**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 | HCS12X |
| Core Size | 16-Bit |
| Speed | 50MHz |
| Connectivity | CANbus, EBI/EMI, I²C, IrDA, SCI, SPI |
| Peripherals | LVD, POR, PWM, WDT |
| Number of I/O | 119 |
| Program Memory Size | 384KB (384K x 8) |
| Program Memory Type | FLASH |
| EEPROM Size | 4K x 8 |
| RAM Size | 24K x 8 |
| Voltage - Supply (Vcc/Vdd) | 1.72V ~ 5.5V |
| Data Converters | A/D 24x12b |
| Oscillator Type | External |
| Operating Temperature | -40°C ~ 125°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/nxp-semiconductors/mc9s12xeq384mag |

## Chapter 3
## Memory Mapping Control (S12XMMCV4)

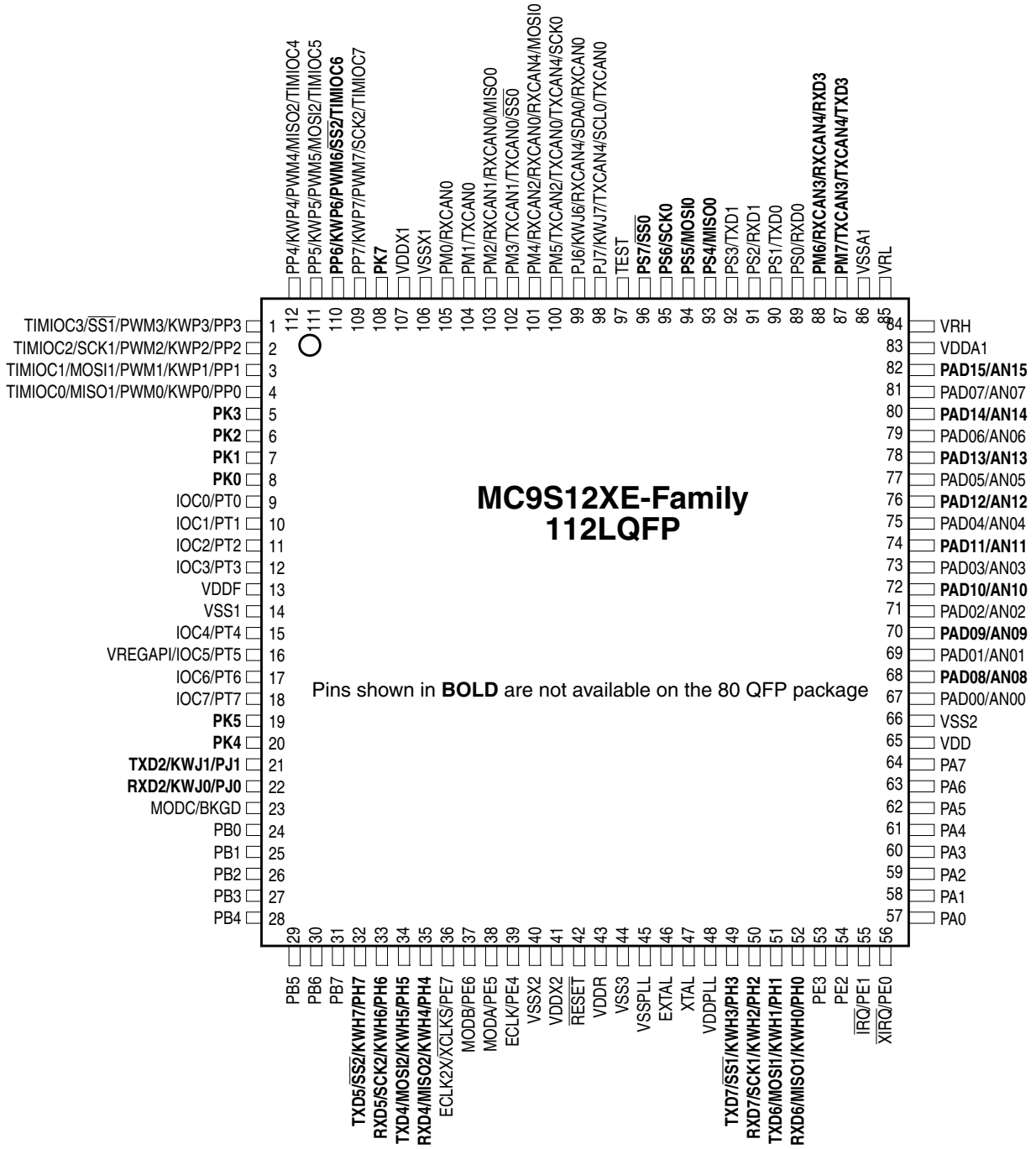



**Figure 1-6. MC9S12XE-Family Pin Assignments 112-pin LQFP Package**

## 2.3.74 Port AD0 Reduced Drive Register 1 (RDR1AD0)

Address 0x0275 Access: User read/write(1)

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| R<br>W | RDR1AD07 | RDR1AD06 | RDR1AD05 | RDR1AD04 | RDR1AD03 | RDR1AD02 | RDR1AD01 | RDR1AD00 |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Figure 2-72. Port AD0 Reduced Drive Register 1 (RDR1AD0)**

1. Read: Anytime.
   Write: Anytime.

**Table 2-70. RDR1AD0 Register Field Descriptions**

| Field | Description |
|---|---|
| 7-0<br>RDR1AD0 | **Port AD0 reduced drive**—Select reduced drive for Port AD0 outputs<br>This register configures the drive strength of Port AD0 output pins 7 through 0 as either full or reduced independent of the function used on the pins. If a pin is used as input this bit has no effect.<br>1 Reduced drive selected (approx. 1/5 of the full drive strength).<br>0 Full drive strength enabled. |

## 2.3.75 Port AD0 Pull Up Enable Register 0 (PER0AD0)

Address 0x0276 Access: User read/write(1)

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| R<br>W | PER0AD07 | PER0AD06 | PER0AD05 | PER0AD04 | PER0AD03 | PER0AD02 | PER0AD01 | PER0AD00 |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Figure 2-73. Port AD0 Pull Device Up Register 0 (PER0AD0)**

1. Read: Anytime.
   Write: Anytime.

**Table 2-71. PER0AD0 Register Field Descriptions**

| Field | Description |
|---|---|
| 7-0<br>PER0AD0 | **Port AD0 pull device enable**—Enable pull devices on input pins<br>These bits configure whether a pull device is activated, if the associated pin is used as an input. This bit has no effect if the pin is used as an output. Out of reset no pull device is enabled.<br>1 Pull device enabled.<br>0 Pull device disabled. |

### 10.3.1.12 XGATE Program Counter Register (XGPC)

The XGPC register (Figure 10-14) provides access to the RISC core's program counter.

