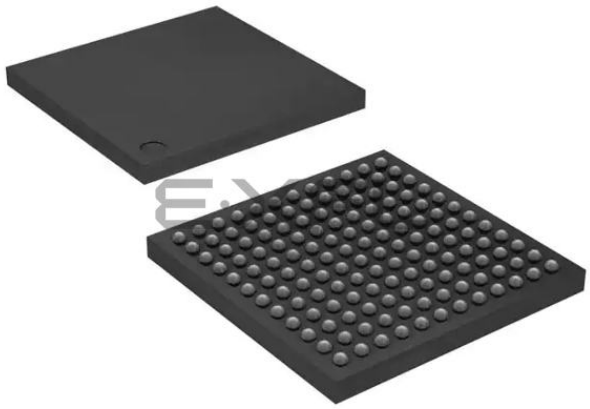


Welcome to E-XFL.COM

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	ARM7®
Core Size	16/32-Bit
Speed	33MHz
Connectivity	EBI/EMI, SPI, UART/USART
Peripherals	POR, WDT
Number of I/O	54
Program Memory Size	-
Program Memory Type	ROMless
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	-
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	144-LFBGA
Supplier Device Package	144-BGA (13x13)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/at91m42800a-33cj-999

1. Description

The AT91M42800A is a member of the Atmel AT91 16/32-bit microcontroller family, which is based on the ARM7TDMI processor core. This processor has a high-performance 32-bit RISC architecture with a high-density 16-bit instruction set and very low power consumption. In addition, a large number of internally banked registers result in very fast exception handling, making the device ideal for real-time control applications. The AT91 ARM-based MCU family also features Atmel's high-density, in-system programmable, nonvolatile memory technology. The AT91M42800A has a direct connection to off-chip memory, including Flash, through the External Bus Interface.

The Power Management Controller allows the user to adjust device activity according to system requirements, and, with the 32.768 kHz low-power oscillator, enables the AT91M42800A to reduce power requirements to an absolute minimum. The AT91M42800A is manufactured using Atmel's high-density CMOS technology. By combining the ARM7TDMI processor core with on-chip SRAM and a wide range of peripheral functions including timers, serial communication controllers and a versatile clock generator on a monolithic chip, the AT91M42800A provides a highly flexible and cost-effective solution to many compute-intensive applications.

Standard RS232 drivers generally contain internal 400 kΩ pull-up resistors. If TXD1 is connected to one of these drivers, this pull-up will ensure normal operation, without the need for an additional external resistor.

7.6.2 Embedded ICE

ARM standard embedded in-circuit emulation is supported via the JTAG/ICE port. It is connected to a host computer via an embedded ICE Interface.

Embedded ICE mode is selected when MODE1 is low.

It is not possible to switch directly between ICE and JTAG operations. A chip reset must be performed (NRST and NTRST) after MODE0 and/or MODE1 have/has been changed. The reset input to the embedded ICE (NTRST) is provided separately to facilitate debug of boot programs.

7.6.3 IEEE 1149.1 JTAG Boundary Scan

IEEE 1149.1 JTAG Boundary Scan is enabled when MODE0 is low and MODE1 is high. The functions SAMPLE, EXTEST and BYPASS are implemented. In ICE Debug mode, the ARM core responds with a non-JTAG chip ID that identifies the core to the ICE system. This is not IEEE 1149.1 JTAG compliant. It is not possible to switch directly between JTAG and ICE operations. A chip reset must be performed (NRST and NTRST) after MODE0 and/or MODE1 have/has been changed.

7.7 Memory Controller

The ARM7TDMI processor address space is 4G bytes. The memory controller decodes the internal 32-bit address bus and defines three address spaces:

- Internal Memories in the four lowest megabytes
- Middle Space reserved for the external devices (memory or peripherals) controlled by the EBI
- Internal Peripherals in the four highest megabytes

In any of these address spaces, the ARM7TDMI operates in little-endian mode only.

7.7.1 Protection Mode

The embedded peripherals can be protected against unwanted access. The PME (Protect Mode Enable) pin must be tied high and validated in its peripheral operation (PIO Disable) to enable the protection mode. When enabled, any peripheral access must be done while the ARM7TDMI is running in Privileged mode (i.e., the accesses in user mode result in an abort). Only the valid peripheral address space is protected and requests to the undefined addresses will lead to a normal operation without abort.

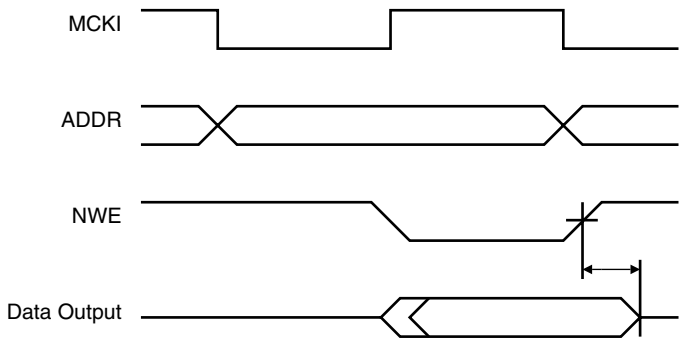
7.7.2 Internal Memories

The AT91M42800A microcontroller integrates an 8-Kbyte primary internal SRAM. All internal memories are 32 bits wide and single-clock cycle accessible. Byte (8-bit), half-word (16-bit) or word (32-bit) accesses are supported and are executed within one cycle. Fetching Thumb or ARM instructions is supported and internal memory can store twice as many Thumb instructions as ARM ones.

