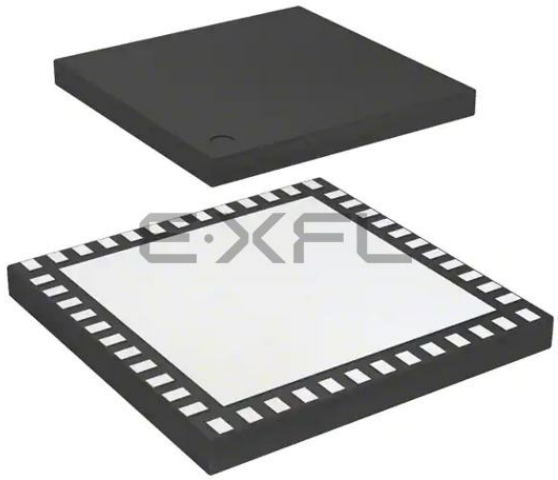


Welcome to E-XFL.COM

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	AVR
Core Size	32-Bit Single-Core
Speed	50MHz
Connectivity	I²C, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	36
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	32K x 8
Voltage - Supply (Vcc/Vdd)	1.62V ~ 3.6V
Data Converters	A/D 8x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-UFLGA Exposed Pad
Supplier Device Package	48-TLLGA (5.5x5.5)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atuc256l4u-d3hes

Table 3-1. GPIO Controller Function Multiplexing

39	51	PA17	17	VDDIO	Normal I/O (TWI)		TC0-A1	USART2-CTS	TWIMS1-TWD	PWMA-PWMA[17]	CAT-SMP	CAT-DIS	CAT-CSB[8]
41	53	PA18	18	VDDIO	Normal I/O	ADCIFB-AD[4]	TC0-B1		GLOC-IN[4]	PWMA-PWMA[18]	CAT-SYNC	EIC-EXTINT[5]	CAT-CSB[0]
40	52	PA19	19	VDDIO	Normal I/O	ADCIFB-AD[5]		TC0-A2	TWIMS1-TWALM	PWMA-PWMA[19]	SCIF-GCLK_IN[0]	CAT-SYNC	CAT-CSA[10]
25	33	PA20	20	VDDIN	Normal I/O	USART2-TXD		TC0-A1	GLOC-IN[3]	PWMA-PWMA[20]	SCIF-RC32OUT		CAT-CSA[12]
24	32	PA21	21	VDDIN	Normal I/O (TWI, 5V tolerant, SMBus)	USART2-RXD	TWIMS0-TWD	TC0-B1	ADCIFB-TRIGGER	PWMA-PWMA[21]	PWMA-PWMAOD[21]	SCIF-GCLK[0]	CAT-SMP
9	13	PA22	22	VDDIO	Normal I/O	USART0-CTS	USART2-CLK	TC0-B2	CAT-SMP	PWMA-PWMA[22]	ACIFB-ACBN[2]		CAT-CSB[10]
6	8	PB00	32	VDDIO	Normal I/O	USART3-TXD	ADCIFB-ADP[0]	SPI-NPCS[0]	TC0-A1	PWMA-PWMA[23]	ACIFB-ACAP[2]	TC1-A0	CAT-CSA[9]
	20	PB01	33	VDDIO	High-drive I/O	USART3-RXD	ADCIFB-ADP[1]	SPI-SCK	TC0-B1	PWMA-PWMA[24]		TC1-A1	CAT-CSB[9]
7	9	PB02	34	VDDIO	Normal I/O	USART3-RTS	USART3-CLK	SPI-MISO	TC0-A2	PWMA-PWMA[25]	ACIFB-ACAN[2]	SCIF-GCLK[1]	CAT-CSB[11]
8	10	PB03	35	VDDIO	Normal I/O	USART3-CTS	USART3-CLK	SPI-MOSI	TC0-B2	PWMA-PWMA[26]	ACIFB-ACBP[2]	TC1-A2	CAT-CSA[11]
21	29	PB04	36	VDDIN	Normal I/O (TWI, 5V tolerant, SMBus)	TC1-A0	USART1-RTS	USART1-CLK	TWIMS0-TWALM	PWMA-PWMA[27]	PWMA-PWMAOD[27]	TWIMS1-TWCK	CAT-CSA[14]
20	28	PB05	37	VDDIN	Normal I/O (TWI, 5V tolerant, SMBus)	TC1-B0	USART1-CTS	USART1-CLK	TWIMS0-TWCK	PWMA-PWMA[28]	PWMA-PWMAOD[28]	SCIF-GCLK[3]	CAT-CSB[14]
30	42	PB06	38	VDDIO	Normal I/O	TC1-A1	USART3-TXD	ADCIFB-AD[6]	GLOC-IN[2]	PWMA-PWMA[29]	ACIFB-ACAN[3]	EIC-NMI (EXTINT[0])	CAT-CSB[13]
31	43	PB07	39	VDDIO	Normal I/O	TC1-B1	USART3-RXD	ADCIFB-AD[7]	GLOC-IN[1]	PWMA-PWMA[30]	ACIFB-ACAP[3]	EIC-EXTINT[1]	CAT-CSA[13]
32	44	PB08	40	VDDIO	Normal I/O	TC1-A2	USART3-RTS	ADCIFB-AD[8]	GLOC-IN[0]	PWMA-PWMA[31]	CAT-SYNC	EIC-EXTINT[2]	CAT-CSB[12]
29	39	PB09	41	VDDIO	Normal I/O	TC1-B2	USART3-CTS	USART3-CLK		PWMA-PWMA[32]	ACIFB-ACBN[1]	EIC-EXTINT[3]	CAT-CSB[15]
23	31	PB10	42	VDDIN	Normal I/O	TC1-CLK0	USART1-TXD	USART3-CLK	GLOC-OUT[1]	PWMA-PWMA[33]	SCIF-GCLK_IN[1]	EIC-EXTINT[4]	CAT-CSB[16]
44	56	PB11	43	VDDIO	Normal I/O	TC1-CLK1	USART1-RXD		ADCIFB-TRIGGER	PWMA-PWMA[34]	CAT-VDIVEN	EIC-EXTINT[5]	CAT-CSA[16]
5	7	PB12	44	VDDIO	Normal I/O	TC1-CLK2		TWIMS1-TWALM	CAT-SYNC	PWMA-PWMA[35]	ACIFB-ACBP[3]	SCIF-GCLK[4]	CAT-CSA[15]
15	22	PB13	45	VDDIN	USB I/O	USBC-DM	USART3-TXD		TC1-A1	PWMA-PWMA[7]	ADCIFB-ADP[1]	SCIF-GCLK[5]	CAT-CSB[2]
16	23	PB14	46	VDDIN	USB I/O	USBC-DP	USART3-RXD		TC1-B1	PWMA-PWMA[24]		SCIF-GCLK[5]	CAT-CSB[9]