Module Base +0x0001E

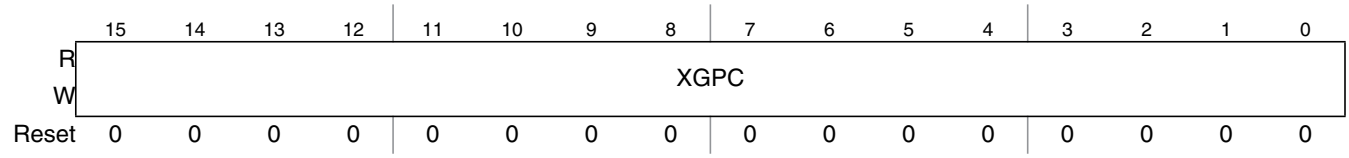



**Figure 10-14. XGATE Program Counter Register (XGPC)**

Read: In debug mode if unsecured and not idle (XGCHID ≠ 0x00)

Write: In debug mode if unsecured and not idle (XGCHID ≠ 0x00)

**Table 10-14. XGPC Field Descriptions**

| Field | Description |
|---|---|
| 15–0<br>XGPC[15:0] | **Program Counter** — The RISC core's program counter |

### 10.3.1.13 XGATE Register 1 (XGR1)

The XGR1 register (Figure 10-15) provides access to the RISC core's register 1.

Module Base +0x00022

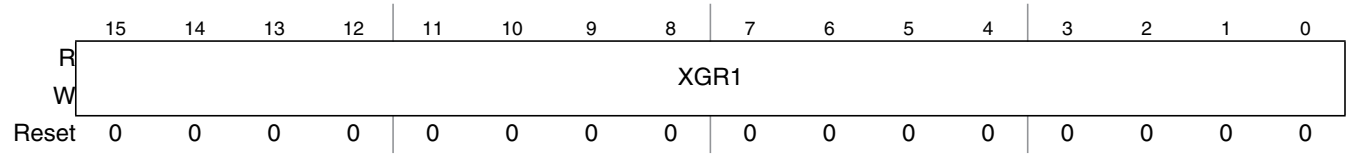



**Figure 10-15. XGATE Register 1 (XGR1)**

Read: In debug mode if unsecured and not idle (XGCHID ≠ 0x00)

Write: In debug mode if unsecured and not idle (XGCHID ≠ 0x00)

**Table 10-15. XGR1 Field Descriptions**

| Field | Description |
|---|---|
| 15–0<br>XGR1[15:0] | **XGATE Register 1** — The RISC core's register 1 |

# ADDH — Add Immediate 8 bit Constant (High Byte) — ADDH

## Operation

RD + IMM8:$00 ⇒ RD

Adds the content of high byte of register RD and a signed immediate 8 bit constant using binary addition and stores the result in the high byte of the destination register RD. This instruction can be used after an ADDL for a 16 bit immediate addition.

Example:

```
ADDL    R2,#LOWBYTE
ADDH    R2,#HIGHBYTE      ; R2 = R2 + 16 bit immediate
```

## CCR Effects

| N | Z | V | C |
|---|---|---|---|
| Δ | Δ | Δ | Δ |

N: Set if bit 15 of the result is set; cleared otherwise.

Z: Set if the result is $0000; cleared otherwise.

V: Set if a two´s complement overflow resulted from the operation; cleared otherwise.
$RD[15]_{old}$ & IMM8[7] & $\overline{RD[15]}_{new}$ | $\overline{RD[15]}_{old}$ & $\overline{IMM8[7]}$ & $RD[15]_{new}$

C: Set if there is a carry from the bit 15 of the result; cleared otherwise.
$RD[15]_{old}$ & IMM8[7] | $RD[15]_{old}$ & $\overline{RD[15]}_{new}$ | IMM8[7] & $\overline{RD[15]}_{new}$

## Code and CPU Cycles

| Source Form | Address Mode | Machine Code | | | | | | | Cycles |
|---|---|---|---|---|---|---|---|---|---|
| ADDH RD, #IMM8 | IMM8 | 1 | 1 | 1 | 0 | 1 | RD | IMM8 | P |

#### 11.4.1.3 Clock Monitor (CM)

If no OSCCLK edges are detected within a certain time, the clock monitor within the oscillator block generates a clock monitor fail event. The S12XECRG then asserts self clock mode or generates a system reset depending on the state of SCME bit. If the clock monitor is disabled or the presence of clocks is detected no failure is indicated by the oscillator block.The clock monitor function is enabled/disabled by the CME control bit.

#### 11.4.1.4 Clock Quality Checker

The clock monitor performs a coarse check on the incoming clock signal. The clock quality checker provides a more accurate check in addition to the clock monitor.

A clock quality check is triggered by any of the following events:

- Power on reset (*POR*)
- Low voltage reset (*LVR*)
- Wake-up from Full Stop Mode (*exit full stop*)
- Clock Monitor fail indication (*CM fail*)

A time window of 50000 PLLCLK cycles[1] is called *check window*.

A number greater equal than 4096 rising OSCCLK edges within a *check window* is called *osc ok*. Note that *osc ok* immediately terminates the current *check window*. See Figure 11-17 as an example.

