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	ARM® Cortex®-M0+
Core Size	32-Bit Single-Core
Speed	48MHz
Connectivity	I²C, LINbus, SCI, SPI, UART/USART, USB
Peripherals	DMA, POR, PWM, WDT
Number of I/O	37
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	32K x 8
Voltage - Supply (Vcc/Vdd)	1.62V ~ 3.63V
Data Converters	A/D 14x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	48-TQFP
Supplier Device Package	48-TQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsaml21g18b-ant

Bits 7:0 – DEVSEL[7:0]: Device Selection

This bit field identifies a device within a product family and product series. Refer to the Ordering Information for device configurations and corresponding values for Flash memory density, pin count and device variant.

16.3. Register Synchronization

16.3.1. Overview

All peripherals are composed of one digital bus interface connected to the APB or AHB bus and running from a corresponding clock in the Main Clock domain, and one peripheral core running from the peripheral Generic Clock (GCLK).

Communication between these clock domains must be synchronized. This mechanism is implemented in hardware, so the synchronization process takes place even if the peripheral generic clock is running from the same clock source and on the same frequency as the bus interface.

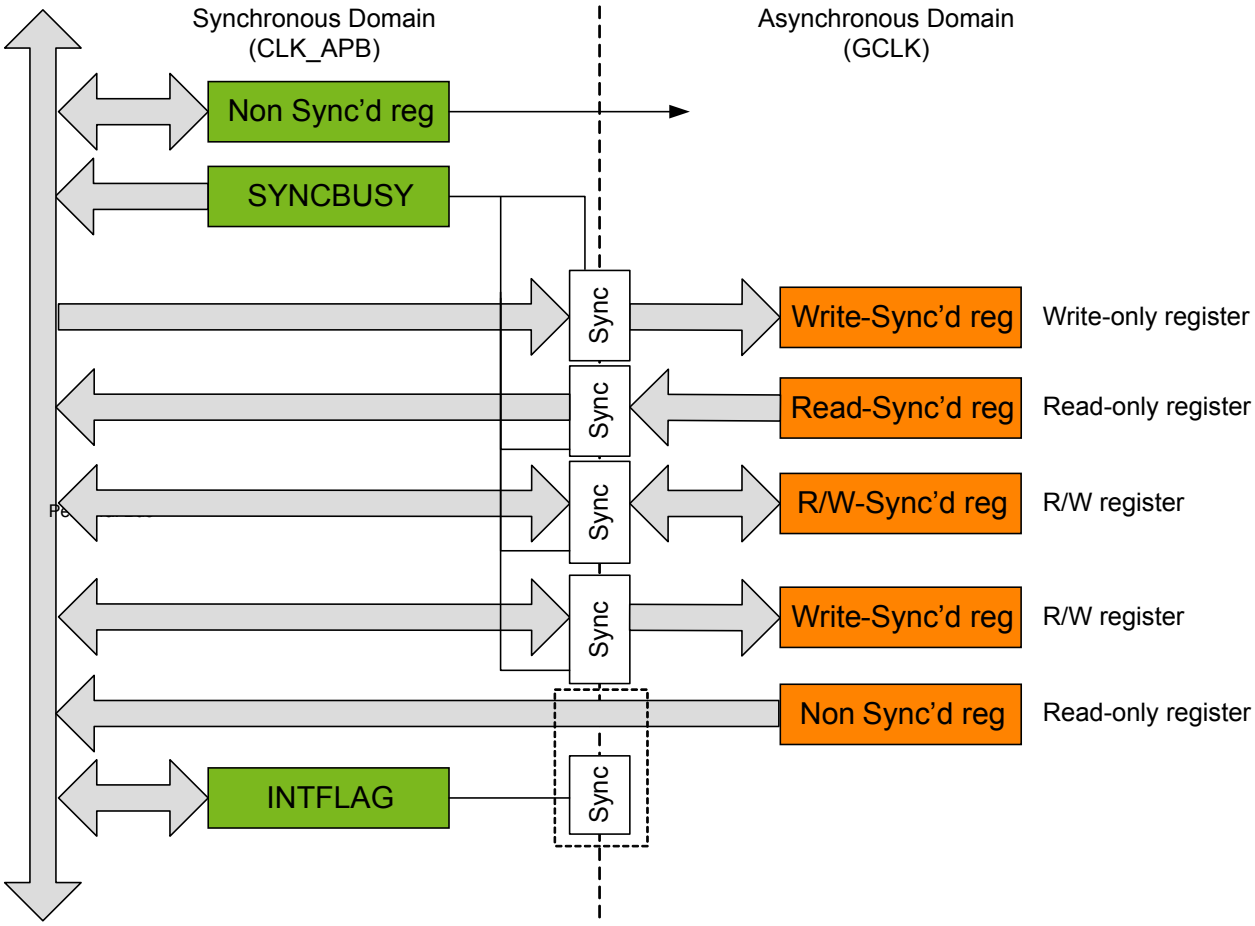
All registers in the bus interface are accessible without synchronization.