4.4 Programming Model

4.4.1 Register File Configuration

The AVR32UC register file is shown below.

Figure 4-3. The AVR32UC Register File

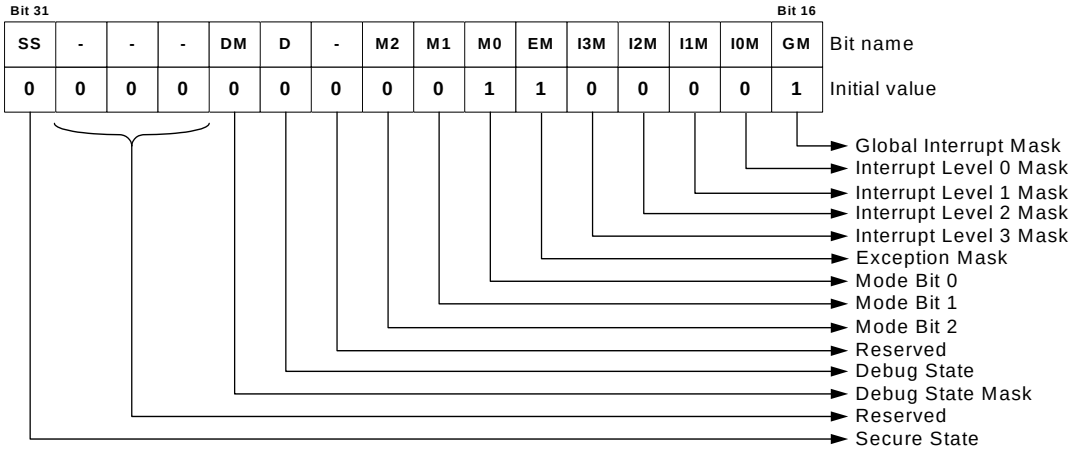
Application	Supervisor	INT0	INT1	INT2	INT3	Exception	NMI	Secure
Bit 31 Bit 0	Bit 31 Bit 0	Bit 31 Bit 0	Bit 31 Bit 0	Bit 31 Bit 0	Bit 31 Bit 0	Bit 31 Bit 0	Bit 31 Bit 0	Bit 31 Bit 0
PC	PC	PC	PC	PC	PC	PC	PC	PC
LR	LR	LR	LR	LR	LR	LR	LR	LR
SP_APP	SP_SYS	SP_SYS	SP_SYS	SP_SYS	SP_SYS	SP_SYS	SP_SYS	SP_SEC
R12	R12	R12	R12	R12	R12	R12	R12	R12
R11	R11	R11	R11	R11	R11	R11	R11	R11
R10	R10	R10	R10	R10	R10	R10	R10	R10
R9	R9	R9	R9	R9	R9	R9	R9	R9
R8	R8	R8	R8	R8	R8	R8	R8	R8
R7	R7	R7	R7	R7	R7	R7	R7	R7
R6	R6	R6	R6	R6	R6	R6	R6	R6
R5	R5	R5	R5	R5	R5	R5	R5	R5
R4	R4	R4	R4	R4	R4	R4	R4	R4
R3	R3	R3	R3	R3	R3	R3	R3	R3
R2	R2	R2	R2	R2	R2	R2	R2	R2
R1	R1	R1	R1	R1	R1	R1	R1	R1
R0	R0	R0	R0	R0	R0	R0	R0	R0
SR	SR	SR	SR	SR	SR	SR	SR	SR

