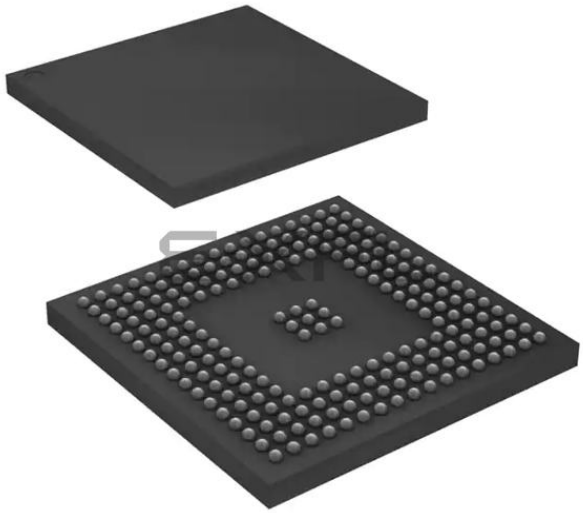
# Microchip Technology - AT91SAM9X35-CU-999 Datasheet



**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 | Active |
| Core Processor | ARM926EJ-S |
| Number of Cores/Bus Width | 1 Core, 32-Bit |
| Speed | 400MHz |
| Co-Processors/DSP | - |
| RAM Controllers | LPDDR, LPDDR2, DDR2, DDR, SDR, SRAM |
| Graphics Acceleration | No |
| Display & Interface Controllers | LCD, Touchscreen |
| Ethernet | 10/100Mbps |
| SATA | - |
| USB | USB 2.0 (3) |
| Voltage - I/O | 1.8V, 2.5V, 3.0V, 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/at91sam9x35-cu-999 |

## 5.1 Memory Mapping

A first level of address decoding is performed by the AHB Bus Matrix, i.e., the implementation of the Advanced High performance Bus (AHB) for its Master and Slave interfaces with additional features.

Decoding breaks up the 4 Gbytes of address space into 16 banks of 256 Mbytes. Banks 1 to 6 are directed to the EBI that associates these banks to the external chip selects, EBI_NCS0 to EBI_NCS5. Bank 0 is reserved for the addressing of the internal memories, and a second level of decoding provides 1 Mbyte of internal memory area. Bank 15 is reserved for the peripherals and provides access to the Advanced Peripheral Bus (APB).

Other areas are unused and performing an access within them provides an abort to the master requesting such an access.

## 5.2 Embedded Memories

### 5.2.1 Internal SRAM

The SAM9X35 embeds a total of 32 Kbytes of high-speed SRAM.

After reset and until the Remap Command is performed, the SRAM is only accessible at address 0x0030 0000.

After Remap, the SRAM also becomes available at address 0x0.

### 5.2.2 Internal ROM

The SAM9X35 embeds an Internal ROM, which contains the SAM-BA® program.

At any time, the ROM is mapped at address 0x0010 0000. It is also accessible at address 0x0 (BMS = 1) after the reset and before the Remap Command.

## 5.3 External Memories

### 5.3.1 External Bus Interface

- Integrates three External Memory Controllers:
  - Static Memory Controller
  - DDR2/SDRAM Controller
  - MLC NAND Flash ECC Controller
- Additional logic for NAND Flash and CompactFlash®
- Up to 26-bit Address Bus (up to 64 Mbytes linear per chip select)
- Up to 6 chip selects, Configurable Assignment:
  - Static Memory Controller on NCS0, NCS1, NCS2, NCS3, NCS4, NCS5
  - DDR2/SDRAM Controller (SDCS) or Static Memory Controller on NCS1
  - Optional NAND Flash support on NCS3

# 12. Advanced Interrupt Controller (AIC)

## 12.1 Description

The Advanced Interrupt Controller (AIC) is an 8-level priority, individually maskable, vectored interrupt controller, providing handling of up to 32 interrupt sources. It is designed to substantially reduce the software and real-time overhead in handling internal and external interrupts.

The AIC drives the nFIQ (fast interrupt request) and the nIRQ (standard interrupt request) inputs of an ARM processor. Inputs of the AIC are either internal peripheral interrupts or external interrupts coming from the product's pins.

The 8-level Priority Controller allows the user to define the priority for each interrupt source, thus permitting higher priority interrupts to be serviced even if a lower priority interrupt is being treated.

Internal interrupt sources can be programmed to be level sensitive or edge triggered. External interrupt sources can be programmed to be positive-edge or negative-edge triggered or high-level or low-level sensitive.

The Fast Forcing feature redirects any internal or external interrupt source to provide a fast interrupt rather than a normal interrupt.