The SRAM bank is mapped at address 0x0 (after the remap command), and ARM7TDMI exception vectors between 0x0 and 0x20 that can be modified by the software. The rest of the

It is therefore necessary to avoid excessive loading of the NWE pins, which could delay the write signal too long and cause a contention with a subsequent read cycle in standard protocol.

Figure 11-12. Data Hold Time



In early read protocol the data can remain valid longer than in standard read protocol due to the additional wait cycle which follows a write access.

11.11 Wait States

The EBI can automatically insert wait states. The different types of wait states are listed below:

- Standard wait states
- Data float wait states
- External wait states
- Chip select change wait states
- Early Read wait states (as described in ”Read Protocols” on page 29)

11.11.1 Standard Wait States

Each chip select can be programmed to insert one or more wait states during an access on the corresponding device. This is done by setting the WSE field in the corresponding EBI_CSR. The number of cycles to insert is programmed in the NWS field in the same register.

Below is the correspondence between the number of standard wait states programmed and the number of cycles during which the NWE pulse is held low:

0 wait states	1/2 cycle
1 wait state	1 cycle

For each additional wait state programmed, an additional cycle is added.

12.13 PMC Clock Generator Mode Register

Register Name: PMC_CGMR
Access Type: Read/Write
Offset: 0x20
Reset Value: 0x0

31	30	29	28	27	26	25	24
PLLCOUNT							

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

15	14	13	12	11	10	9	8
MUL							

7	6	5	4	3	2	1	0
CSS	MCKODS	MCKOSS		PLLS	PRES		

- **PRES: Prescaler Selection**

PRES			Prescaler Selected	Code Label `PMC_PRES`
0	0	0	None. The Prescaler is bypassed.	`PMC_PRES_NONE`
0	0	1	Divide by 2	`PMC_PRES_DIV2`
0	1	0	Divide by 4	`PMC_PRES_DIV4`
0	1	1	Divide by 8	`PMC_PRES_DIV8`
1	0	0	Divide by 16	`PMC_PRES_DIV16`
1	0	1	Divide by 32	`PMC_PRES_DIV32`
1	1	0	Divide by 64	`PMC_PRES_DIV64`
1	1	1	Reserved	–

- **PLLS: PLL Selection**

0 = The PLL A with 5 - 20 MHz output range is selected as PLL source. (Code Label `PMC_PLL_A`)

- **1 = The PLL B with 20 - 80 MHz output range is selected as PLL source. (Code Label `PMC_PLL_B`)**

Note: This bit can be written only once after the reset. Any write of a different value than this one written the first time has no effect on the bit.

- **MCKOSS: Master Clock Output Source Selection**

MCKOSS		Master Clock Output Source Select	Code Label: `PMC_MCKOSS`
0	0	Slow Clock	`PMC_MCKOSS_SLCK`
0	1	Master Clock	`PMC_MCKOSS_MCK`
1	0	Master Clock inverted	`PMC_MCKOSS_MCKINV`
1	1	Master Clock divided by 2	`PMC_MCKOSS_MCK_DIV2`

- **MCKODS: Master Clock Output Disable (Code Label `PMC_MCKO_DIS`)**

0 = The pin MCKO is driven with the clock selected by MCKOSS.

1 = The pin MCKO is tri-stated.

12.17 PMC Interrupt Mask Register

Register Name: PMC_IMR
Access Type: Read-only
Offset: 0x3C
Reset Value: 0x0

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
–	–	–	–	–	–	–	LOCK

- **LOCK: PLL Lock Interrupt Mask**

0 = The PLL Lock Interrupt is disabled.

1 = The PLL Lock Interrupt is enabled.

loads the program counter with the interrupt handler address stored in the AIC_IVR register. Execution is then vectored to the interrupt handler corresponding to the current interrupt.

```
ldr PC,[PC,# -&F20]
```

The current interrupt is the interrupt with the highest priority when the Interrupt Vector Register (AIC_IVR) is read. The value read in the AIC_IVR corresponds to the address stored in the Source Vector Register (AIC_SVR) of the current interrupt. Each interrupt source has its corresponding AIC_SVR. In order to take advantage of the hardware interrupt vectoring it is necessary to store the address of each interrupt handler in the corresponding AIC_SVR, at system initialization.

14.2 Priority Controller

The NIRQ line is controlled by an 8-level priority encoder. Each source has a programmable priority level of 7 to 0. Level 7 is the highest priority and level 0 the lowest.

When the AIC receives more than one unmasked interrupt at a time, the interrupt with the highest priority is serviced first. If both interrupts have equal priority, the interrupt with the lowest interrupt source number (see Table 14-1) is serviced first.

The current priority level is defined as the priority level of the current interrupt at the time the register AIC_IVR is read (the interrupt which will be serviced).

In the case when a higher priority unmasked interrupt occurs while an interrupt already exists, there are two possible outcomes depending on whether the AIC_IVR has been read.

- If the NIRQ line has been asserted but the AIC_IVR has not been read, then the processor will read the new higher priority interrupt handler address in the AIC_IVR register and the current interrupt level is updated.
- If the processor has already read the AIC_IVR then the NIRQ line is reasserted. When the processor has authorized nested interrupts to occur and reads the AIC_IVR again, it reads the new, higher priority interrupt handler address. At the same time the current priority value is pushed onto a first-in last-out stack and the current priority is updated to the higher priority.