All registers in the peripheral core are synchronized when written. Some registers in the peripheral core are synchronized when read.

Each individual register description will have the properties "Read-Synchronized" and/or "Write-Synchronized" if a register is synchronized.

As shown in the figure below, each register that requires synchronization has its individual synchronizer and its individual synchronization status bit in the Synchronization Busy register (SYNCBUSY).
Note: For registers requiring both read- and write-synchronization, the corresponding bit in SYNCBUSY is shared.

Figure 16-3. Register Synchronization Overview



20.8.2. Sleep Configuration

Name: SLEEPCFG
Offset: 0x01
Reset: 0x2
Property: PAC Write-Protection

Bit	7	6	5	4	3	2	1	0
						SLEEPMODE[2:0]		
Access						R/W	R/W	R/W
Reset						0	0	0

Bits 2:0 – SLEEPMODE[2:0]: Sleep Mode
Note: A small latency happens between the store instruction and actual writing of the SLEEPCFG register due to bridges. Software has to make sure the SLEEPCFG register reads the wanted value before issuing WFI instruction.

Value	Name	Definition
0x0	Reserved	Reserved
0x1	Reserved	Reserved
0x2	IDLE	CPU, AHBx, and APBx clocks are OFF
0x3	Reserved	Reserved
0x4	STANDBY	ALL clocks are OFF, unless requested by sleepwalking peripheral
0x5	BACKUP	Only Backup domain is powered ON
0x6	OFF	All power domains are powered OFF
0x7	Reserved	Reserved

Bit 4 – OSC16MRDY: OSC16M Ready
This flag is cleared by writing '1' to it.

This flag is set on 0-to-1 transition of the OSC16M Ready bit in the Status register (STATUS.OSC16MRDY) and will generate an interrupt request if INTENSET.OSC16MRDY is '1'.

Writing '0' to this bit has no effect.

Writing '1' to this bit clears the OSC16M Ready interrupt flag.

Bit 0 – XOSCRDY: XOSC Ready
This flag is cleared by writing '1' to it.

This flag is set on a 0-to-1 transition of the XOSC Ready bit in the Status register (STATUS.XOSCRDY) and will generate an interrupt request if INTENSET.XOSCRDY is '1'.

Writing '0' to this bit has no effect.

Writing '1' to this bit clears the XOSC Ready interrupt flag.

21.8.15. DPLL Synchronization Busy

Name: DPLLSYNCBUSY
Offset: 0x38
Reset: 0x00
Property: –

Bit	7	6	5	4	3	2	1	0
					DPLLPRESC	DPLLRATIO	ENABLE	
Access					R	R	R	
Reset					0	0	0	

Bit 3 – DPLLPRESC: DPLL Prescaler Synchronization Status

Value	Description
0	The DPLLRESC register has been synchronized.
1	The DPLLRESC register value has changed and its synchronization is in progress.

Bit 2 – DPLLRATIO: DPLL Loop Divider Ratio Synchronization Status

Value	Description
0	The DPLLRATIO register has been synchronized.
1	The DPLLRATIO register value has changed and its synchronization is in progress.

Bit 1 – ENABLE: DPLL Enable Synchronization Status

Value	Description
0	The DPLLCTRLA.ENABLE bit has been synchronized.
1	The DPLLCTRLA.ENABLE bit value has changed and its synchronization is in progress.

Value	Name	Description
0xB	DIV1024	CLK_RTC_CNT = GCLK_RTC/1024
0xC-0xF	-	Reserved

Bit 7 – MATCHCLR: Clear on Match
This bit defines if the counter is cleared or not on a match.