## 12.2 Embedded Characteristics

- Controls the Interrupt Lines (nIRQ and nFIQ) of an ARM® Processor
- 32 Individually Maskable and Vectored Interrupt Sources
  - Source 0 is Reserved for the Fast Interrupt Input (FIQ)
  - Source 1 is Reserved for System Peripherals
  - Source 2 to Source 31 Control up to 30 Embedded Peripheral Interrupts or External Interrupts
  - Programmable Edge-triggered or Level-sensitive Internal Sources
  - Programmable Positive/Negative Edge-triggered or High/Low Level-sensitive External Sources
- 8-level Priority Controller
  - Drives the Normal Interrupt of the Processor
  - Handles Priority of the Interrupt Sources 1 to 31
  - Higher Priority Interrupts Can Be Served During Service of Lower Priority Interrupt
- Vectoring
  - Optimizes Interrupt Service Routine Branch and Execution
  - One 32-bit Vector Register per Interrupt Source
  - Interrupt Vector Register Reads the Corresponding Current Interrupt Vector
- Protect Mode
  - Easy Debugging by Preventing Automatic Operations when Protect Models Are Enabled
- Fast Forcing
  - Permits Redirecting any Normal Interrupt Source to the Fast Interrupt of the Processor
- General Interrupt Mask
  - Provides Processor Synchronization on Events Without Triggering an Interrupt
- Register Write Protection

### 12.9.13 AIC End of Interrupt Command Register

**Name:** AIC_EOICR

**Address:** 0xFFFFF130

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 |
| – | – | – | – | – | – | – | ENDIT |

- **ENDIT: Interrupt Processing Complete Command**

The End of Interrupt Command Register is used by the interrupt routine to indicate that the interrupt treatment is complete. Any value can be written because it is only necessary to make a write to this register location to signal the end of interrupt treatment.

### 12.9.17 AIC Fast Forcing Disable Register

**Name:** AIC_FFDR

**Address:** 0xFFFFF144

Access: Write-only

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|
| PID31 | PID30 | PID29 | PID28 | PID27 | PID26 | PID25 | PID24 |
| 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| PID23 | PID22 | PID21 | PID20 | PID19 | PID18 | PID17 | PID16 |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| PID15 | PID14 | PID13 | PID12 | PID11 | PID10 | PID9 | PID8 |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PID7 | PID6 | PID5 | PID4 | PID3 | PID2 | SYS | – |

- **SYS: Fast Forcing Disable**

0: No effect.

1: Disables the Fast Forcing feature on the corresponding interrupt.

- **PID2–PID31: Fast Forcing Disable**

0: No effect.

1: Disables the Fast Forcing feature on the corresponding interrupt.

### 20.5.4 Bypassing the 12 to 16 MHz Crystal Oscillator

Prior to bypassing the 12 to 16 MHz crystal oscillator, the external clock frequency provided on the XIN pin must be stable and within the values specified in the XIN Clock characteristics in the section “Electrical Characteristics”.

The sequence to bypass the crystal oscillator is the following:

1. Ensure that an external clock is connected on XIN.
2. Enable the bypass by setting CKGR_MOR.MOSCXTBY.
3. Disable the 12 to 16 MHz crystal oscillator by clearing CKGR_MOR.MOSCXTEN.

### 20.5.5 Main Clock Frequency Counter

The frequency counter is managed by CKGR_MCFR.

During the measurement period, the frequency counter increments at the main clock speed.

A measurement is started in the following cases:

- When the 12 MHz RC oscillator is selected as the source of the main clock and when this oscillator becomes stable (i.e., when the MOSCRCS bit is set)
- When the 12 to 16 MHz crystal oscillator is selected as the source of the main clock and when this oscillator becomes stable (i.e., when the MOSCXTS bit is set)
- When the main clock source selection is modified

The measurement period ends at the 16th falling edge of slow clock, the MAINFRDY bit in the CKGR_MCFR is set and the counter stops counting. Its value can be read in the MAINF field of CKGR_MCFR and gives the number of main clock cycles during 16 periods of slow clock, so that the frequency of the 12 MHz RC oscillator or the crystal oscillator can be determined.

### 20.5.6 Switching Main Clock Between the RC Oscillator and the Crystal Oscillator

For USB operations an external 12 MHz crystal is required for better accuracy.

The programmer controls the main clock switching by software and so must take precautions during the switching phase.

To switch from internal 12 MHz RC oscillator to the 12 MHz crystal, the programmer must execute the following sequence:

1. Enable the 12 MHz oscillator by setting the bit MOSCXTEN to 1.
2. Wait that the 12 MHz oscillator status bit MOSCXTS is 1.
3. Switch from internal 12 MHz RC oscillator to the 12 MHz oscillator by setting the bit MOSCSEL to 1.
4. If not the bit MOSCSEL is set to 0 by the PMC.
5. Disable the 12 MHz RC oscillator by setting the bit MOSCRCEN to 0.

- **BAT: Byte Access Type**

This field is used only if DBW defines a 16- or 32-bit data bus.

| Value | Name | Description |
|---|---|---|
| 0 | BYTE_SELECT | Byte select access type:<br>- Write operation is controlled using NCS, NWE, NBS0, NBS1, NBS2 and NBS3<br>- Read operation is controlled using NCS, NRD, NBS0, NBS1, NBS2 and NBS3 |
| 1 | BYTE_WRITE | Byte write access type:<br>- Write operation is controlled using NCS, NWR0, NWR1, NWR2, NWR3<br>- Read operation is controlled using NCS and NRD |

- **DBW: Data Bus Width**

| Value | Name | Description |
|---|---|---|
| 00 | BIT_8 | 8-bit bus |
| 01 | BIT_16 | 16-bit bus |
| 10 | BIT_32 | 32-bit bus |
| 11 | — | Reserved |

- **TDF_CYCLES: Data Float Time**

This field gives the integer number of clock cycles required by the external device to release the data after the rising edge of the read controlling signal. The SMC always provides one full cycle of bus turnaround after the TDF_CYCLES period. The external bus cannot be used by another chip select during TDF_CYCLES + 1 cycles. From 0 up to 15 TDF_CYCLES can be set.

