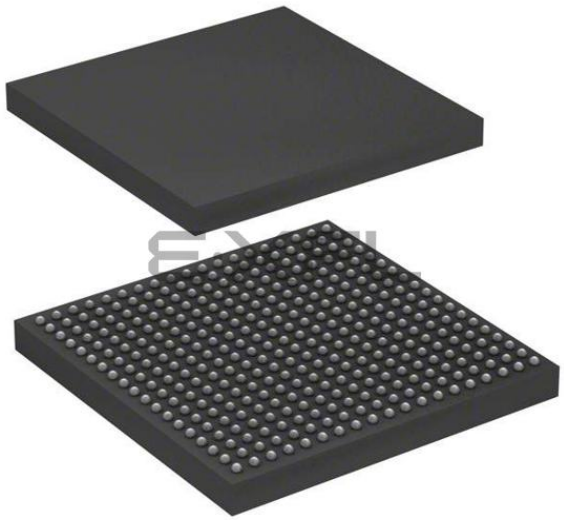


Welcome to E-XFL.COM

Understanding Embedded - Microprocessors

Embedded microprocessors are specialized computing chips designed to perform specific tasks within an embedded system. Unlike general-purpose microprocessors found in personal computers, embedded microprocessors are tailored for dedicated functions within larger systems, offering optimized performance, efficiency, and reliability. These microprocessors are integral to the operation of countless electronic devices, providing the computational power necessary for controlling processes, handling data, and managing communications.

Applications of Embedded - Microprocessors

Embedded microprocessors are utilized across a broad spectrum of applications, making them indispensable in

Details	
Product Status	Active
Core Processor	ARM® Cortex®-A5
Number of Cores/Bus Width	1 Core, 32-Bit
Speed	600MHz
Co-Processors/DSP	Multimedia; NEON™ SIMD
RAM Controllers	LPDDR, LPDDR2, DDR2
Graphics Acceleration	Yes
Display & Interface Controllers	LCD, Touchscreen
Ethernet	10/100Mbps (1)
SATA	-
USB	USB 2.0 (3)
Voltage - I/O	1.2V, 1.8V, 3.3V
Operating Temperature	-40°C ~ 85°C (TA)
Security Features	AES, SHA, TDES, TRNG
Package / Case	361-TFBGA
Supplier Device Package	361-TFBGA (16x16)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsama5d42b-cur

Table 3-1: TFBGA361 Pin Description (Continued)

Pin	Power Rail	I/O Type	Primary		Alternate		PIO Peripheral A		PIO Peripheral B		PIO Peripheral C		Reset State(1)
			Signal	Dir	Signal	Dir	Signal	Dir	Signal	Dir	Signal	Dir	Signal, Dir, PU, PD, HiZ, ST
C13 C17 E11 F10 G11 G14 G17 J13 K18 L13	GNDIODDR	Ground	GNDIODDR	I	–	–	–	–	–	–	–	–	I
M8 N7 P15 R9	VDDIOM	Power supply	VDDIOM	I	–	–	–	–	–	–	–	–	I
M7 M12 N8 P9	GNDIOM	Ground	GNDIOM	I	–	–	–	–	–	–	–	–	I
B8 C8 E8 F8	GNDIOP	Ground	GNDIOP	I	–	–	–	–	–	–	–	–	I
H7 K6 L5 M6	VDDIOP	Power supply	VDDIOP	I	–	–	–	–	–	–	–	–	I
F7 G6 G7 J6 L6 N6	GNDIOP	Ground	GNDIOP	I	–	–	–	–	–	–	–	–	I
V13	VDDUTMIC	Power supply	VDDUTMIC	I	–	–	–	–	–	–	–	–	I
W13 W17	VDDUTMII	Power supply	VDDUTMII	I	–	–	–	–	–	–	–	–	I
P13	GNDUTMI	Ground	GNDUTMI	I	–	–	–	–	–	–	–	–	I
W10	VDDPLLA	Power supply	VDDPLLA	I	–	–	–	–	–	–	–	–	I
L7	GNDPLL	Ground	GNDPLL	I	–	–	–	–	–	–	–	–	I
P14	VDDOSC	Power supply	VDDOSC	I	–	–	–	–	–	–	–	–	I
N13	GNDOSC	Ground	GNDOSC	I	–	–	–	–	–	–	–	–	I
A11	GNDIOP	Ground	GNDIOP	I	–	–	–	–	–	–	–	–	I
C9 N11 P12	VDDANA	Power supply	VDDANA	I	–	–	–	–	–	–	–	–	I
C10 H11 N12	GNDANA	Ground	GNDANA	I	–	–	–	–	–	–	–	–	I
R14	VDDFUSE	Power supply	VDDFUSE	I	–	–	–	–	–	–	–	–	I
R15	GNDFUSE	Ground	GNDFUSE	I	–	–	–	–	–	–	–	–	I

