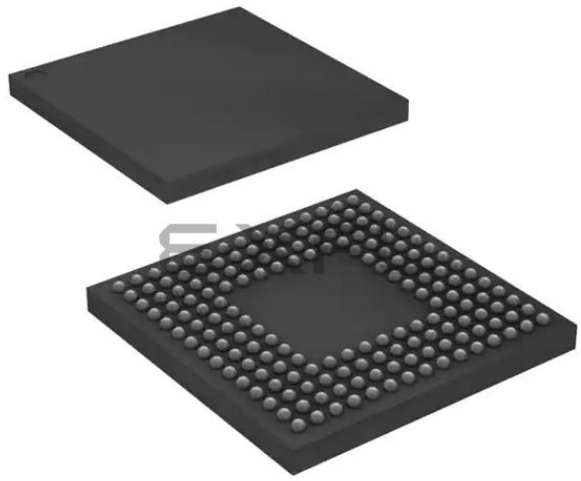


Welcome to E-XFL.COM

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	58
Program Memory Size	-
Program Memory Type	ROMless
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 8x10b; D/A 2x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	176-LFBGA
Supplier Device Package	176-BGA (13x13)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/at91m55800a-33cj

PIO Controllers called PIOA and PIOB. The PIO controller enables the generation of an interrupt on input change and insertion of a simple input glitch filter on any of the PIO pins.

8.4.5 WD: Watchdog

The Watchdog is built around a 16-bit counter, and is used to prevent system lock-up if the software becomes trapped in a deadlock. It can generate an internal reset or interrupt, or assert an active level on the dedicated pin NWDOVF. All programming registers are password-protected to prevent unintentional programming.

8.4.6 SF: Special Function

The AT91M55800A provides registers which implement the following special functions.

- Chip identification
- RESET status

8.5 User Peripherals

8.5.1 USART: Universal Synchronous Asynchronous Receiver Transmitter

The AT91M55800A provides three identical, full-duplex, universal synchronous/asynchronous receiver/transmitters.

Each USART has its own baud rate generator, and two dedicated Peripheral Data Controller channels. The data format includes a start bit, up to 8 data bits, an optional programmable parity bit and up to 2 stop bits.

The USART also features a Receiver Timeout register, facilitating variable-length frame support when it is working with the PDC, and a Time-guard register, used when interfacing with slow remote equipment.

8.5.2 TC: Timer Counter

The AT91M55800A features two Timer Counter blocks that include three identical 16-bit timer counter channels. Each channel can be independently programmed to perform a wide range of functions including frequency measurement, event counting, interval measurement, pulse generation, delay timing and pulse-width modulation.

The Timer Counters can be used in Capture or Waveform mode, and all three counter channels can be started simultaneously and chained together.

8.5.3 SPI: Serial Peripheral Interface

The SPI provides communication with external devices in master or slave mode. It has four external chip selects that can be connected to up to 15 devices. The data length is programmable, from 8- to 16-bit.

8.5.4 ADC: Analog-to-digital Converter

The two identical 4-channel 10-bit analog-to-digital converters (ADC) are based on a Successive Approximation Register (SAR) approach.

Each ADC has 4 analog input pins, AD0 to AD3 and AD4 to AD7, digital trigger input pins AD0TRIG and AD1TRIG, and provides an interrupt signal to the AIC. Both ADCs share the analog power supply pins VDDA and GNDA, and the input reference voltage pin ADVREF.

Each channel can be enabled or disabled independently, and has its own data register. The ADC can be configured to automatically enter Sleep mode after a conversion sequence, and can be triggered by the software, the Timer Counter, or an external signal.

8.5.5 DAC: Digital-to-analog Converter

Each DAC has an analog output pin, DA0 and DA1, and provides an interrupt signal to the AIC DA0IRQ and DA1IRQ. Both DACs share the analog power supply pins VDDA and GNDA, and the input reference DAVREF.

