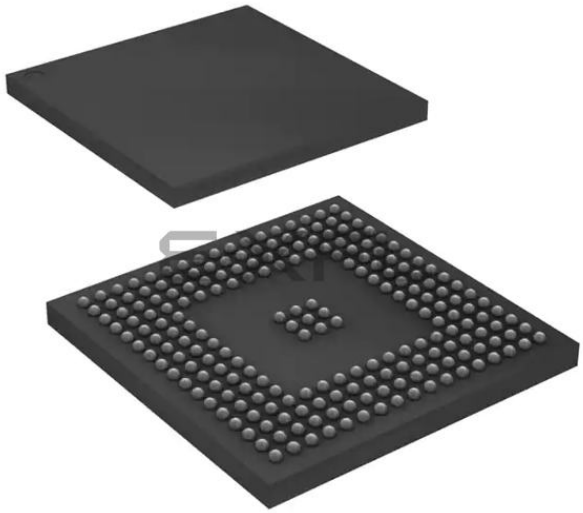
Microchip Technology - AT91SAM9G10-CU-999 Datasheet



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	Obsolete
Core Processor	ARM926EJ-S
Number of Cores/Bus Width	1 Core, 32-Bit
Speed	266MHz
Co-Processors/DSP	-
RAM Controllers	SDRAM, SRAM
Graphics Acceleration	No
Display & Interface Controllers	LCD
Ethernet	-
SATA	-
USB	USB 2.0 (2)
Voltage - I/O	1.8V, 3.3V
Operating Temperature	-40°C ~ 85°C (TA)
Security Features	-
Package / Case	217-LFBGA
Supplier Device Package	217-LFBGA (15x15)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/at91sam9g10-cu-999

Table 12-2. SAM9G10 JTAG Boundary Scan Register (Continued)

Bit Number	Pin Name	Pin Type	Associated BSR Cells
473	NCS2	OUT	OUTPUT
472	NCS3	OUT	OUTPUT
471	NRD	OUT	OUTPUT
470	NWR0	IN/OUT	INPUT
469			OUTPUT
468	NWR1	IN/OUT	INPUT
467			OUTPUT
466		internal	
465	NWR3	OUT	OUTPUT
464	SDRAMCKE	OUT	OUTPUT
463	SDRAMCKE/RAS/CAS SDA10/SDWE		CONTROL
462	SDRAMCLK	IN/OUT	INPUT
461			OUTPUT
460			CONTROL
459	RAS	OUT	OUTPUT
458	CAS	OUT	OUTPUT
457	SDWE	OUT	OUTPUT
456	D0	IN/OUT	INPUT
455			OUTPUT
454			CONTROL
453		internal	
452	D1	IN/OUT	INPUT
451			OUTPUT
450			CONTROL
449	D2	IN/OUT	INPUT
448			OUTPUT
447			CONTROL
446	D3	IN/OUT	INPUT
445			OUTPUT
444			CONTROL
443	D4	IN/OUT	INPUT
442			OUTPUT
441			CONTROL
440		internal	

Table 12-2. SAM9G10 JTAG Boundary Scan Register (Continued)

Bit Number	Pin Name	Pin Type	Associated BSR Cells
08	A10	OUT	OUTPUT
07	SDA10	OUT	OUTPUT
06	A11	OUT	OUTPUT
05	A12	OUT	OUTPUT
04	A13	OUT	OUTPUT
03	A14	OUT	OUTPUT
02	A15	OUT	OUTPUT
01	A16	OUT	OUTPUT
00	A17	OUT	OUTPUT

12.5.6 ID Code Register

Access: Read-only

31	30	29	28	27	26	25	24
VERSION				PART NUMBER			

23	22	21	20	19	18	17	16
PART NUMBER							

15	14	13	12	11	10	9	8
PART NUMBER				MANUFACTURER IDENTITY			

7	6	5	4	3	2	1	0
MANUFACTURER IDENTITY							1

- **VERSION[31:28]: Product Version Number**

Set to 0x0.

