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	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	16K 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-TQFP
Supplier Device Package	48-TQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/atmel/atuc64l4u-aut

8.7.1.7 Address Size Register

Register Name: UADDRSIZE

Access Type: Read-Only

Offset: 0x0820

Reset Value: -

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

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

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

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

- **UADDRSIZE: IP PB Address Size**

  This field indicates the size of the PB address space reserved for the USBC IP interface.

This bit is set, for control endpoints, when the current bank contains a bulk OUT packet (data or status stage). This triggers an EPnINT interrupt if RXOUTE is one.

This bit is set for isochronous, bulk and, interrupt OUT endpoints, at the same time as FIFOCON when the current bank is full. This triggers an EPnINT interrupt if RXOUTE is one.

This bit is inactive (cleared) for isochronous, bulk and interrupt IN endpoints.

- **TXINI: Transmitted IN Data Interrupt**

This bit is cleared when the TXINIC bit is written to one, acknowledging the interrupt. For control endpoints, this will send the packet. For other endpoint types, the user should clear the FIFOCON to allow the USBC to send the data. TXINI shall always be cleared before clearing FIFOCON to avoid missing an interrupt.

This bit is set for control endpoints, when the current bank is ready to accept a new IN packet. This triggers an EPnINT interrupt if TXINE is one.

This bit is set for isochronous, bulk and interrupt IN endpoints, at the same time as FIFOCON when the current bank is free. This triggers an EPnINT interrupt if TXINE is one.

This bit is inactive (cleared) for isochronous, bulk and interrupt OUT endpoints.

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.

14.6.39 3.3V Supply Monitor Version Register

Name: SM33IFAVERSION

Access Type: Read-only

Offset: 0x03E0

Reset Value: -

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

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

15	14	13	12	11	10	9	8
-	-	-	-	VERSION[11:8]			

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

- **VARIANT: Variant number**
  Reserved. No functionality associated.
- **VERSION: Version number**
  Version number of the module. No functionality associated.

16.6.2 Clear Register

Name: CLR

Access Type: Write-only

Offset: 0x004

Reset Value: 0x00000000

31	30	29	28	27	26	25	24
KEY							

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

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

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	WDTCLR

When the Watchdog Timer is enabled, this Register must be periodically written within the window time frame or within the watchdog timeout period, to prevent a watchdog reset.

- **KEY**

  This field must be written twice, first with key value 0x55, then 0xAA, for a write operation to be effective.

- **WDTCLR: Watchdog Clear**

  Writing a zero to this bit has no effect.

  Writing a one to this bit clears the WDT counter.

17.7.12 Enable Register

Name: EN

Access Type: Write-only

Offset: 0x030

Reset Value: 0x00000000

31	30	29	28	27	26	25	24
-	INT30	INT29	INT28	INT27	INT26	INT25	INT24

23	22	21	20	19	18	17	16
INT23	INT22	INT21	INT20	INT19	INT18	INT17	INT16

15	14	13	12	11	10	9	8
INT15	INT14	INT13	INT12	INT11	INT10	INT9	INT8

7	6	5	4	3	2	1	0
INT7	INT6	INT5	INT4	INT3	INT2	INT1	NMI

- **INTn: External Interrupt n**

  Writing a zero to this bit has no effect.

  Writing a one to this bit will enable the corresponding external interrupt.

  Please refer to the Module Configuration section for the number of external interrupts.

- **NMI: Non-Maskable Interrupt**

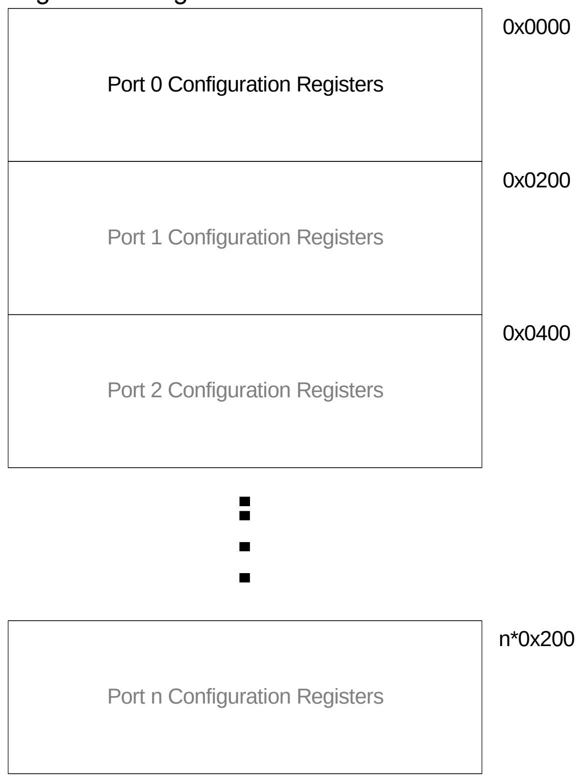
  Writing a zero to this bit has no effect.