11.2 EBI Pin Description

Name	Description	Type
A0 - A23	Address bus (output)	Output
D0 - D15	Data bus (input/output)	I/O
NCS0 - NCS7	Active low chip selects (output)	Output
NRD	Read Enable (output)	Output
NWR0 - NWR1	Lower and upper write enable (output)	Output
NOE	Output enable (output)	Output
NWE	Write enable (output)	Output
NUB, NLB	Upper and lower byte-select (output)	Output
NWAIT	Wait request (input)	Input

The following table shows how certain EBI signals are multiplexed:

Multiplexed Signals		Functions
A0	NLB	8- or 16-bit data bus
NRD	NOE	Byte-write or byte-select access
NWR0	NWE	Byte-write or byte-select access
NWR1	NUB	Byte-write or byte-select access

11.6 Read Protocols

The EBI provides two alternative protocols for external memory read access: standard and early read. The difference between the two protocols lies in the timing of the NRD (read cycle) waveform.

The protocol is selected by the DRP field in EBI_MCR (Memory Control Register) and is valid for all memory devices. Standard read protocol is the default protocol after reset.

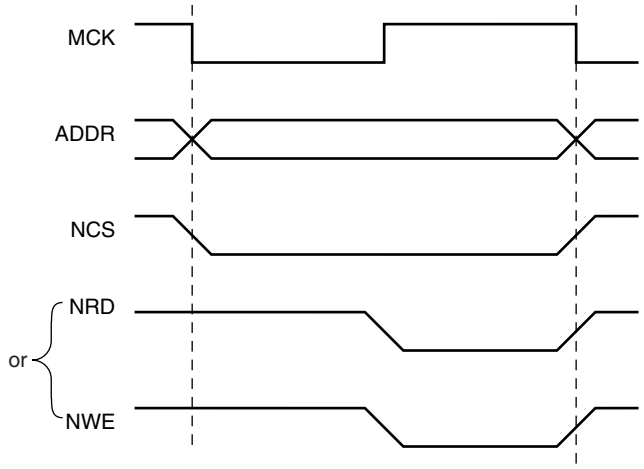
Note: In the following waveforms and descriptions, **NRD** represents NRD and NOE since the two signals have the same waveform. Likewise, **NWE** represents NWE, NWR0 and NWR1 unless NWR0 and NWR1 are otherwise represented. **ADDR** represents A0 - A23 and/or A1 - A23.

11.6.1 Standard Read Protocol

Standard read protocol implements a read cycle in which NRD and NWE are similar. Both are active during the second half of the clock cycle. The first half of the clock cycle allows time to ensure completion of the previous access as well as the output of address and NCS before the read cycle begins.

During a standard read protocol, external memory access, NCS is set low and ADDR is valid at the beginning of the access while NRD goes low only in the second half of the master clock cycle to avoid bus conflict (see Figure 11-7). NWE is the same in both protocols. NWE always goes low in the second half of the master clock cycle (see Figure 11-8).

Figure 11-7. Standard Read Protocol



11.6.2 Early Read Protocol

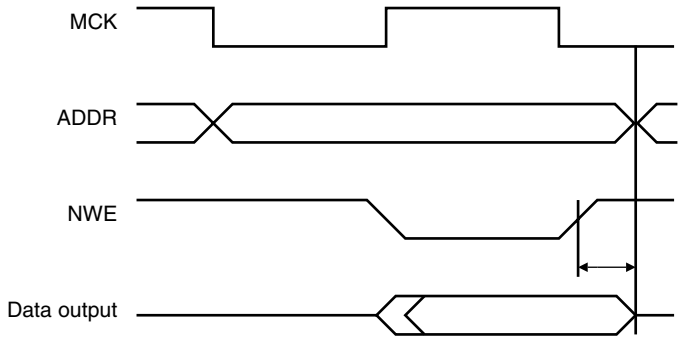
Early read protocol provides more time for a read access from the memory by asserting NRD at the beginning of the clock cycle. In the case of successive read cycles in the same memory, NRD remains active continuously. Since a read cycle normally limits the speed of operation of the external memory system, early read protocol can allow a faster clock frequency to be used. However, an extra wait state is required in some cases to avoid contentions on the external bus.

11.7 Write Data Hold Time

During write cycles in both protocols, output data becomes valid after the falling edge of the NWE signal and remains valid after the rising edge of NWE, as illustrated in the figure below. The external NWE waveform (on the NWE pin) is used to control the output data timing to guarantee this operation.

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 11-10. 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.

- **MUL: Phase Lock Loop Factor**

0 = The PLL is deactivated, reducing power consumption to a minimum.

1 - 63 = The PLL output is at a higher frequency (MUL+1) than the input if the bit lock is set in APMC_SR.