This bit is not synchronized.

Value	Description
0	The counter is not cleared on a Compare/Alarm 0 match
1	The counter is cleared on a Compare/Alarm 0 match

Bits 3:2 – MODE[1:0]: Operating Mode
This bit group defines the operating mode of the RTC.

This bit is not synchronized.

Value	Name	Description
0x0	COUNT32	Mode 0: 32-bit counter
0x1	COUNT16	Mode 1: 16-bit counter
0x2	CLOCK	Mode 2: Clock/calendar
0x3	-	Reserved

Bit 1 – ENABLE: Enable
Due to synchronization there is a delay between writing CTRLA.ENABLE and until the peripheral is enabled/disabled. The value written to CTRLA.ENABLE will read back immediately and the Enable bit in the Synchronization Busy register (SYNCBUSY.ENABLE) will be set. SYNCBUSY.ENABLE will be cleared when the operation is complete.

Value	Description
0	The peripheral is disabled
1	The peripheral is enabled

Bit 0 – SWRST: Software Reset
Writing a '0' to this bit has no effect.

Writing a '1' to this bit resets all registers in the RTC (except DBGCTRL) to their initial state, and the RTC will be disabled.

Writing a '1' to CTRLA.SWRST will always take precedence, meaning that all other writes in the same write-operation will be discarded.

Due to synchronization there is a delay between writing CTRLA.SWRST and until the reset is complete. CTRLA.SWRST will be cleared when the reset is complete.

Value	Description
0	There is not reset operation ongoing
1	The reset operation is ongoing

Value	Name	Description
0x3		Reserved

several events are output to the same pin, the lowest event line will get the access. All other events will be ignored.

Related Links

EVSYS – Event System on page 544

29.6.5. PORT Access Priority

The PORT is accessed by different systems:

- The ARM® CPU through the ARM® single-cycle I/O port (IOBUS)
- The ARM® CPU through the high-speed matrix and the AHB/APB bridge (APB)
- EVSYS through four asynchronous input events

The following priority is adopted:

1. ARM® CPU IOBUS (No wait tolerated)
2. APB
3. EVSYS input events

For input events that require different actions on the same I/O pin, refer to Events.

29.8.2. Data Direction Clear

This register allows the user to set one or more I/O pins as an input, without doing a read-modify-write operation. Changes in this register will also be reflected in the Data Direction (DIR), Data Direction Toggle (DIRTGL) and Data Direction Set (DIRSET) registers.

Name: DIRCLR
Offset: 0x04
Reset: 0x00000000
Property: PAC Write-Protection

Bit	31	30	29	28	27	26	25	24
	DIRCLR[31:24]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	DIRCLR[23:16]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	DIRCLR[15:8]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	DIRCLR[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 31:0 – DIRCLR[31:0]: Port Data Direction Clear
Writing a '0' to a bit has no effect.

Writing a '1' to a bit will clear the corresponding bit in the DIR register, which configures the I/O pin as an input.

Value	Description
0	The corresponding I/O pin in the PORT group will keep its configuration.
1	The corresponding I/O pin in the PORT group is configured as input.

30.8.7. Channel

This register allows the user to configure channel n. To write to this register, do a single, 32-bit write of all the configuration data.

Name: CHANNELn
Offset: 0x20+n*0x4 [n=0..11]
Reset: 0x00000000
Property: PAC Write-Protection

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
	ONDEMAND	RUNSTDBY			EDGSEL[1:0]		PATH[1:0]	
Access	R/W	R/W			R/W	R/W	R/W	R/W
Reset	0	0			0	0	0	0

Bit	7	6	5	4	3	2	1	0
		EVGEN[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bit 15 – ONDEMAND: Generic Clock On Demand

Value	Description
0	Generic clock for a channel is always on, if the channel is configured and generic clock source is enabled.
1	Generic clock is requested on demand while an event is handled

Bit 14 – RUNSTDBY: Run in Standby

This bit is used to define the behavior during standby sleep mode.

Value	Description
0	The channel is disabled in standby sleep mode.
1	The channel is not stopped in standby sleep mode and depends on the CHANNEL.ONDEMAND

Bits 11:10 – EDGSEL[1:0]: Edge Detection Selection

These bits set the type of edge detection to be used on the channel.