When the end of interrupt command register (AIC_EOICR) is written, the current interrupt level is updated with the last stored interrupt level from the stack (if any). Hence at the end of a higher priority interrupt, the AIC returns to the previous state corresponding to the preceding lower priority interrupt which had been interrupted.

14.3 Interrupt Handling

The interrupt handler must read the AIC_IVR as soon as possible. This de-asserts the NIRQ request to the processor and clears the interrupt in case it is programmed to be edge triggered. This permits the AIC to assert the NIRQ line again when a higher priority unmasked interrupt occurs.

At the end of the interrupt service routine, the end of interrupt command register (AIC_EOICR) must be written. This allows pending interrupts to be serviced.

14.4 Interrupt Masking

Each interrupt source, including FIQ, can be enabled or disabled using the command registers AIC_IECR and AIC_IDCR. The interrupt mask can be read in the Read-only register AIC_IMR. A disabled interrupt does not affect the servicing of other interrupts.

14.11 AIC Source Mode Register

Register Name: AIC_SMR0..AIC_SMR31
Access Type: Read/Write
Reset Value: 0

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
–	SRCTYPE		–	–	PRIOR		

- **PRIOR: Priority Level (Code Label `AIC_PRIOR`)**

Program the priority level for all sources except source 0 (FIQ).
The priority level can be between 0 (lowest) and 7 (highest).
The priority level is not used for the FIQ, in the SMR0.

- **SRCTYPE: Interrupt Source Type (Code Label `AIC_SRCTYPE`)**

Program the input to be positive or negative edge-triggered or positive or negative level sensitive.
The active level or edge is not programmable for the internal sources.

SRCTYPE		External Sources	Code Label
0	0	Low-level Sensitive	AIC_SRCTYPE_EXT_LOW_LEVEL
0	1	Negative Edge triggered	AIC_SRCTYPE_EXT_NEGATIVE_EDGE
1	0	High-level Sensitive	AIC_SRCTYPE_EXT_HIGH_LEVEL
1	1	Positive Edge triggered	AIC_SRCTYPE_EXT_POSITIVE_EDGE

SRCTYPE		Internal Sources	Code Label
X	0	Level Sensitive	AIC_SRCTYPE_INT_LEVEL_SENSITIVE
X	1	Edge triggered	AIC_SRCTYPE_INT_EDGE_TRIGGERED

14.19 AIC Interrupt Enable Command Register

Register Name: AIC_IECR
Access Type: Write-only
Offset: 0x120

31	30	29	28	27	26	25	24
IRQ0	IRQ1	IRQ2	IRQ3	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–
15	14	13	12	11	10	9	8
PMC	PIOB	PIOA	ST	TC5	TC4	TC3	TC2
7	6	5	4	3	2	1	0
TC1	TC0	SPIB	SPIA	US1	US0	SW	FIQ

- **Interrupt Enable**

0 = No effect.

1 = Enables corresponding interrupt.

14.20 AIC Interrupt Disable Command Register

Register Name: AIC_IDCR
Access Type: Write-only
Offset: 0x124

31	30	29	28	27	26	25	24
IRQ0	IRQ1	IRQ2	IRQ3	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–
15	14	13	12	11	10	9	8
PMC	PIOB	PIOA	ST	TC5	TC4	TC3	TC2
7	6	5	4	3	2	1	0
TC1	TC0	SPIB	SPIA	US1	US0	SW	FIQ

- **Interrupt Disable**

0 = No effect.

1 = Disables corresponding interrupt.

15.8 PIO Connection Tables

Table 15-1. PIO Controller A Connection Table

PIO Controller		Peripheral					
Bit Number	Port Name	Port Name	Signal Description	Signal Direction	OFF[1] Value	Reset State	Pin Number
0	PA0	IRQ0	External Interrupt 0	Input	0	PIO Input	77
1	PA1	IRQ1	External Interrupt 1	Input	0	PIO Input	78
2	PA2	IRQ2	External Interrupt 2	Input	0	PIO Input	79
3	PA3	IRQ3	External Interrupt 3	Input	0	PIO Input	80
4	PA4	FIQ	Fast Interrupt	Input	0	PIO Input	81
5	PA5	SCK0	USART0 Clock Signal	Bi-directional	0	PIO Input	82
6	PA6	TXD0	USART0 Transmit Data Signal	Output	–	PIO Input	83
7	PA7	RXD0	USART0 Receive Data Signal	Input	0	PIO Input	86
8	PA8	SCK1	USART1 Clock Signal	Bi-directional	0	PIO Input	87
9	PA9	TXD1/NTRI	USART1 Transmit Data Signal	Output	–	PIO Input	88
10	PA10	RXD1	USART1 Receive Data Signal	Input	0	PIO Input	89
11	PA11	SPCKA	SPIA Clock Signal	Bi-directional	0	PIO Input	90
12	PA12	MISOA	SPIA Master In Slave Out	Bi-directional	0	PIO Input	91
13	PA13	MOSIA	SPIA Master Out Slave In	Bi-directional	0	PIO Input	92
14	PA14	NPCSA0/NSSA	SPIA Peripheral Chip Select 0	Bi-directional	1	PIO Input	93
15	PA15	NPCSA1	SPIA Peripheral Chip Select 1	Output	–	PIO Input	94
16	PA16	NPCSA2	SPIA Peripheral Chip Select 2	Output	–	PIO Input	95
17	PA17	NPCSA3	SPIA Peripheral Chip Select 3	Output	–	PIO Input	98
18	PA18	SPCKB	SPIB Clock Signal	Bi-directional	0	PIO Input	99
19	PA19	MISOB	SPIB Master In Slave Out	Bi-directional	0	PIO Input	100
20	PA20	MOSIB	SPIB Master Out Slave In	Bi-directional	0	PIO Input	101
21	PA21	NPCSB0/NSSB	SPIB Peripheral Chip Select 0	Bi-directional	1	PIO Input	102
22	PA22	NPCSB1	SPIB Peripheral Chip Select 1	Output	–	PIO Input	103
23	PA23	NPCSB2	SPIB Peripheral Chip Select 2	Output	–	PIO Input	104
24	PA24	NPCSB3	SPIB Peripheral Chip Select 3	Output	–	PIO Input	105
25	PA25	MCKO	Master Clock Output	Output	–	MCKO	106
26	PA26	–	–	–	–	PIO Input	111
27	PA27	BMS	Boot Mode Select	Input	0	PIO Input	123
28	PA28	–	–	Output	–	PIO Input	131
29	PA29	PME	Protect Mode Enable	Input	0	PIO Input	134