- **CSS: Clock Source Selection**

CSS		Clock Source Selection	Code Label
0	0	Low-frequency clock provided by the RTC	APMC_CSS_LF
0	1	Main oscillator Output or external clock	APMC_CSS_MOSC
1	0	Phase Lock Loop Output	APMC_CSS_PLL
1	1	Reserved	–

- **OSCOUNT: Main Oscillator Counter**

Specifies the number of 32,768 Hz divided by 8 clock cycles for the main oscillator start-up timer to count before the main oscillator is stabilized, after the oscillator is enabled. The main oscillator counter is a down-counter which is preloaded with the OSCOUNT value when the MOSCEN bit in the Clock Generator Mode register (CGMR) is set, but only if the OSCOUNT value is different from 0x0.

- **PLLCOUNT: PLL Lock Counter**

Specifies the number of 32,768 Hz clock cycles for the PLL lock timer to count before the PLL is locked, after the PLL is started. The PLL counter is a down-counter which is preloaded with the PLLCOUNT value when the MUL field in the Clock Generator Mode register (CGMR) is modified, but only if the MUL value is different from 0 (PLL disabled) and also the PLLCOUNT value itself different from 0x0. PLLCOUNT must be loaded with a minimum value of 2 in order to guarantee a time of at least one slow clock period.

13.3.6 RTC Calendar Alarm Register

Register Name: RTC_CAR

Access Type: Read/Write

Reset State: 0x0

Offset: 0x14

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

- **MONTH: Month Alarm**

This field is the alarm field corresponding to the BCD-coded month counter.

- **MTHEN: Month Alarm Enable**

MTHEN	Selected MTHEN	Code Label
0	The month matching alarm is disabled.	RTC_MONTH_ALARM_DIS
1	The month matching alarm is enabled.	RTC_MONTH_ALARM_EN

- **DATE: Date Alarm**

This field is the alarm field corresponding to the BCD-coded date counter.

- **DATEN: Date Alarm Enable**

DATEN	Selected DATEN	Code Label
0	The date matching alarm is disabled.	RTC_DATE_ALARM_DIS
1	The date matching alarm is enabled.	RTC_DATE_ALARM_EN

13.3.12 RTC Valid Entry Register

Register Name: RTC_VER

Access Type: Read-only

Reset State: 0x0

Offset: 0x2C

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
–	–	–	–	NVCAL	NVTAL	NVC	NVT

- **NVT: Non-Valid Time (Code Label `RTC_NVT`)**

0 = No invalid data has been detected in RTC_TIMR.

1 = RTC_TIMR has contained invalid data since it was last programmed.

- **NVC: Non-Valid Calendar (Code Label `RTC_NVC`)**

0 = No invalid data has been detected in RTC_CALR.

1 = RTC_CALR has contained invalid data since it was last programmed.

- **NVTAL: Non-Valid Time Alarm (Code Label `RTC_NVTAL`)**

0 = No invalid data has been detected in RTC_TAR.

1 = RTC_TAR has contained invalid data since it was last programmed.

- **NVCAL: Non-Valid Calendar Alarm (Code Label `RTC_NVCAL`)**

0 = No invalid data has been detected in RTC_CAR.

1 = RTC_CAR has contained invalid data since it was last programmed.

16.9.9 PIO Input Filter Status Register

Register Name: PIO_IFSR

Access Type: Read-only

Offset: 0x28

Reset Value: 0

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

This register indicates which pins have glitch filters selected. It is updated when PIO outputs are enabled or disabled by writing to PIO_IFER or PIO_IFDR.

1 = Filter is selected on the corresponding input (peripheral and PIO).

0 = Filter is not selected on the corresponding input.

Note: When the glitch filter is selected, and the PIO Controller clock is disabled, either the signal on the peripheral input or the corresponding bit in PIO_PDSR remains at the current state.

16.9.10 PIO Set Output Data Register

Register Name: PIO_SODR

Access Type: Write-only

Offset: 0x30

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

This register is used to set PIO output data. It affects the pin only if the corresponding PIO output line is enabled and if the pin is controlled by the PIO. Otherwise, the information is stored.

1 = PIO output data on the corresponding pin is set.

0 = No effect.

16.9.11 PIO Clear Output Data Register

Register Name: PIO_CODR

Access Type: Write-only

Offset: 0x34

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

This register is used to clear PIO output data. It affects the pin only if the corresponding PIO output line is enabled and if the pin is controlled by the PIO. Otherwise, the information is stored.