- **PART NUMBER[27:12]: Product Part Number**

Product part Number is 0x5B25

- **MANUFACTURER IDENTITY[11:1]**

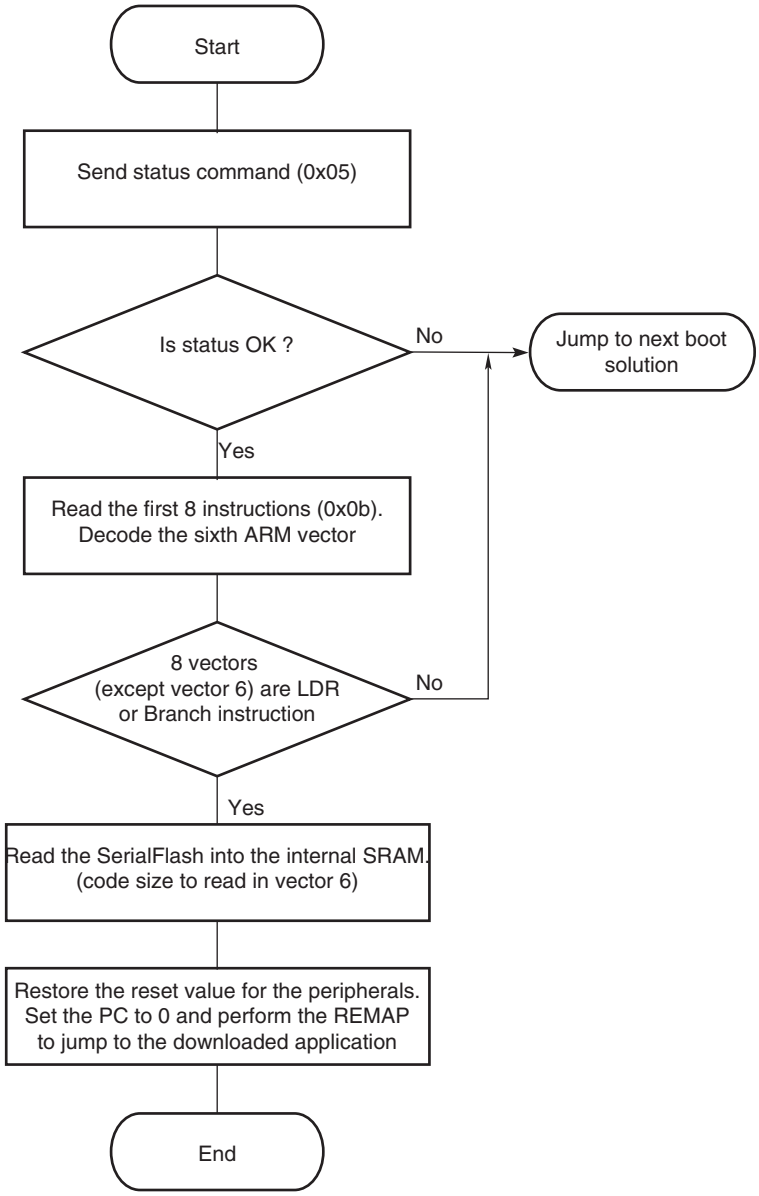
Set to 0x01F.

Bit[0] Required by IEEE Std. 1149.1.

Set to 0x1.

JTAG ID Code value is 0x05B2_503F.

Figure 13-6. Serial Flash Download



13.6 DataFlash Boot Sequence

The Dataflash boot looks for a valid application in the SPI DataFlash memory.

SPI0 is configured in master mode to generate a SPCK at 8MHz. Serial Flash shall be connected to NPCS0.

The DataFlash boot reads the dataflash flash status register (Instruction code 0xD7). The data flash is considered as ready if bit 7 of the returned status register is set.

If no dataflash is connected or if it does not answer, DataFlash boots exits after a 1000 attempts.

16.4.1 Periodic Interval Timer Mode Register

Register Name: PIT_MR

Address: 0xFFFFFD30

Access Type: Read/Write

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

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

15	14	13	12	11	10	9	8
PIV							

7	6	5	4	3	2	1	0
PIV							

- **PIV: Periodic Interval Value**

Defines the value compared with the primary 20-bit counter of the Periodic Interval Timer (CPIV). The period is equal to (PIV + 1).

- **PITEN: Period Interval Timer Enabled**

0 = The Periodic Interval Timer is disabled when the PIV value is reached.

1 = The Periodic Interval Timer is enabled.