  Writing a one to this bit will enable the Non-Maskable Interrupt.

19.7 User Interface

The GPIO controller manages all the GPIO pins on the 32-bit AVR microcontroller. The pins are managed as 32-bit ports that are configurable through a Peripheral Bus (PB) interface. Each port has a set of configuration registers. The overall memory map of the GPIO is shown below. The number of pins and hence the number of ports is product specific.

Figure 19-6. Port Configuration Registers



In the peripheral muxing table in the Package and Pinout chapter each GPIO pin has a unique number. Note that the PA, PB, PC, and PX ports do not necessarily directly correspond to the GPIO ports. To find the corresponding port and pin the following formulas can be used:

GPIO port = floor((GPIO number) / 32), example: floor((36)/32) = 1

GPIO pin = GPIO number % 32, example: 36 % 32 = 4

Table 19-2 shows the configuration registers for one port. Addresses shown are relative to the port address offset. The specific address of a configuration register is found by adding the register offset and the port offset to the GPIO start address. One bit in each of the configuration registers corresponds to a GPIO pin.

19.7.1 Access Types

Most configuration register can be accessed in four different ways. The first address location can be used to write the register directly. This address can also be used to read the register value. The following addresses facilitate three different types of write access to the register. Performing a “set” access, all bits written to one will be set. Bits written to zero will be unchanged by the operation. Performing a “clear” access, all bits written to one will be cleared. Bits written to zero will be unchanged by the operation. Finally, a toggle access will toggle the value of all bits writ-

19.7.18 Lock Register

Name: LOCK

Access: Read/Write, Set, Clear, Toggle

Offset: 0x1A0, 0x1A4, 0x1A8, 0x1AC

Reset Value: -

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

- **P0-31: Lock State**

  0: Pin is unlocked. The corresponding bit can be changed in any GPIO register for this port.

  1: Pin is locked. The corresponding bit can not be changed in any GPIO register for this port.

  The value of LOCK determines which bits are locked in the lockable registers.

  The LOCK, LOCKC, and LOCKT registers are protected, which means they can only be written immediately after a write to the UNLOCK register with the proper KEY and OFFSET.

  LOCKS is not protected, and can be written at any time.

20. Universal Synchronous Asynchronous Receiver Transmitter (USART)

Rev: 4.4.0.6

20.1 Features

- **Configurable baud rate generator**
- **5- to 9-bit full-duplex, synchronous and asynchronous, serial communication**
  - **1, 1.5, or 2 stop bits in asynchronous mode, and 1 or 2 in synchronous mode**
  - **Parity generation and error detection**
  - **Framing- and overrun error detection**
  - **MSB- or LSB-first**
  - **Optional break generation and detection**
  - **Receiver frequency over-sampling by 8 or 16 times**
  - **Optional RTS-CTS hardware handshaking**
  - **Receiver Time-out and transmitter Timeguard**
  - **Optional Multidrop mode with address generation and detection**
- **SPI Mode**
  - **Master or slave**
  - **Configurable serial clock phase and polarity**
  - **CLK SPI serial clock frequency up to a quarter of the CLK_USART internal clock frequency**
- **LIN Mode**
  - **Compliant with LIN 1.3 and LIN 2.0 specifications**
  - **Master or slave**
  - **Processing of Frames with up to 256 data bytes**
  - **Configurable response data length, optionally defined automatically by the Identifier**
  - **Self synchronization in slave node configuration**
  - **Automatic processing and verification of the “Break Field” and “Sync Field”**
  - **The “Break Field” is detected even if it is partially superimposed with a data byte**
  - **Optional, automatic identifier parity management**
  - **Optional, automatic checksum management**
  - **Supports both “Classic” and “Enhanced” checksum types**
  - **Full LIN error checking and reporting**
  - **Frame Slot Mode: the master allocates slots to scheduled frames automatically.**
  - **Wakeup signal generation**
- **Test Modes**
  - **Automatic echo, remote- and local loopback**
- **Supports two Peripheral DMA Controller channels**
  - **Buffer transfers without processor intervention**

