burden

- The flag RHC[i], i being the region index, is set (if RHIEN is '0') when the hash result is available at address defined in HASH.
- The flag REC[i], i being the region index, is set (if ECIEN is '0') when the hash result is available at the address defined in HASH.

An interrupt is generated if the bit RHC[i] is written to '1' in the IER (if RHC[i] is set in RCTRL of region i) or if the bit REC[i] is written to 1 in the IER (if REC[i] is set in RCTRL of region i).

#### 26.6.4.2 Processing Period

The SHA engine processing period can be configured by writing to the Region Configuration Structure Member register (RCFGn).

The short processing period allows to allocate bandwidth to the SHA module whereas the long processing period allocates more bandwidth on the system bus to other applications.

In SHA mode, the shortest processing period is 85 clock cycles + 2 clock cycles for start command synchronization. The longest period is 209 clock cycles + 2 clock cycles.

In SHA256 and SHA224 modes, the shortest processing period is 72 clock cycles + 2 clock cycles for start command synchronization. The longest period is 194 clock cycles + 2 clock cycles.

### 26.6.5 ICM Automatic Monitoring Mode

The ASCD bit of the CFG register is used to activate the ICM Automatic Mode. When CFG.ASCD is set, the ICM performs the following actions:

- The ICM controller passes through the Main List once with CDWBN bit in RCFGn at 0 (i.e., Write Back activated) and EOM bit in the RCFGn context register at 0.
- When RCFGn.WRAP=1, the ICM controller enters active monitoring, with CDWBN bit in context register now set, and EOM bit in context register cleared. Writing to the CDWBN and EOM bits in RCFGn has no effect.

### 26.6.6 ICM Configuration Parameters

| Transfer Type | | Main List | RCFG | | | RNEXT | Comments |
|---|---|---|---|---|---|---|---|
| | | | CDWBN | WRAP | EOM | NEXT | |
| Single Region | Contiguous list of blocks<br><br>Digest written to memory<br><br>Monitoring disabled | 1 item | 0 | 0 | 1 | 0 | The Main List contains only one descriptor. The Secondary List is empty for that descriptor. The digest is computed and saved to memory. |
| | Non-contiguous list of blocks<br><br>Digest written to memory | 1 item | 0 | 0 | 1 | Secondary List address of the current region identifier | The Main List contains only one descriptor. The Secondary List describes the layout of the non- |

# 33. SERCOM – Serial Communication Interface

## 33.1 Overview

There are up to eight instances of the serial communication interface (SERCOM) peripheral.

A SERCOM can be configured to support a number of modes: $I^2C$, SPI, and USART. When an instance of SERCOM is configured and enabled, all of the resources of that SERCOM instance will be dedicated to the selected mode.

The SERCOM serial engine consists of a transmitter and receiver, baud-rate generator and address matching functionality. It can use the internal generic clock or an external clock. Using an external clock allows the SERCOM to be operated in all Sleep modes.

**Related Links**

34. SERCOM USART - SERCOM Synchronous and Asynchronous Receiver and Transmitter
35. SERCOM SPI – SERCOM Serial Peripheral Interface
36. SERCOM I2C – Inter-Integrated Circuit
6.2.6 SERCOM I2C Configurations

## 33.2 Features

- Interface for configuring into one of the following:
  - Inter-Integrated Circuit ($I^2C$) Two-wire Serial Interface
  - System Management Bus (SMBus™) compatible
  - Serial Peripheral Interface (SPI)
  - Universal Synchronous/Asynchronous Receiver/Transmitter (USART)
- Single transmit buffer and double receive buffer
- Baud-rate generator
- Address match/mask logic
- Operational in all Sleep modes with an external clock source
- Can be used with DMA
- 32-bit extension for better system bus utilization

See the Related Links for full feature lists of the interface configurations.