13.5.15 L2CC Interrupt Clear Register

Name:L2CC_ICR

Address:0x00A00220

Access: Programmable in Auxiliary Control Register

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–

23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–

15	14	13	12	11	10	9	8
–	–	–	–	–	–	–	DECERR

7	6	5	4	3	2	1	0
SLVERR	ERRRD	ERRRT	ERRWD	ERRWT	PARRD	PARRT	ECNTR

ECNTR: Event Counter 1/0 Overflow Increment

PARRT: Parity Error on L2 Tag RAM, Read

PARRD: Parity Error on L2 Data RAM, Read

ERRWT: Error on L2 Tag RAM, Write

ERRWD: Error on L2 Data RAM, Write

ERRRT: Error on L2 Tag RAM, Read

ERRRD: Error on L2 Data RAM, Read

SLVERR: SLVERR from L3 memory

DECERR: DECERR from L3 memory

0: No effect. Read returns zero.

1: Clears the corresponding bit in the Raw Interrupt Status Register.

15.10 Arbitration

The Bus Matrix provides an arbitration mechanism that reduces latency when conflicts occur, i.e., when two or more masters try to access the same slave at the same time. One arbiter per AHB slave is provided, thus arbitrating each slave specifically.

The Bus Matrix provides the user with the possibility of choosing between two arbitration types or mixing them for each slave:

- Round-robin Arbitration (default)
- Fixed Priority Arbitration

The resulting algorithm may be complemented by selecting a default master configuration for each slave.

When re-arbitration must be done, specific conditions apply. Refer to Section 15.10.1 “Arbitration Scheduling”.

15.10.1 Arbitration Scheduling

Each arbiter has the ability to arbitrate between two or more master requests. In order to avoid burst breaking and also to provide the maximum throughput for slave interfaces, arbitration may only take place during the following cycles:

- Idle Cycles: when a slave is not connected to any master or is connected to a master which is not currently accessing it.
- Single Cycles: when a slave is currently performing a single access.
- End of Burst Cycles: when the current cycle is the last cycle of a burst transfer. For defined burst length, predicted end of burst matches the size of the transfer but is managed differently for undefined burst length. Refer to Section 15.10.1.1 “Undefined Length Burst Arbitration”.
- Slot Cycle Limit: when the slot cycle counter has reached the limit value indicating that the current master access is too long and must be broken. Refer to Section 15.10.1.2 “Slot Cycle Limit Arbitration”.

15.10.1.1 Undefined Length Burst Arbitration

In order to prevent long AHB burst lengths that can lock the access to the slave for an excessive period of time, the user can trigger the re-arbitration before the end of the incremental bursts. The re-arbitration period can be selected from the following Undefined Length Burst Type (ULBT) possibilities:

- Unlimited: no predetermined end of burst is generated. This value enables 1 Kbyte burst lengths.
- 1-beat bursts: predetermined end of burst is generated at each single transfer during the INCR transfer.
- 4-beat bursts: predetermined end of burst is generated at the end of each 4-beat boundary during INCR transfer.
- 8-beat bursts: predetermined end of burst is generated at the end of each 8-beat boundary during INCR transfer.
- 16-beat bursts: predetermined end of burst is generated at the end of each 16-beat boundary during INCR transfer.
- 32-beat bursts: predetermined end of burst is generated at the end of each 32-beat boundary during INCR transfer.
- 64-beat bursts: predetermined end of burst is generated at the end of each 64-beat boundary during INCR transfer.
- 128-beat bursts: predetermined end of burst is generated at the end of each 128-beat boundary during INCR transfer.

The use of undefined length 8-beat bursts, or less, is discouraged since this may decrease the overall bus bandwidth due to arbitration and slave latencies at each first access of a burst.

However, if the usual length of undefined length bursts is known for a master it is recommended to configure the ULBT accordingly.

This selection can be done through the ULBT field of the Master Configuration Registers (MATRIX_MCFG).

15.10.1.2 Slot Cycle Limit Arbitration

The Bus Matrix contains specific logic to break long accesses, such as very long bursts on a very slow slave (e.g., an external low speed memory). At each arbitration time, a counter is loaded with the value previously written in the SLOT_CYCLE field of the related Slave Configuration Register (MATRIX_SCFG) and decreased at each clock cycle. When the counter elapses, the arbiter has the ability to re-arbitrate at the end of the current AHB access cycle.