SS_STATUS
SS_ADRF
SS_ADRR
SS_ADR0
SS_ADR1
SS_SP_SYS
SS_SP_APP
SS_RAR
SS_RSR

4.4.2 Status Register Configuration

The Status Register (SR) is split into two halfwords, one upper and one lower, see Figure 4-4 and Figure 4-5. The lower word contains the C, Z, N, V, and Q condition code flags and the R, T, and L bits, while the upper halfword contains information about the mode and state the processor executes in. Refer to the *AVR32 Architecture Manual* for details.

Figure 4-4. The Status Register High Halfword



5. Memories

5.1 Embedded Memories

- **Internal high-speed flash**
  - **256 Kbytes (ATUC256L3U, ATUC256L4U)**
  - **128 Kbytes (ATUC128L3U, ATUC128L4U)**
  - **64 Kbytes (ATUC64L3U, ATUC64L4U)**
    - **0 wait state access at up to 25 MHz in worst case conditions**
    - **1 wait state access at up to 50 MHz in worst case conditions**
    - **Pipelined flash architecture, allowing burst reads from sequential flash locations, hiding penalty of 1 wait state access**
    - **Pipelined flash architecture typically reduces the cycle penalty of 1 wait state operation to only 8% compared to 0 wait state operation**
    - **100 000 write cycles, 15-year data retention capability**
    - **Sector lock capabilities, bootloader protection, security bit**
    - **32 fuses, erased during chip erase**
    - **User page for data to be preserved during chip erase**
- **Internal high-speed SRAM, single-cycle access at full speed**
  - **32 Kbytes (ATUC256L3U, ATUC256L4U, ATUC128L3U, ATUC128L4U)**
  - **16 Kbytes (ATUC64L3U, ATUC64L4U)**

5.2 Physical Memory Map

The system bus is implemented as a bus matrix. All system bus addresses are fixed, and they are never remapped in any way, not even during boot. Note that AVR32 UC CPU uses unsegmented translation, as described in the AVR32 Architecture Manual. The 32-bit physical address space is mapped as follows:

Table 5-1. ATUC64/128/256L3/4U Physical Memory Map

Memory	Start Address	Size		
		ATUC256L3U, ATUC256L4U	ATUC128L3U, ATUC128L4U	ATUC64L3U, ATUC64L4U
Embedded SRAM	0x00000000	32 Kbytes	32 Kbytes	16 Kbytes
Embedded Flash	0x80000000	256 Kbytes	128 Kbytes	64 Kbytes
SAU Channels	0x90000000	256 bytes	256 bytes	256 bytes
HSB-PB Bridge B	0xFFFE0000	64 Kbytes	64 Kbytes	64 Kbytes
HSB-PB Bridge A	0xFFFF0000	64 Kbytes	64 Kbytes	64 Kbytes

Table 5-2. Flash Memory Parameters

Device	Flash Size (*FLASH_PW*)	Number of Pages (*FLASH_P*)	Page Size (*FLASH_W*)
ATUC256L3U, ATUC256L4U	256 Kbytes	512	512 bytes
ATUC128L3U, ATUC128L4U	128 Kbytes	256	512 bytes
ATUC64L3U, ATUC64L4U	64 Kbytes	128	512 bytes

8.7.1.8 IP Name Register 1

Register Name: UNAME1

Access Type: Read-Only

Offset: 0x0824

Reset Value: -

31	30	29	28	27	26	25	24
UNAME1[31:24]							

23	22	21	20	19	18	17	16
UNAME1[23:16]							

15	14	13	12	11	10	9	8
UNAME1[15:8]							

7	6	5	4	3	2	1	0
UNAME1[7:0]							

- **UNAME1: IP Name Part One**
  This field indicates the first part of the ASCII-encoded name of the USBC IP.

8.7.2 USB Device Registers

8.7.2.1 Device General Control Register

Register Name: UDCON

Access Type: Read/Write

Offset: 0x0000

Reset Value: 0x00000100

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

- **GNAK: Global NAK**
  0: Normal mode.
  1: A NAK handshake is answered for each USB transaction regardless of the current endpoint memory bank status.
- **LS: low-speed mode force**
  0: The full-speed mode is active.
  1: The low-speed mode is active.
  This bit can be written to even if USBE is zero or FRZCLK is one. Disabling the USBC (by writing a zero to the USBE bit) does not reset this bit.
- **RMWKUP: Remote wakeup**
  Writing a zero to this bit has no effect.
  Writing a one to this bit will send an upstream resume to the host for a remote wakeup.
  This bit is cleared when the USBC receives a USB reset or once the upstream resume has been sent.
- **DETACH: Detach**
  Writing a zero to this bit will reconnect the device.
  Writing a one to this bit will physically detach the device (disconnect internal pull-up resistor from DP and DM).
- **ADDEN: Address Enable**
  Writing a zero to this bit has no effect.
  Writing a one to this bit will activate the UADD field (USB address).
  This bit is cleared when a USB reset is received.
- **UADD: USB Address**
  This field contains the device address.
  This field is cleared when a USB reset is received.