**Related Links**

34. SERCOM USART - SERCOM Synchronous and Asynchronous Receiver and Transmitter
35. SERCOM SPI – SERCOM Serial Peripheral Interface
36. SERCOM I2C – Inter-Integrated Circuit
6.2.6 SERCOM I2C Configurations

**Bit 2 – RXC** Receive Complete Interrupt Enable
Writing '0' to this bit has no effect.

Writing '1' to this bit will set the Receive Complete Interrupt Enable bit, which enables the Receive Complete interrupt.

| Value | Description |
|---|---|
| 0 | Receive Complete interrupt is disabled. |
| 1 | Receive Complete interrupt is enabled. |

**Bit 1 – TXC** Transmit Complete Interrupt Enable
Writing '0' to this bit has no effect.

Writing '1' to this bit will set the Transmit Complete Interrupt Enable bit, which enables the Transmit Complete interrupt.

| Value | Description |
|---|---|
| 0 | Transmit Complete interrupt is disabled. |
| 1 | Transmit Complete interrupt is enabled. |

**Bit 0 – DRE** Data Register Empty Interrupt Enable
Writing '0' to this bit has no effect.

Writing '1' to this bit will set the Data Register Empty Interrupt Enable bit, which enables the Data Register Empty interrupt.

| Value | Description |
|---|---|
| 0 | Data Register Empty interrupt is disabled. |
| 1 | Data Register Empty interrupt is enabled. |

**Figure 36-2. Transaction Diagram Symbols**

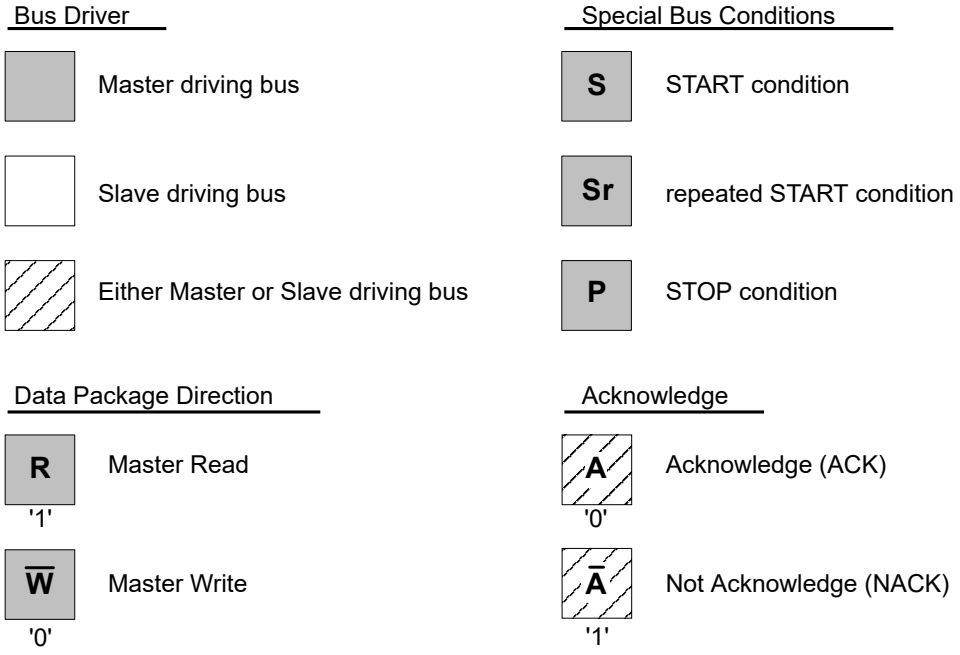
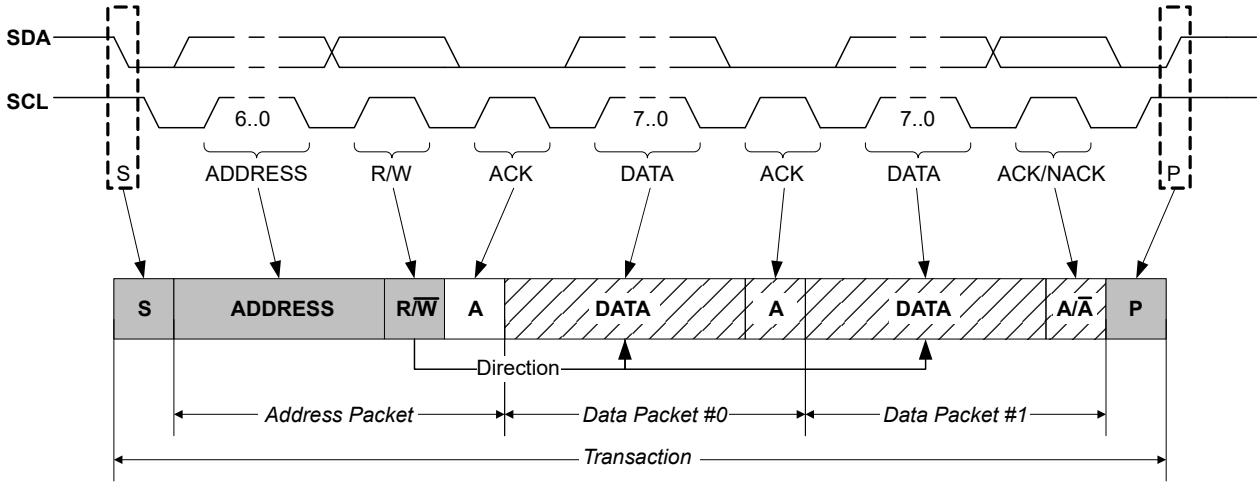