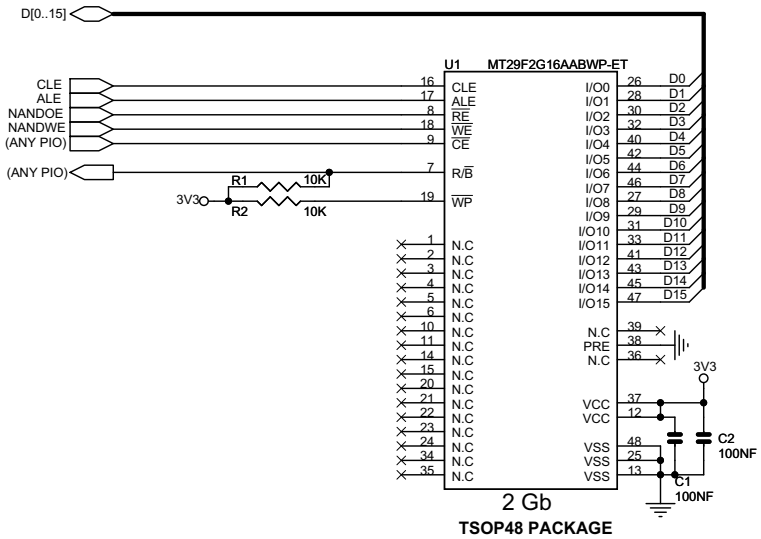
- **PITIEN: Periodic Interval Timer Interrupt Enable**

0 = The bit PITS in PIT_SR has no effect on interrupt.

1 = The bit PITS in PIT_SR asserts interrupt.

21.7.4 16-bit NAND Flash

21.7.4.1 Hardware Configuration



21.7.4.2 Software Configuration

The software configuration is the same as for an 8-bit NAND Flash except the data bus width programmed in the mode register of the Static Memory Controller.

21.7.7.2 Software Configuration

The following configuration has to be performed:

- Assign the EBI CS4 and/or EBI_CS5 to the CompactFlash Slot 0 or/and Slot 1 by setting the bit EBI_CS4A or/and EBI_CS5A in the EBI Chip Select Assignment Register located in the bus matrix memory space.
- The address line A21 is to select Alternate True IDE (A21=1) or True IDE (A21=0) modes.
- CFRNW, CFS0, CFCS1, CFCE1 and CFCE2 signals are multiplexed with PIO lines and thus the dedicated PIOs must be programmed in peripheral mode in the PIO controller.
- Configure a PIO line as an output for CFRST and two others as an input for CFIRQ and CARD DETECT functions respectively.
- Configure SMC CS4 and/or SMC_CS5 (for Slot 0 or 1) Setup, Pulse, Cycle and Mode accordingly to Compact Flash timings and system bus frequency.

plify the description of the functional operations and the user interface, the interrupt sources are named FIQ, SYS, and PID2 to PID31.

25.7 Functional Description

25.7.1 Interrupt Source Control

25.7.1.1 Interrupt Source Mode

The Advanced Interrupt Controller independently programs each interrupt source. The SRCTYPE field of the corresponding AIC_SMR (Source Mode Register) selects the interrupt condition of each source.

The internal interrupt sources wired on the interrupt outputs of the embedded peripherals can be programmed either in level-sensitive mode or in edge-triggered mode. The active level of the internal interrupts is not important for the user.

The external interrupt sources can be programmed either in high level-sensitive or low level-sensitive modes, or in positive edge-triggered or negative edge-triggered modes.

25.7.1.2 Interrupt Source Enabling

Each interrupt source, including the FIQ in source 0, can be enabled or disabled by using the command registers; AIC_IECR (Interrupt Enable Command Register) and AIC_IDCR (Interrupt Disable Command Register). This set of registers conducts enabling or disabling in one instruction. The interrupt mask can be read in the AIC_IMR register. A disabled interrupt does not affect servicing of other interrupts.

25.7.1.3 Interrupt Clearing and Setting

All interrupt sources programmed to be edge-triggered (including the FIQ in source 0) can be individually set or cleared by writing respectively the AIC_ISCR and AIC_ICCR registers. Clearing or setting interrupt sources programmed in level-sensitive mode has no effect.

The clear operation is perfunctory, as the software must perform an action to reinitialize the “memorization” circuitry activated when the source is programmed in edge-triggered mode. However, the set operation is available for auto-test or software debug purposes. It can also be used to execute an AIC-implementation of a software interrupt.

The AIC features an automatic clear of the current interrupt when the AIC_IVR (Interrupt Vector Register) is read. Only the interrupt source being detected by the AIC as the current interrupt is affected by this operation. (See “Priority Controller” on page 243.) The automatic clear reduces the operations required by the interrupt service routine entry code to reading the AIC_IVR. Note that the automatic interrupt clear is disabled if the interrupt source has the Fast Forcing feature enabled as it is considered uniquely as a FIQ source. (For further details, See “Fast Forcing” on page 247.)

The automatic clear of the interrupt source 0 is performed when AIC_FVR is read.

25.7.1.4 Interrupt Status

For each interrupt, the AIC operation originates in AIC_IPR (Interrupt Pending Register) and its mask in AIC_IMR (Interrupt Mask Register). AIC_IPR enables the actual activity of the sources, whether masked or not.