These bits must be written to zero when using the asynchronous path.

Figure 31-3. Baud Rate Generator

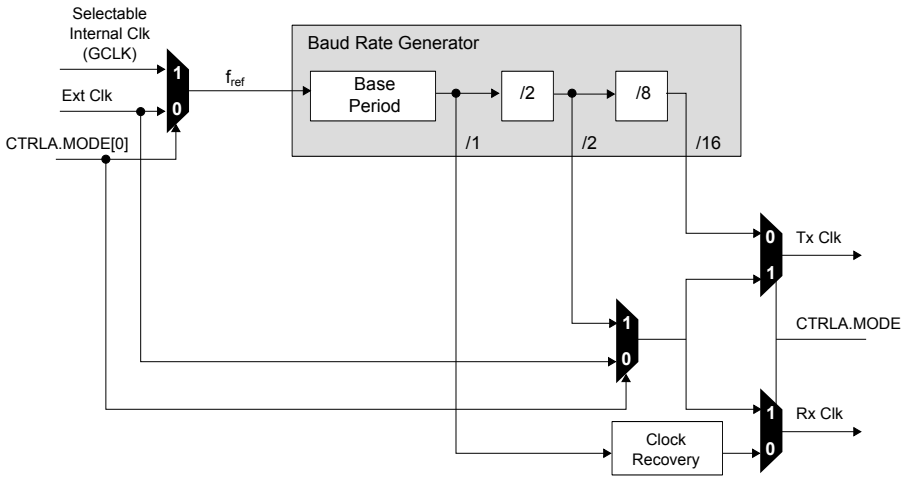


Table 31-2 contains equations for the baud rate (in bits per second) and the BAUD register value for each operating mode.

For asynchronous operation, there are two different modes: In *arithmetic mode*, the BAUD register value is 16 bits (0 to 65,535). In *fractional mode*, the BAUD register is 13 bits, while the fractional adjustment is 3 bits. In this mode the BAUD setting must be greater than or equal to 1.

For synchronous operation, the BAUD register value is 8 bits (0 to 255).

Table 31-2. Baud Rate Equations

Operating Mode	Condition	Baud Rate (Bits Per Second)	BAUD Register Value Calculation
Asynchronous Arithmetic	$f_{BAUD} \leq \frac{f_{ref}}{S}$	$f_{BAUD} = \frac{f_{ref}}{S}\left(1 - \frac{BAUD}{65536}\right)$	$BAUD = 65536 \cdot \left(1 - S \cdot \frac{f_{BAUD}}{f_{ref}}\right)$
Asynchronous Fractional	$f_{BAUD} \leq \frac{f_{ref}}{S}$	$f_{BAUD} = \frac{f_{ref}}{S \cdot \left(BAUD + \frac{FP}{8}\right)}$	$BAUD = \frac{f_{ref}}{S \cdot f_{BAUD}} - \frac{FP}{8}$
Synchronous	$f_{BAUD} \leq \frac{f_{ref}}{2}$	$f_{BAUD} = \frac{f_{ref}}{2 \cdot (BAUD + 1)}$	$BAUD = \frac{f_{ref}}{2 \cdot f_{BAUD}} - 1$

S - Number of samples per bit. Can be 16, 8, or 3.

The Asynchronous Fractional option is used for auto-baud detection.

The baud rate error is represented by the following formula:

$$\text{Error} = 1 - \left(\frac{\text{ExpectedBaudRate}}{\text{ActualBaudRate}}\right)$$

Asynchronous Arithmetic Mode BAUD Value Selection

The formula given for f_{BAUD} calculates the average frequency over 65536 f_{ref} cycles. Although the BAUD register can be set to any value between 0 and 65536, the actual average frequency of f_{BAUD} over a single frame is more granular. The BAUD register values that will affect the average frequency over a single frame lead to an integer increase in the cycles per frame (CPF)