**Figure 36-3. Basic I²C Transaction Diagram**




### 36.6.2 Basic Operation

#### 36.6.2.1 Initialization

The following registers are enable-protected, meaning they can be written only when the I²C interface is disabled (CTRLA.ENABLE is '0'):

- Control A register (CTRLA), except Enable (CTRLA.ENABLE) and Software Reset (CTRLA.SWRST) bits
- Control B register (CTRLB), except Acknowledge Action (CTRLB.ACKACT) and Command (CTRLB.CMD) bits
- Baud register (BAUD)
- Address register (ADDR) in slave operation.

When the I²C is enabled or is being enabled (CTRLA.ENABLE=1), writing to these registers will be discarded. If the I²C is being disabled, writing to these registers will be completed after the disabling.

## 36.8 Register Description - I²C Slave

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16- and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers are optionally write-protected by the Peripheral Access Controller (PAC). Optional PAC write-protection is denoted by the "PAC Write-Protection" property in each individual register description. For details, refer to 36.5.8 Register Access Protection.

Some registers are synchronized when read and/or written. Synchronization is denoted by the "Write-Synchronized" or the "Read-Synchronized" property in each individual register description. For details, refer to 36.6.6 Synchronization.

Some registers are enable-protected, meaning they can only be written when the peripheral is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.

### 36.10.3 Control C

**Name:** CTRLC
**Offset:** 0x08
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | DATA32B |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

| 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 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

**Bit 24 – DATA32B** Data 32 Bit
This bit enables 32-bit data writes and reads to/from the DATA register.

| Value | Description |
|---|---|
| 0 | Data transactions to/from DATA are 8-bit in size |
| 1 | Data transactions to/from DATA are 32-bit in size |

#### 38.8.2.1 Control B

**Name:** CTRLB
**Offset:** 0x08
**Reset:** 0x0000
**Property:** PAC Write-Protection

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

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | NREPLY | SPDCONF[1:0] | | UPRSM | DETACH |
| Access | | | | R | R/W | R/W | R/W | R/W |
| Reset | | | | 0 | 0 | 0 | 0 | 0 |

**Bits 11:10 – LPMHDSK[1:0]** Link Power Management Handshake
These bits select the Link Power Management Handshake configuration.