10.6.5 Status Register

Name: SR

Access Type: Read-only

Offset: 0x10

Reset Value: 0x00000400

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

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

15	14	13	12	11	10	9	8
-	-	-	-	-	IDLE	SEN	EN

7	6	5	4	3	2	1	0
RTRADR	MBERROR	URES	URKEY	URREAD	CAU	CAS	EXP

- **IDLE**
  This bit is cleared when a read or write operation to the SAU channel is started.
  This bit is set when the operation is completed and no SAU bus operations are pending.
- **SEN: SAU Setup Mode Enable**
  This bit is cleared when the SAU exits setup mode.
  This bit is set when the SAU enters setup mode.
- **EN: SAU Enabled**
  This bit is cleared when the SAU is disabled.
  This bit is set when the SAU is enabled.
- **RTRADR: RTR Address Error**
  This bit is cleared when the corresponding bit in ICR is written to one.
  This bit is set if, in the configuration phase, an RTR was written with an illegal address, i.e. the upper 16 bits in the address were different from 0xFFFC, 0xFFFD, 0xFFFE or 0xFFFF.
- **MBERROR: Master Interface Bus Error**
  This bit is cleared when the corresponding bit in ICR is written to one.
  This bit is set if a channel access generated a transfer on the master interface that received a bus error response from the addressed slave.
- **URES: Unlock Register Error Status**
  This bit is cleared when the corresponding bit in ICR is written to one.
  This bit is set if an attempt was made to unlock a channel by writing to the Unlock Register while one or more error bits were set in SR. The unlock operation was aborted.
- **URKEY: Unlock Register Key Error**
  This bit is cleared when the corresponding bit in ICR is written to one.
  This bit is set if the Unlock Register was attempted written with an invalid key.
- **URREAD: Unlock Register Read**
  This bit is cleared when the corresponding bit in ICR is written to one.
  This bit is set if the Unlock Register was read.

12. Interrupt Controller (INTC)

Rev: 1.0.2.5

12.1 Features

- **Autovectored low latency interrupt service with programmable priority**
  - **4 priority levels for regular, maskable interrupts**
  - **One Non-Maskable Interrupt**
- **Up to 64 groups of interrupts with up to 32 interrupt requests in each group**

12.2 Overview

The INTC collects interrupt requests from the peripherals, prioritizes them, and delivers an interrupt request and an autovector to the CPU. The AVR32 architecture supports 4 priority levels for regular, maskable interrupts, and a Non-Maskable Interrupt (NMI).

The INTC supports up to 64 groups of interrupts. Each group can have up to 32 interrupt request lines, these lines are connected to the peripherals. Each group has an Interrupt Priority Register (IPR) and an Interrupt Request Register (IRR). The IPRs are used to assign a priority level and an autovector to each group, and the IRRs are used to identify the active interrupt request within each group. If a group has only one interrupt request line, an active interrupt group uniquely identifies the active interrupt request line, and the corresponding IRR is not needed. The INTC also provides one Interrupt Cause Register (ICR) per priority level. These registers identify the group that has a pending interrupt of the corresponding priority level. If several groups have a pending interrupt of the same level, the group with the lowest number takes priority.

12.3 Block Diagram

Figure 12-1 gives an overview of the INTC. The grey boxes represent registers that can be accessed via the user interface. The interrupt requests from the peripherals (IREQn) and the NMI are input on the left side of the figure. Signals to and from the CPU are on the right side of the figure.

13.7.3 HSB Clock Select

Name: HSBSEL

Access Type: Read

Offset: 0x008

Reset Value: 0x00000000

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

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

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

7	6	5	4	3	2	1	0
HSBDIV	-	-	-	-	HSBSEL		

This register is read-only and its content is always equal to CPUSEL.

13.7.9 Interrupt Enable Register

Name: IER

Access Type: Write-only

Offset: 0x0C0

Reset Value: 0x00000000

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

Writing a zero to a bit in this register has no effect.
Writing a one to a bit in this register will set the corresponding bit in IMR.