1 = PIO output data on the corresponding pin is cleared.

0 = No effect.

16.9.12 PIO Output Data Status Register

Register Name: PIO_ODSR

Access Type: Read-only

Offset: 0x38

Reset Value: 0

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

This register shows the output data status which is programmed in PIO_SODR or PIO_CODR. The defined value is effective only if the pin is controlled by the PIO Controller and only if the pin is defined as an output.

1 = The output data for the corresponding line is programmed to 1.

0 = The output data for the corresponding line is programmed to 0.

17. SF: Special Function Registers

The AT91M55800A provides registers which implement the following special functions.

- Chip identification
- RESET status

17.1 Chip Identifier

The following chip identifier values are covered in this datasheet:

Product	Revision	Chip ID
AT91M55800A	A	0x15580040

17.2 SF User Interface

Chip ID Base Address = 0xFFF00000 (Code Label `SF_BASE`)

Table 17-1. Register Mapping

Offset	Register	Name	Access	Reset State
0x00	Chip ID Register	SF_CIDR	Read-only	Hardwired
0x04	Chip ID Extension Register	SF_EXID	Read-only	Hardwired
0x08	Reset Status Register	SF_RSR	Read-only	See register description
0x0C	Reserved	–	–	–
0x10	Reserved	–	–	–
0x14	Reserved	–	–	–
0x18	Protect Mode Register	SF_PMR	Read/Write	0x0

17.2.4 SF Protect Mode Register

Register Name: SF_PMR

Access Type: Read/Write

Reset Value: 0x0

Offset: 0x18

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

- **PMRKEY: Protect Mode Register Key**

Used only when writing SF_PMR. PMRKEY is reads 0.

0x27A8: Write access in SF_PMR is allowed.

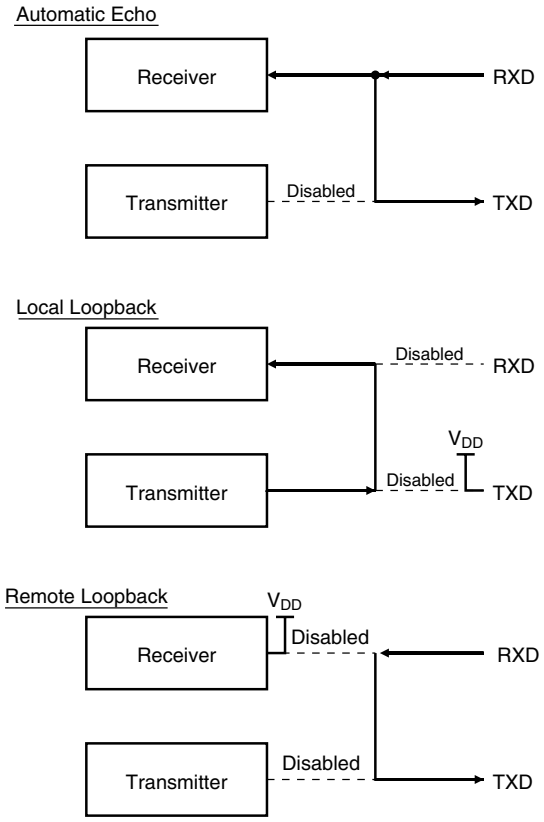
Other value: Write access in SF_PMR is prohibited.

- **AIC: AIC Protect Mode Enable (Code Label `SF_AIC`)**

0 = The Advanced Interrupt Controller runs in Normal Mode.

1 = The Advanced Interrupt Controller runs in Protect Mode.

Figure 18-7. Channel Modes



18.10.12 USART Receive Pointer Register

Name: US_RPR
Access Type: Read/Write
Reset State: 0
Offset: 0x30

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

- **RXPTR: Receive Pointer**

RXPTR must be loaded with the address of the receive buffer.

18.10.13 USART Receive Counter Register

Name: US_RCR
Access Type: Read/Write
Reset State: 0
Offset: 0x34

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

- **RXCTR: Receive Counter**

RXCTR must be loaded with the size of the receive buffer.

0 = Stop Peripheral Data Transfer dedicated to the receiver.

1 - 65535 = Start Peripheral Data transfer if RXRDY is active.

19.2 Timer Counter Description

Each Timer Counter channel is identical in operation. The registers for channel programming are listed in Table 19-1 on page 164.