- **TDF_MODE: TDF Optimization**

1: TDF optimization enabled—The number of TDF wait states is optimized using the setup period of the next read/write access.

0: TDF optimization disabled—The number of TDF wait states is inserted before the next access begins.

- **PMEN: Page Mode Enabled**

1: Asynchronous burst read in page mode is applied on the corresponding chip select.

0: Standard read is applied.

- **PS: Page Size**

If page mode is enabled, this field indicates the size of the page in bytes.

| Value | Name | Description |
|---|---|---|
| 00 | BYTE_4 | 4-byte page |
| 01 | BYTE_8 | 8-byte page |
| 10 | BYTE_16 | 16-byte page |
| 11 | BYTE_32 | 32-byte page |

## 30.7 DMAC Software Requirements

- There must not be any write operation to channel registers in an active channel after the channel enable is made HIGH. If any channel parameters must be reprogrammed, this can only be done after disabling the DMAC channel.
- The channel registers DMAC_SADDRx and DMAC_DADDRx must be programmed with a byte, half-word and word aligned address depending on the source width and destination width.
- After the software disables a channel by writing into the DMAC Channel Handler Disable Register, it must re-enable the channel only after it has polled a ‘0’ in the DMAC Channel Handler Status Register. This is because the current AHB Burst must terminate properly.
- If the value of field DMAC_CTRLAx.BTSIZE is configured to zero and the DMAC has been defined as the flow controller, the channel is automatically disabled.
- Multiple transfers involving the same peripheral must not be programmed and enabled on different channels, unless this peripheral integrates several hardware handshaking interfaces.
- When a peripheral has been defined as the flow controller, the targeted DMAC channel must be enabled before the peripheral. If you do not ensure this and the first DMAC request is also the last transfer, the DMAC channel might miss a Last Transfer Flag.
- When the AUTO bit is set to TRUE, the BTSIZE field is automatically reloaded from its previous value. BTSIZE must be initialized to a non zero value if the first transfer is initiated with the AUTO bit set to TRUE, even if LLI mode is enabled, because the LLI fetch operation will not update this field.

### 31.7.24 UDPHS DMA Channel Control Register

**Name:** UDPHS_DMACONTROLx [x = 0..5]

**Address:** 0xF803C308 [0], 0xF803C318 [1], 0xF803C328 [2], 0xF803C338 [3], 0xF803C348 [4], 0xF803C358 [5]

**Access:** Read/Write

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|
| BUFF_LENGTH | | | | | | | |
| 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| BUFF_LENGTH | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| – | – | – | – | – | – | – | – |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| BURST_LCK | DESC_LD_IT | END_BUFFIT | END_TR_IT | END_B_EN | END_TR_EN | LDNXT_DSC | CHANN_ENB |

Note: Channel 0 is not used.

- **CHANN_ENB: (Channel Enable Command)**

0: DMA channel is disabled at and no transfer will occur upon request. This bit is also cleared by hardware when the channel source bus is disabled at end of buffer.

If the UDPHS_DMACONTROL register LDNXT_DSC bit has been cleared by descriptor loading, the firmware will have to set the corresponding CHANN_ENB bit to start the described transfer, if needed.

If the UDPHS_DMACONTROL register LDNXT_DSC bit is cleared, the channel is frozen and the channel registers may then be read and/or written reliably as soon as both UDPHS_DMASTATUS register CHANN_ENB and CHANN_ACT flags read as 0.

If a channel request is currently serviced when this bit is cleared, the DMA FIFO buffer is drained until it is empty, then the UDPHS_DMASTATUS register CHANN_ENB bit is cleared.

If the LDNXT_DSC bit is set at or after this bit clearing, then the currently loaded descriptor is skipped (no data transfer occurs) and the next descriptor is immediately loaded.

1: UDPHS_DMASTATUS register CHANN_ENB bit will be set, thus enabling DMA channel data transfer. Then any pending request will start the transfer. This may be used to start or resume any requested transfer.

- **LDNXT_DSC: Load Next Channel Transfer Descriptor Enable (Command)**

0: No channel register is loaded after the end of the channel transfer.

1: The channel controller loads the next descriptor after the end of the current transfer, i.e., when the UDPHS_DMASTATUS/CHANN_ENB bit is reset.

If the UDPHS_DMA CONTROL/CHANN_ENB bit is cleared, the next descriptor is immediately loaded upon transfer request.

DMA Channel Control Command Summary

| LDNXT_DSC | CHANN_ENB | Description |
|---|---|---|
| 0 | 0 | Stop now |
| 0 | 1 | Run and stop at end of buffer |
| 1 | 0 | Load next descriptor now |
| 1 | 1 | Run and link at end of buffer |

### 31.7.25 UDPHS DMA Channel Status Register

**Name:** UDPHS_DMASTATUSx [x = 0..5]

**Address:** 0xF803C30C [0], 0xF803C31C [1], 0xF803C32C [2], 0xF803C33C [3], 0xF803C34C [4], 0xF803C35C [5]

**Access:** Read/Write

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|
| BUFF_COUNT | | | | | | | |
| 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| BUFF_COUNT | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| – | – | – | – | – | – | – | – |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| – | DESC_LDST | END_BF_ST | END_TR_ST | – | – | CHANN_ACT | CHANN_ENB |

Note: Channel 0 is not used.