Unless a master has a very tight access latency constraint, which could lead to data overflow or underflow due to a badly undersized internal FIFO with respect to its throughput, the Slot Cycle Limit should be disabled (SLOT_CYCLE = 0) or set to its default maximum value in order not to inefficiently break long bursts performed by some Microchip masters.

In most cases, this feature is not needed and should be disabled for power saving.

Warning: This feature cannot prevent any slave from locking its access indefinitely.

15.10.2 Arbitration Priority Scheme

The bus Matrix arbitration scheme is organized in priority pools, each corresponding to an access criticality class as shown in the “Latency Quality of Service” column in Table 15-7.

Access violations may be reported either by an AHB slave through the bus error response (example from the AHB/APB Bridge), or by the Bus Matrix embedded TrustZone controller. In both cases, a bus error response is sent to the offending master and the error is flagged in the Master Error Status Register. An interrupt can be sent to the Secure world, if it has been enabled for that master by writing into the Master Error Interrupt Enable Register. Thus, the offending master is identified. The offending address is registered in the Master Error Address Registers, so that the slave and the targeted security region are also known.

Depending on the hardware parameters and software configuration, the address space of each AHB slave security region may or may not be split into two parts, one belonging to the Secure world and the other one to the Normal world.

Five different security types of AHB slaves are supported. The number of security regions is fixed by design for each slave, independently, from 1 to 8, totalling from 1 up to 16 security areas for security configurable slaves.

15.12.1 Security Types of AHB Slaves

15.12.1.1 Principles

The Bus Matrix supports five different security types of AHB slaves: two fixed types and three configurable types. The security type of an AHB slave is set at hardware design among the following:

- Always Non-secured
- Always Secured
- Internal Securable
- External Securable
- Scalable Securable

The security type is fixed at hardware design on a per-master and a per-slave basis. **Always Non-secured** and **Always Secured** security types are not software configurable.

The different security types have the following characteristics:

- **Always Non-secured** slaves have no security mode access restriction. Their address space is precisely fixed by design. Any out-of-address range access is denied and reported.
- **Always Secured** slaves can only be accessed by a secure master request. Their address space is precisely fixed by design. Any non-secure or out-of-address range access is denied and reported.
- **Internal Securable** is intended for internal memories such as RAM, ROM or embedded Flash. The Internal Securable slave has one slave region which has a hardware fixed base address and Security Region Top. This slave region may be split through software configuration into one Non-secured area plus one Securable area. Inside the slave security region, the split boundary is programmable in powers of 2 from 4 Kbytes up to the full slave security region address space. The security area located below the split boundary may be configured as the Non-secured or the Securable one. The Securable area may be independently configured as Read Secured and/or Write Secured. Any access with security or address range violation is denied and reported.
- **External Securable** is intended for external memories on the EBI, such as DDR, SDRAM, external ROM or NAND Flash. The External Securable slave has identical features as the Internal Securable slave, plus the ability to configure each of its slave security region address space sizes according to the external memory parts used. This avoids mirroring Secured areas into Non-secured areas, and further restricts the overall accessible address range. Any access with security or configured address range violation is denied and reported.
- **Scalable Securable** is intended for external memories with a dedicated slave, such as DDR. The Scalable Securable slave is divided into a fixed number of scalable, equally sized, and contiguous security regions. Each of them can be split in the same way as for Internal or External Securable slaves. The security region size must be configured by software, so that the equally-sized regions fill the actual available memory. This avoids mirroring Secured areas into Non-secured areas, and further restricts the overall accessible address range. Any access with security or configured address range violation is denied and reported.

As the security type is fixed at hardware design on a per-master and per-slave basis, it is possible to set some slave access security as configurable from one or some particular masters, and to fix the access as Always Secured from all the other masters.

As the security type is fixed by design at the slave region level, different security region types can be mixed inside a single slave.

Likewise, the mapping base address and the accessible address range of each AHB slave or slave region may have been hardware-restricted on a per-master basis from no access to full slave address space.

17.8.8 Register Write Protection

To prevent any single software error from corrupting AIC behavior, certain registers in the address space can be write-protected by setting the WPEN bit in the AIC Write Protection Mode Register (AIC_WPMR).

If a write access to a write-protected register is detected, the WPVS flag in the AIC Write Protection Status Register (AIC_WPSR) is set and the field WPVSRC indicates the register in which the write access has been attempted.