Note: 1. The OFF value is the default level seen on the peripheral input when the PIO line is enabled.

Table 15-2. PIO Controller B Connection Table

PIO Controller		Peripheral					
Bit Number	Port Name	Port Name	Signal Description	Signal Direction	OFF[1] Value	Reset State	Pin Number
0	PB0	NCS2	Chip Select 2	Output	–	NCS2	141
1	PB1	NCS3	Chip Select 3	Output	–	NCS3	142
2	PB2	A20/CS7	Address 20/Chip Select 7	Output	–	A20	27
3	PB3	A21/CS6	Address 21/Chip Select 6	Output	–	A21	28
4	PB4	A22/CS5	Address 22/Chip Select 5	Output	–	A22	29
5	PB5	A23/CS4	Address 23/Chip Select 4	Output	–	A23	30
6	PB6	TCLK0	Timer0 Clock Signal	Input	0	PIO Input	53
7	PB7	TIOA0	Timer0 Signal A	Bi-directional	0	PIO Input	54
8	PB8	TIOB0	Timer0 Signal B	Bi-directional	0	PIO Input	55
9	PB9	TCLK1	Timer1 Clock Signal	Input	0	PIO Input	56
10	PB10	TIOA1	Timer1 Signal A	Bi-directional	0	PIO Input	57
11	PB11	TIOB1	Timer1 Signal B	Bi-directional	0	PIO Input	58
12	PB12	TCLK2	Timer2 Clock Signal	Input	0	PIO Input	59
13	PB13	TIOA2	Timer2 Signal A	Bi-directional	0	PIO Input	62
14	PB14	TIOB2	Timer2 Signal B	Bi-directional	0	PIO Input	63
15	PB15	TCLK3	Timer3 Clock Signal	Input	0	PIO Input	64
16	PB16	TIOA3	Timer3 Signal A	Bi-directional	0	PIO Input	65
17	PB17	TIOB3	Timer3 Signal B	Bi-directional	0	PIO Input	66
18	PB18	TCLK4	Timer4 Clock Signal	Input	0	PIO Input	67
19	PB19	TIOA4	Timer4 Signal A	Bi-directional	0	PIO Input	68
20	PB20	TIOB4	Timer4 Signal B	Bi-directional	0	PIO Input	69
21	PB21	TCLK5	Timer5 Clock Signal	Input	0	PIO Input	70
22	PB22	TIOA5	Timer5 Signal A	Bi-directional	0	PIO Input	75
23	PB23	TIOB5	Timer5 Signal B	Bi-directional	0	PIO Input	76

Note: 1. The OFF value is the default level seen on the peripheral input when the PIO line is enabled.

17.1 Pin Description

Each USART channel has the following external signals:

Name	Description
SCK	USART Serial clock can be configured as input or output: SCK is configured as input if an External clock is selected (USCLKS = 3) SCK is driven as output if the External Clock is disabled (USCLKS ≠ 3) and Clock output is enabled (CLKO = 1)
TXD	Transmit Serial Data is an output
RXD	Receive Serial Data is an input

Notes:
1. After a hardware reset, the USART clock is disabled by default (see "PMC: Power Management Controller" on page 55). The user must configure the Power Management Controller before any access to the User Interface of the USART.
2. After a hardware reset, the USART pins are deselected by default (see "PIO: Parallel I/O Controller" on page 97). The user must configure the PIO Controller before enabling the transmitter or receiver. If the user selects one of the internal clocks, SCK can be configured as a PIO.

17.2 Baud Rate Generator

The Baud Rate Generator provides the bit period clock (the Baud Rate clock) to both the Receiver and the Transmitter.

The Baud Rate Generator can select between external and internal clock sources. The external clock source is SCK. The internal clock sources can be either the master clock MCK or the master clock divided by 8 (MCK/8).

Note: In all cases, if an external clock is used, the duration of each of its levels must be longer than the system clock (MCK) period. The external clock frequency must be at least 2.5 times lower than the system clock.