The PLL allows multiplication of the divider's outputs. The PLL clock signal has a frequency that depends on the respective source signal frequency and on the parameters DIV and MUL. The factor applied to the source signal frequency is (MUL + 1)/DIV. When MUL is written to 0, the corresponding PLL is disabled and its power consumption is saved. Re-enabling the PLL can be performed by writing a value higher than 0 in the MUL field.

Whenever the PLL is re-enabled or one of its parameters is changed, the LOCK bit (LOCKA or LOCKB) in PMC_SR is automatically cleared. The values written in the PLLCOUNT field (PLLACOUNT or PLLBCOUNT) in CKGR_PLLR (CKGR_PLLAR or CKGR_PLLBR), are loaded in the PLL counter. The PLL counter then decrements at the speed of the Slow Clock until it reaches 0. At this time, the LOCK bit is set in PMC_SR and can trigger an interrupt to the processor. The user has to load the number of Slow Clock cycles required to cover the PLL transient time into the PLLCOUNT field. The transient time depends on the PLL filter. The initial state of the PLL and its target frequency can be calculated using a specific tool provided by Atmel.

During the PLLA or PLLB initialization, the PMC_PLLICPR register must be programmed correctly.

28.5.10 Debug Unit Chip ID Register

Name: DBGU_CIDR

Address: 0xFFFFF240

Access Type: Read-only

31	30	29	28	27	26	25	24
EXT	NVPTYP			ARCH			
23	22	21	20	19	18	17	16
ARCH				SRAMSIZ			
15	14	13	12	11	10	9	8
NVPSIZ2				NVPSIZ			
7	6	5	4	3	2	1	0
EPROC			VERSION				

- **VERSION: Version of the Device**

Values depend upon the version of the device.

- **EPROC: Embedded Processor**

EPROC			Processor
0	0	1	ARM946ES
0	1	0	ARM7TDMI
1	0	0	ARM920T
1	0	1	ARM926EJS

- **NVPSIZ: Nonvolatile Program Memory Size**

NVPSIZ				Size
0	0	0	0	None
0	0	0	1	8K bytes
0	0	1	0	16K bytes
0	0	1	1	32K bytes
0	1	0	0	Reserved
0	1	0	1	64K bytes
0	1	1	0	Reserved
0	1	1	1	128K bytes
1	0	0	0	Reserved
1	0	0	1	256K bytes
1	0	1	0	512K bytes
1	0	1	1	Reserved
1	1	0	0	1024K bytes

31.4.1 I/O Lines Description

Table 31-3. I/O Lines Description

Pin Name	Pin Description	Type
TWD	Two-wire Serial Data	Input/Output
TWCK	Two-wire Serial Clock	Input/Output

31.5 Product Dependencies

31.5.1 I/O Lines

Both TWD and TWCK are bidirectional lines, connected to a positive supply voltage via a current source or pull-up resistor (see Figure 31-2 on page 384). When the bus is free, both lines are high. The output stages of devices connected to the bus must have an open-drain or open-collector to perform the wired-AND function.

TWD and TWCK pins may be multiplexed with PIO lines. To enable the TWI, the programmer must perform the following step:

- Program the PIO controller to dedicate TWD and TWCK as peripheral lines.

The user must not program TWD and TWCK as open-drain. It is already done by the hardware.

Table 31-4. I/O Lines

Instance	Signal	I/O Line	Peripheral
TWI	TWCK	PA8	A
TWI	TWD	PA7	A

31.5.2 Power Management

- Enable the peripheral clock.

The TWI interface may be clocked through the Power Management Controller (PMC), thus the programmer must first configure the PMC to enable the TWI clock.

31.5.3 Interrupt

The TWI interface has an interrupt line connected to the Advanced Interrupt Controller (AIC). In order to handle interrupts, the AIC must be programmed before configuring the TWI.

Table 31-5. Peripheral IDs

Instance	ID
TWI	11

31.6 Functional Description

31.6.1 Transfer Format

The data put on the TWD line must be 8 bits long. Data is transferred MSB first; each byte must be followed by an acknowledgement. The number of bytes per transfer is unlimited (see Figure 31-4).

Each transfer begins with a START condition and terminates with a STOP condition (see Figure 31-3).

After a Master Write transfer, the Serial Clock line is stretched (tied low) while no new data is written in the TWI_THR or until a STOP command is performed.

See Figure 31-6, Figure 31-7, and Figure 31-8.