The WPVS bit is automatically cleared after reading AIC_WPSR.

The following registers can be write-protected:

- AIC Source Mode Register
- AIC Source Vector Register
- AIC Spurious Interrupt Vector Register
- AIC Debug Control Register

25.5.7 SFC Data Register x

Name:SFC_DRx [x=0..15]

Address:0xFC060020

Access: Read/Write

31	30	29	28	27	26	25	24
DATA							
23	22	21	20	19	18	17	16
DATA							
15	14	13	12	11	10	9	8
DATA							
7	6	5	4	3	2	1	0
DATA							

DATA: Fuse Data

READ: Reports the state of the corresponding fuses.

WRITE: The data to be programmed in the corresponding fuses. Only bits with a value of ‘1’ are programmed. Writing this register automatically triggers a programming sequence of the corresponding fuses. Note that a write to the Key Register (SFC_KR) with the correct key code must always precede any write to SFC_DRx.

28.6.45 PIO Fall/Rise - Low/High Status Register

Name:PIO_FRLHSR

Address:0xFC06A0D8 (PIOA), 0xFC06B0D8 (PIOB), 0xFC06C0D8 (PIOC), 0xFC0680D8 (PIOD), 0xFC06D0D8 (PIOE)

Access: Read-only

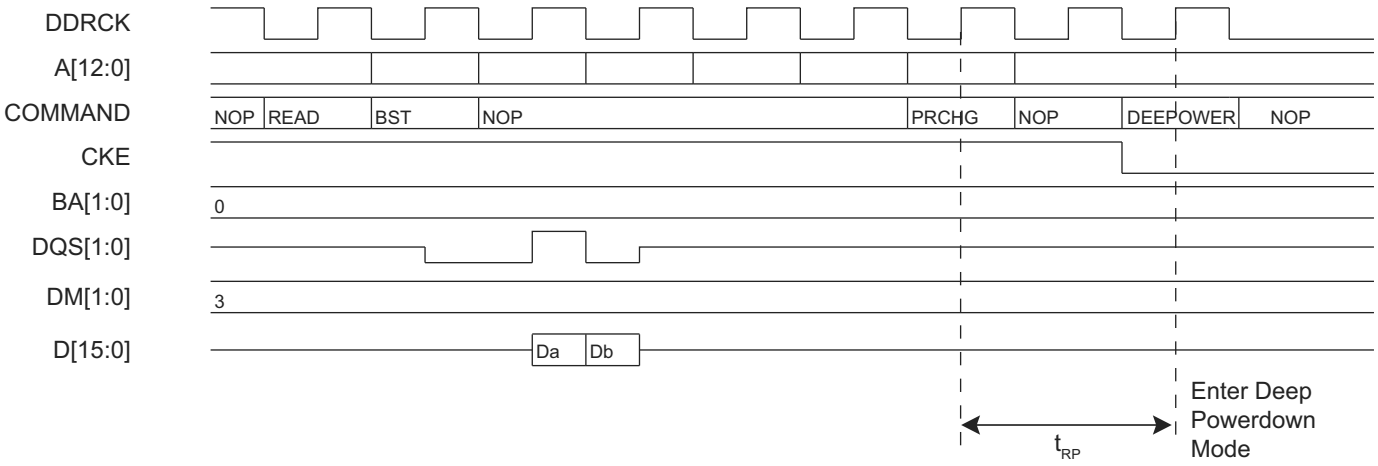
31	30	29	28	27	26	25	24
P31	P30	P29	P28	P27	P26	P25	P24
23	22	21	20	19	18	17	16
P23	P22	P21	P20	P19	P18	P17	P16
15	14	13	12	11	10	9	8
P15	P14	P13	P12	P11	P10	P9	P8
7	6	5	4	3	2	1	0
P7	P6	P5	P4	P3	P2	P1	P0

P0–P31: Edge/Level Interrupt Source Selection

0: The interrupt source is a falling edge detection (if PIO_ELSR = 0) or low-level detection event (if PIO_ELSR = 1).

1: The interrupt source is a rising edge detection (if PIO_ELSR = 0) or high-level detection event (if PIO_ELSR = 1).

Figure 29-17: Deep Powerdown Mode Entry



29.5.3.4 Reset Mode

The Reset mode is a feature of DDR2-SDRAM. This mode is activated by writing a 3 to the Low-power Command bit (LPCB) and a one to the Clock Frozen Command bit (CLK_FR) in the MPDDRC Low-power Register.