| Value | Description |
|---|---|
| 0x0 | No handshake. LPM is not supported. |
| 0x1 | ACK |
| 0x2 | NYET |
| 0x3 | Reserved |

**Bit 9 – GNAK** Global NAK
This bit configures the operating mode of the NAK.

This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | The handshake packet reports the status of the USB transaction |
| 1 | A NAK handshake is answered for each USB transaction regardless of the current endpoint memory bank status |

**Bit 4 – NREPLY** No reply excepted SETUP Token
This bit is cleared by hardware when receiving a SETUP packet.

This bit has no effect for any other endpoint but endpoint 0.

| Value | Description |
|---|---|
| 0 | Disable the “NO_REPLY” feature: Any transaction to endpoint 0 will be handled according to the USB2.0 standard. |
| 1 | Enable the “NO_REPLY” feature: Any transaction to endpoint 0 will be ignored except SETUP. |

**Bits 3:2 – SPDCONF[1:0]** Speed Configuration
These bits select the speed configuration.

#### 38.8.3.5 Device EndPoint Interrupt Flag n

**Name:** EPINTFLAGn
**Offset:** 0x107 + (n x 0x20)
**Reset:** 0x00
**Property:** -

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | STALL | RXSTP | | TRFAIL | | TRCPT |
| Access | | | R/W | R/W | | R/W | | R/W |
| Reset | | | 2 | 0 | | 2 | | 2 |

**Bit 5 – STALL** Transmit Stall x Interrupt Flag
This flag is cleared by writing a one to the flag.

This flag is set when a Transmit Stall occurs and will generate an interrupt if EPINTENCLR/SET.STALL is one.

EPINTFLAG.STALL is set for a single bank OUT endpoint or double bank IN/OUT endpoint when current bank is "0".

Writing a zero to this bit has no effect.

Writing a one to this bit clears the STALL Interrupt Flag.

**Bit 4 – RXSTP** Received Setup Interrupt Flag
This flag is cleared by writing a one to the flag.

This flag is set when a Received Setup occurs and will generate an interrupt if EPINTENCLR/SET.RXSTP is one.

Writing a zero to this bit has no effect.

Writing a one to this bit clears the RXSTP Interrupt Flag.

**Bit 2 – TRFAIL** Transfer Fail x Interrupt Flag
This flag is cleared by writing a one to the flag.

This flag is set when a transfer fail occurs and will generate an interrupt if EPINTENCLR/SET.TRFAIL is one.

EPINTFLAG.TRFAIL is set for a single bank OUT endpoint or double bank IN/OUT endpoint when current bank is "0".

Writing a zero to this bit has no effect.

Writing a one to this bit clears the TRFAIL Interrupt Flag.

**Bit 0 – TRCPT** Transfer Complete x interrupt Flag
This flag is cleared by writing a one to the flag.

This flag is set when a Transfer complete occurs and will generate an interrupt if EPINTENCLR/SET.TRCPT is one. EPINTFLAG.TRCPT is set for a single bank OUT endpoint or double bank IN/OUT endpoint when current bank is "0".

### 38.8.7.4 Extended Register

**Name:** EXTREG
**Offset:** 0x08
**Reset:** 0xxxxxxxx
**Property:** NA

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | VARIABLE[10:4] | | | | | | |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | VARIABLE[3:0] | | | | SUBPID[3:0] | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | x | 0 | 0 | 0 | x |

**Bits 14:4 – VARIABLE[10:0]** Variable field send with extended token
These bits define the VARIABLE field sent with extended token. See “Section 2.1.1 Protocol Extension Token in the reference document ENGINEERING CHANGE NOTICE, USB 2.0 Link Power Management Addendum.”