- **CHANN_ENB: Channel Enable Status**

0: The DMA channel no longer transfers data, and may load the next descriptor if the UDPHS_DMACONTROLx register LDNXT_DSC bit is set.

When any transfer is ended either due to an elapsed byte count or a UDPHS device initiated transfer end, this bit is automatically reset.

1: The DMA channel is currently enabled and transfers data upon request.

This bit is normally set or cleared by writing into the UDPHS_DMACONTROLx register CHANN_ENB bit either by software or descriptor loading.

If a channel request is currently serviced when the UDPHS_DMACONTROLx register CHANN_ENB bit is cleared, the DMA FIFO buffer is drained until it is empty, then this status bit is cleared.

- **CHANN_ACT: Channel Active Status**

0: The DMA channel is no longer trying to source the packet data.

When a packet transfer is ended this bit is automatically reset.

1: The DMA channel is currently trying to source packet data, i.e., selected as the highest-priority requesting channel.

When a packet transfer cannot be completed due to an END_BF_ST, this flag stays set during the next channel descriptor load (if any) and potentially until UDPHS packet transfer completion, if allowed by the new descriptor.

- **END_TR_ST: End of Channel Transfer Status**

0: Cleared automatically when read by software.

1: Set by hardware when the last packet transfer is complete, if the UDPHS device has ended the transfer.

Valid until the CHANN_ENB flag is cleared at the end of the next buffer transfer.

- **END_BF_ST: End of Channel Buffer Status**

0: Cleared automatically when read by software.

1: Set by hardware when the BUFF_COUNT downcount reach zero.

Valid until the CHANN_ENB flag is cleared at the end of the next buffer transfer.

## 35.6 Functional Description

### 35.6.1 Description

All channels of the Timer Counter are independent and identical in operation. The registers for channel programming are listed in Table 35-6 “Register Mapping”.

### 35.6.2 32-bit Counter

Each 32-bit channel is organized around a 32-bit counter. The value of the counter is incremented at each positive edge of the selected clock. When the counter has reached the value $2^{32}$-1 and passes to zero, an overflow occurs and the COVFS bit in the TC Status Register (TC_SR) is set.

The current value of the counter is accessible in real time by reading the TC Counter Value Register (TC_CV). The counter can be reset by a trigger. In this case, the counter value passes to zero on the next valid edge of the selected clock.

### 35.6.3 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 internal I/O signals TIOA0, TIOA1 or TIOA2 for chaining by programming the TC Block Mode Register (TC_BMR). See Figure 35-2.

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

- External clock signals(1): XC0, XC1 or XC2
- Internal clock signals: MCK/2, MCK/8, MCK/32, MCK/128, SLCK

This selection is made by the TCCLKS bits in the TC Channel Mode Register (TC_CMR).

The selected clock can be inverted with the CLKI bit in the TC_CMR. 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 TC_CMR defines this signal (none, XC0, XC1, XC2). See Figure 35-3.

Note: 1. In all cases, if an external clock is used, the duration of each of its levels must be longer than the peripheral clock period. The external clock frequency must be at least 2.5 times lower than the peripheral clock.

**Figure 35-2. Clock Chaining Selection**

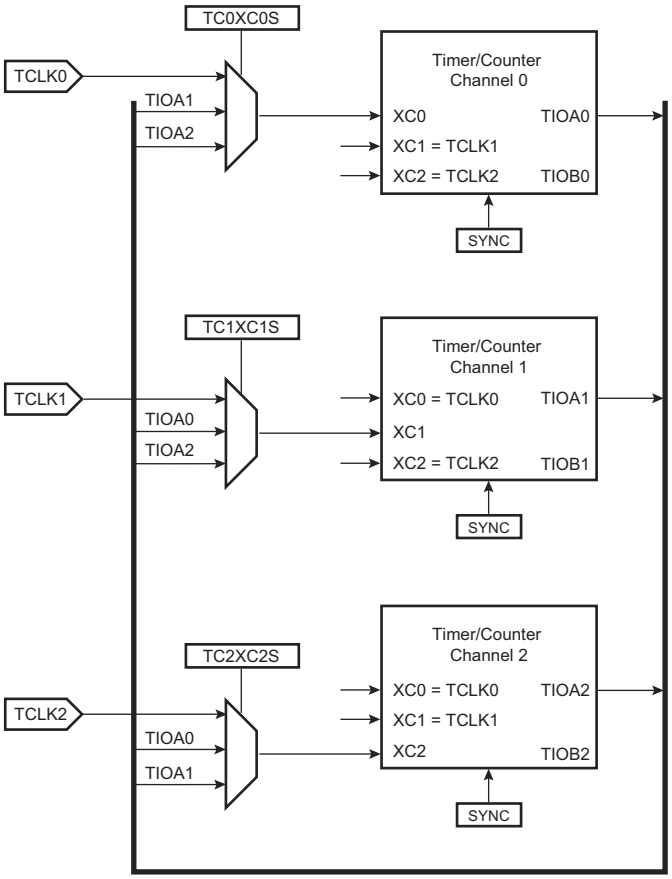
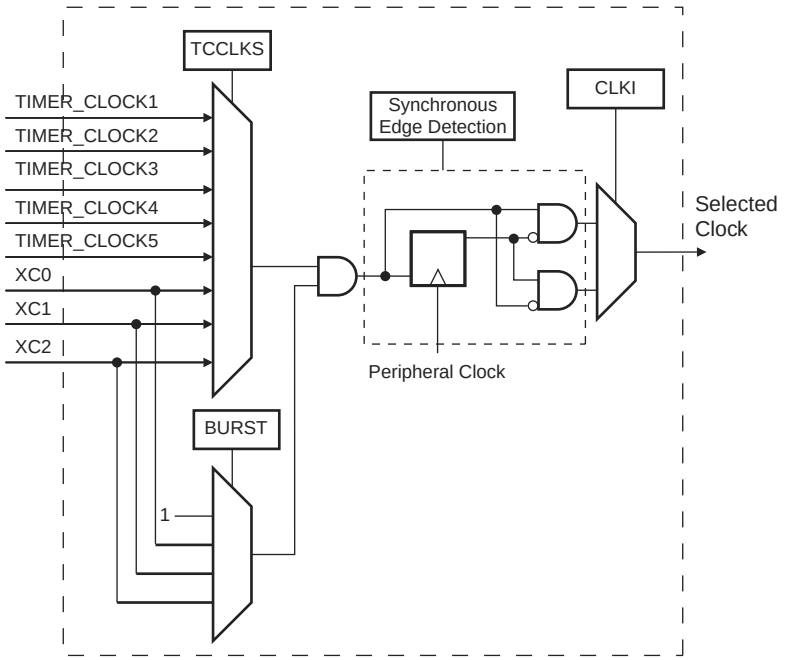