19.2.1 Counter

Each Timer Counter channel is organized around a 16-bit counter. The value of the counter is incremented at each positive edge of the input clock. When the counter reaches the value 0xFFFF and passes to 0x0000, an overflow occurs and the bit COVFS in TC_SR (Status Register) is set.

The current value of the counter is accessible in real-time by reading TC_CV. The counter can be reset by a trigger. In this case, the counter value passes to 0x0000 on the next valid edge of the clock.

19.2.2 Clock Selection

At block level, input clock signals of each channel can either be connected to the external inputs TCLK0, TCLK1 or TCLK2, or be connected to the configurable I/O signals TIOA0, TIOA1 or TIOA2 for chaining by programming the TC_BMR (Block Mode).

Each channel can independently select an internal or external clock source for its counter:

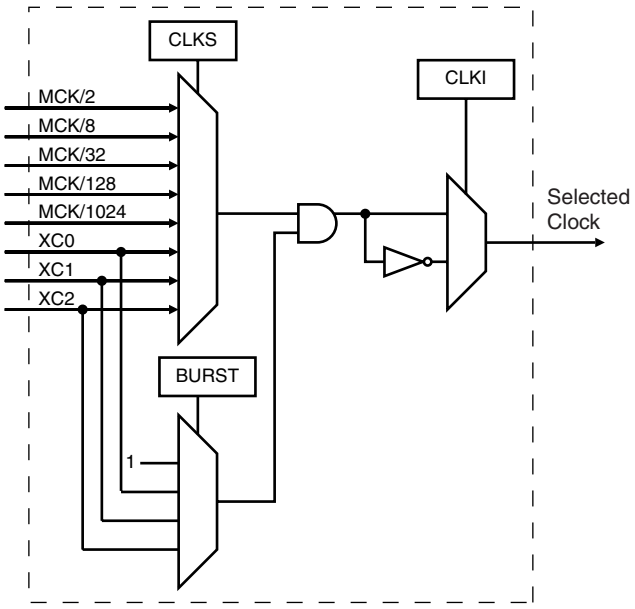
- Internal clock signals: MCK/2, MCK/8, MCK/32, MCK/128, MCK/1024
- External clock signals: XC0, XC1 or XC2

The selected clock can be inverted with the CLKI bit in TC_CMR (Channel Mode). This allows counting on the opposite edges of the clock.

The burst function allows the clock to be validated when an external signal is high. The BURST parameter in the Mode Register defines this signal (none, XC0, XC1, XC2).

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.

Figure 19-2. Clock Selection



19.5.5 TC Channel Mode Register: Waveform Mode

Register Name: TC_CMR
Access Type: Read/Write
Reset State: 0
Offset: 0x4

31	30	29	28	27	26	25	24
BSWTRG		BEEVT		BCPC		BCPB	
23	22	21	20	19	18	17	16
ASWTRG		AEEVT		ACPC		ACPA	
15	14	13	12	11	10	9	8
WAVE=1	CPCTRG	–	ENETRG	EEVT		EEVTEDG	
7	6	5	4	3	2	1	0
CPCDIS	CPCSTOP	BURST		CLKI	TCCLKS		

- **TCCLKS: Clock Selection**

TCCLKS			Clock Selected	Code Label: TC_CLKS
0	0	0	MCK/2	TC_CLKS_MCK2
0	0	1	MCK/8	TC_CLKS_MCK8
0	1	0	MCK/32	TC_CLKS_MCK32
0	1	1	MCK/128	TC_CLKS_MCK128
1	0	0	MCK/1024	TC_CLKS_MCK1024
1	0	1	XC0	TC_CLKS_XC0
1	1	0	XC1	TC_CLKS_XC1
1	1	1	XC2	TC_CLKS_XC2

- **CLKI: Clock Invert (Code Label TC_CLKI)**

0 = Counter is incremented on rising edge of the clock.

1 = Counter is incremented on falling edge of the clock.