20.2 Overview

The Universal Synchronous Asynchronous Receiver Transceiver (USART) provides a full duplex, universal, synchronous/asynchronous serial link. Data frame format is widely configurable, including basic length, parity, and stop bit settings, maximizing standards support. The receiver implements parity-, framing-, and overrun error detection, and can handle un-fixed frame lengths with the time-out feature. The USART supports several operating modes, providing an interface to, LIN, and SPI buses and infrared transceivers. Communication with slow and remote devices is eased by the timeguard. Duplex multidrop communication is supported by address and data differentiation through the parity bit. The hardware handshaking feature enables an out-of-band flow control, automatically managing RTS and CTS pins. The Peripheral DMA Controller connection enables memory transactions, and the USART supports chained

21.8.5 Status Register

Name: SR

Access Type: Read-only

Offset: 0x10

Reset Value: 0x00000000

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

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

15	14	13	12	11	10	9	8
-	-	-	-	-	UNDES	TXEMPTY	NSSR

7	6	5	4	3	2	1	0
-	-	-	-	OVRES	MODF	TDRE	RDRF

- **SPIENS: SPI Enable Status**

  1: This bit is set when the SPI is enabled.

  0: This bit is cleared when the SPI is disabled.
- **UNDES: Underrun Error Status (Slave Mode Only)**

  1: This bit is set when a transfer begins whereas no data has been loaded in the TDR register.

  0: This bit is cleared when the SR register is read.
- **TXEMPTY: Transmission Registers Empty**

  1: This bit is set when TDR and internal shifter are empty. If a transfer delay has been defined, TXEMPTY is set after the completion of such delay.

  0: This bit is cleared as soon as data is written in TDR.
- **NSSR: NSS Rising**

  1: A rising edge occurred on NSS pin since last read.

  0: This bit is cleared when the SR register is read.
- **OVRES: Overrun Error Status**

  1: This bit is set when an overrun has occurred. An overrun occurs when RDR is loaded at least twice from the serializer since the last read of the RDR.

  0: This bit is cleared when the SR register is read.
- **MODF: Mode Fault Error**

  1: This bit is set when a Mode Fault occurred.

  0: This bit is cleared when the SR register is read.
- **TDRE: Transmit Data Register Empty**

  1: This bit is set when the last data written in the TDR register has been transferred to the serializer.

  0: This bit is cleared when data has been written to TDR and not yet transferred to the serializer.

  TDRE equals zero when the SPI is disabled or at reset. The SPI enable command sets this bit to one.
- **RDRF: Receive Data Register Full**

  1: Data has been received and the received data has been transferred from the serializer to RDR since the last read of RDR.

  0: No data has been received since the last read of RDR

22.9 User Interface

Table 22-6. TWIM Register Memory Map

Offset	Register	Register Name	Access	Reset
0x00	Control Register	CR	Write-only	0x00000000
0x04	Clock Waveform Generator Register	CWGR	Read/Write	0x00000000
0x08	SMBus Timing Register	SMBTR	Read/Write	0x00000000
0x0C	Command Register	CMDR	Read/Write	0x00000000
0x10	Next Command Register	NCMDR	Read/Write	0x00000000
0x14	Receive Holding Register	RHR	Read-only	0x00000000
0x18	Transmit Holding Register	THR	Write-only	0x00000000
0x1C	Status Register	SR	Read-only	0x00000002
0x20	Interrupt Enable Register	IER	Write-only	0x00000000
0x24	Interrupt Disable Register	IDR	Write-only	0x00000000
0x28	Interrupt Mask Register	IMR	Read-only	0x00000000
0x2C	Status Clear Register	SCR	Write-only	0x00000000
0x30	Parameter Register	PR	Read-only	-[(1)]
0x34	Version Register	VR	Read-only	-[(1)]

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

23.9 User Interface

Table 23-6. TWIS Register Memory Map

Offset	Register	Register Name	Access	Reset
0x00	Control Register	CR	Read/Write	0x00000000
0x04	NBYTES Register	NBYTES	Read/Write	0x00000000
0x08	Timing Register	TR	Read/Write	0x00000000
0x0C	Receive Holding Register	RHR	Read-only	0x00000000
0x10	Transmit Holding Register	THR	Write-only	0x00000000
0x14	Packet Error Check Register	PECR	Read-only	0x00000000
0x18	Status Register	SR	Read-only	0x00000002
0x1C	Interrupt Enable Register	IER	Write-only	0x00000000
0x20	Interrupt Disable Register	IDR	Write-only	0x00000000
0x24	Interrupt Mask Register	IMR	Read-only	0x00000000
0x28	Status Clear Register	SCR	Write-only	0x00000000
0x2C	Parameter Register	PR	Read-only	-[(1)]
0x30	Version Register	VR	Read-only	-[(1)]

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

23.9.4 Receive Holding Register

Name: RHR

Access Type: Read-only

Offset: 0x0C

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
RXDATA							

- **RXDATA: Received Data Byte**

  When the RXRDY bit in the Status Register (SR) is one, this field contains a byte received from the TWI bus.