To support the USB2.0 Link Power Management addition the VARIABLE field should be set as described below.

| VARIABLE | Description |
|---|---|
| VARIABLE[3:0] | bLinkState(1) |
| VARIABLE[7:4] | BESL (See LPM ECN)(2) |
| VARIABLE[8] | bRemoteWake(1) |
| VARIABLE[10:9] | Reserved |

(1) for a definition of LPM Token bRemoteWake and bLinkState fields, refer to "Table 2-3 in the reference document ENGINEERING CHANGE NOTICE, USB 2.0 Link Power Management Addendum"
(2) for a definition of LPM Token BESL field, refer to "Table 2-3 in the reference document ENGINEERING CHANGE NOTICE, USB 2.0 Link Power Management Addendum" and "Table X-X1 in Errata for ECN USB 2.0 Link Power Management.

**Bits 3:0 – SUBPID[3:0]** SUBPID field send with extended token
These bits define the SUBPID field sent with extended token. See “Section 2.1.1 Protocol Extension Token in the reference document ENGINEERING CHANGE NOTICE, USB 2.0 Link Power Management Addendum”.

To support the USB2.0 Link Power Management addition the SUBPID field should be set as described in “Table 2.2 SubPID Types in the reference document ENGINEERING CHANGE NOTICE, USB 2.0 Link Power Management Addendum”.

**Bits 15:0 – F1SA[15:0]** Rx FIFO 1 Start Address
Start address of Rx FIFO 1 in Message RAM. When the CAN module addresses the Message RAM it addresses 32-bit words, not single bytes. The configurable start addresses are 32-bit word addresses, i.e. only bits 15 to 2 are evaluated, the two least significant bits are ignored. Bits 1 to 0 will always be read back as “00”.

### 40.8.2 Block Size Register

**Name:** BSR
**Offset:** 0x04
**Reset:** 0x0000
**Property:** -

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | BOUNDARY[2:0] | | | | | BLKSIZE[9:8] | |
| Access | | | | | | | R/W | R/W |
| Reset | | 0 | 0 | 0 | | | 0 | 0 |

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

**Bits 14:12 – BOUNDARY[2:0]** SDMA Buffer Boundary
This field specifies the size of the contiguous buffer in the system memory. The SDMA transfer waits at every boundary specified by this field and the peripheral generates the DMA Interrupt to instruct the software to update SSAR. If this field is set to 0 (buffer size = 4 Kbytes), the lowest 12 bits of SSAR.ADDRESS point to data in the contiguous buffer, and the upper 20 bits point to the location of the buffer in the system memory. This function is active when the DMA Enable bit in the Transfer Mode Register (TMR.DMAEN) is '1'.