15.6 User Interface

Table 15-1. AST Register Memory Map

Offset	Register	Register Name	Access	Reset
0x00	Control Register	CR	Read/Write	0x00000000
0x04	Counter Value	CV	Read/Write	0x00000000
0x08	Status Register	SR	Read-only	0x00000000
0x0C	Status Clear Register	SCR	Write-only	0x00000000
0x10	Interrupt Enable Register	IER	Write-only	0x00000000
0x14	Interrupt Disable Register	IDR	Write-only	0x00000000
0x18	Interrupt Mask Register	IMR	Read-only	0x00000000
0x1C	Wake Enable Register	WER	Read/write	0x00000000
0x20	Alarm Register 0(2)	AR0	Read/Write	0x00000000
0x24	Alarm Register 1(2)	AR1	Read/Write	0x00000000
0x30	Periodic Interval Register 0(2)	PIR0	Read/Write	0x00000000
0x34	Periodic Interval Register 1(2)	PIR1	Read/Write	0x00000000
0x40	Clock Control Register	CLOCK	Read/Write	0x00000000
0x44	Digital Tuner Register	DTR	Read/Write	0x00000000
0x48	Event Enable	EVE	Write-only	0x00000000
0x4C	Event Disable	EVD	Write-only	0x00000000
0x50	Event Mask	EVM	Read-only	0x00000000
0x54	Calendar Value	CALV	Read/Write	0x00000000
0xF0	Parameter Register	PARAMETER	Read-only	-(1)
0xFC	Version Register	VERSION	Read-only	-(1)

Note: 1. The reset values are device specific. Please refer to the Module Configuration section at the end of this chapter.

2. The number of Alarm and Periodic Interval registers are device specific. Please refer to the Module Configuration section at the end of this chapter.

18.6.7 Interrupt Mask Register

Name: IMR

Access Type: Read-only

Offset: 0x018

Reset Value: 0x00000000

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

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

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

7	6	5	4	3	2	1	0
-	-	-	-	-	-	RCLKRDY	DONE

0: The corresponding interrupt is disabled.

1: The corresponding interrupt is enabled.

A bit in this register is cleared when the corresponding bit in IDR is written to one.

A bit in this register is set when the corresponding bit in IER is written to one.