Figure 31-6. Master Write with One Data Byte

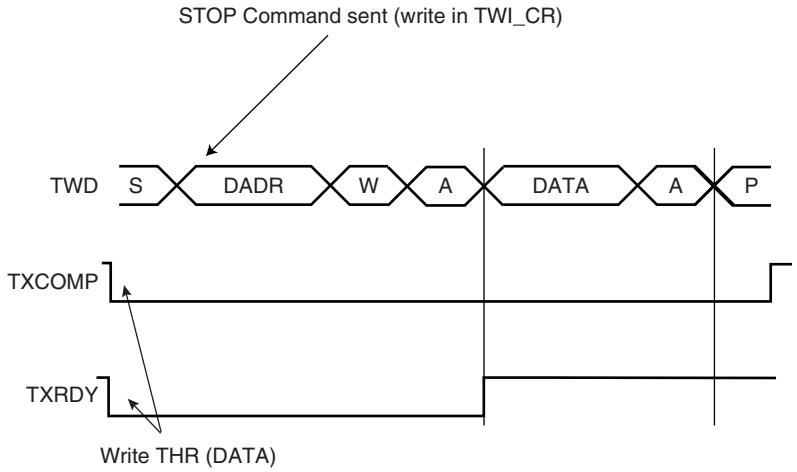
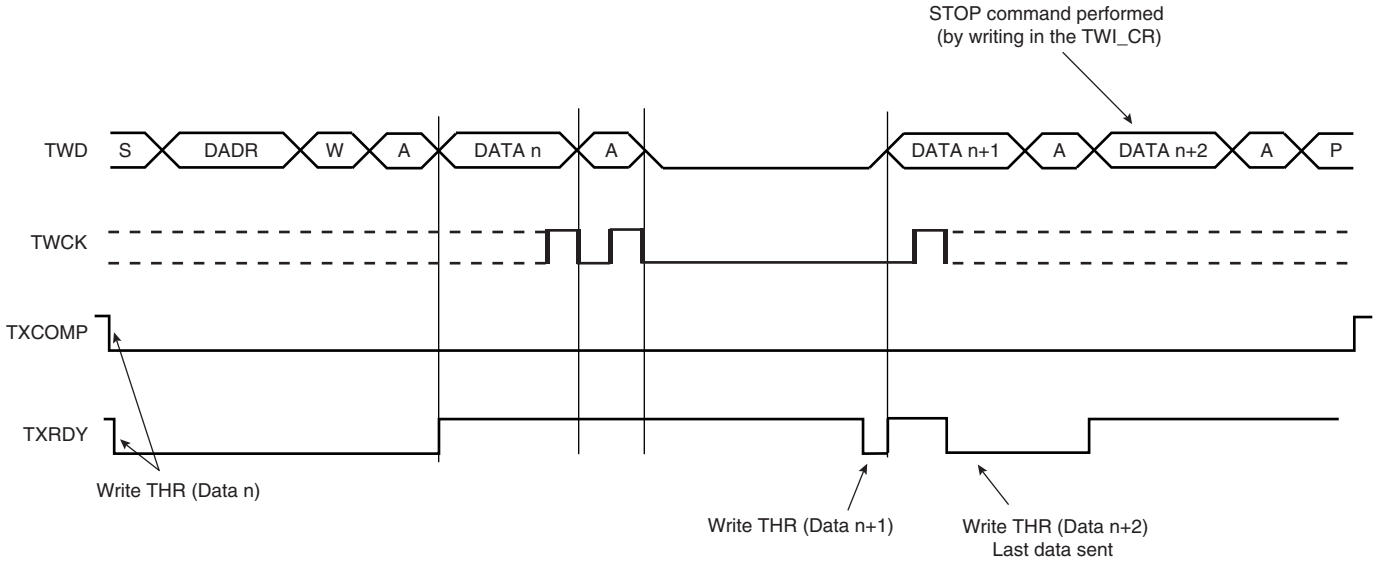