- **BURST: Burst Signal Selection**

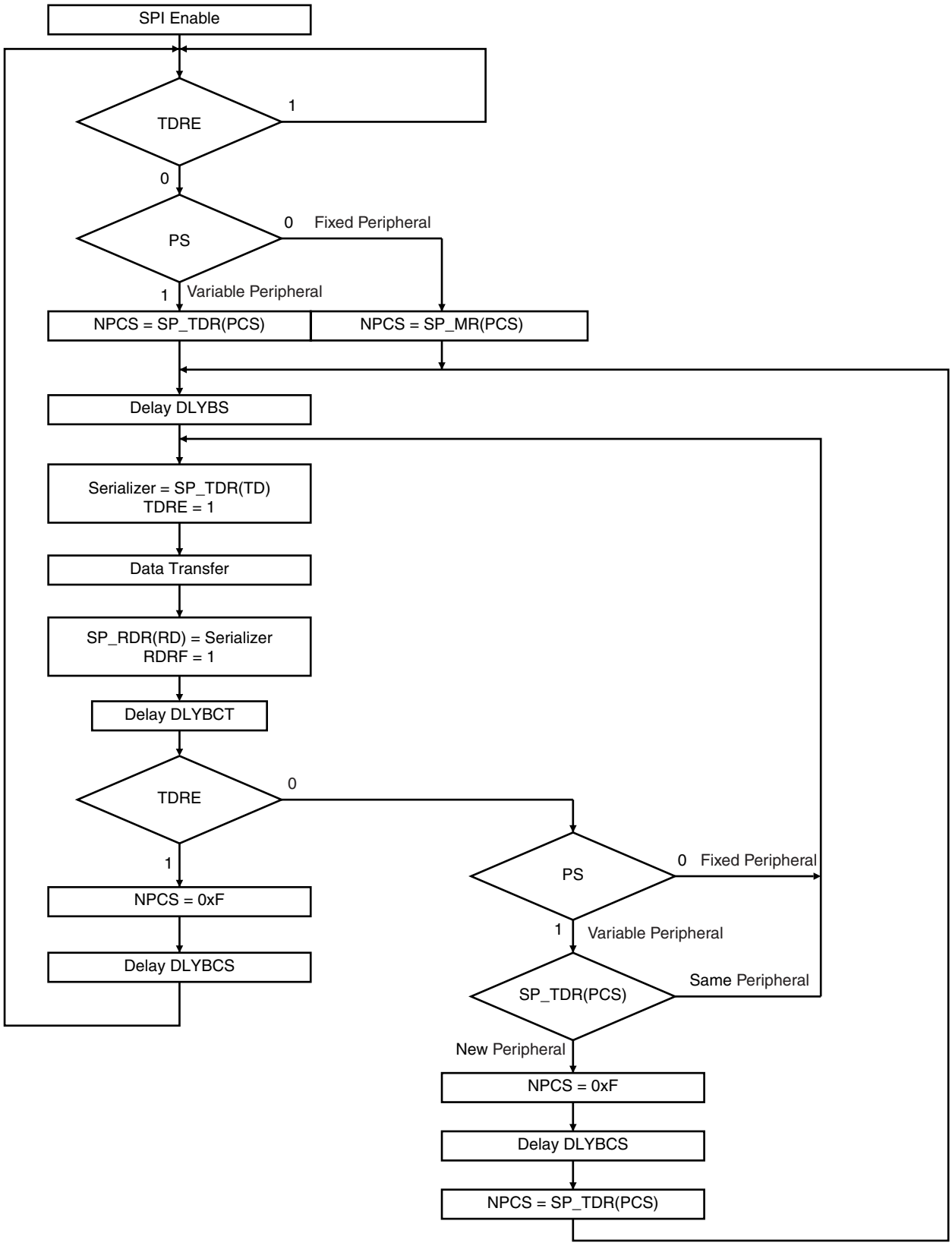
BURST		Selected BURST	Code Label: TC_BURST
0	0	The clock is not gated by an external signal.	TC_BURST_NONE
0	1	XC0 is ANDed with the selected clock.	TC_BURST_XC0
1	0	XC1 is ANDed with the selected clock.	TC_BURST_XC1
1	1	XC2 is ANDed with the selected clock.	TC_BURST_XC2

- **CPCSTOP: Counter Clock Stopped with RC Compare (Code Label TC_CPCSTOP)**

0 = Counter clock is not stopped when counter reaches RC.

1 = Counter clock is stopped when counter reaches RC.

Figure 20-2. Functional Flow Diagram in Master Mode



20.4 Data Transfer

The following waveforms show examples of data transfers.