When the USART is programmed to operate in Asynchronous Mode (SYNC = 0 in the Mode Register US_MR), the selected clock is divided by 16 times the value (CD) written in US_BRGR (Baud Rate Generator Register). If US_BRGR is set to 0, the Baud Rate Clock is disabled.

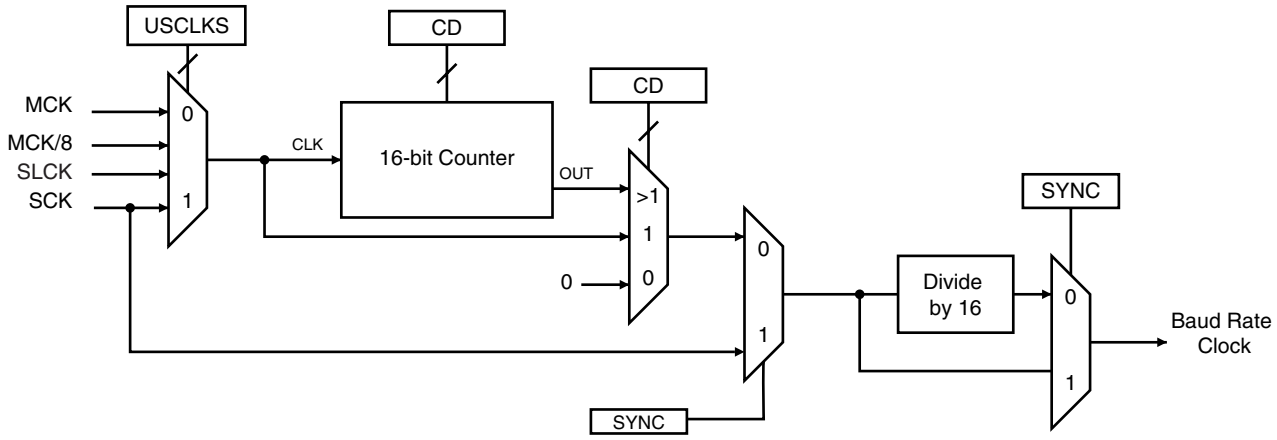
$$\text{Baud Rate} = \frac{\text{Selected Clock}}{16 \times \text{CD}}$$

When the USART is programmed to operate in Synchronous Mode (SYNC = 1) and the selected clock is internal (USCLKS ≠ 3 in the Mode Register US_MR), the Baud Rate Clock is the internal selected clock divided by the value written in US_BRGR. If US_BRGR is set to 0, the Baud Rate Clock is disabled.

$$\text{Baud Rate} = \frac{\text{Selected Clock}}{\text{CD}}$$

In Synchronous Mode with external clock selected (USCLKS = 3), the clock is provided directly by the signal on the SCK pin. No division is active. The value written in US_BRGR has no effect.

Figure 17-2. Baud Rate Generator



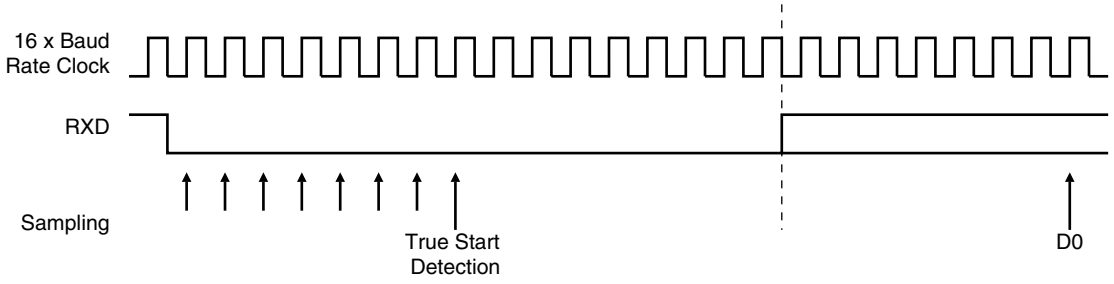
17.3 Receiver

17.3.1 Asynchronous Receiver

The USART is configured for asynchronous operation when SYNC = 0 (bit 7 of US_MR). In asynchronous mode, the USART detects the start of a received character by sampling the RXD signal until it detects a valid start bit. A low level (space) on RXD is interpreted as a valid start bit if it is detected for more than 7 cycles of the sampling clock, which is 16 times the baud rate. Hence a space which is longer than 7/16 of the bit period is detected as a valid start bit. A space which is 7/16 of a bit period or shorter is ignored and the receiver continues to wait for a valid start bit.

When a valid start bit has been detected, the receiver samples the RXD at the theoretical mid-point of each bit. It is assumed that each bit lasts 16 cycles of the sampling clock (one bit period) so the sampling point is 8 cycles (0.5 bit periods) after the start of the bit. The first sampling point is therefore 24 cycles (1.5 bit periods) after the falling edge of the start bit was detected. Each subsequent bit is sampled 16 cycles (1 bit period) after the previous one.

Figure 17-3. Asynchronous Mode: Start Bit Detection



- **TIMEOUT: Receiver Time-out (Code Label `US_TIMEOUT`)**

0 = There has not been a time-out since the last “Start Time-out” command or the Time-out Register is 0.

1 = There has been a time-out since the last “Start Time-out” command.

- **TXEMPTY: Transmitter Empty (Code Label `US_TXEMPTY`)**

0 = There are characters in either US_THR or the Transmit Shift Register, or the transmitter is disabled.