Figure 31-7. Master Write with Multiple Data Bytes



31.9.5.4 Clock Synchronization

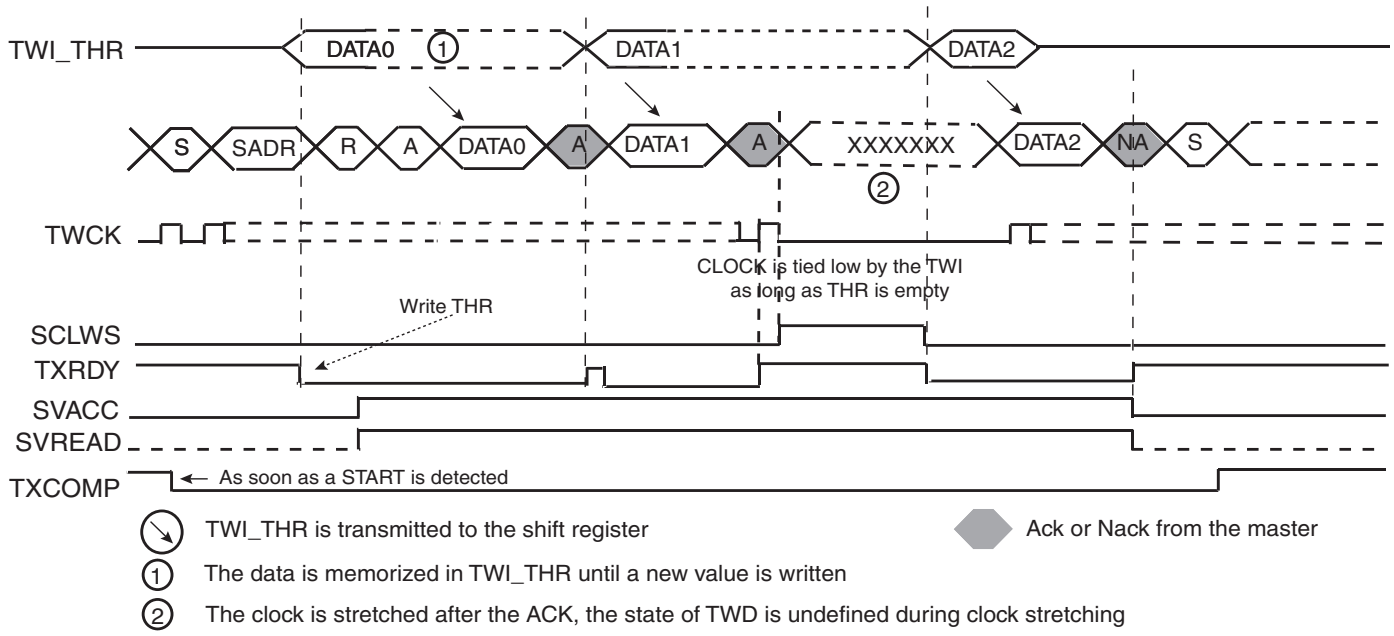
In both read and write modes, it may happen that TWI_THR/TWI_RHR buffer is not filled /emptied before the emission/reception of a new character. In this case, to avoid sending/receiving undesired data, a clock stretching mechanism is implemented.

31.9.5.5 Clock Synchronization in Read Mode

The clock is tied low if the shift register is empty and if a STOP or REPEATED START condition was not detected. It is tied low until the shift register is loaded.

Figure 31-28 on page 406 describes the clock synchronization in Read mode.

Figure 31-28. Clock Synchronization in Read Mode



Notes:
1. TXRDY is reset when data has been written in the TWI_THR to the shift register and set when this data has been acknowledged or non acknowledged.
2. At the end of the read sequence, TXCOMP is set after a STOP or after a REPEATED_START + an address different from SADR.
3. SCLWS is automatically set when the clock synchronization mechanism is started.

32.7.11 USART Transmitter Timeguard Register

Name: US_TTGR

Addresses: 0xFFFB0028 (0), 0xFFFB4028 (1), 0xFFFB8028 (2)

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
TG							

- **TG: Timeguard Value**

0: The Transmitter Timeguard is disabled.

1 - 255: The Transmitter timeguard is enabled and the timeguard delay is TG x Bit Period.

33.4 Pin Name List

Table 33-1. I/O Lines Description

Pin Name	Pin Description	Type
RF	Receiver Frame Synchro	Input/Output
RK	Receiver Clock	Input/Output
RD	Receiver Data	Input
TF	Transmitter Frame Synchro	Input/Output
TK	Transmitter Clock	Input/Output
TD	Transmitter Data	Output

33.5 Product Dependencies

33.5.1 I/O Lines

The pins used for interfacing the compliant external devices may be multiplexed with PIO lines.

Before using the SSC receiver, the PIO controller must be configured to dedicate the SSC receiver I/O lines to the SSC peripheral mode.

Before using the SSC transmitter, the PIO controller must be configured to dedicate the SSC transmitter I/O lines to the SSC peripheral mode.