$$CPF = \frac{f_{ref}}{f_{BAUD}}(D + S)$$

where

Offset	Name	Bit Pos.								
0x14	COUNT	7:0	COUNT[7:0]							
0x15		15:8	COUNT[15:8]							
0x16		23:16	COUNT[23:16]							
0x17		31:24	COUNT[31:24]							
0x18	Reserved									
0x19	Reserved									
0x1A	Reserved									
0x1B	Reserved									
0x1C	CC0	7:0	CC[7:0]							
0x1D		15:8	CC[15:8]							
0x1E		23:16	CC[23:16]							
0x1F		31:24	CC[31:24]							
0x20	CC1	7:0	CC[7:0]							
0x21		15:8	CC[15:8]							
0x22		23:16	CC[23:16]							
0x23		31:24	CC[31:24]							
0x24	Reserved									
0x25	Reserved									
0x26	Reserved									
0x27	Reserved									
0x28	Reserved									
0x29	Reserved									
0x2A	Reserved									
0x2B	Reserved									
0x2C	Reserved									
0x2D	Reserved									
0x2E	Reserved									
0x2F	Reserved									
0x30	CCBUF0	7:0	CCBUF[7:0]							
0x31		15:8	CCBUF[15:8]							
0x32		23:16	CCBUF[23:16]							
0x33		31:24	CCBUF[31:24]							
0x34	CCBUF1	7:0	CCBUF[7:0]							
0x35		15:8	CCBUF[15:8]							
0x36		23:16	CCBUF[23:16]							
0x37		31:24	CCBUF[31:24]							
0x38	Reserved									
0x39	Reserved									
0x3A	Reserved									
0x3B	Reserved									
0x3C	Reserved									
0x3D	Reserved									
0x3E	Reserved									
0x3F	Reserved									

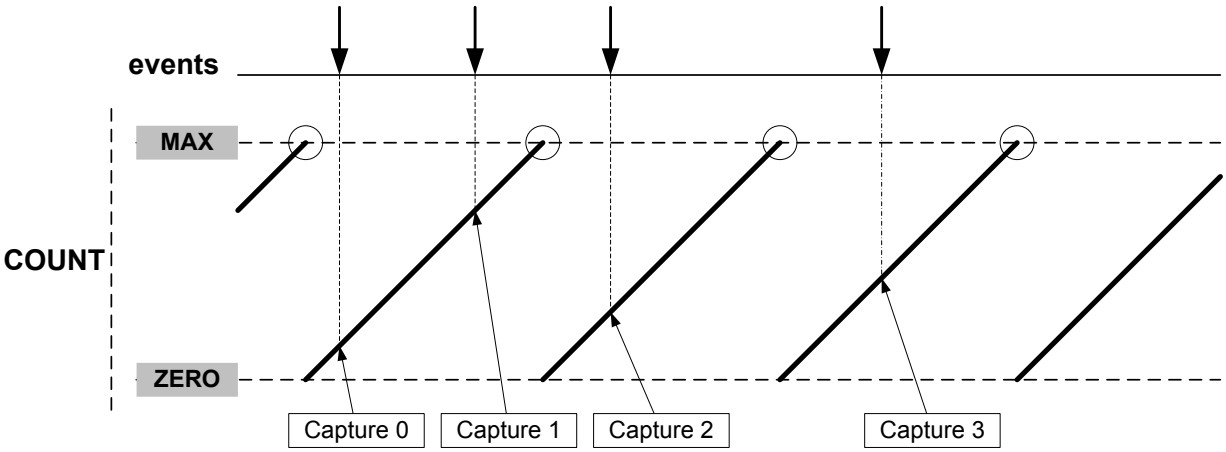
36.6.2.7. Capture Operations

To enable and use capture operations, the Match or Capture Channel x Event Input Enable bit in the Event Control register (EVCTRL.MCEIx) must be written to '1'. The capture channels to be used must also be enabled in the Capture Channel x Enable bit in the Control A register (CTRLA.CPTENx) before capturing can be performed.

Event Capture Action

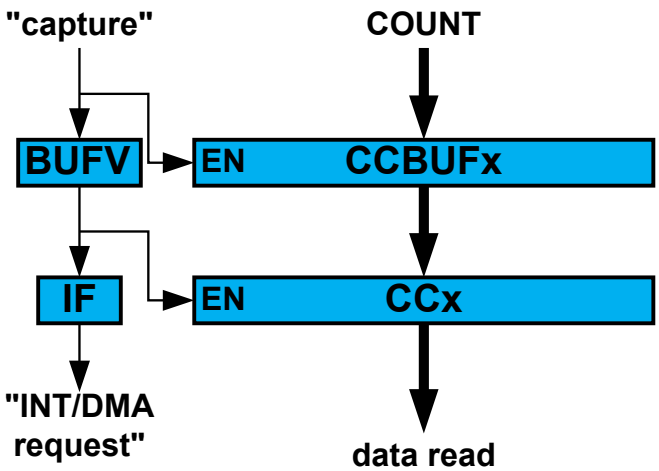
The compare/capture channels can be used as input capture channels to capture events from the Event System, and give them a timestamp. The following figure shows four capture events for one capture channel.