1 = There are no characters in either US_THR or the Transmit Shift Register. TXEMPTY is 1 after Parity, Stop Bit and Time-guard have been transmitted. TXEMPTY is 1 after stop bit has been sent, or after Time-guard has been sent if US_TTGR is not 0.

Equal to zero when the USART is disabled or at reset. Transmitter Enable command (in US_CR) sets this bit to one, if the transmitter is disabled.

- **COMMTX: ARM7TDMI ICE Debug Communication Channel Transmit Status**

For USART0 only. Refer to the ARM7TDMI Datasheet for a complete description of this flag.

- **COMMRX: ARM7TDMI ICE Debug Communication Channel Receive Status**

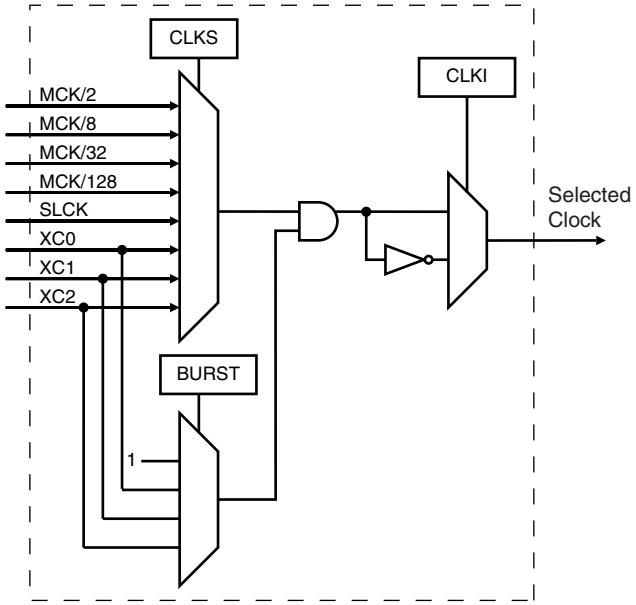
For USART0 only. Refer to the ARM7TDMI Datasheet for a complete description of this flag.

18.1 Signal Name Description[1, 2]

Channel Signals	Description
XC0, XC1, XC2	External Clock Inputs
TIOA	Capture Mode: General-purpose Input Waveform Mode: General-purpose Output
TIOB	Capture Mode: General-purpose Input Waveform Mode: General-purpose Input/Output
INT	Interrupt Signal Output
SYNC	Synchronization Input Signal
Block 0 Signals	**Description**
TCLK0, TCLK1, TCLK2	External Clock Inputs for Channels 0, 1, 2
TIOA0	TIOA Signal for Channel 0
TIOB0	TIOB Signal for Channel 0
TIOA1	TIOA Signal for Channel 1
TIOB1	TIOB Signal for Channel 1
TIOA2	TIOA Signal for Channel 2
TIOB2	TIOB Signal for Channel 2
Block 1 Signals	**Description**
TCLK3, TCLK4, TCLK5	External Clock Inputs for Channels 3, 4, 5
TIOA3	TIOA Signal for Channel 3
TIOB3	TIOB Signal for Channel 3
TIOA4	TIOA Signal for Channel 4
TIOB4	TIOB Signal for Channel 4
TIOA5	TIOA Signal for Channel 5
TIOB5	TIOB Signal for Channel 5

Notes:
1. After a hardware reset, the TC clock is disabled by default (see ”PMC: Power Management Controller” on page 55). The user must configure the Power Management Controller before any access to the User Interface of the TC.
2. After a hardware reset, the Timer/Counter block pins are controlled by the PIO Controller. They must be configured to be controlled by the peripheral before being used.

Figure 18-2. Clock Selection



18.2.3 Clock Control

The clock of each counter can be controlled in two different ways: it can be enabled/disabled and started/stopped.

- The clock can be **enabled** or **disabled** by the user with the CLKEN and the CLKDIS commands in the Control Register. In Capture Mode it can be disabled by an RB load event if LDBDIS is set to 1 in TC_CMR. In Waveform Mode, it can be disabled by an RC Compare event if CPCDIS is set to 1 in TC_CMR. When disabled, the start or the stop actions have no effect: only a CLKEN command in the Control Register can re-enable the clock. When the clock is enabled, the CLKSTA bit is set in the Status Register.
- The clock can also be **started** or **stopped**: a trigger (software, synchro, external or compare) always starts the clock. The clock can be stopped by an RB load event in Capture Mode (LDBSTOP = 1 in TC_CMR) or a RC compare event in Waveform Mode (CPCSTOP = 1 in TC_CMR). The start and the stop commands have effect only if the clock is enabled.

19.9 SPI Mode Register

Register Name: SP_MR

Access Type: Read/Write

Offset: 0x04

Reset Value: 0x0

31	30	29	28	27	26	25	24
DLYBCS							
23	22	21	20	19	18	17	16
–	–	–	–	PCS			
15	14	13	12	11	10	9	8
–	–	–	–	–	–	–	–
7	6	5	4	3	2	1	0
LLB	–	–	–	MCK32	PCSDEC	PS	MSTR

- **MSTR: Master/Slave Mode (Code Label `SP_MSTR`)**

0 = SPI is in Slave mode.

1 = SPI is in Master mode.

MSTR configures the SPI Interface for either master or slave mode operation.