**Figure 35-3. Clock Selection**



# 41. Analog-to-Digital Converter (ADC)

## 41.1 Description

The ADC is based on a 10-bit Analog-to-Digital Converter (ADC) managed by an ADC Controller. Refer to Figure 41-1 “Analog-to-Digital Converter Block Diagram with Touchscreen Mode”. It also integrates a 12-to-1 analog multiplexer, making possible the analog-to-digital conversions of 12 analog lines. The conversions extend from 0V to the voltage carried on pin ADVREF.

The ADC digital controller embeds circuitry to reduce the resolution down to 8 bits. The 8-bit resolution mode prevents using 16-bit Peripheral DMA transfer into memory when only 8-bit resolution is required by the application. Note that using this low resolution mode does not increase the conversion rate.

Conversion results are reported in a common register for all channels, as well as in a channel-dedicated register.

Software trigger, external trigger on rising edge of the ADTRG pin or internal triggers from Timer Counter output(s) are configurable.

The comparison circuitry allows automatic detection of values below a threshold, higher than a threshold, in a given range or outside the range, thresholds and ranges being fully configurable.

The ADC also integrates a Sleep mode and a conversion sequencer and connects with a DMA channel. These features reduce both power consumption and processor intervention.

Finally, the user can configure ADC timings, such as startup time and tracking time.

This ADC Controller includes a Resistive Touchscreen Controller. It supports 4-wire and 5-wire technologies.

## 41.2 Embedded Characteristics

- 10-bit Resolution
- 440 kHz Conversion Rate
- Wide Range of Power Supply Operation
- Resistive 4-wire and 5-wire Touchscreen Controller
  - Position and Pressure Measurement for 4-wire Screens
  - Position Measurement for 5-wire Screens
  - Average of Up to 8 Measures for Noise Filtering
- Programmable Pen Detection Sensitivity
- Integrated Multiplexer Offering Up to 12 Independent Analog Inputs
- Individual Enable and Disable of Each Channel
- Hardware or Software Trigger
  - External Trigger Pin
  - Internal Trigger Counter
  - Trigger on Pen Contact Detection
- DMA Support
- Possibility of ADC Timings Configuration
- Two Sleep Modes and Conversion Sequencer
  - Automatic Wakeup on Trigger and Back to Sleep Mode after Conversions of all Enabled Channels
  - Possibility of Customized Channel Sequence
- Standby Mode for Fast Wakeup Time Response
  - Power Down Capability
- Automatic Window Comparison of Converted Values
- Register Write Protection