Figure 36-14. Input Capture Timing



For input capture, the buffer register and the corresponding CCx act like a FIFO. When CCx is empty or read, any content in CCBUFx is transferred to CCx. The buffer valid flag is passed to set the CCx interrupt flag (IF) and generate the optional interrupt, event or DMA request. CCBUFx register value can't be read, all captured data must be read from CCx register.

Figure 36-15. Capture Double Buffering



The TCC can detect capture overflow of the input capture channels: When a new capture event is detected while the Capture Buffer Valid flag (STATUS.CCBUFV) is still set, the new timestamp will not be stored and INTFLAG.ERR will be set.

Period and Pulse-Width (PPW) Capture Action

39.8.2.2. Device Address

Name: DADD
Offset: 0x0A
Reset: 0x00
Property: PAC Write-Protection

Bit	7	6	5	4	3	2	1	0
	ADDEN	DADD[6:0]						
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit 7 – ADDEN: Device Address Enable
This bit is cleared when a USB reset is received.

Value	Description
0	Writing a zero will deactivate the DADD field (USB device address) and return the device to default address 0.
1	Writing a one will activate the DADD field (USB device address).

Bits 6:0 – DADD[6:0]: Device Address
These bits define the device address. The DADD register is reset when a USB reset is received.

39.8.4.5. Device Status Bank

Name: STATUS_BK
Offset: 0x0A & 0x1A
Reset: 0xxxxxxxx
Property: NA

Bit	7	6	5	4	3	2	1	0
							ERROFLOW	CRCERR
Access							R/W	R/W
Reset							x	x

Bit 1 – ERROFLOW: Error Flow Status
This bit defines the Error Flow Status.

This bit is set when a Error Flow has been detected during transfer from/towards this bank.

For OUT transfer, a NAK handshake has been sent.

For Isochronous OUT transfer, an overrun condition has occurred.

For IN transfer, this bit is not valid. EPSTATUS.TRFAIL0 and EPSTATUS.TRFAIL1 should reflect the flow errors.

Value	Description
0	No Error Flow detected.
1	A Error Flow has been detected.

Bit 0 – CRCERR: CRC Error
This bit defines the CRC Error Status.

This bit is set when a CRC error has been detected in an isochronous OUT endpoint bank.

0.2.5 Host Registers - Common

Value	Description
0	No CRC Error.
1	CRC Error detected.

39.8.5. Host Registers - Common

39.8.6.6. Host Pipe Interrupt Flag Register

Name: PINTFLAG
Offset: 0x107 + (n x 0x20)
Reset: 0x0000
Property: -

Bit	7	6	5	4	3	2	1	0
			STALL	TXSTP	PERR	TRFAIL	TRCPT1	TRCPT0
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit 5 – STALL: STALL Received Interrupt Flag
This flag is cleared by writing a one to the flag.

This flag is set when a stall occurs and will generate an interrupt if PINTENCLR/SET.STALL is one.

Writing a zero to this bit has no effect.

Writing a one to this bit clears the STALL Interrupt Flag.

Bit 4 – TXSTP: Transmitted Setup Interrupt Flag
This flag is cleared by writing a one to the flag.

This flag is set when a Transfer Complete occurs and will generate an interrupt if PINTENCLR/SET.TXSTP is one.

Writing a zero to this bit has no effect.

Writing a one to this bit clears the TXSTP Interrupt Flag.

Bit 3 – PERR: Pipe Error Interrupt Flag
This flag is cleared by writing a one to the flag.

This flag is set when a pipe error occurs and will generate an interrupt if PINTENCLR/SET.PERR is one.

Writing a zero to this bit has no effect.

Writing a one to this bit clears the PERR Interrupt Flag.

Bit 2 – TRFAIL: Transfer Fail Interrupt Flag
This flag is cleared by writing a one to the flag.