When this mode is enabled, the MPDDRC leaves Normal mode (MPDDRC_MR.MODE = 0) and the controller is frozen. Before enabling this mode, the user must make sure there is no access in progress.

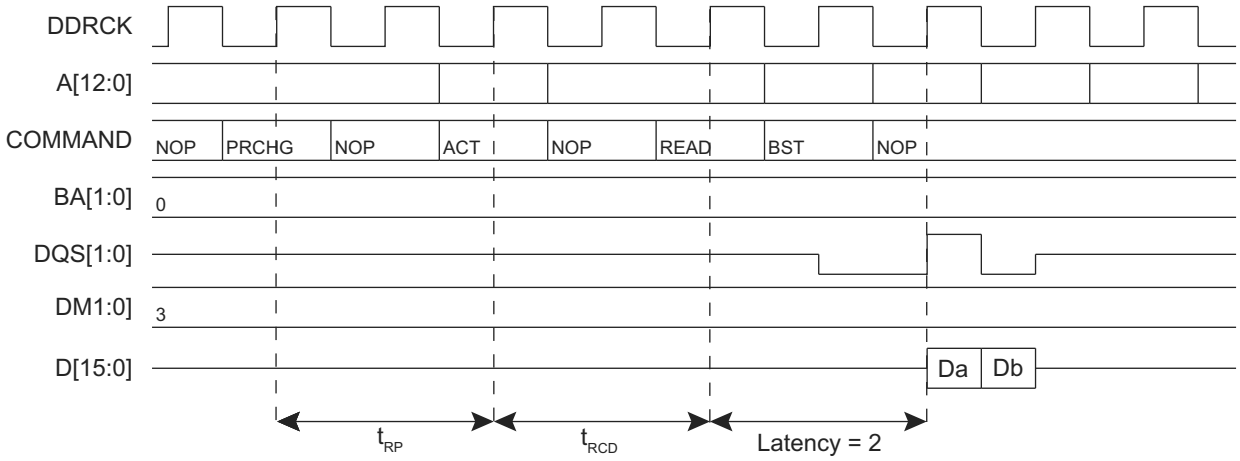
To exit Reset mode, the Low-power Command bit (LPCB) must be written to zero, the Clock Frozen Command bit (CLK_FR) must be written to zero and the initialization sequence must be generated by software (refer to Section 29.4.2 “DDR2-SDRAM Initialization”).

29.5.4 Multiport Functionality

The DDR-SDRAM protocol imposes a check of timings prior to performing a read or a write access, thus decreasing system performance. An access to DDR-SDRAM is performed if banks and rows are open (or active). To activate a row in a particular bank, the last open row must be deactivated and a new row must be open. Two DDR-SDRAM commands must be performed to open a bank: Precharge command and Activate command with respect to t_{RP} timing. Before performing a read or write command, t_{RCD} timing must be checked.

This operation generates a significant bandwidth loss (refer to Figure 29-18).

Figure 29-18: t_{RP} and t_{RCD} Timings



4 cycles before performing a read command

- XFACTOR and YFACTOR fields—Define the scaling ratio.

The position and size attributes are to be programmed to keep the window within the display area.

When the Color Lookup Table mode is enabled, the restrictions detailed in the following table apply on the horizontal and vertical window sizes.

Table 32-8: Color Lookup Table Mode and Window Size

CLUT Mode	X-Y Size Requirement
1 bpp	Multiple of 8 pixels
2 bpp	Multiple of 4 pixels
4 bpp	Multiple of 2 pixels
8 bpp	Free size

Pixel striding is disabled when CLUT mode is enabled.

When YUV mode is enabled, the restrictions detailed in the following table apply on the window size.

Table 32-9: YUV Mode and Window Size

YUV Mode	X-Y Requirement, Scaling Turned Off	X-Y Requirement, Scaling Turned On
AYUV	Free size	X-Y size is greater than 5
YUV 4:2:2 packed	XSIZE is greater than 2 pixels	X-Y size is greater than 5
YUV 4:2:2 semiplanar	XSIZE is greater than 2 pixels	X-Y size is greater than 5
YUV 4:2:2 planar	XSIZE is greater than 2 pixels	X-Y size is greater than 5
YUV 4:2:0 semiplanar	XSIZE is greater that 2 pixels	X-Y size is greater than 5
YUV 4:2:0 planar	XSIZE is greater than 2 pixels	X-Y size is greater than 5

In RGB mode, there is no restriction on the line length.

32.6.3.4 Overlay Blender Attributes

When two or more video layers are used, alpha blending is performed to define the final image displayed. Each window has its own blending attributes.