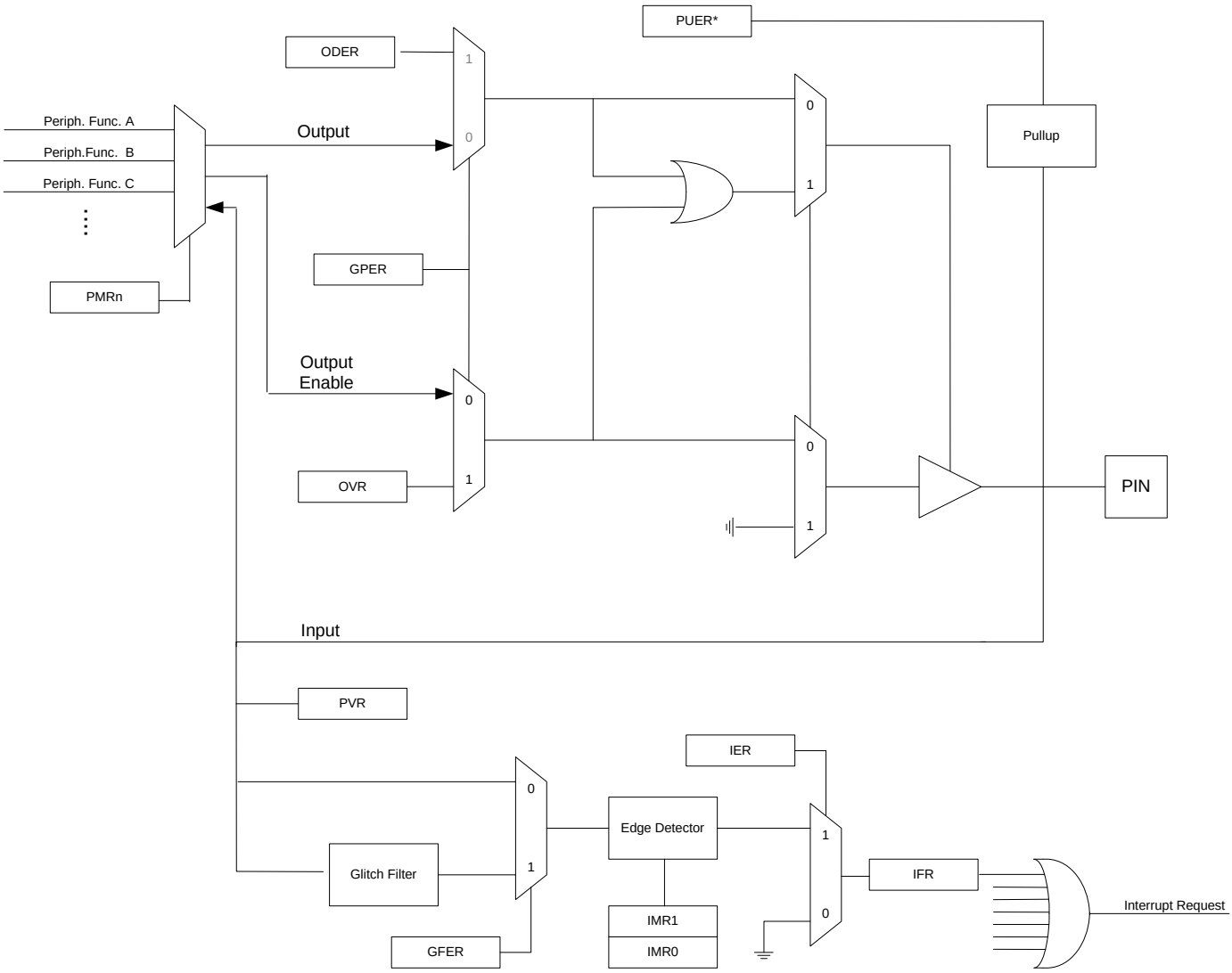
Table 18-5. Clock Sources for CLK_REF

REFSEL	Clock/Oscillator	Description
1	OSC32K	Output clock form OSC32K
2	GCLK9	Generic clock 9
3-7	Reserved	

19.6 Functional Description

The GPIO controls the I/O pins of the microcontroller. The control logic associated with each pin is shown in the figure below.

Figure 19-2. Overview of the GPIO



*) Register value is overrided if a peripheral function that support this function is enabled

- **CHRL: Character Length.**

Table 20-14.

CHRL		Character Length
0	0	5 bits
0	1	6 bits
1	0	7 bits
1	1	8 bits

- **USCLKS: Clock Selection**

Table 20-15.

USCLKS		Selected Clock
0	0	CLK_USART
0	1	CLK_USART/DIV(1)
1	0	Reserved
1	1	CLK

Note: 1. The value of DIV is device dependent. Please refer to the Module Configuration section at the end of this chapter.