- **PS: Peripheral Select**

PS	Selected PS	Code Label: `SP_PS`
0	Fixed Peripheral Select	SP_PS_FIXED
1	Variable Peripheral Select	SP_PS_VARIABLE

- **PCSDEC: Chip Select Decode (Code Label `SP_PCSDEC`)**

0 = The chip selects are directly connected to a peripheral device.

1 = The four chip select lines are connected to a 4- to 16-bit decoder.

When PCSDEC equals one, up to 16 Chip Select signals can be generated with the four lines using an external 4- to 16-bit decoder.

The Chip Select Registers define the characteristics of the 16 chip selects according to the following rules:

SP_CSR0 defines peripheral chip select signals 0 to 3.

SP_CSR1 defines peripheral chip select signals 4 to 7.

SP_CSR2 defines peripheral chip select signals 8 to 11.

SP_CSR3 defines peripheral chip select signals 12 to 15[(1)].

Note: 1. The 16th state corresponds to a state in which all chip selects are inactive. This allows a different clock configuration to be defined by each chip select register.

- **MCK32: Clock Selection (Code Label `SP_DIV32`)**

0 = SPI Master Clock equals MCK

1 = SPI Master Clock equals MCK/32

- **LLB: Local Loopback Enable (Code Label `SP_LLB`)**

0 = Local loopback path disabled

1 = Local loopback path enabled

LLB controls the local loopback on the data serializer for testing in master mode only.