This flag is set when a Transfer Fail occurs and will generate an interrupt if PINTENCLR/SET.TRFAIL is one.

Writing a zero to this bit has no effect.

Writing a one to this bit clears the TRFAIL Interrupt Flag.

Bit 1 – TRCPT1: Transfer Complete 1 interrupt Flag
This flag is cleared by writing a one to the flag.

This flag is set when a Transfer Complete occurs and will generate an interrupt if PINTENCLR/SET.TRCPT1 is one. PINTFLAG.TRCPT1 is set for a double bank IN/OUT pipe when current bank is 1.

Writing a zero to this bit has no effect.

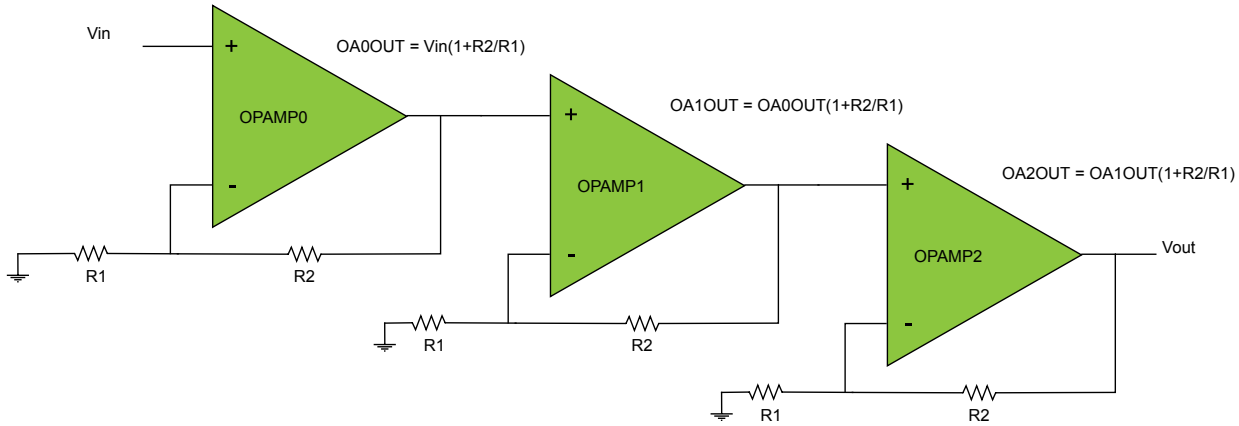
EVSYS – Event System on page 544

40.6.4. Sleep Mode Operation

When using the GCLK_CCL internal clocking, writing the Run In Standby bit in the Control register (CTRL.RUNSTDBY) to '1' will allow GCLK_CCL to be enabled in all sleep modes.

If CTRL.RUNSTDBY=0, the GCLK_CCL will be disabled. If the Filter, Edge Detector or Sequential logic are enabled, the LUT output will be forced to zero in STANDBY mode. In all other cases, the TRUTH table decoder will continue operation and the LUT output will be refreshed accordingly.

Figure 41-6. Cascaded Non-Inverting PGA



41.6.10.6. Two OPAMPs Differential Amplifier

In this mode, OPAMP0 can be coupled with OPAMP1 or OPAMP1 with OPAMP2 in order to amplify a differtial signal.

To configure OPAMP0 and OPAMP1 as differential amplifier, the OPAMPCTRLx register can be configured as follows:

Table 41-7. OPAMP0 OPAMP1 Differential Amplifier (Example: R1=4R, R2=12R)

	MUXPOS	MUXNEG	RES1MUX	POTMUX	RES2VCC	RES2OUT	RES1EN	ANAOUT
OPAMP0	000	010	00	000	0	0	0	0
OPAMP1	000	001	10	100	0	1	1	0
OPAMP2	000	000	00	000	0	0	0	0

To configure OPAMP1 and OPAMP2 as differential amplifier, the OPAMPCTRLx register can be configured as follows:

Table 41-8. OPAMP1 OPAMP2 Differential Amplifier (Example: R1=4R, R2=12R)

	MUXPOS	MUXNEG	RES1MUX	POTMUX	RES2VCC	RES2OUT	RES1EN	ANAOUT
OPAMP0	000	000	00	000	0	0	0	0
OPAMP1	000	010	00	000	0	1	0	0
OPAMP2	000	001	10	100	0	1	1	0