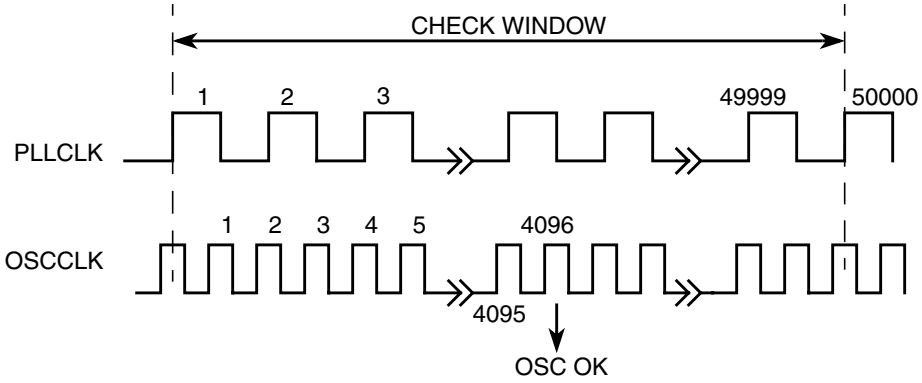



**Figure 11-17. Check Window Example**

1. IPLL is running at self clock mode frequency $f_{SCM}$.

| Address | Name | | Bit 7 | 6 | 5 | 4 | 3 | 2 | 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x0024 | ATDDR10 | R<br>W | See Section 13.3.2.12.1, “Left Justified Result Data (DJM=0)” and Section 13.3.2.12.2, “Right Justified Result Data (DJM=1)” | | | | | | | |
| 0x0026 | ATDDR11 | R<br>W | See Section 13.3.2.12.1, “Left Justified Result Data (DJM=0)” and Section 13.3.2.12.2, “Right Justified Result Data (DJM=1)” | | | | | | | |
| 0x0028 | ATDDR12 | R<br>W | See Section 13.3.2.12.1, “Left Justified Result Data (DJM=0)” and Section 13.3.2.12.2, “Right Justified Result Data (DJM=1)” | | | | | | | |
| 0x002A | ATDDR13 | R<br>W | See Section 13.3.2.12.1, “Left Justified Result Data (DJM=0)” and Section 13.3.2.12.2, “Right Justified Result Data (DJM=1)” | | | | | | | |
| 0x002C | ATDDR14 | R<br>W | See Section 13.3.2.12.1, “Left Justified Result Data (DJM=0)” and Section 13.3.2.12.2, “Right Justified Result Data (DJM=1)” | | | | | | | |
| 0x002E | ATDDR15 | R<br>W | See Section 13.3.2.12.1, “Left Justified Result Data (DJM=0)” and Section 13.3.2.12.2, “Right Justified Result Data (DJM=1)” | | | | | | | |

= Unimplemented or Reserved

**Figure 13-2. ADC12B16C Register Summary (Sheet 3 of 3)**

## 13.3.2 Register Descriptions

This section describes in address order all the ADC12B16C registers and their individual bits.

### 13.3.2.1 ATD Control Register 0 (ATDCTL0)

Writes to this register will abort current conversion sequence.

Module Base + 0x0000

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| R | Reserved | 0 | 0 | 0 | WRAP3 | WRAP2 | WRAP1 | WRAP0 |
| W | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |

= Unimplemented or Reserved

**Figure 13-3. ATD Control Register 0 (ATDCTL0)**

Read: Anytime

Write: Anytime, in special modes always write 0 to Reserved Bit 7.

**Table 13-2. ATDCTL0 Field Descriptions**

| Field | Description |
|---|---|
| | |
| 3-0<br>WRAP[3-0] | **Wrap Around Channel Select Bits** — These bits determine the channel for wrap around when doing multi-channel conversions. The coding is summarized in Table 13-3. |

**Table 13-3. Multi-Channel Wrap Around Coding**

| WRAP3 | WRAP2 | WRAP1 | WRAP0 | Multiple Channel Conversions (MULT = 1) Wraparound to AN0 after Converting |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | Reserved(1) |

**Table 13-11. Conversion Sequence Length Coding**

| S8C | S4C | S2C | S1C | Number of Conversions per Sequence |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 16 |
| 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 2 |
| 0 | 0 | 1 | 1 | 3 |
| 0 | 1 | 0 | 0 | 4 |
| 0 | 1 | 0 | 1 | 5 |
| 0 | 1 | 1 | 0 | 6 |
| 0 | 1 | 1 | 1 | 7 |
| 1 | 0 | 0 | 0 | 8 |
| 1 | 0 | 0 | 1 | 9 |
| 1 | 0 | 1 | 0 | 10 |
| 1 | 0 | 1 | 1 | 11 |
| 1 | 1 | 0 | 0 | 12 |
| 1 | 1 | 0 | 1 | 13 |
| 1 | 1 | 1 | 0 | 14 |
| 1 | 1 | 1 | 1 | 15 |

**Table 13-12. ATD Behavior in Freeze Mode (Breakpoint)**

| FRZ1 | FRZ0 | Behavior in Freeze Mode |
|---|---|---|
| 0 | 0 | Continue conversion |
| 0 | 1 | Reserved |
| 1 | 0 | Finish current conversion, then freeze |
| 1 | 1 | Freeze Immediately |

### 13.3.2.5 ATD Control Register 4 (ATDCTL4)

Writes to this register will abort current conversion sequence.