Table 33-2. I/O Lines

Instance	Signal	I/O Line	Peripheral
SSC0	RD0	PB24	A
SSC0	RF0	PB26	A
SSC0	RK0	PB25	A
SSC0	TD0	PB23	A
SSC0	TF0	PB21	A
SSC0	TK0	PB22	A
SSC1	RD1	PA20	B
SSC1	RF1	PA22	B
SSC1	RK1	PA21	B
SSC1	TD1	PA19	B
SSC1	TF1	PA17	B
SSC1	TK1	PA18	B
SSC2	RD2	PC28	B
SSC2	RF2	PC30	B
SSC2	RK2	PC29	B
SSC2	TD2	PC27	B
SSC2	TF2	PC25	B
SSC2	TK2	PC26	B

34.6 Timer Counter (TC) User Interface

Table 34-5. Register Mapping

Offset[1]	Register	Name	Access	Reset
0x00 + channel * 0x40 + 0x00	Channel Control Register	TC_CCR	Write-only	–
0x00 + channel * 0x40 + 0x04	Channel Mode Register	TC_CMR	Read-write	0
0x00 + channel * 0x40 + 0x08	Reserved			
0x00 + channel * 0x40 + 0x0C	Reserved			
0x00 + channel * 0x40 + 0x10	Counter Value	TC_CV	Read-only	0
0x00 + channel * 0x40 + 0x14	Register A	TC_RA	Read-write[2]	0
0x00 + channel * 0x40 + 0x18	Register B	TC_RB	Read-write[2]	0
0x00 + channel * 0x40 + 0x1C	Register C	TC_RC	Read-write	0
0x00 + channel * 0x40 + 0x20	Status Register	TC_SR	Read-only	0
0x00 + channel * 0x40 + 0x24	Interrupt Enable Register	TC_IER	Write-only	–
0x00 + channel * 0x40 + 0x28	Interrupt Disable Register	TC_IDR	Write-only	–
0x00 + channel * 0x40 + 0x2C	Interrupt Mask Register	TC_IMR	Read-only	0
0xC0	Block Control Register	TC_BCR	Write-only	–
0xC4	Block Mode Register	TC_BMR	Read-write	0
0xD8	Reserved			
0xE4	Reserved			
0xFC	Reserved	–	–	–

Notes: 1. Channel index ranges from 0 to 2.

2. Read-only if WAVE = 0

36. USB Host Port (UHP)

36.1 Description

The USB Host Port (UHP) interfaces the USB with the host application. It handles Open HCI protocol (Open Host Controller Interface) as well as USB v2.0 Full-speed and Low-speed protocols.

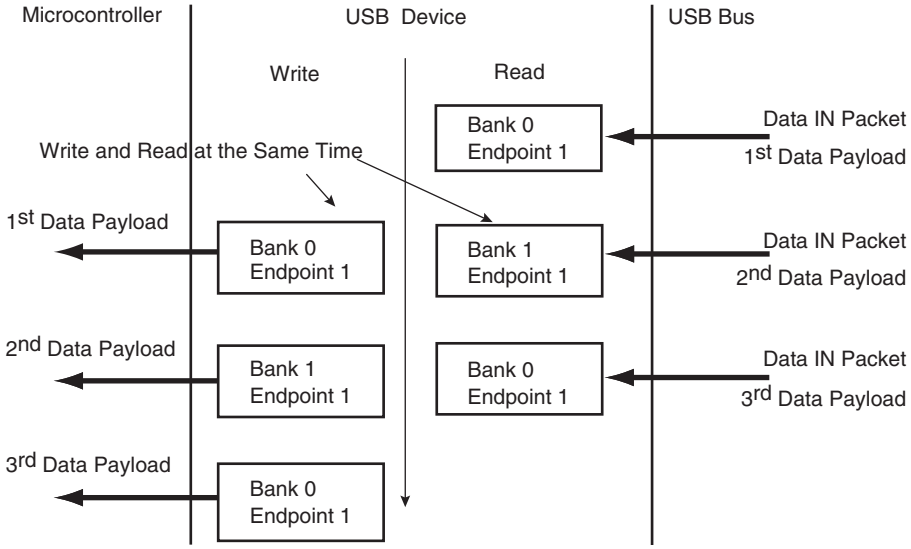
The USB Host Port integrates a root hub and transceivers on downstream ports. It provides several high-speed half-duplex serial communication ports at a baud rate of 12 Mbit/s. Up to 127 USB devices (printer, camera, mouse, keyboard, disk, etc.) and the USB hub can be connected to the USB host in the USB “tiered star” topology.

The USB Host Port controller is fully compliant with the OpenHCI specification. The USB Host Port User Interface (registers description) can be found in the Open HCI Rev 1.0 Specification available on http://h18000.www1.hp.com/productinfo/development/openhci.html. The standard OHCI USB stack driver can be easily ported to ATMEL’s architecture in the same way all existing class drivers run without hardware specialization.