- CRKEY bit—Enables the chroma keying and match logic.
- INV bit—Performs bit inversion at pixel level.
- ITER2BL bit—When written to ‘1’, the iterated data path is selected.
- ITER bit—When written to ‘1’, the iterated value is used in the iterated data path, otherwise the iterated value is set to 0.
- REVALPHA bit—Uses the reverse alpha value.
- GAEN bit—Enables the global alpha value in the data path.
- LAEN bit—Enables the local alpha value from the pixel.
- OVR bit—When written to ‘1’, the overlay is selected as an input of the blender.
- DMA bit—The DMA data path is activated.
- REP bit—Enables the bit replication to fill the 24-bit internal data path.
- DSTKEY bit—When written to ‘1’, Destination keying is enabled.
- GA field—Defines the global alpha value.

32.6.3.5 Overlay Attributes Software Operation

1. When required, write the overlay attributes configuration registers.
2. Set UPDATEEN field of the CHXCHER register.
3. Poll UPDATESR field in the CHXCHSR, the update applies when that field is reset.

37.8.4 GMAC User Register

Name:GMAC_UR

Address:0xF802000C (0), 0xFC02800C (1)

Access: Read/Write

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–
15	14	13	12	11	10	9	8
–	–	–	–	–	–	–	–
7	6	5	4	3	2	1	0
–	–	–	–	–	–	–	RMII

RMII: Reduced MII Mode

0: MII mode is selected (default).

1: RMII mode is selected.

37.8.33 GMAC IPG Stretch Register

Name:GMAC_IPGS

Address:0xF80200BC (0), 0xFC0280BC (1)

Access: Read/Write

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–

23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–

15	14	13	12	11	10	9	8
FL							

7	6	5	4	3	2	1	0
FL							

FL: Frame Length

Bits 7:0 are multiplied with the previously transmitted frame length (including preamble). Bits 15:8 +1 divide the frame length. If the resulting number is greater than 96 and bit 28 is set in the Network Configuration Register then the resulting number is used for the transmit inter-packet-gap. 1 is added to bits 15:8 to prevent a divide by zero. Refer to Section 37.6.4 “MAC Transmit Block”.

39.7.5 Register Write Protection

To prevent any single software error from corrupting SPI behavior, certain registers in the address space can be write-protected in the SPI Write Protection Mode Register (SPI_WPMR).

If a write access to a write-protected register is detected, the WPVS flag in the SPI Write Protection Status Register (SPI_WPSR) is set and the WPVSRC field indicates the register in which the write access has been attempted.

The WPVS bit is automatically cleared after reading SPI_WPSR.

The following registers are write-protected when WPEN is set in SPI_WPMR:

- SPI Mode Register
- SPI Chip Select Register

39.8.10 SPI Write Protection Mode Register

Name:SPI_WPMR

Address:0xF80100E4 (0), 0xFC0180E4 (1), 0xFC01C0E4 (2)

Access: Read/Write

31	30	29	28	27	26	25	24
WPKEY							
23	22	21	20	19	18	17	16
WPKEY							
15	14	13	12	11	10	9	8
WPKEY							
7	6	5	4	3	2	1	0
–	–	–	–	–	–	–	WPEN

WPEN: Write Protection Enable

0: Disables the write protection if WPKEY corresponds to 0x535049 (“SPI” in ASCII)

1: Enables the write protection if WPKEY corresponds to 0x535049 (“SPI” in ASCII)

WPKEY: Write Protection Key

Value	Name	Description
0x535049	PASSWD	Writing any other value in this field aborts the write operation of the WPEN bit. Always reads as 0.

Refer to Section 39.7.5 “Register Write Protection” for the list of registers that can be write-protected.

40.8.5 TWI Clock Waveform Generator Register

Name: TWI_CWGR

Address: 0xF8014010 (0), 0xF8018010 (1), 0xF8024010 (2), 0xFC038010 (3)

Access: Read/Write

31	30	29	28	27	26	25	24
–	–	–	HOLD				

23	22	21	20	19	18	17	16
–	–	–	–	–	CKDIV		

15	14	13	12	11	10	9	8
CHDIV							

7	6	5	4	3	2	1	0
CLDIV							

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

TWI_CWGR is only used in Master mode.

CLDIV: Clock Low Divider

The TWCK low period is defined as follows: $t_{low} = ((CLDIV \times 2^{CKDIV}) + 4 \times t_{peripheral\ clock}$

CHDIV: Clock High Divider

The TWCK high period is defined as follows: $t_{high} = ((CHDIV \times 2^{CKDIV}) + 4 \times t_{peripheral\ clock}$

CKDIV: Clock Divider

The CKDIV field is used to increase both TWCK high and low periods.