Module Base + 0x0004

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| R<br>W | SMP2 | SMP1 | SMP0 | PRS[4:0] | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

**Figure 13-7. ATD Control Register 4 (ATDCTL4)**

Read: Anytime

Write: Anytime

**Table 15-5. Prescale Divider Encoding**

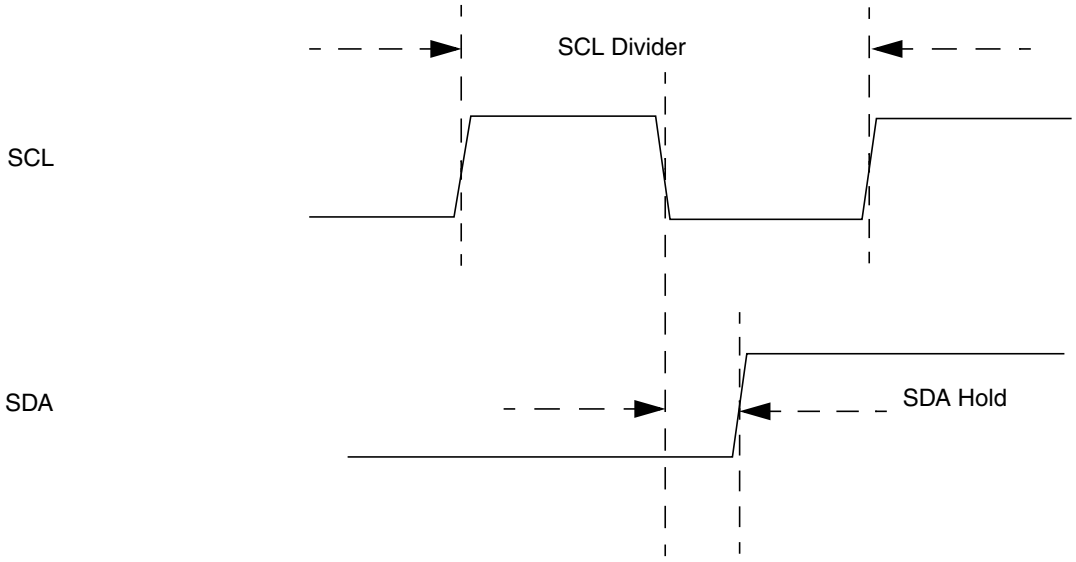
| IBC5-3 (bin) | scl2start (clocks) | scl2stop (clocks) | scl2tap (clocks) | tap2tap (clocks) |
|---|---|---|---|---|
| 000 | 2 | 7 | 4 | 1 |
| 001 | 2 | 7 | 4 | 2 |
| 010 | 2 | 9 | 6 | 4 |
| 011 | 6 | 9 | 6 | 8 |
| 100 | 14 | 17 | 14 | 16 |
| 101 | 30 | 33 | 30 | 32 |
| 110 | 62 | 65 | 62 | 64 |
| 111 | 126 | 129 | 126 | 128 |

**Table 15-6. Multiplier Factor**

| IBC7-6 | MUL |
|---|---|
| 00 | 01 |
| 01 | 02 |
| 10 | 04 |
| 11 | RESERVED |

The number of clocks from the falling edge of SCL to the first tap (Tap[1]) is defined by the values shown in the scl2tap column of Table 15-4, all subsequent tap points are separated by $2^{IBC5-3}$ as shown in the tap2tap column in Table 15-4. The SCL Tap is used to generated the SCL period and the SDA Tap is used to determine the delay from the falling edge of SCL to SDA changing, the SDA hold time.

IBC7–6 defines the multiplier factor MUL. The values of MUL are shown in the Table 15-6.



| Register Name | | Bit 7 | 6 | 5 | 4 | 3 | 2 | 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|---|
| 0x000F CANTXERR | R | TXERR7 | TXERR6 | TXERR5 | TXERR4 | TXERR3 | TXERR2 | TXERR1 | TXERR0 |
| | W | | | | | | | | |
| 0x0010–0x0013 CANIDAR0–3 | R | AC7 | AC6 | AC5 | AC4 | AC3 | AC2 | AC1 | AC0 |
| | W | | | | | | | | |
| 0x0014–0x0017 CANIDMRx | R | AM7 | AM6 | AM5 | AM4 | AM3 | AM2 | AM1 | AM0 |
| | W | | | | | | | | |
| 0x0018–0x001B CANIDAR4–7 | R | AC7 | AC6 | AC5 | AC4 | AC3 | AC2 | AC1 | AC0 |
| | W | | | | | | | | |
| 0x001C–0x001F CANIDMR4–7 | R | AM7 | AM6 | AM5 | AM4 | AM3 | AM2 | AM1 | AM0 |
| | W | | | | | | | | |
| 0x0020–0x002F CANRXFG | R | See Section 16.3.3, "Programmer's Model of Message Storage" | | | | | | | |
| | W | | | | | | | | |
| 0x0030–0x003F CANTXFG | R | See Section 16.3.3, "Programmer's Model of Message Storage" | | | | | | | |
| | W | | | | | | | | |

= Unimplemented or Reserved

**Figure 16-3. MSCAN Register Summary (continued)**

## 16.3.2 Register Descriptions

This section describes in detail all the registers and register bits in the MSCAN module. Each description includes a standard register diagram with an associated figure number. Details of register bit and field function follow the register diagrams, in bit order. All bits of all registers in this module are completely synchronous to internal clocks during a register read.

### 16.3.2.1 MSCAN Control Register 0 (CANCTL0)

The CANCTL0 register provides various control bits of the MSCAN module as described below.

Module Base + 0x0000 — Access: User read/write(1)

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| R | RXFRM | RXACT | CSWAI | SYNCH | TIME | WUPE | SLPRQ | INITRQ |
| W | | | | | | | | |
| Reset: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

= Unimplemented

**Figure 16-4. MSCAN Control Register 0 (CANCTL0)**

**Table 26-14. FECCRIX Field Descriptions**

| Field | Description |
|---|---|
| 2-0<br>ECCRIX[2:0] | **ECC Error Register Index**— The ECCRIX bits are used to select which word of the FECCR register array is being read. See Section 26.3.2.13, “Flash ECC Error Results Register (FECCR),” for more details. |

### 26.3.2.5 Flash Configuration Register (FCNFG)

The FCNFG register enables the Flash command complete interrupt and forces ECC faults on Flash array read access from the CPU or XGATE.

Offset Module Base + 0x0004

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| R | CCIE | 0 | 0 | IGNSF | 0 | 0 | FDFD | FSFD |
| W | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

= Unimplemented or Reserved

**Figure 26-9. Flash Configuration Register (FCNFG)**

CCIE, IGNSF, FDFD, and FSFD bits are readable and writable while remaining bits read 0 and are not writable.