#### 44.6.26.3 Single Collision Frames Register

**Name:** EMAC_SCF

**Address:** 0xF802C044

**Access:** Read-write

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

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

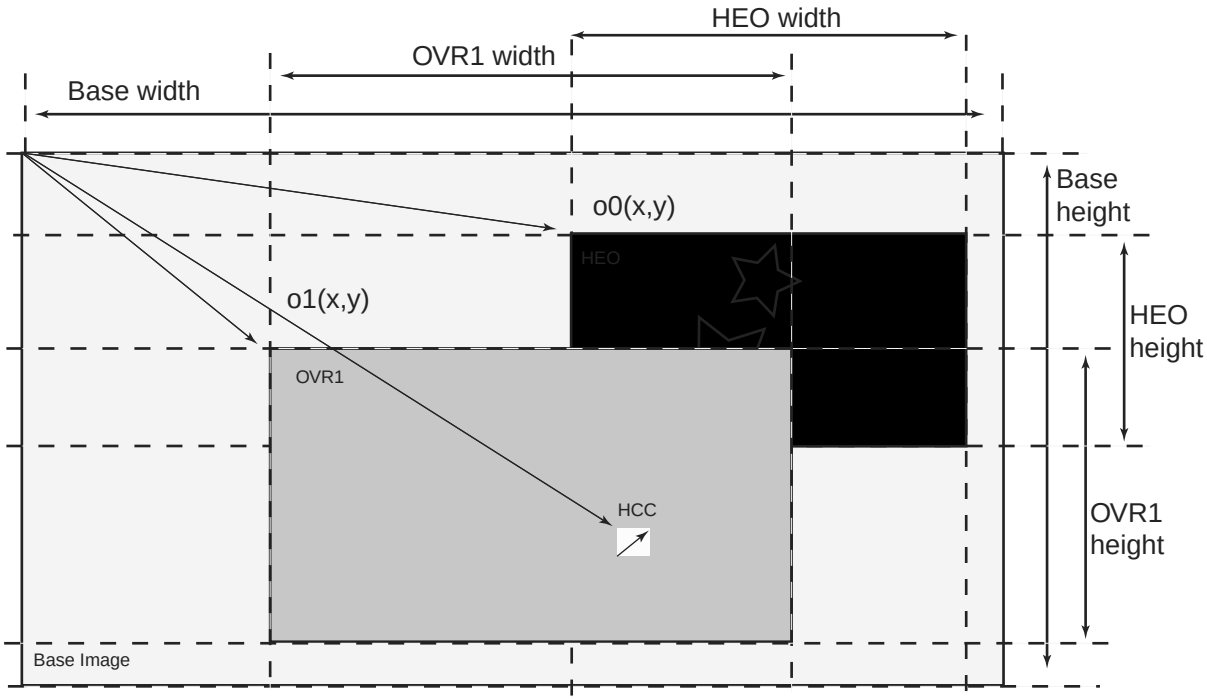
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|
| SCF | | | | | | | |

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| SCF | | | | | | | |

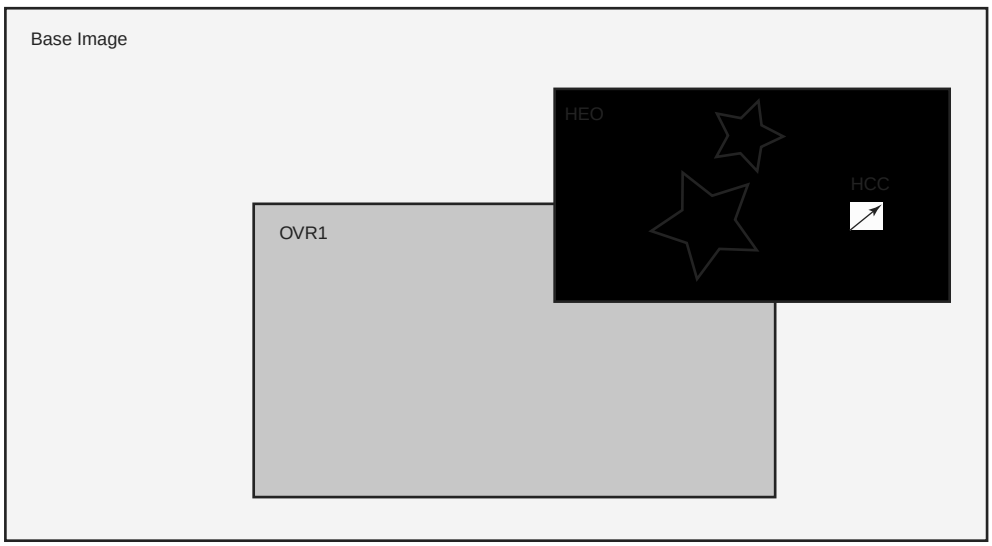
- **SCF: Single Collision Frames**

A 16-bit register counting the number of frames experiencing a single collision before being successfully transmitted, i.e., no underrun.