HOLD: TWD Hold Time versus TWCK falling

TWD is kept unchanged after TWCK falling edge for a period of (HOLD + 3) $\times$ $t_{peripheral\ clock}$.

41.9.14 SSC Interrupt Enable Register

Name:SSC_IER

Address:0xF8008044 (0), 0xFC014044 (1)

Access: Write-only

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–
15	14	13	12	11	10	9	8
–	–	–	–	RXSYN	TXSYN	CP1	CP0
7	6	5	4	3	2	1	0
–	–	OVRUN	RXRDY	–	–	TXEMPTY	TXRDY

TXRDY: Transmit Ready Interrupt Enable

0: No effect.

1: Enables the Transmit Ready Interrupt.

TXEMPTY: Transmit Empty Interrupt Enable

0: No effect.

1: Enables the Transmit Empty Interrupt.

RXRDY: Receive Ready Interrupt Enable

0: No effect.

1: Enables the Receive Ready Interrupt.

OVRUN: Receive Overrun Interrupt Enable

0: No effect.

1: Enables the Receive Overrun Interrupt.

CP0: Compare 0 Interrupt Enable

0: No effect.

1: Enables the Compare 0 Interrupt.

CP1: Compare 1 Interrupt Enable

0: No effect.

1: Enables the Compare 1 Interrupt.

TXSYN: Tx Sync Interrupt Enable

0: No effect.

1: Enables the Tx Sync Interrupt.

RXSYN: Rx Sync Interrupt Enable

0: No effect.

1: Enables the Rx Sync Interrupt.

RX_MPOL: Receiver Manchester Polarity

0: Logic zero is coded as a zero-to-one transition, Logic one is coded as a one-to-zero transition.

1: Logic zero is coded as a one-to-zero transition, Logic one is coded as a zero-to-one transition.

ONE: Must Be Set to 1

Bit 29 must always be set to 1 when programming the US_MAN register.

DRIFT: Drift Compensation

0: The USART cannot recover from an important clock drift

1: The USART can recover from clock drift. The 16X clock mode must be enabled.

46.7.17 TC QDEC Interrupt Disable Register

Name:TC_QIDR

Address:0xF801C0CC (0), 0xFC0200CC (1), 0xFC0240CC (2)

Access: Write-only

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–
15	14	13	12	11	10	9	8
–	–	–	–	–	–	–	–
7	6	5	4	3	2	1	0
–	–	–	–	–	QERR	DIRCHG	IDX

IDX: Index

0: No effect.

1: Disables the interrupt when a rising edge occurs on IDX input.

DIRCHG: Direction Change

0: No effect.

1: Disables the interrupt when a change on rotation direction is detected.

QERR: Quadrature Error

0: No effect.

1: Disables the interrupt when a quadrature error occurs on PHA, PHB.

51.5 Triple Data Encryption Standard (TDES) User Interface

Table 51-6: Register Mapping

Offset	Register	Name	Access	Reset
0x00	Control Register	TDES_CR	Write-only	–
0x04	Mode Register	TDES_MR	Read/Write	0x2
0x08–0x0C	Reserved	–	–	–
0x10	Interrupt Enable Register	TDES_IER	Write-only	–
0x14	Interrupt Disable Register	TDES_IDR	Write-only	–
0x18	Interrupt Mask Register	TDES_IMR	Read-only	0x0
0x1C	Interrupt Status Register	TDES_ISR	Read-only	0x0000001E
0x20	Key 1 Word Register 0	TDES_KEY1WR0	Write-only	–
0x24	Key 1 Word Register 1	TDES_KEY1WR1	Write-only	–
0x28	Key 2 Word Register 0	TDES_KEY2WR0	Write-only	–
0x2C	Key 2 Word Register 1	TDES_KEY2WR1	Write-only	–
0x30	Key 3 Word Register 0	TDES_KEY3WR0	Write-only	–
0x34	Key 3 Word Register 1	TDES_KEY3WR1	Write-only	–
0x38–0x3C	Reserved	–	–	–
0x40	Input Data Register 0	TDES_IDATAR0	Write-only	–
0x44	Input Data Register 1	TDES_IDATAR1	Write-only	–
0x48–0x4C	Reserved	–	–	–
0x50	Output Data Register 0	TDES_ODATAR0	Read-only	0x0
0x54	Output Data Register 1	TDES_ODATAR1	Read-only	0x0
0x58–0x5C	Reserved	–	–	–
0x60	Initialization Vector Register 0	TDES_IVR0	Write-only	–
0x64	Initialization Vector Register 1	TDES_IVR1	Write-only	–
0x68–0x6C	Reserved	–	–	–
0x70	XTEA Rounds Register	TDES_XTEA_RNDR	Read/Write	0x0
0x74–0xFC	Reserved	–	–	–