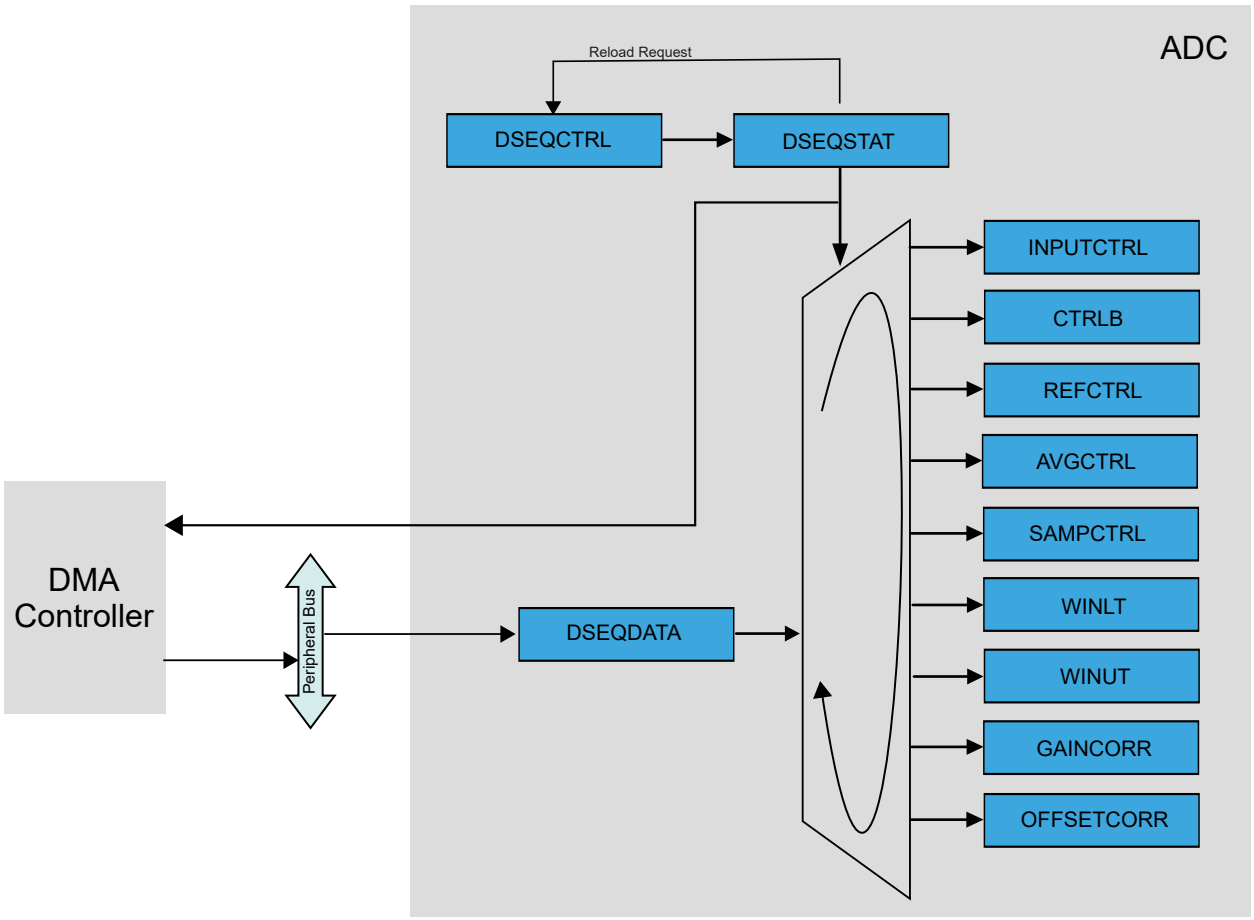
| Value | Name | Description |
|---|---|---|
| 0 | 4K | 4-Kbyte boundary |
| 1 | 8K | 8-Kbyte boundary |
| 2 | 16K | 16-Kbyte boundary |
| 3 | 32K | 32-Kbyte boundary |
| 4 | 64K | 64-Kbyte boundary |
| 5 | 128K | 128-Kbyte boundary |
| 6 | 256k | 256-Kbyte boundary |
| 7 | 512K | 512-Kbyte boundary |

**Bits 9:0 – BLKSIZE[9:0]** Transfer Block Size
This field specifies the block size of data transfers for CMD17, CMD18, CMD24, CMD25 and CMD53. Values ranging from 1 to 512 can be set. It can be accessed only if no transaction is executing (i.e., after a transaction has stopped). Read operations during transfers may return an invalid value, and write operations are ignored.

When one of these registers is written, the data is stored in the corresponding buffer as long as the current conversion is not impacted, and the corresponding busy status will be set in the Synchronization Busy register (SYNCBUSY). When a new RESULT is available, data stored in the buffer registers will be transfered to the ADC and a new conversion can start.

#### 45.6.3.3 DMA Sequencing

The ADC can sequence a series of conversion. When DMA sequencing is enabled, a set of ADC configuration registers can be automatically refreshed using the DMA controller.




**Enabling DMA Sequencing**

DMA Sequencing is enabled when at least one bit in the DMA Sequence Control register (DSEQCTRL) is '1'.

When this is the case, the BUSY status bit in the DMA Sequential Status register (DSEQSTAT.BUSY) is set to '1'.