**Figure 45-6. Overlay Example with two different video prioritization algorithms**




Video Prioritization Algorithm 1: HCC > OVR1 > HEO > BASE

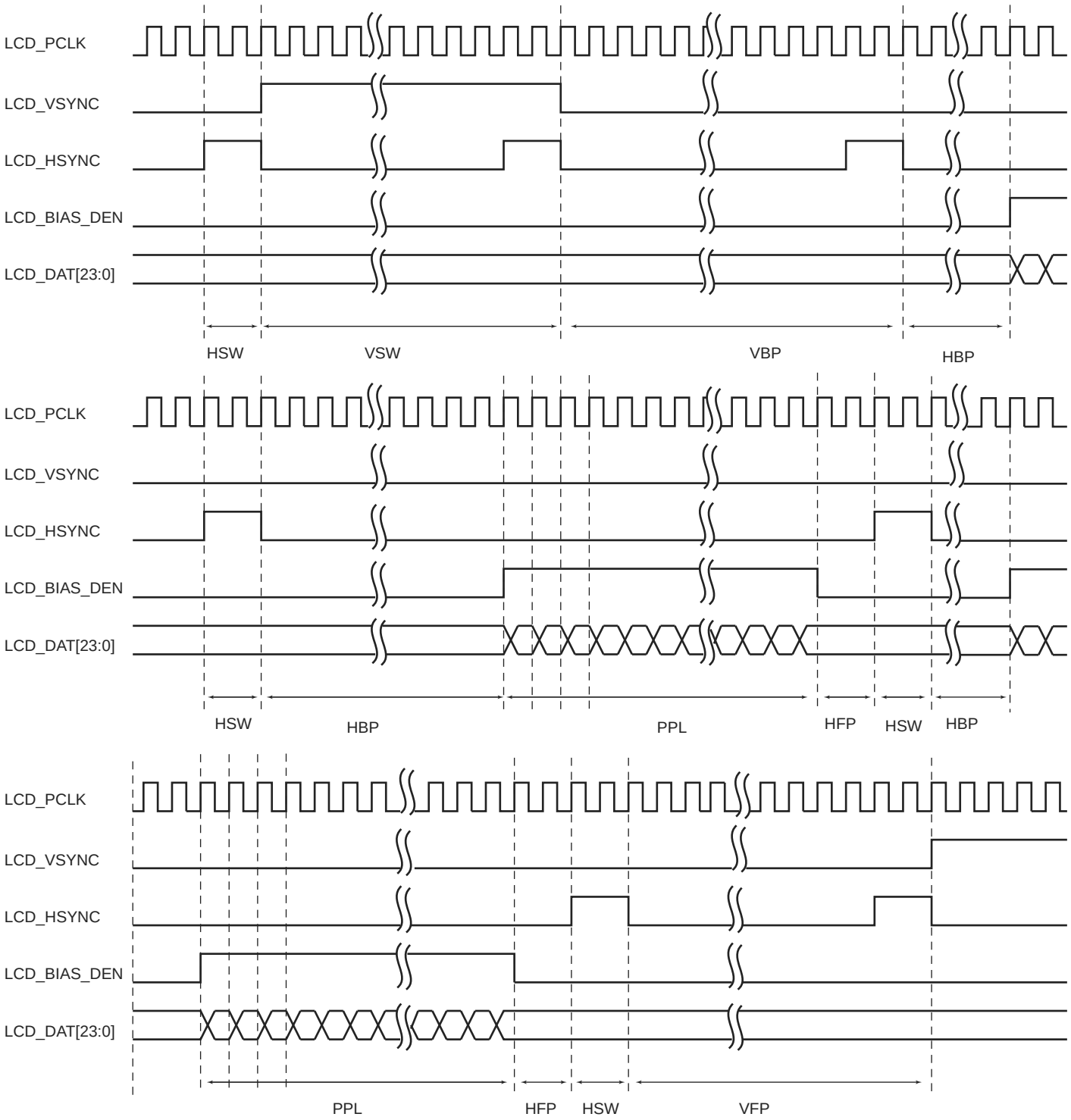



Video Prioritization Algorithm 2: HCC > HEO > OVR1 > BASE

### 45.6.13 Output Timing Generation

#### 45.6.13.1 Active Display Timing Mode

**Figure 45-11. Active Display Timing**



### 45.7.6 LCD Controller Configuration Register 5

**Name:** LCDC_LCDCFG5

**Address:** 0xF8038014

**Access:** Read/Write

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|
| – | – | – | – | – | – | – | – |
| 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| – | – | – | GUARDTIME | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| – | – | VSPHO | VSPSU | – | – | MODE | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| DISPDLY | DITHER | – | DISPPOL | VSPDLYE | VSPDLYS | VSPOL | HSPOL |

- **HSPOL: Horizontal Synchronization Pulse Polarity**

0: Active High

1: Active Low

- **VSPOL: Vertical Synchronization Pulse Polarity**

0: Active High

1: Active Low

- **VSPDLYS: Vertical Synchronization Pulse Start**

0: The first active edge of the Vertical synchronization pulse is synchronous with the second edge of the horizontal pulse.

1: The first active edge of the Vertical synchronization pulse is synchronous with the first edge of the horizontal pulse.

- **VSPDLYE: Vertical Synchronization Pulse End**

0: The second active edge of the Vertical synchronization pulse is synchronous with the second edge of the horizontal pulse.

1: The second active edge of the Vertical synchronization pulse is synchronous with the first edge of the horizontal pulse.

- **DISPPOL: Display Signal Polarity**

0: Active High

1: Active Low

- **DITHER: LCD Controller Dithering**

0: Dithering logical unit is disabled.

1: Dithering logical unit is activated.

- **DISPDLY: LCD Controller Display Power Signal Synchronization**

0: the LCD_DISP signal is asserted synchronously with the second active edge of the horizontal pulse.

1: the LCD_DISP signal is asserted asynchronously with both edges of the horizontal pulse.

### 45.7.45 Overlay 1 Layer Configuration 3 Register

**Name:** LCDC_OVR1CFG3

**Address:** 0xF8038138

**Access:** Read/Write

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

- **XSIZE: Horizontal Window Size**

Overlay 1 window width in pixels. The window width is set to (XSIZE + 1).

The following constraint must be met: XPOS + XSIZE ≤PPL

- **YSIZE: Vertical Window Size**

Overlay 1 window height in pixels. The window height is set to (YSIZE + 1).

The following constrain must be met: YPOS + YSIZE ≤RPF

### 45.7.56 High End Overlay Layer Interrupt Disable Register

**Name:** LCDC_HEOIDR

**Address:** 0xF8038290

**Access:** Write-only

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|
| – | – | – | – | – | – | – | – |
| 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| – | VOVR | VDONE | VADD | VDSCR | VDMA | – | – |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| – | UOVR | UDONE | UADD | UDSCR | UDMA | – | – |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| – | OVR | DONE | ADD | DSCR | DMA | – | – |

- **DMA: End of DMA Transfer Interrupt Disable Register**

0: No effect.