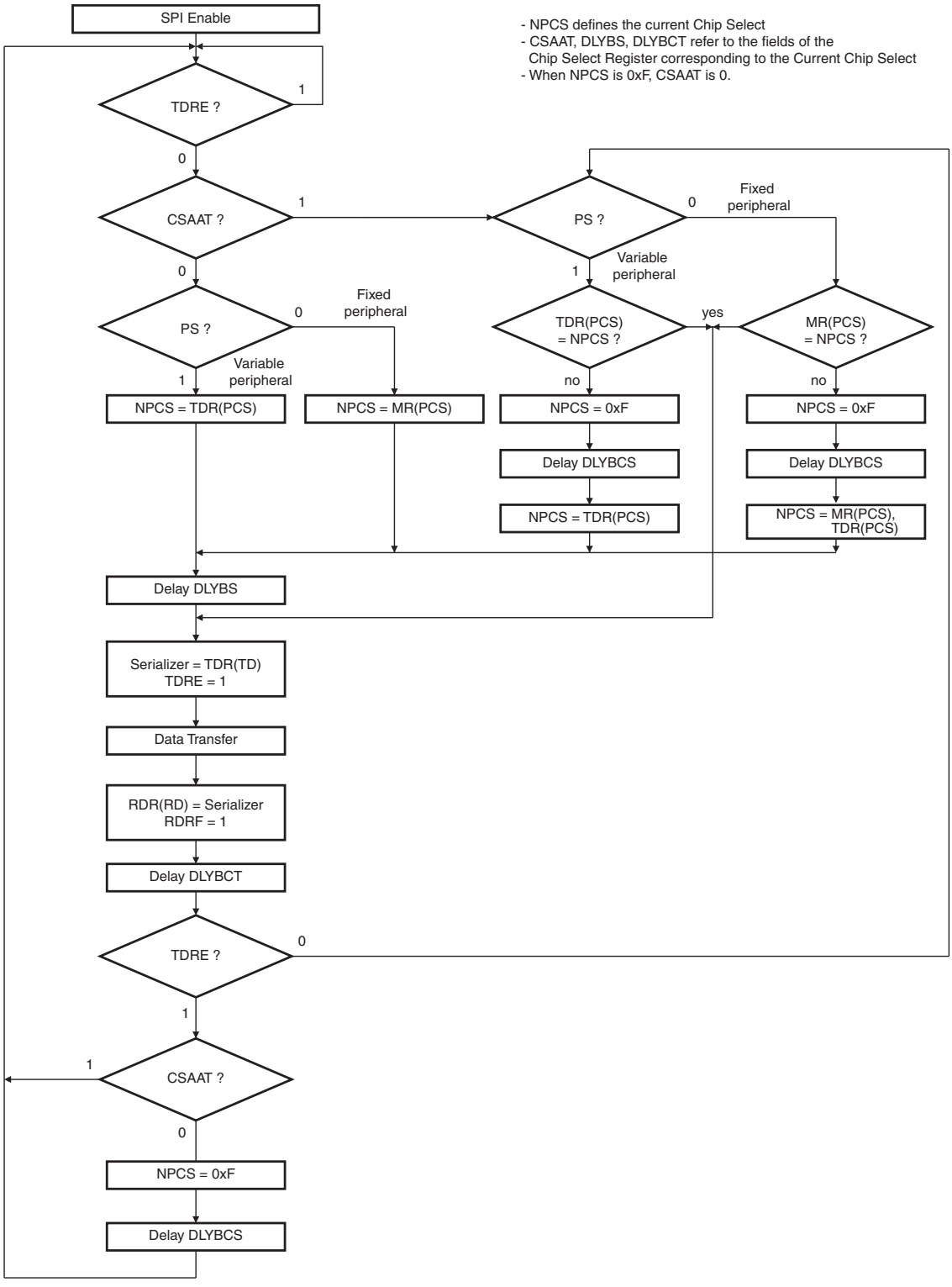
- **MODE**

Table 20-16.

MODE				Mode of the USART
0	0	0	0	Normal
0	0	1	0	Hardware Handshaking
1	0	1	0	LIN Master
1	0	1	1	LIN Slave
1	1	1	0	SPI Master
1	1	1	1	SPI Slave
Others				Reserved

21.7.3.2 *Master mode flow diagram*

Figure 21-6. Master Mode Flow Diagram



29.9.6 Status Register

Name: SR

Access Type: Read-only

Offset: 0x14

Reset Value: 0x00000000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	EN

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

15	14	13	12	11	10	9	8
-	CELSE	CGT	CLT	-	-	BUSY	READY

7	6	5	4	3	2	1	0
-	-	NOCNT	PENCNT	-	-	OVRE	DRDY

- **EN: Enable Status**
  0: The ADCIFB is disabled.
  1: The ADCIFB is enabled.
  This bit is cleared when CR.DIS is written to one.
  This bit is set when CR.EN is written to one.
- **CELSE: Compare Else Status**
  This bit is cleared when either CLT or CGT are detected or when analog compare is disabled.
  This bit is set when no CLT or CGT are detected on the last converted data and analog compare is enabled.
- **CGT: Compare Greater Than Status**
  This bit is cleared when no compare greater than CVR.HV is detected on the last converted data or when analog compare is disabled.
  This bit is set when compare greater than CVR.HV is detected on the last converted data and analog compare is enabled.
- **CLT: Compare Lesser Than Status**
  This bit is cleared when no compare lesser than CVR.LV is detected on the last converted data or when analog compare is disabled.
  This bit is set when compare lesser than CVR.LV is detected on the last converted data and analog compare is enabled.
- **BUSY: Busy Status**
  This bit is cleared when the ADCIFB is ready to perform a conversion sequence.
  This bit is set when the ADCIFB is busy performing a convention sequence.
- **READY: Ready Status**
  This bit is cleared when the ADCIFB is busy performing a conversion sequence
  This bit is set when the ADCIFB is ready to perform a conversion sequence.
- **NOCNT: No Contact Status**
  This bit is cleared when no contact loss is detected or pen detect is disabled
  This bit is set when contact loss is detected and pen detect is enabled.
- **PENCNT: Pen Contact Status**
  This bit is cleared when no contact is detected or pen detect is disabled.

29.10 Module Configuration

The specific configuration for each ADCIFB instance is listed in the following tables.The module bus clocks listed here are connected to the system bus clocks. Please refer to the Power Manager chapter for details.

Table 29-5. Module Configuration

Feature	ADCIFB
Number of ADC channels	9 (8 + 1 internal temperature sensor channel)

Table 29-6. ADCIFB Clocks

Clock Name	Description
CLK_ADCIFB	Clock for the ADCIFB bus interface

Table 29-7. Register Reset Values

Register	Reset Value
VERSION	0x00000110
PARAMETER	0x000003FF

Table 29-8. ADC Input Channels[1]

Channel	Input
CH0	AD0
CH1	AD1
CH2	AD2
CH4	AD4
CH5	AD5
CH6	AD6
CH7	AD7
CH8	AD8
CH9	Temperature sensor

Note: 1. AD3 does not exist

Table 31-1. I/O Lines Description

Name	Description	Type
SMP	SMP line (only used for QMatrix)	Output
SYNC	Synchronize signal	Input
VDIVEN	Voltage divider enable (only used for QMatrix)	Output

31.5 Product Dependencies

In order to use the CAT module, other parts of the system must be configured correctly, as described below.

31.5.1 I/O Lines

The CAT pins may be multiplexed with other peripherals. The user must first program the I/O Controller to give control of the pins to the CAT module. In QMatrix mode, the Y lines must be driven by the CAT and analog comparators sense the voltage on the Y lines. Thus, the CAT (not the Analog Comparator Interface) must be the selected function for the Y lines in the I/O Controller.

By writing ones and zeros to bits in the Pin Mode Registers (PINMODEx), most of the CAT pins can be individually selected to implement the QTouch method or the QMatrix method. Each pin has a different name and function depending on whether it is implementing the QTouch method or the QMatrix method. The following table shows the pin names for each method and the bits in the PINMODEx registers which control the selection of the QTouch or QMatrix method.