Symbol	Description	Conditions	Max.		Units
			PL0	PL2	
$f_{GCLK\_EVSYS\_CHANNEL\_0}$	EVSYS channel 0 input clock frequency	-	12	48	MHz
$f_{GCLK\_EVSYS\_CHANNEL\_1}$	EVSYS channel 1 input clock frequency	-			
$f_{GCLK\_EVSYS\_CHANNEL\_2}$	EVSYS channel 2 input clock frequency	-			
$f_{GCLK\_EVSYS\_CHANNEL\_3}$	EVSYS channel 3 input clock frequency	-			
$f_{GCLK\_EVSYS\_CHANNEL\_4}$	EVSYS channel 4 input clock frequency	-			
$f_{GCLK\_EVSYS\_CHANNEL\_5}$	EVSYS channel 5 input clock frequency	-			
$f_{GCLK\_EVSYS\_CHANNEL\_6}$	EVSYS channel 6 input clock frequency	-			
$f_{GCLK\_EVSYS\_CHANNEL\_7}$	EVSYS channel 7 input clock frequency	-			
$f_{GCLK\_EVSYS\_CHANNEL\_8}$	EVSYS channel 8 input clock frequency	-			
$f_{GCLK\_EVSYS\_CHANNEL\_9}$	EVSYS channel 9 input clock frequency	-			
$f_{GCLK\_EVSYS\_CHANNEL\_10}$	EVSYS channel 10 input clock frequency	-			
$f_{GCLK\_EVSYS\_CHANNEL\_11}$	EVSYS channel 11 input clock frequency	-			
$f_{GCLK\_SERCOMx\_SLOW}$	Common SERCOM slow input clock frequency	-	1	5	MHz
$f_{GCLK\_SERCOM0\_CORE}$	SERCOM0 input clock frequency	-	12	48	MHz
$f_{GCLK\_SERCOM1\_CORE}$	SERCOM1 input clock frequency	-			
$f_{GCLK\_SERCOM2\_CORE}$	SERCOM2 input clock frequency	-			
$f_{GCLK\_SERCOM3\_CORE}$	SERCOM3 input clock frequency	-			
$f_{GCLK\_SERCOM4\_CORE}$	SERCOM4 input clock frequency	-			
$f_{GCLK\_SERCOM5\_CORE}$	SERCOM5 input clock frequency	-			
$f_{GCLK\_TCC0,\ GCLK\_TCC1}$	TCC0,TCC1 input clock frequency	-	24	96	MHz
$f_{GCLK\_TCC2,\ GCLK\_TC0}$	TCC2,TC0 input clock frequency	-	12	48	MHz
$f_{GCLK\_TC1,\ GCLK\_TC2}$	TC1,TC2 input clock frequency	-			
$f_{GCLK\_TC3,\ GCLK\_TC4}$	TC3,TC4 input clock frequency	-			
f_{GCLK_ADC}	ADC input clock frequency	-	12	48	MHz
f_{GCLK_AC}	AC digital input clock frequency	-			
f_{GCLK_DAC}	DAC input clock frequency	-			
f_{GCLK_PTC}	PTC input clock frequency	-			
f_{GCLK_CCL}	CCL input clock frequency	-			

Note:

1. These values are based on simulation. They are not covered by production test limits or characterization.

Electrical Characteristics	• Voltage Regulator Characteristics – Table External Components Requirements in Switching Mode updated – Table External Components Requirements in Linear Mode updated – Table POR33 Characteristics updated – Table BOD33 Characteristics updated – Table BOD33 Power Consumption added • Analog-to-Digital Converter (ADC) Characteristics – Table Operating Conditions updated – Table Power Consumption added – Values updated, Min. and Max. added – Figure ADC Analog Input AINx added • Digital-to-Analog Converter (DAC) Characteristics – Table Operating Conditions updated – Values updated, Min. and Max. added – Table Power Consumption added • Analog Comparator (AC) Characteristics – Table Electrical and Timing updated – Values updated, Min. and Max. added – Table Power Consumption added • DETREF added • Temperature Sensor Characteristics: ADCTsAcc removed • OPAMP: Power Consumption added • NVM Characteristics: DSU Chip erase removed • External Reset Pin: section added • USB Characteristics: editorial update