- **PCS: Peripheral Chip Select (Code Label `SP_PCS`)**

21. Packaging Information

Figure 21-1. 144-lead LQFP Package Drawing

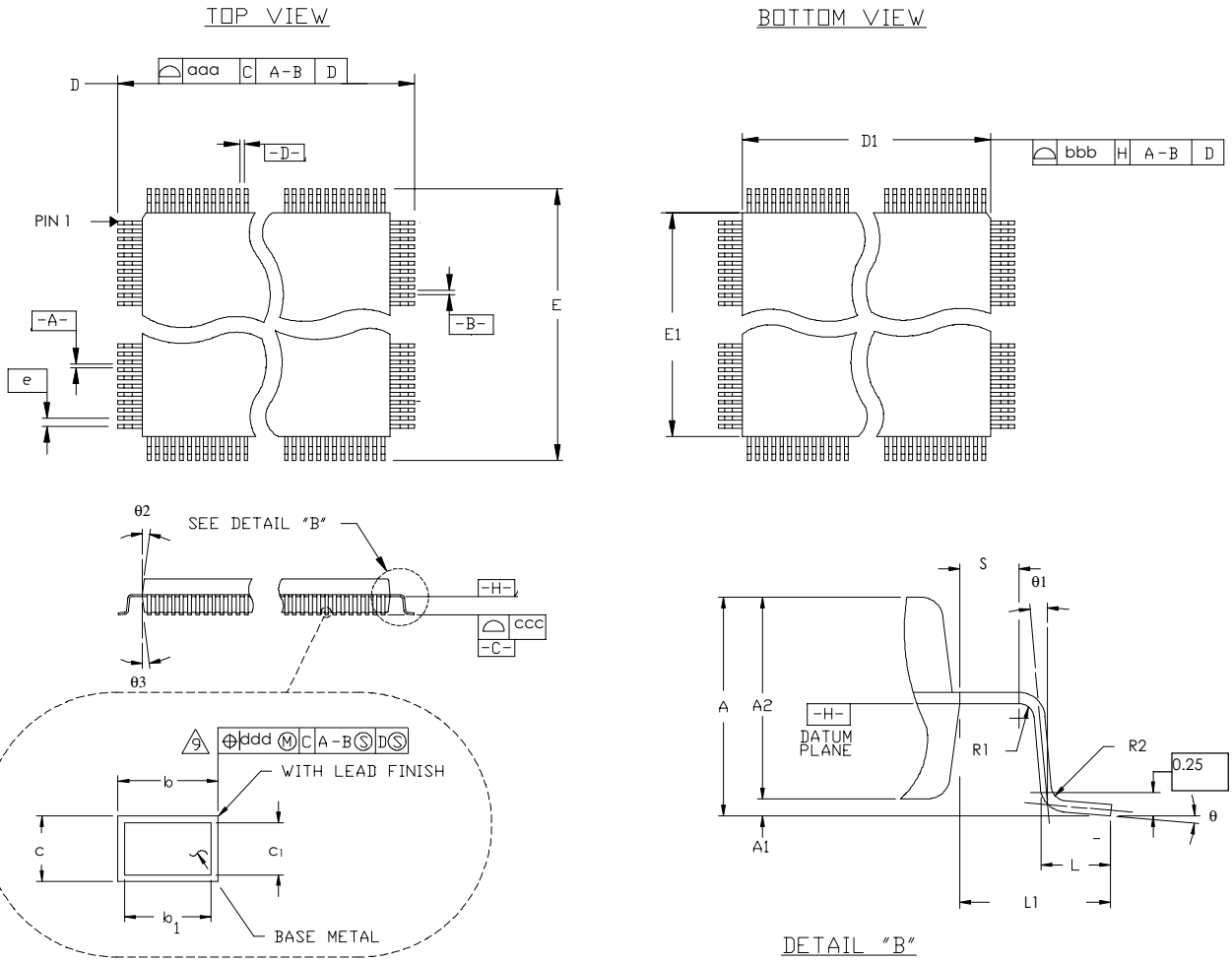
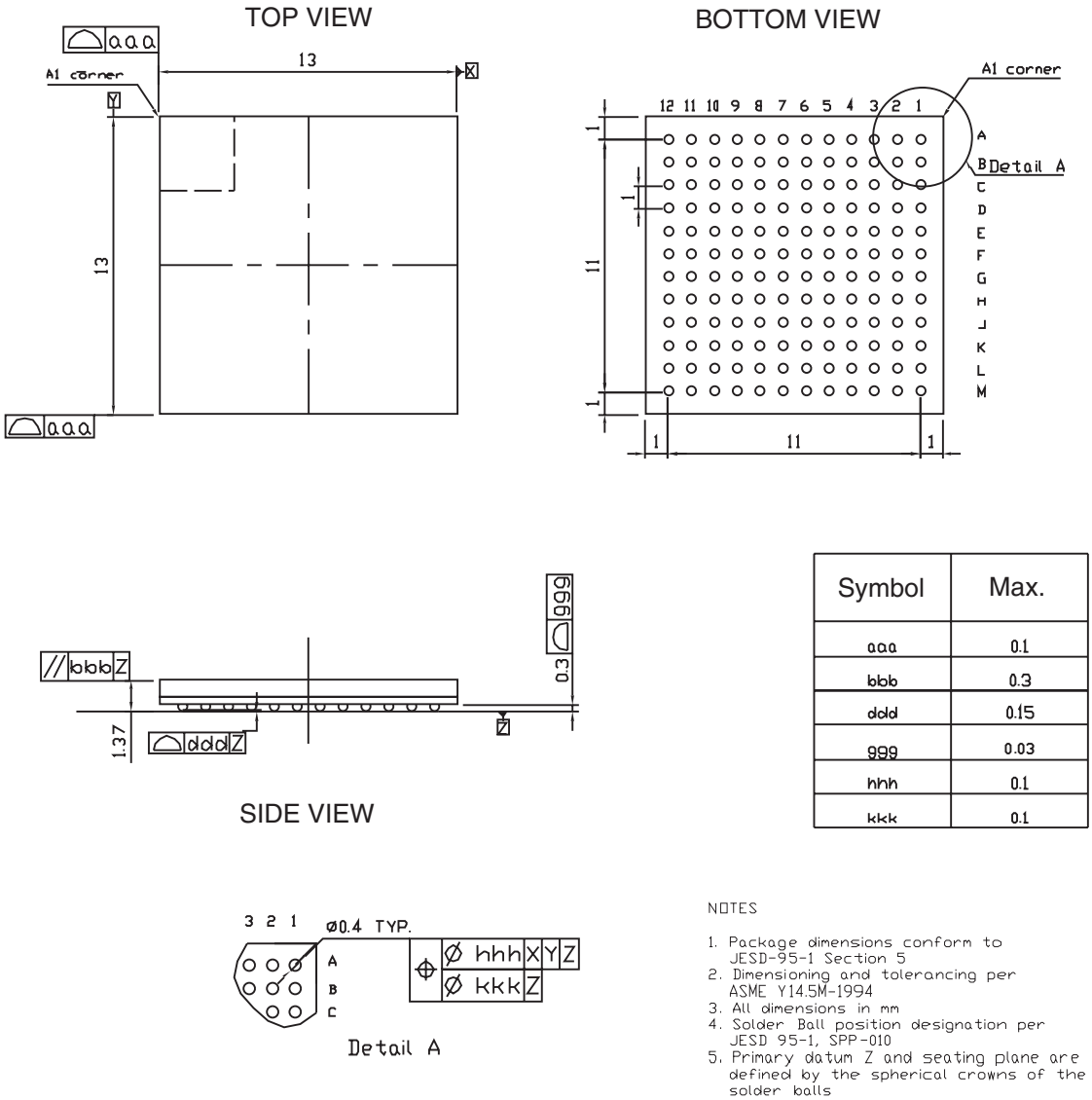


Figure 21-2. 144-ball Ball Grid Array Package Drawing



Symbol	Max.
aaa	0.1
bbb	0.3
ddd	0.15
eee	0.03
fff	0.1
ggg	0.1

Table 21-4. Device and 144-ball BGA Package Maximum Weight

584	mg

25. Revision History

Table 25-1. Revision History

1779A	First Issue. Publication Date: Oct-01	
1779B	Publication Date: 22-Mar-02	
	Change in Table 2 on page 4	
	Change in Table 3 on page 5	
	Change in Table 4 on page 10	
	Change in section Power SupplyChange in section Clock Generator on page 11	
	Added section Protection ModeChange in section Internal MemoriesDeleted section Protect Mode on page 13	
1779C	Added Section 7.5.2 ”NTRST Pin” on page 13.	05-473
	Changed number of wait states in Section 11.8 ”Boot on NCS0” on page 29. Change in reset state for EBI_CSR0 in Table 11-4 on page 48.	03-245
	Added Section 21. ”Packaging Information” on page 205, Section 22. ”Soldering Profile” on page 208 and Section 23. ”Ordering Information” on page 209.	
	Added Section 24. ”AT91M42800A Errata” on page 210 to replace Lit. No. 1782, AT91M42800A Errata Sheet.	
1779D	Updated Section 22. ”Soldering Profile” on page 208 and Section 23. ”Ordering Information” on page 209 to remove leaded packages.	2600