**Table 26-15. FCNFG Field Descriptions**

| Field | Description |
|---|---|
| 7<br>CCIE | **Command Complete Interrupt Enable** — The CCIE bit controls interrupt generation when a Flash command has completed.<br>0 Command complete interrupt disabled<br>1 An interrupt will be requested whenever the CCIF flag in the FSTAT register is set (see Section 26.3.2.7) |
| 4<br>IGNSF | **Ignore Single Bit Fault** — The IGNSF controls single bit fault reporting in the FERSTAT register (see Section 26.3.2.8).<br>0 All single bit faults detected during array reads are reported<br>1 Single bit faults detected during array reads are not reported and the single bit fault interrupt will not be generated |

**Table 27-26. FCCOB - NVM Command Mode (Typical Usage)**

| CCOBIX[2:0] | Byte | FCCOB Parameter Fields (NVM Command Mode) |
|---|---|---|
| 011 | HI | Data 1 [15:8] |
| | LO | Data 1 [7:0] |
| 100 | HI | Data 2 [15:8] |
| | LO | Data 2 [7:0] |
| 101 | HI | Data 3 [15:8] |
| | LO | Data 3 [7:0] |

### 27.3.2.12 EEE Tag Counter Register (ETAG)

The ETAG register contains the number of outstanding words in the buffer RAM EEE partition that need to be programmed into the D-Flash EEE partition. The ETAG register is decremented prior to the related tagged word being programmed into the D-Flash EEE partition. All tagged words have been programmed into the D-Flash EEE partition once all bits in the ETAG register read 0 and the MGBUSY flag in the FSTAT register reads 0.

Offset Module Base + 0x000C

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| R | ETAG[15:8] | | | | | | | |
| W | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

= Unimplemented or Reserved

**Figure 27-18. EEE Tag Counter High Register (ETAGHI)**

Offset Module Base + 0x000D

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| R | ETAG[7:0] | | | | | | | |
| W | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

= Unimplemented or Reserved

**Figure 27-19. EEE Tag Counter Low Register (ETAGLO)**

All ETAG bits are readable but not writable and are cleared by the Memory Controller.

### 27.3.2.13 Flash ECC Error Results Register (FECCR)

The FECCR registers contain the result of a detected ECC fault for both single bit and double bit faults. The FECCR register provides access to several ECC related fields as defined by the ECCRIX index bits in the FECCRIX register (see Section 27.3.2.4). Once ECC fault information has been stored, no other

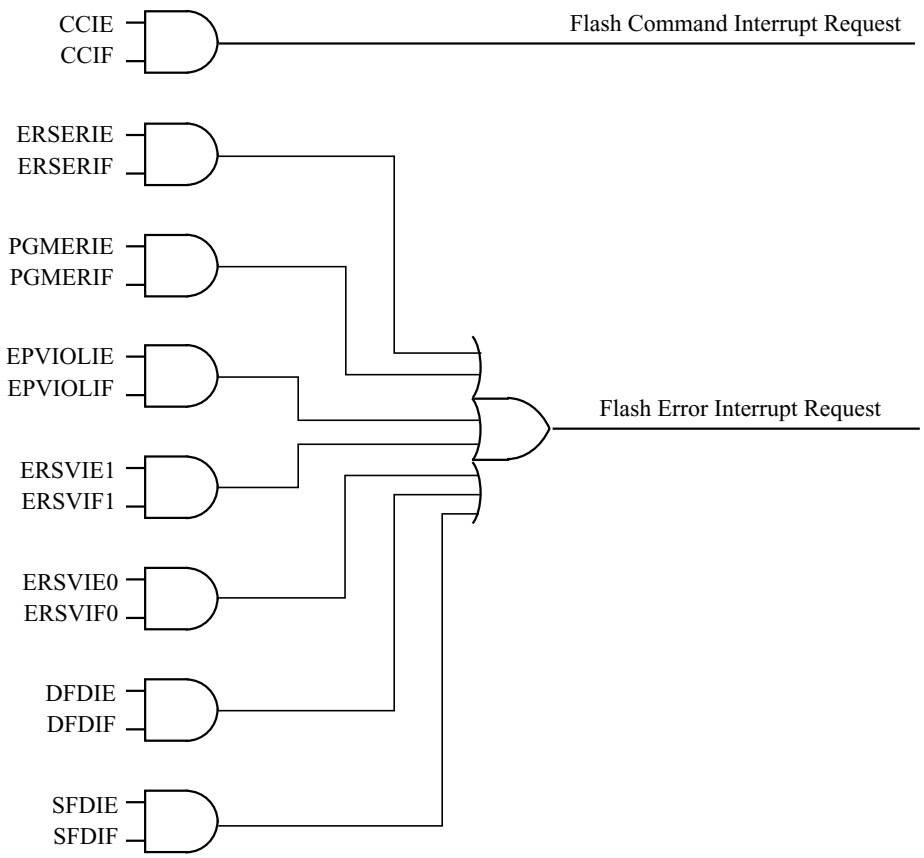



**Figure 27-27. Flash Module Interrupts Implementation**

### 27.4.4 Wait Mode

The Flash module is not affected if the MCU enters wait mode. The Flash module can recover the MCU from wait via the CCIF interrupt (see Section 27.4.3, “Interrupts”).

### 27.4.5 Stop Mode

If a Flash command is active (CCIF = 0) or an EE-Emulation operation is pending when the MCU requests stop mode, the current Flash operation will be completed before the CPU is allowed to enter stop mode.

## 27.5 Security

The Flash module provides security information to the MCU. The Flash security state is defined by the SEC bits of the FSEC register (see Table 27-12). During reset, the Flash module initializes the FSEC register using data read from the security byte of the Flash configuration field at global address 0x7F_FF0F.

**D-Flash Sector** — The D-Flash sector is the smallest portion of the D-Flash memory that can be erased. The D-Flash sector consists of four 64 byte rows for a total of 256 bytes.