1: Interrupt source is disabled.

- **DSCR: Descriptor Loaded Interrupt Disable Register**

0: No effect.

1: Interrupt source is disabled.

- **ADD: Head Descriptor Loaded Interrupt Disable Register**

0: No effect.

1: Interrupt source is disabled.

- **DONE: End of List Interrupt Disable Register**

0: No effect.

1: Interrupt source is disabled.

- **OVR: Overflow Interrupt Disable Register**

0: No effect.

1: Interrupt source is disabled.

- **UDMA: End of DMA Transfer for U or UV Chrominance Component Interrupt Disable Register**

0: No effect.

1: Interrupt source is disabled.

- **UDSCR: Descriptor Loaded for U or UV Chrominance Component Interrupt Disable Register**

0: No effect.

1: Interrupt source is disabled.

- **UADD: Head Descriptor Loaded for U or UV Chrominance Component Interrupt Disable Register**

0: No effect.

1: Interrupt source is disabled.

### 45.7.109 Overlay 1 CLUT Register x Register

**Name:** LCDC_OVR1CLUTx [x=0..255]

**Address:** 0xF8038800

**Access:** Read/Write

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

- **BCLUT: Blue Color entry**

This field indicates the 8-bit width Blue color of the color lookup table.

- **GCLUT: Green Color entry**

This field indicates the 8-bit width Green color of the color lookup table.

- **RCLUT: Red Color entry**

This field indicates the 8-bit width Red color of the color lookup table.

- **ACLUT: Alpha Color entry**

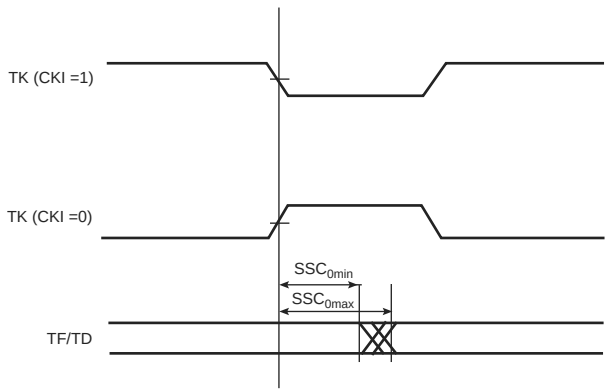
This field indicates the 8-bit width Alpha channel of the color lookup table.

**Table 46-43. SSC Timings - 3.3V Peripheral Supply**

| Symbol | Parameter | Conditions | Min | Max | Unit |
|---|---|---|---|---|---|
| | | Transmitter | | | |
| $SSC_0$ | TK edge to TF/TD (TK output, TF output) | | -4.6(1) | 4.9(1) | ns |
| $SSC_1$ | TK edge to TF/TD (TK input, TF output) | | 2.3(1) | 11.4(1) | ns |
| $SSC_2$ | TF setup time before TK edge (TK output) | | 9.9 | | ns |
| $SSC_3$ | TF hold time after TK edge (TK output) | | 0 | | ns |
| $SSC_4$ | TK edge to TD (TK output, TF input) | | -4.6 (1) | 4.7 (1) | ns |
| | | STTDLY = 0<br>START = 4, 5 or 7 | -4.6 (+2 × $t_{CPMCK}$)(1) | 4.7 (+2 × $t_{CPMCK}$)(1) | |
| $SSC_5$ | TF setup time before TK edge (TK input) | | 0 | | ns |
| $SSC_6$ | TF hold time after TK edge (TK input) | | $t_{CPMCK}$ | | ns |
| $SSC_7$ | TK edge to TD (TK input, TF input) | | 2.3 (1) | 11.1 (1) | ns |
| | | STTDLY = 0<br>START = 4, 5 or 7 | 2.3 (+3 × $t_{CPMCK}$)(1) | 11.1 (+3 × $t_{CPMCK}$)(1) | |
| | | Receiver | | | |
| $SSC_8$ | RF/RD setup time before RK edge (RK input) | | 0 | | ns |
| $SSC_9$ | RF/RD hold time after RK edge (RK input) | | $t_{CPMCK}$ | | ns |
| $SSC_{10}$ | RK edge to RF (RK input) | | 2.0(1) | 10.9(1) | ns |
| $SSC_{11}$ | RF/RD setup time before RK edge (RK output) | | 10.0 - $t_{CPMCK}$ | | ns |
| $SSC_{12}$ | RF/RD hold time after RK edge (RK output) | | $t_{CPMCK}$ - 1.8 | | ns |
| $SSC_{13}$ | RK edge to RF (RK output) | | -4.9(1) | 4.3(1) | ns |

Notes: 1. For output signals (TF, TD, RF), minimum and maximum access times are defined. The minimum access time is the time between the TK (or RK) edge and the signal change. The maximum access time is the time between the TK edge and the signal stabilization. Figure 46-24 illustrates minimum and maximum accesses for SSC0. The same applies to SSC1, SSC4, and SSC7, SSC10 and SSC13.

**Figure 46-24. Minimum and Maximum Access Time of Output Signals**




### 46.18.3 HSMCI

The High Speed MultiMedia Card Interface (HSMCI) supports the MultiMedia Card (MMC) Specification V4.3, the SD Memory Card Specification V2.0, the SDIO V2.0 specification and CE-ATA V1.1.