This means that all standard class devices are automatically detected and available to the user application. As an example, integrating an HID (Human Interface Device) class driver provides a plug & play feature for all USB keyboards and mouses.

load sent by the host, while the current data payload is received by the USB device. Thus two banks of memory are used. While one is available for the microcontroller, the other one is locked by the USB device.

Figure 37-10. Bank Swapping in Data OUT Transfers for Ping-pong Endpoints



When using a ping-pong endpoint, the following procedures are required to perform Data OUT transactions:

1. The host generates a Data OUT packet.
2. This packet is received by the USB device endpoint. It is written in the endpoint's FIFO Bank 0.
3. The USB device sends an ACK PID packet to the host. The host can immediately send a second Data OUT packet. It is accepted by the device and copied to FIFO Bank 1.
4. The microcontroller is notified that the USB device has received a data payload, polling RX_DATA_BK0 in the endpoint's UDP_CSRx register. An interrupt is pending for this endpoint while RX_DATA_BK0 is set.
5. The number of bytes available in the FIFO is made available by reading RXBYTECNT in the endpoint's UDP_CSRx register.
6. The microcontroller transfers out data received from the endpoint's memory to the microcontroller's memory. Data received is made available by reading the endpoint's UDP_FDRx register.
7. The microcontroller notifies the USB peripheral device that it has finished the transfer by clearing RX_DATA_BK0 in the endpoint's UDP_CSRx register.
8. A third Data OUT packet can be accepted by the USB peripheral device and copied in the FIFO Bank 0.
9. If a second Data OUT packet has been received, the microcontroller is notified by the flag RX_DATA_BK1 set in the endpoint's UDP_CSRx register. An interrupt is pending for this endpoint while RX_DATA_BK1 is set.
10. The microcontroller transfers out data received from the endpoint's memory to the microcontroller's memory. Data received is available by reading the endpoint's UDP_FDRx register.

Table 38-2. I/O Lines (Continued)

LCDC	LCDD9	PB14	A
LCDC	LCDD10	PB10	B
LCDC	LCDD10	PB15	A
LCDC	LCDD11	PB11	B
LCDC	LCDD11	PB16	A
LCDC	LCDD12	PB12	B
LCDC	LCDD12	PB17	A
LCDC	LCDD13	PB13	B
LCDC	LCDD13	PB18	A
LCDC	LCDD14	PB14	B
LCDC	LCDD14	PB19	A
LCDC	LCDD15	PB15	B
LCDC	LCDD15	PB20	A
LCDC	LCDD16	PB21	B
LCDC	LCDD17	PB22	B
LCDC	LCDD18	PB23	B
LCDC	LCDD19	PB16	B
LCDC	LCDD19	PB24	B
LCDC	LCDD20	PB17	B
LCDC	LCDD20	PB25	B
LCDC	LCDD21	PB18	B
LCDC	LCDD21	PB26	B
LCDC	LCDD22	PB19	B
LCDC	LCDD22	PB27	B
LCDC	LCDD23	PB20	B
LCDC	LCDD23	PB28	B
LCDC	LCDHSYNC	PB1	A
LCDC	LCDVSYNC	PB0	A

38.5.2 Power Management

The LCD Controller is not continuously clocked. The user must first enable the LCD Controller clock in the Power Management Controller before using it (PMC_PCER).

38.11.13 LCD Frame Configuration Register

Name: LCDFRMCFG

Address: 0x00600810

Access: Read-write

Reset: 0x0000000

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

- **LINEVAL: Vertical size of LCD module**

LINEVAL = (Vertical display size) - 1

In dual scan mode, vertical display size refers to the size of each panel.

- **HOZVAL: Horizontal size of LCD module**

In STN Mode:

– HOZVAL = (Horizontal display size / Number of valid LCDD data line) - 1

– In STN monochrome mode, Horizontal display size = Number of horizontal pixels

– In STN color mode, Horizontal display size = 3*Number of horizontal pixels

– In 4-bit single scan or 8-bit dual scan STN display mode, number of valid LCDD data lines = 4

– In 8-bit single scan or 16-bit dual scan STN display mode, number of valid LCDD data lines = 8

– If the value calculated for HOZVAL with the above formula is not an integer, it must be rounded up to the next integer value.

In TFT mode:

– HOZVAL = Horizontal display size

38.11.26 Contrast Value Register

Name: CONSTRAST_VAL

Access: Read-write

Reset: 0x00000000

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
CVAL							

- **CVAL**

PWM compare value. Used to adjust the analog value obtained after an external filter to control the contrast of the display.