53.6.3 AESB Interrupt Enable Register

Name: AESB_IER

Address:0xF0020010

Access: Write-only

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–
15	14	13	12	11	10	9	8
–	–	–	–	–	–	–	URAD
7	6	5	4	3	2	1	0
–	–	–	–	–	–	–	DATRDY

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

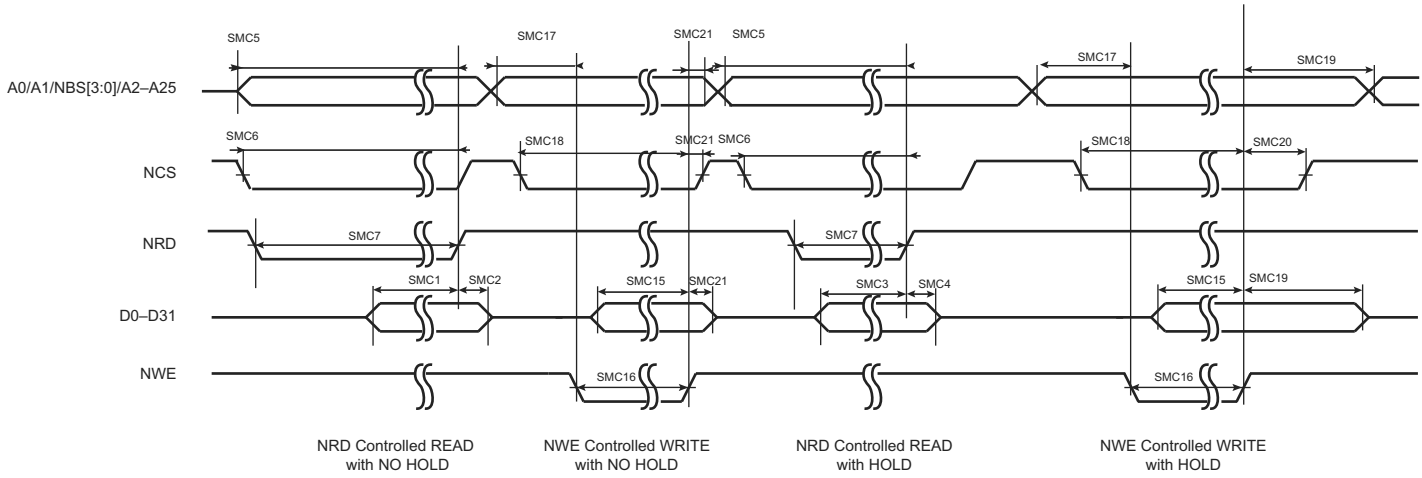
0: No effect.

1: Enables the corresponding interrupt.

DATRDY: Data Ready Interrupt Enable

URAD: Unspecified Register Access Detection Interrupt Enable

Figure 56-5: SMC Timings - NRD Controlled Read and NWE Controlled Write



56.14 SPI Timings

56.14.1 Maximum SPI Frequency

The following formulas give maximum SPI frequency in Master read and write modes and in Slave read and write modes.

- Master Write Mode

  The SPI is only sending data to a slave device such as an LCD, for example. The limit is given by SPI_2 (or SPI_5) timing.

- Master Read Mode

$$f_{SPCK}Max = \frac{1}{SPI_0(\text{or } SPI_3) + t_{valid}}$$

  t_{valid} is the slave time response to output data after deleting an SPCK edge. For a non-volatile memory with t_{valid} (or t_v) = 12 ns, f_{SPCK}max = 45 MHz at V_{DDIO} = 3.3V.

- Slave Read Mode

  In slave mode, SPCK is the input clock for the SPI. The max SPCK frequency is given by setup and hold timings SPI_7/SPI_8 (or SPI_{10}/SPI_{11}). Since this gives a frequency well above the pad limit, the limit in slave read mode is given by SPCK pad.

- Slave Write Mode

$$f_{SPCK}Max = \frac{1}{SPI_6(\text{or } SPI_9) + t_{setup}}$$

  t_{setup} is the setup time from the master before sampling data (12 ns).

  This gives f_{SPCK}Max = 45 MHz @ V_{DDIO} = 3.3V.