**Disabling DMA Sequencing**

DMA Sequencing is disabled when at least one of the following conditions is valid:

- The ADC is disabled (CTRLA.ENABLE = 0).
- The ADC is reset (CTRLA.SWRST = 1).
- The DMA Sequence Control register (DSEQCTRL) is written '0' and the ongoing DMA sequence is completed.

| Value | Description |
|---|---|
| 0 | The On Demand is disabled. If On Demand is disabled, the TC will continue to request the clock when its operation is stopped (STATUS.STOP=1). |
| 1 | The On Demand is enabled. When On Demand is enabled, the stopped TC will not request the clock. The clock is requested when a software re-trigger command is applied or when an event with start/re-trigger action is detected. |

**Bit 6 – RUNSTDBY** Run in Standby
This bit is used to keep the TC running in standby mode.

This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | The TC is halted in standby. |
| 1 | The TC continues to run in standby. |

**Bits 5:4 – PRESCSYNC[1:0]** Prescaler and Counter Synchronization
These bits select whether the counter should wrap around on the next GCLK_TCx clock or the next prescaled GCLK_TCx clock. It also makes it possible to reset the prescaler.

These bits are not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | GCLK | Reload or reset the counter on next generic clock |
| 0x1 | PRESC | Reload or reset the counter on next prescaler clock |
| 0x2 | RESYNC | Reload or reset the counter on next generic clock. Reset the prescaler counter |
| 0x3 | - | Reserved |

**Bits 3:2 – MODE[1:0]** Timer Counter Mode
These bits select the counter mode.

These bits are not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | COUNT16 | Counter in 16-bit mode |
| 0x1 | COUNT8 | Counter in 8-bit mode |
| 0x2 | COUNT32 | Counter in 32-bit mode |
| 0x3 | - | Reserved |

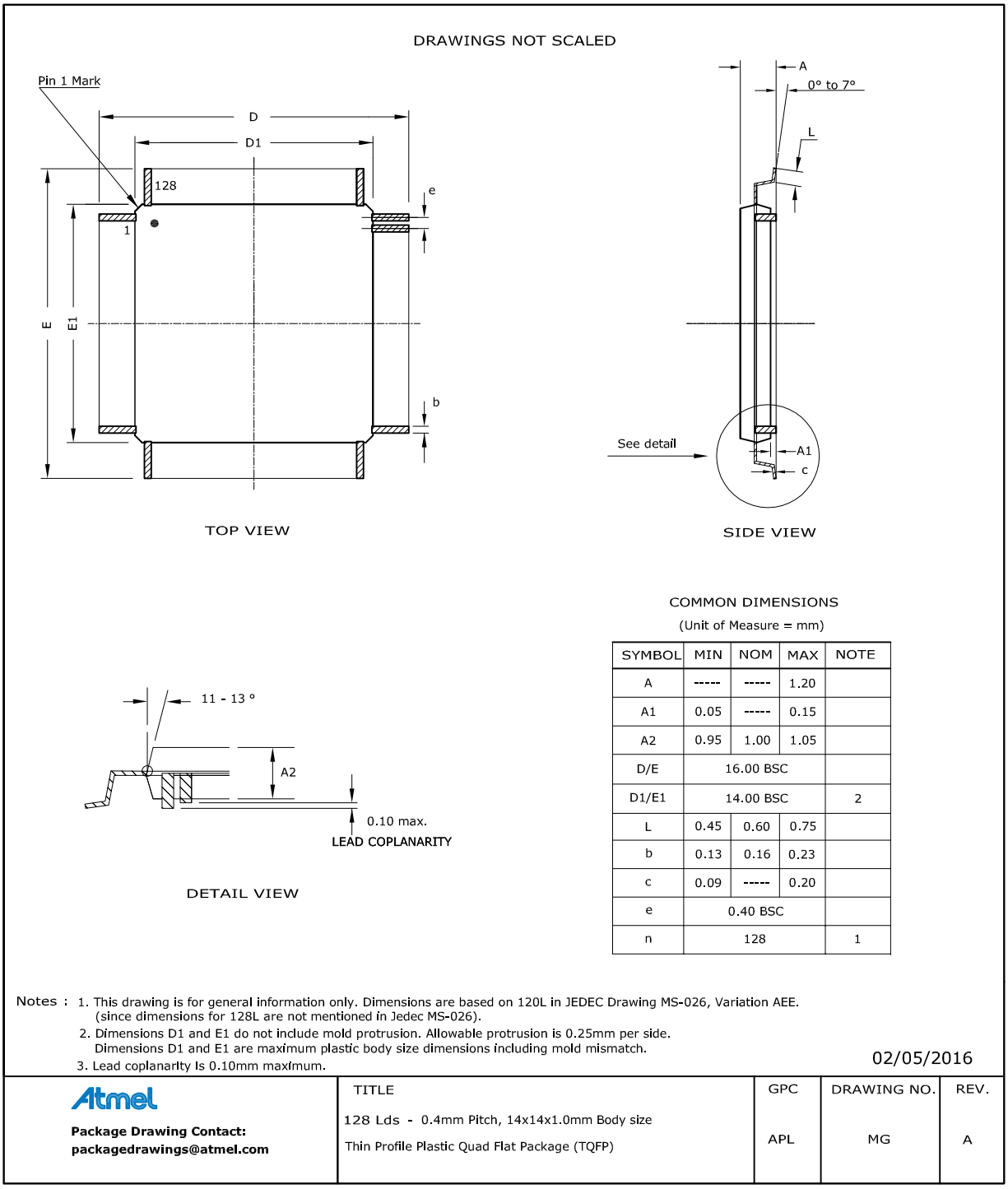
**Bit 1 – ENABLE** Enable
Due to synchronization, there is delay from writing CTRLA.ENABLE until the peripheral is enabled/disabled. The value written to CTRLA.ENABLE will read back immediately, and the ENABLE Synchronization Busy bit in the SYNCBUSY register (SYNCBUSY.ENABLE) will be set. SYNCBUSY.ENABLE will be cleared when the operation is complete.