25.7.5 Interrupt Enable Register

Name: IER

Access Type: Write-only

Offset: 0x10

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
-	-	-	-	-	READY	-	TOFL

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.

31.7.8 Touch Group x Configuration Register 0

Name: TGxCFG0

Access Type: Read/Write

Offset: 0x20, 0x28

Reset Value: 0x00000000

31	30	29	28	27	26	25	24
DIV[15:8]							

23	22	21	20	19	18	17	16
DIV[7:0]							

15	14	13	12	11	10	9	8
CHLEN							

7	6	5	4	3	2	1	0
SELEN							

- **DIV: Clock Divider**

  The prescaler is used to ensure that the CLK_CAT clock is divided to around 1 MHz to produce the sampling clock.The prescaler uses the following formula to generate the sampling clock:

  Sampling clock = CLK_CAT / (2(DIV+1))

- **CHLEN: Charge Length**

  For the QTouch method, specifies how many sample clock cycles should be used for transferring charge to the sense capacitor.

- **SELEN: Settle Length**

  For the QTouch method, specifies how many sample clock cycles should be used for settling after charge transfer.

31.8 Module Configuration

The specific configuration the CAT module 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 31-4. CAT Configuration

Feature	CAT
Number of touch sensors/Size of matrix	Allows up to 17 touch sensors, or up to 16 by 8 matrix sensors to be interfaced.

Table 31-5. CAT Clocks

Clock Name	Description
CLK_CAT	Clock for the CAT bus interface
GCLK	The generic clock used for the CAT is GCLK4

Table 31-6. Register Reset Values

Register	Reset Value
VERSION	0x00000400
PARAMETER	0x0001FFFF

31.8.1 Resistive Drive

By default, the CAT pins are driven with normal I/O drive properties. Some of the CSA and CSB pins can optionally drive with a 1k output resistance for improved EMC.

To enable resistive drive on a pin, the user must write a one to the corresponding bit in the CSA Resistor Control Register (CSARES) or CSB Resistor Control Register (CSBRES) register.

33.7.8 Transmit Holding Register

Name: THR

Access Type: Read/Write

Offset: 0x1C

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
TXDATA							

- **TXDATA: Transmit Data**
  The data to send.

continue shifting the same instruction until the busy bit clears, or start shifting data. If shifting data, you must be prepared that the data shift may also report busy.

- During Shift-DR of an address: The new address is ignored. The SAB stays in address mode, so no data must be shifted. Repeat the address until the busy bit clears.
- During Shift-DR of read data: The read data is invalid. The SAB stays in data mode. Repeat scanning until the busy bit clears.
- During Shift-DR of write data: The write data is ignored. The SAB stays in data mode. Repeat scanning until the busy bit clears.

34.4.11.5 Error Reporting

The Service Access Bus may not be able to complete all accesses as requested. This may be because the address is invalid, the addressed area is read-only or cannot handle byte/halfword accesses, or because the chip is set in a protected mode where only limited accesses are allowed.

The error bit is updated when an access completes, and is cleared when a new access starts.

What to do if the error bit is set:

- During Shift-IR: The new instruction is selected. The last operation performed using the old instruction did not complete successfully.
- During Shift-DR of an address: The previous operation failed. The new address is accepted. If the read bit is set, a read operation is started.
- During Shift-DR of read data: The read operation failed, and the read data is invalid.
- During Shift-DR of write data: The previous write operation failed. The new data is accepted and a write operation started. This should only occur during block writes or stream writes. No error can occur between scanning a write address and the following write data.
- While polling with CANCEL_ACCESS: The previous access was cancelled. It may or may not have actually completed.
- After power-up: The error bit is set after power up, but there has been no previous SAB instruction so this error can be discarded.

34.4.11.6 Protected Reporting

A protected status may be reported during Shift-IR or Shift-DR. This indicates that the security bit in the Flash Controller is set and that the chip is locked for access, according to Section 34.5.1.