Figure 20-4. SPI Transfer Format (NCPHA equals One, 8 bits per transfer)

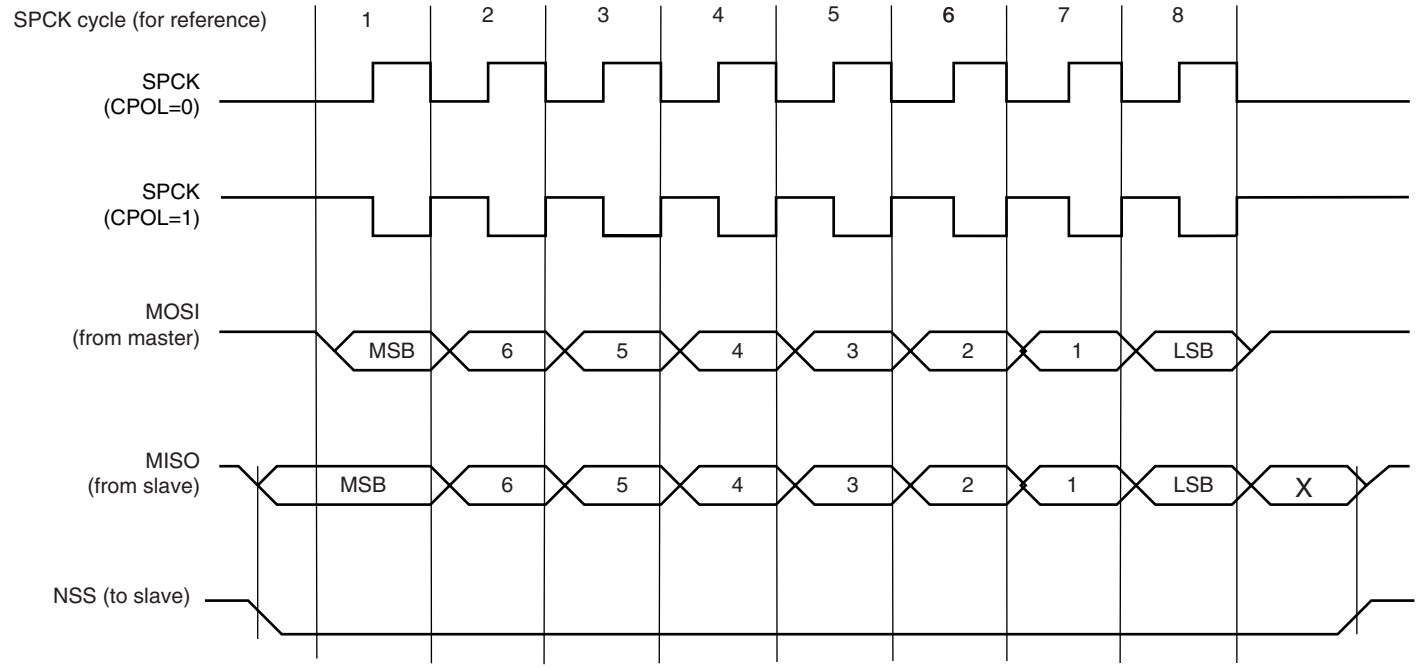
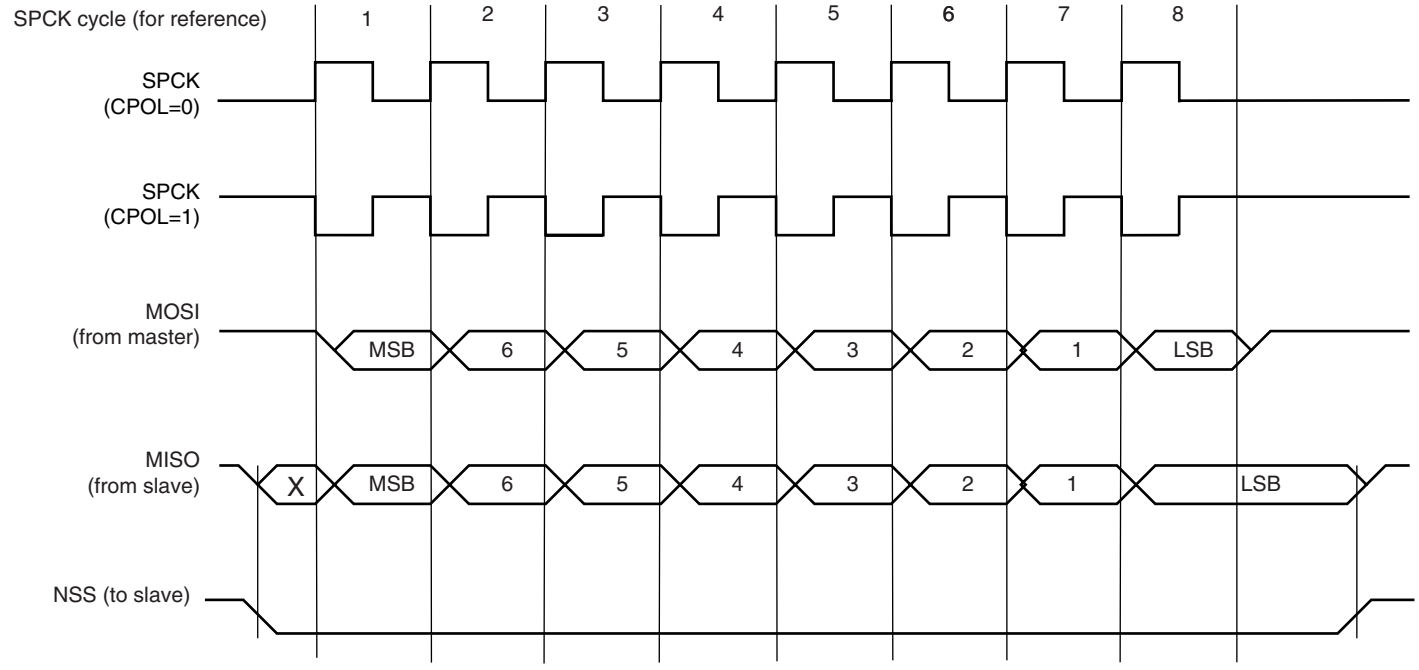