**EEE (Emulated EEPROM)** — A method to emulate the small sector size features and endurance characteristics associated with an EEPROM.

**EEE IFR** — Nonvolatile information register located in the D-Flash block that contains data required to partition the D-Flash memory and buffer RAM for EEE. The EEE IFR is visible in the global memory map by setting the EEEIFRON bit in the MMCCTL1 register.

**NVM Command Mode** — An NVM mode using the CPU to setup the FCCOB register to pass parameters required for Flash command execution.

**Phrase** — An aligned group of four 16-bit words within the P-Flash memory. Each phrase includes eight ECC bits for single bit fault correction and double bit fault detection within the phrase.

**P-Flash Memory** — The P-Flash memory constitutes the main nonvolatile memory store for applications.

**P-Flash Sector** — The P-Flash sector is the smallest portion of the P-Flash memory that can be erased. Each P-Flash sector contains 1024 bytes.

**Program IFR** — Nonvolatile information register located in the P-Flash block that contains the Device ID, Version ID, and the Program Once field. The Program IFR is visible in the global memory map by setting the PGMIFRON bit in the MMCCTL1 register.

## 28.1.2 Features

### 28.1.2.1 P-Flash Features

- 768 Kbytes of P-Flash memory composed of two 256 Kbyte Flash blocks and two 128 Kbyte Flash blocks. The 256 Kbyte Flash block consists of two 128 Kbyte sections each divided into 128 sectors of 1024 bytes. The 128 Kbyte Flash blocks are each divided into 128 sectors of 1024 bytes.
- Single bit fault correction and double bit fault detection within a 64-bit phrase during read operations
- Automated program and erase algorithm with verify and generation of ECC parity bits
- Fast sector erase and phrase program operation
- Ability to program up to one phrase in each P-Flash block simultaneously
- Flexible protection scheme to prevent accidental program or erase of P-Flash memory

### 28.1.2.2 D-Flash Features

- Up to 32 Kbytes of D-Flash memory with 256 byte sectors for user access
- Dedicated commands to control access to the D-Flash memory over EEE operation
- Single bit fault correction and double bit fault detection within a word during read operations
- Automated program and erase algorithm with verify and generation of ECC parity bits
- Fast sector erase and word program operation

**Table 28-15. FCNFG Field Descriptions (continued)**

| Field | Description |
|---|---|
| 1<br>FDFD | **Force Double Bit Fault Detect** — The FDFD bit allows the user to simulate a double bit fault during Flash array read operations and check the associated interrupt routine. The FDFD bit is cleared by writing a 0 to FDFD. The FECCR registers will not be updated during the Flash array read operation with FDFD set unless an actual double bit fault is detected.<br>0 Flash array read operations will set the DFDIF flag in the FERSTAT register only if a double bit fault is detected<br>1 Any Flash array read operation will force the DFDIF flag in the FERSTAT register to be set (see Section 28.3.2.7) and an interrupt will be generated as long as the DFDIE interrupt enable in the FERCNFG register is set (see Section 28.3.2.6) |
| 0<br>FSFD | **Force Single Bit Fault Detect** — The FSFD bit allows the user to simulate a single bit fault during Flash array read operations and check the associated interrupt routine. The FSFD bit is cleared by writing a 0 to FSFD. The FECCR registers will not be updated during the Flash array read operation with FSFD set unless an actual single bit fault is detected.<br>0 Flash array read operations will set the SFDIF flag in the FERSTAT register only if a single bit fault is detected<br>1 Flash array read operation will force the SFDIF flag in the FERSTAT register to be set (see Section 28.3.2.7) and an interrupt will be generated as long as the SFDIE interrupt enable in the FERCNFG register is set (see Section 28.3.2.6) |

## 28.3.2.6 Flash Error Configuration Register (FERCNFG)

The FERCNFG register enables the Flash error interrupts for the FERSTAT flags.

Offset Module Base + 0x0005

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| R | ERSERIE | PGMERIE | 0 | EPVIOLIE | ERSVIE1 | ERSVIE0 | DFDIE | SFDIE |
| W | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

= Unimplemented or Reserved

**Figure 28-10. Flash Error Configuration Register (FERCNFG)**

All assigned bits in the FERCNFG register are readable and writable.

**Table 28-16. FERCNFG Field Descriptions**

| Field | Description |
|---|---|
| 7<br>ERSERIE | **EEE Erase Error Interrupt Enable** — The ERSERIE bit controls interrupt generation when a failure is detected during an EEE erase operation.<br>0 ERSERIF interrupt disabled<br>1 An interrupt will be requested whenever the ERSERIF flag is set (see Section 28.3.2.8) |
| 6<br>PGMERIE | **EEE Program Error Interrupt Enable** — The PGMERIE bit controls interrupt generation when a failure is detected during an EEE program operation.<br>0 PGMERIF interrupt disabled<br>1 An interrupt will be requested whenever the PGMERIF flag is set (see Section 28.3.2.8) |
| 4<br>EPVIOLIE | **EEE Protection Violation Interrupt Enable** — The EPVIOLIE bit controls interrupt generation when a protection violation is detected during a write to the buffer RAM EEE partition.<br>0 EPVIOLIF interrupt disabled<br>1 An interrupt will be requested whenever the EPVIOLIF flag is set (see Section 28.3.2.8) |

#### 28.3.2.9.1 P-Flash Protection Restrictions

The general guideline is that P-Flash protection can only be added and not removed. Table 28-23 specifies all valid transitions between P-Flash protection scenarios. Any attempt to write an invalid scenario to the FPROT register will be ignored. The contents of the FPROT register reflect the active protection scenario. See the FPHS and FPLS bit descriptions for additional restrictions.

**Table 28-23. P-Flash Protection Scenario Transitions**

| From Protection Scenario | To Protection Scenario[1] | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 0 | X | X | X | X | | | | |
| 1 | | X | | X | | | | |
| 2 | | | X | X | | | | |
| 3 | | | | X | | | | |
| 4 | | | | X | X | | | |
| 5 | | | X | X | X | X | | |
| 6 | | X | | X | X | | X | |
| 7 | X | X | X | X | X | X | X | X |

1. Allowed transitions marked with X, see Figure 28-14 for a definition of the scenarios.

### 28.3.2.10 EEE Protection Register (EPROT)

The EPROT register defines which buffer RAM EEE partition areas are protected against writes.