The protected state is reported when:

- The Flash Controller is under reset. This can be due to the AVR_RESET command or the RESET_N line.
- The Flash Controller has not read the security bit from the flash yet (This will take a a few ms). Happens after the Flash Controller reset has been released.
- The security bit in the Flash Controller is set.

What to do if the protected bit is set:

- Release all active AVR_RESET domains, if any.
- Release the RESET_N line.
- Wait a few ms for the security bit to clear. It can be set temporarily due to a reset.

Table 34-30. aWire Packet Format

Field	Number of bytes	Description	Comment	Optional
LENGTH	2	The number of bytes in the DATA field.		Yes
DATA	LENGTH	Data according to command/ response.		Yes
CRC	2	CRC calculated with the FCS16 polynomial.	CRC value of 0x0000 makes the aWire disregard the CRC if the master does not support it.	No

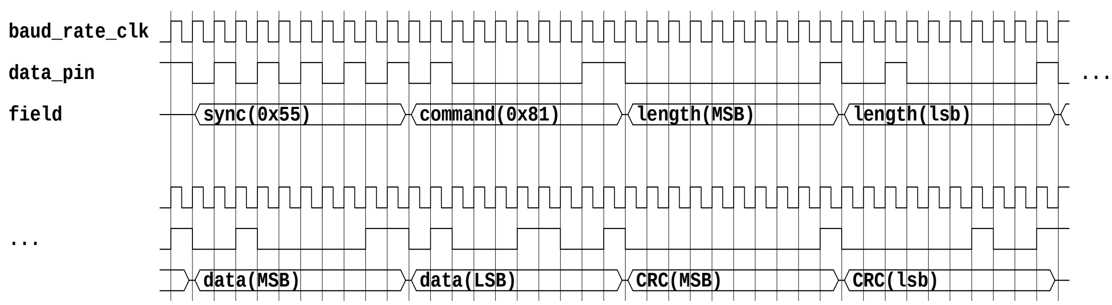
CRC calculation

The CRC is calculated from the command/response, length, and data fields. The polynomial used is the FCS16 (or CRC-16-CCIT) in reverse mode (0x8408) and the starting value is 0x0000.

Example command

Below is an example command from the master with additional data.

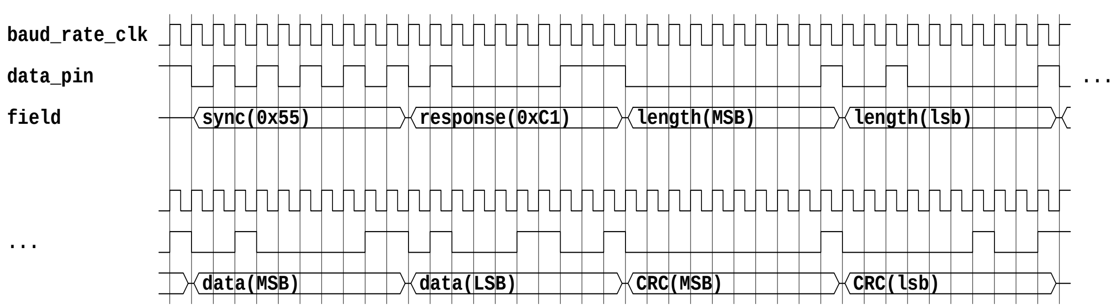
Figure 34-9. Example Command



Example response

Below is an example response from the slave with additional data.

Figure 34-10. Example Response



  - $V_{VDDCORE}$ = 1.62V, supplied by the internal regulator
  - Corresponds to the 3.3V supply mode with 1.8V regulated I/O lines, please refer to the Supply and Startup Considerations section for more details
    - Equivalent to the 3.3V single supply mode
    - Consumption in 1.8V single supply mode can be estimated by subtracting the regulator static current
- Operating conditions, external core supply (Figure 35-2) - used only when noted
  - V_{VDDIN} = $V_{VDDCORE}$ = 1.8V
  - Corresponds to the 1.8V single supply mode, please refer to the Supply and Startup Considerations section for more details
- T_A = 25°C
- Oscillators
  - OSC0 (crystal oscillator) stopped
  - OSC32K (32KHz crystal oscillator) running with external 32KHz crystal
  - DFLL running at 50MHz with OSC32K as reference
- Clocks
  - DFLL used as main clock source
  - CPU, HSB, and PBB clocks undivided
  - PBA clock divided by 4
  - The following peripheral clocks running
    - PM, SCIF, AST, FLASHCDW, PBA bridge
  - All other peripheral clocks stopped
- I/Os are inactive with internal pull-up
- Flash enabled in high speed mode
- POR18 enabled
- POR33 disabled