Table 31-2. Pin Selection Guide

CAT Module Pin Name	QTouch Method Pin Name	QMatrix Method Pin Name	Selection Bit in PINMODEx Register
CSA0	SNS0	X0	SP0
CSB0	SNSK0	X1	SP0
CSA1	SNS1	Y0	SP1
CSB1	SNSK1	YK0	SP1
CSA2	SNS2	X2	SP2
CSB2	SNSK2	X3	SP2
CSA3	SNS3	Y1	SP3
CSB3	SNSK3	YK1	SP3
CSA4	SNS4	X4	SP4
CSB4	SNSK4	X5	SP4
CSA5	SNS5	Y2	SP5
CSB5	SNSK5	YK2	SP5
CSA6	SNS6	X6	SP6
CSB6	SNSK6	X7	SP6
CSA7	SNS7	Y3	SP7
CSB7	SNSK7	YK3	SP7
CSA8	SNS8	X8	SP8

34.6.8 aWire Response Summary

The implemented aWire responses are shown in the table below.

Table 34-48. aWire Response Summary

RESPONSE	Instruction	Description
0x40	ACK	Acknowledge.
0x41	NACK	Not acknowledge. Sent after CRC errors and after unknown commands.
0xC0	IDCODE	The JTAG idcode.
0xC1	MEMDATA	Values read from memory.
0xC2	MEMORY_READWRITE_STATUS	Status after a MEMORY_WRITE or a MEMORY_READ command. OK, busy, error.
0xC3	BAUD_RATE	The current baudrate.
0xC4	STATUS_INFO	Status information.
0xC5	MEMORY_SPEED	SAB to aWire speed information.

34.6.8.1 ACK

The AW has received the command successfully and performed the operation.

Table 34-49. ACK Details

Response	Details
Response value	0x40
Additional data	N/A

34.6.8.2 NACK

The AW has received the command, but got a CRC mismatch.

Table 34-50. NACK Details

Response	Details
Response value	0x41
Additional data	N/A

34.6.8.3 IDCODE

The JTAG idcode for this device.

Table 34-51. IDCODE Details

Response	Details
Response value	0xC0
Additional data	JTAG idcode

34.6.8.4 MEMDATA

The data read from the address specified by the MEMORY_READ command. The last 3 bytes are status bytes from the read. The first status byte is the status of the command described in the table below. The last 2 bytes are the number of remaining data bytes to be sent in the data field of the packet when the error occurred. If the read was not successful all data bytes after the failure are undefined. A successful word read (4 bytes) will look like this:

35.10 Timing Characteristics

35.10.1 Startup, Reset, and Wake-up Timing

The startup, reset, and wake-up timings are calculated using the following formula:

$$t = t_{CONST} + N_{CPU} \times t_{CPU}$$

Where t_{CONST} and N_{CPU} are found in Table 35-39. t_{CPU} is the period of the CPU clock. If a clock source other than RCSYS is selected as the CPU clock, the oscillator startup time, $t_{OSCSTART}$, must be added to the wake-up time from the stop, deepstop, and static sleep modes. Please refer to the source for the CPU clock in the "Oscillator Characteristics" on page 905 for more details about oscillator startup times.

Table 35-39. Maximum Reset and Wake-up Timing[1]

Parameter		Measuring	Max t_{CONST} (in µs)	Max N_{CPU}
Startup time from power-up, using regulator		Time from VDDIN crossing the V_{POT+} threshold of POR33 to the first instruction entering the decode stage of CPU. VDDCORE is supplied by the internal regulator.	2210	0
Startup time from power-up, no regulator		Time from VDDIN crossing the V_{POT+} threshold of POR33 to the first instruction entering the decode stage of CPU. VDDCORE is connected to VDDIN.	1810	0
Startup time from reset release		Time from releasing a reset source (except POR18, POR33, and SM33) to the first instruction entering the decode stage of CPU.	170	0
Wake-up	Idle	From wake-up event to the first instruction of an interrupt routine entering the decode stage of the CPU.	0	19
	Frozen		0	110
	Standby		0	110
	Stop		27 + $t_{OSCSTART}$	116
	Deepstop		27 + $t_{OSCSTART}$	116
	Static		97 + $t_{OSCSTART}$	116
Wake-up from shutdown		From wake-up event to the first instruction entering the decode stage of the CPU.	1180	0

Note: 1. These values are based on simulation and characterization of other AVR microcontrollers manufactured in the same process technology. These values are not covered by test limits in production.

35.10.2 RESET_N Timing

Table 35-40. RESET_N Waveform Parameters[1]

Symbol	Parameter	Conditions	Min	Max	Units
t_{RESET}	RESET_N minimum pulse length		10		ns

Note: 1. These values are based on simulation and characterization of other AVR microcontrollers manufactured in the same process technology. These values are not covered by test limits in production.