Offset Module Base + 0x0009

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| R | EPOPEN | RNV[6:4] | | | EPDIS | EPS[2:0] | | |
| W | | | | | | | | |
| Reset | F | F | F | F | F | F | F | F |

= Unimplemented or Reserved

**Figure 28-15. EEE Protection Register (EPROT)**

All bits in the EPROT register are readable and writable except for RNV[6:4] which are only readable. The EPOPEN and EPDIS bits can only be written to the protected state. The EPS bits can be written anytime until the EPDIS bit is cleared. If the EPOPEN bit is cleared, the state of the EPDIS and EPS bits is irrelevant.

During the reset sequence, the EPROT register is loaded from the EEE protection byte in the Flash configuration field at global address 0x7F_FF0D located in P-Flash memory (see Table 28-3) as indicated by reset condition F in Figure 28-15. To change the EEE protection that will be loaded during the reset sequence, the P-Flash sector containing the EEE protection byte must be unprotected, then the EEE protection byte must be programmed. If a double bit fault is detected while reading the P-Flash phrase

containing the EEE protection byte during the reset sequence, the EPOPEN bit will be cleared and remaining bits in the EPROT register will be set to leave the buffer RAM EEE partition fully protected.

Trying to write data to any protected area in the buffer RAM EEE partition will result in a protection violation error and the EPVIOLIF flag will be set in the FERSTAT register. Trying to write data to any protected area in the buffer RAM partitioned for user access will not be prevented and the EPVIOLIF flag in the FERSTAT register will not set.

**Table 28-24. EPROT Field Descriptions**

| Field | Description |
|---|---|
| 7<br>EPOPEN | **Enables writes to the Buffer RAM partitioned for EEE**<br>0 The entire buffer RAM EEE partition is protected from writes<br>1 Unprotected buffer RAM EEE partition areas are enabled for writes |
| 6–4<br>RNV[6:4] | **Reserved Nonvolatile Bits** — The RNV bits should remain in the erased state for future enhancements |
| 3<br>EPDIS | **Buffer RAM Protection Address Range Disable** — The EPDIS bit determines whether there is a protected area in a specific region of the buffer RAM EEE partition.<br>0 Protection enabled<br>1 Protection disabled |
| 2–0<br>EPS[2:0] | **Buffer RAM Protection Size** — The EPS[2:0] bits determine the size of the protected area in the buffer RAM EEE partition as shown inTable 28-21. The EPS bits can only be written to while the EPDIS bit is set. |

**Table 28-25. Buffer RAM EEE Partition Protection Address Range**

| EPS[2:0] | Global Address Range | Protected Size |
|---|---|---|
| 000 | 0x13_FFC0 – 0x13_FFFF | 64 bytes |
| 001 | 0x13_FF80 – 0x13_FFFF | 128 bytes |
| 010 | 0x13_FF40 – 0x13_FFFF | 192 bytes |
| 011 | 0x13_FF00 – 0x13_FFFF | 256 bytes |
| 100 | 0x13_FEC0 – 0x13_FFFF | 320 bytes |
| 101 | 0x13_FE80 – 0x13_FFFF | 384 bytes |
| 110 | 0x13_FE40 – 0x13_FFFF | 448 bytes |
| 111 | 0x13_FE00 – 0x13_FFFF | 512 bytes |

### 28.3.2.11 Flash Common Command Object Register (FCCOB)

The FCCOB is an array of six words addressed via the CCOBIX index found in the FCCOBIX register. Byte wide reads and writes are allowed to the FCCOB register.

| Address & Name | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|

= Unimplemented or Reserved

**Figure 29-4. FTM1024K5 Register Summary (continued)**

### 29.3.2.1 Flash Clock Divider Register (FCLKDIV)

The FCLKDIV register is used to control timed events in program and erase algorithms.

Offset Module Base + 0x0000

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| R | FDIVLD | FDIV[6:0] | | | | | | |
| W | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

= Unimplemented or Reserved

**Figure 29-5. Flash Clock Divider Register (FCLKDIV)**

All bits in the FCLKDIV register are readable, bits 6–0 are write once and bit 7 is not writable.

**Table 29-8. FCLKDIV Field Descriptions**

| Field | Description |
|---|---|
| 7<br>FDIVLD | **Clock Divider Loaded**<br>0 FCLKDIV register has not been written<br>1 FCLKDIV register has been written since the last reset |
| 6–0<br>FDIV[6:0] | **Clock Divider Bits** — FDIV[6:0] must be set to effectively divide OSCCLK down to generate an internal Flash clock, FCLK, with a target frequency of 1 MHz for use by the Flash module to control timed events during program and erase algorithms. Table 29-9 shows recommended values for FDIV[6:0] based on OSCCLK frequency. Please refer to Section 29.4.1, “Flash Command Operations,” for more information. |

**CAUTION**

The FCLKDIV register should never be written while a Flash command is executing (CCIF=0). The FCLKDIV register is writable during the Flash reset sequence even though CCIF is clear.

fault information will be recorded until the specific ECC fault flag has been cleared. In the event of simultaneous ECC faults, the priority for fault recording is:

1. Double bit fault over single bit fault
2. CPU over XGATE

Offset Module Base + 0x000E

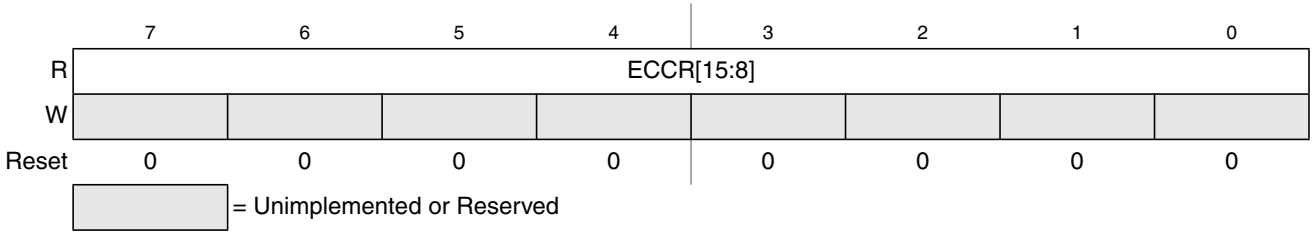



**Figure 29-20. Flash ECC Error Results High Register (FECCRHI)**

Offset Module Base + 0x000F

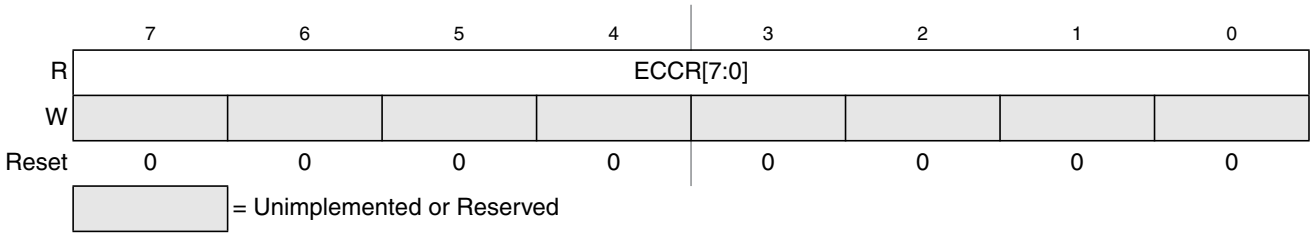



**Figure 29-21. Flash ECC Error Results Low Register (FECCRLO)**

All FECCR bits are readable but not writable.

**Table 29-27. FECCR Index Settings**

| ECCRIX[2:0] | FECCR Register Content | | |
|---|---|---|---|
| | **Bits [15:8]** | **Bit[7]** | **Bits[6:0]** |
| 000 | Parity bits read from Flash block | CPU or XGATE source identity | Global address [22:16] |
| 001 | Global address [15:0] | | |
| 010 | Data 0 [15:0] | | |
| 011 | Data 1 [15:0] (P-Flash only) | | |
| 100 | Data 2 [15:0] (P-Flash only) | | |
| 101 | Data 3 [15:0] (P-Flash only) | | |
| 110 | Not used, returns 0x0000 when read | | |
| 111 | Not used, returns 0x0000 when read | | |

## 0x02C0–0x02EF Analog-to-Digital Converter 12-Bit 16-Channel (ATD0) Map (continued)

| Address | Name | | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x02D6 | ATD0DR3H | R | Bit15 | 14 | 13 | 12 | 11 | 10 | 9 | Bit8 |
| | | W | | | | | | | | |
| 0x02D7 | ATD0DR3L | R | Bit7 | Bit6 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | W | | | | | | | | |
| 0x02D8 | ATD0DR4H | R | Bit15 | 14 | 13 | 12 | 11 | 10 | 9 | Bit8 |
| | | W | | | | | | | | |
| 0x02D9 | ATD0DR4L | R | Bit7 | Bit6 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | W | | | | | | | | |
| 0x02DA | ATD0DR5H | R | Bit15 | 14 | 13 | 12 | 11 | 10 | 9 | Bit8 |
| | | W | | | | | | | | |
| 0x02DB | ATD0DR5L | R | Bit7 | Bit6 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | W | | | | | | | | |
| 0x02DC | ATD0DR6H | R | Bit15 | 14 | 13 | 12 | 11 | 10 | 9 | Bit8 |
| | | W | | | | | | | | |
| 0x02DD | ATD0DR6L | R | Bit7 | Bit6 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | W | | | | | | | | |
| 0x02DE | ATD0DR7H | R | Bit15 | 14 | 13 | 12 | 11 | 10 | 9 | Bit8 |
| | | W | | | | | | | | |
| 0x02DF | ATD0DR7L | R | Bit7 | Bit6 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | W | | | | | | | | |
| 0x02E0 | ATD0DR8H | R | Bit15 | 14 | 13 | 12 | 11 | 10 | 9 | Bit8 |
| | | W | | | | | | | | |
| 0x02E1 | ATD0DR8L | R | Bit7 | Bit6 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | W | | | | | | | | |
| 0x02E2 | ATD0DR9H | R | Bit15 | 14 | 13 | 12 | 11 | 10 | 9 | Bit8 |
| | | W | | | | | | | | |
| 0x02E3 | ATD0DR9L | R | Bit7 | Bit6 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | W | | | | | | | | |
| 0x02E4 | ATD0DR10H | R | Bit15 | 14 | 13 | 12 | 11 | 10 | 9 | Bit8 |
| | | W | | | | | | | | |
| 0x02E5 | ATD0DR10L | R | Bit7 | Bit6 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | W | | | | | | | | |
| 0x02E6 | ATD0DR11H | R | Bit15 | 14 | 13 | 12 | 11 | 10 | 9 | Bit8 |
| | | W | | | | | | | | |
| 0x02E7 | ATD0DR11L | R | Bit7 | Bit6 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | W | | | | | | | | |
| 0x02E8 | ATD0DR12H | R | Bit15 | 14 | 13 | 12 | 11 | 10 | 9 | Bit8 |
| | | W | | | | | | | | |
| 0x02E9 | ATD0DR12L | R | Bit7 | Bit6 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | W | | | | | | | | |
| 0x02EA | ATD0DR13H | R | Bit15 | 14 | 13 | 12 | 11 | 10 | 9 | Bit8 |
| | | W | | | | | | | | |
| 0x02EB | ATD0DR13L | R | Bit7 | Bit6 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | W | | | | | | | | |
| 0x02EC | ATD0DR14H | R | Bit15 | 14 | 13 | 12 | 11 | 10 | 9 | Bit8 |
| | | W | | | | | | | | |