This bit is not enable protected.

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

**Bit 0 – SWRST** Software Reset
Writing a '0' to this bit has no effect.

### 55.3.7 128 pin TQFP




COMMON DIMENSIONS

(Unit of Measure = mm)

| SYMBOL | MIN | NOM | MAX | NOTE |
|---|---|---|---|---|
| A | ----- | ----- | 1.20 | |
| A1 | 0.05 | ----- | 0.15 | |
| A2 | 0.95 | 1.00 | 1.05 | |
| D/E | | 16.00 BSC | | |
| D1/E1 | | 14.00 BSC | | 2 |
| L | 0.45 | 0.60 | 0.75 | |
| b | 0.13 | 0.16 | 0.23 | |
| c | 0.09 | ----- | 0.20 | |
| e | | 0.40 BSC | | |
| n | | 128 | | 1 |

Notes :
1. This drawing is for general information only. Dimensions are based on 120L in JEDEC Drawing MS-026, Variation AEE. (since dimensions for 128L are not mentioned in Jedec MS-026).
2. Dimensions D1 and E1 do not include mold protrusion. Allowable protrusion is 0.25mm per side. Dimensions D1 and E1 are maximum plastic body size dimensions including mold mismatch.
3. Lead coplanarity is 0.10mm maximum.

02/05/2016

| Atmel | TITLE | GPC | DRAWING NO. | REV. |
|---|---|---|---|---|
| **Package Drawing Contact:** **packagedrawings@atmel.com** | 128 Lds - 0.4mm Pitch, 14x14x1.0mm Body size Thin Profile Plastic Quad Flat Package (TQFP) | APL | MG | A |

**Table 55-14. Device and Package Maximum Weight**

| | |
|---|---|
| 520 | mg |

**Table 55-15. Package Characteristics**

| | |
|---|---|
| Moisture Sensitivity Level | MSL3 |