Figure 20-5. SPI Transfer Format (NCPHA equals Zero, 8 bits per transfer)



- **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`)**

This field is only used if Fixed Peripheral Select is active (PS=0).

If PCSDEC=0:

PCS = xxx0NPCS[3:0] = 1110 (Code Label `SP_PCS0`)

PCS = xx01NPCS[3:0] = 1101 (Code Label `SP_PCS1`)

PCS = x011NPCS[3:0] = 1011 (Code Label `SP_PCS2`)

PCS = 0111NPCS[3:0] = 0111 (Code Label `SP_PCS3`)

PCS = 1111forbidden (no peripheral is selected)

(x = don't care)

If PCSDEC=1:

NPCS[3:0] output signals = PCS.

- **DLYBCS: Delay Between Chip Selects (Code Label `SP_DLYBCS`)**

This field defines the delay from NPCS inactive to the activation of another NPCS. The DLYBCS time guarantees non-overlapping chip selects and solves bus contentions in case of peripherals having long data float times.

If DLYBCS is less than or equal to six, six SPI Master Clock periods will be inserted by default.

Otherwise, the following equation determines the delay:

Delay_ Between_Chip_Selects = DLYBCS * SPI_Master_Clock_period

21.0.5 ADC User Interface

Base Address ADC 0:0xFFFB0000 (Code Label `ADC0_BASE`)
Base Address ADC 1:0xFFFB4000 (Code Label `ADC1_BASE`)

Table 21-2. Register Mapping

Offset	Register	Name	Access	Reset
0x00	Control Register	ADC_CR	Write-only	–
0x04	Mode Register	ADC_MR	Read/Write	0
0x08	Reserved	–	–	–
0x0C	Reserved	–	–	–
0x10	Channel Enable Register	ADC_CHER	Write-only	–
0x14	Channel Disable Register	ADC_CHDR	Write-only	–
0x18	Channel Status Register	ADC_CHSR	Read-only	0
0x1C	Reserved	–	–	–
0x20	Status Register	ADC_SR	Read-only	0
0x24	Interrupt Enable Register	ADC_IER	Write-only	–
0x28	Interrupt Disable Register	ADC_IDR	Write-only	–
0x2C	Interrupt Mask Register	ADC_IMR	Read-only	0
0x30	Convert Data Register 0	ADC_CDR0	Read-only	0
0x34	Convert Data Register 1	ADC_CDR1	Read-only	0
0x38	Convert Data Register 2	ADC_CDR2	Read-only	0
0x3C	Convert Data Register 3	ADC_CDR3	Read-only	0