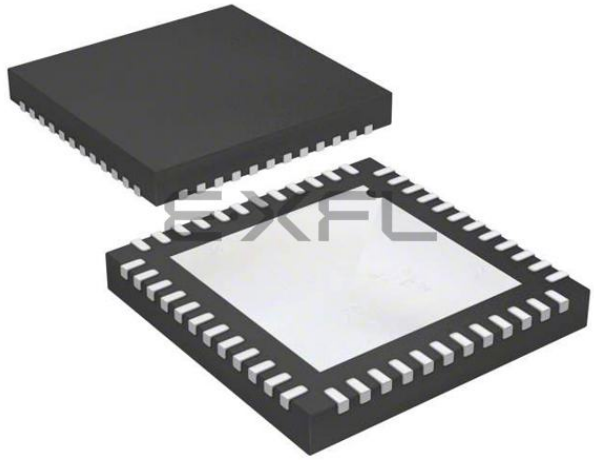

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	ARM7®
Core Size	16/32-Bit
Speed	20.48MHz
Connectivity	LINbus, SPI, UART/USART
Peripherals	POR, PSM, Temp Sensor, WDT
Number of I/O	9
Program Memory Size	32KB (16K x 16)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 32
Voltage - Supply (Vcc/Vdd)	3.5V ~ 18V
Data Converters	A/D 2x16b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 115°C (TA)
Mounting Type	Surface Mount
Package / Case	48-VFQFN Exposed Pad, CSP
Supplier Device Package	48-LFCSP-VQ (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/aduc7034bcpz

Parameter	Test Conditions/Comments	Min	Typ	Max	Unit
R_{SLAVE}	Slave termination resistance	20	30	47	kΩ
$V_{SERIAL\ DIODE}$[28]	Voltage drop at the internal diode	0.4	0.7	1	V
Symmetry of Transmit Propagation Delay[1]	VDD (minimum) = 7 V	−4		+4	µs
Receive Propagation Delay[1]	VDD (minimum) = 7 V			6	µs
Symmetry of Receive Propagation Delay[1]	VDD (minimum) = 7 V	−2		+2	µs
LIN VERSION1.3 SPECIFICATION	Bus load conditions (C_{BUS}||R_{BUS}): 1 nF||1 kΩ ; 6.8 nF||660 Ω; 10 nF||500 Ω				
$\left\|\frac{dV}{dt}\right\|$[1]	Slew rate				
	Dominant and recessive edges, VBAT = 18 V	1	2	3	V/ µs
$\left\|\frac{dV}{dt}\right\|$[1]	Slew rate				
	Dominant and recessive edges, VBAT = 7 V	0.5		3	V/ µs
t_{SYM}[1]	Symmetry of rising and falling edge, VBAT = 18 V	−5		+5	µs
	Symmetry of rising and falling edge, VBAT = 7 V	−4		+4	µs
LIN VERSION 2.0 SPECIFICATION	Bus load conditions (CBUS||RBUS): 1 nF||1 kΩ, 6.8 nF||660 Ω, 10 nF||500 Ω				
D1	Duty Cycle 1, $TH_{REC(MAX)} = 0.744 \times VBAT$, $TH_{DOM(MAX)} = 0.581 \times VBAT$, V_{SUP} = 7.0 V … 18 V, t_{BIT} = 50 µs, D1 = $t_{BUS\_REC(MIN)}/(2 \times t_{BIT})$	0.396			
D2	Duty Cycle 2, $TH_{REC(MIN)} = 0.284 \times VBAT$, $TH_{DOM(MIN)} = 0.422 \times VBAT$, V_{SUP} = 7.0 V … 18 V; t_{BIT} = 50 µs, D2 = $t_{BUS\_REC(MAX)}/(2 \times t_{BIT})$			0.581	
BSD INPUT/OUTPUT[29]					
Baud Rate		1164	1200	1236	bps
Input Leakage Current	Input high = VDD, or input low = IO_VSS	−50		+50	µA
Output Low Voltage (V_{OL})				1.2	V
Output High Voltage (V_{OH})		0.8 VDD			V
Short-Circuit Output Current ($I_{o(sc)}$)	V_{BSD} = VDD = 12 V	40	80	200	mA
Input Low Voltage (V_{INL})				1.8	V
Input High Voltage (V_{INH})		0.7 VDD			V
WAKE-UP	R_{LOAD} = 300 Ω, C_{BUS} = 91 nF, R_{LIMIT} = 39 Ω				
VDD[1]	Supply voltage range at which the WU pin is functional	7		18	V
Input Leakage Current	Input high = VDD	0.4		2.1	mA
	Input low = IO_VSS	−50		+50	µA
V_{OH}[30]	Output high level	5			V
V_{OL}[30]	Output low level			2	V
V_{IH}	Input high level	4.6			V
V_{IL}	Input low level			1.2	V
Monoflop Timeout	Timeout period	0.6	1.3	2	sec
Short-Circuit Output Current ($I_{o(sc)}$)		100	140		mA
SERIAL TEST INTERFACE	R_{LOAD} = 500 Ω, C_{BUS} = 2.4 nF, R_{LIMIT} = 39 Ω				
Baud Rate				40	kbps
Input Leakage Current	Input high = VDD, or input low = IO_VSS	−50		+70	µA
VDD	Supply voltage range for which STI is functional	7		18	V
V_{OH}	Output high level	0.6 VDD			V
V_{OL}	Output low level			0.4 VDD	V
V_{IH}	Input high level	0.6 VDD			V
V_{IL}	Input low level			0.4 VDD	V
PACKAGE THERMAL SPECIFICATIONS					
Thermal Shutdown[1, 31]		140	150	160	°C
Thermal Impedance (θ_{JA})[32]	48-lead LFCSP, stacked die		45		°C/W

Table 4. SPI Slave Mode Timing—PHASE Mode = 1

Parameter	Description	Min	Typ	Max	Unit
$t_{\overline{SS}}$	$\overline{SS}$ to SCLK edge		½ t_{SL}		ns
t_{SL}	SCLK low pulse width[1]		(SPIDIV + 1) × t_{HCLK}		ns
t_{SH}	SCLK high pulse width[1]		(SPIDIV + 1) × t_{HCLK}		ns
t_{DAV}	Data output valid after SCLK edge[2]			(3 × t_{UCLK}) + (2 × t_{HCLK})	ns
t_{DSU}	Data input setup time before SCLK edge	0			ns
t_{DHD}	Data input hold time after SCLK edge[2]	4 × t_{UCLK}			ns
t_{DF}	Data output fall time		3.5		ns
t_{DR}	Data output rise time		3.5		ns
t_{SR}	SCLK rise time		3.5		ns
t_{SF}	SCLK fall time		3.5		ns
t_{SFS}	$\overline{SS}$ high after SCLK edge		½ t_{SL}		ns

[1] t_{HCLK} depends on the clock divider (CD) bits in the POWCON MMR. $t_{HCLK} = t_{UCLK}/2^{CD}$.
[2] t_{UCLK} = 48.8 ns and corresponds to the 20.48 MHz internal clock from the PLL before the clock divider.

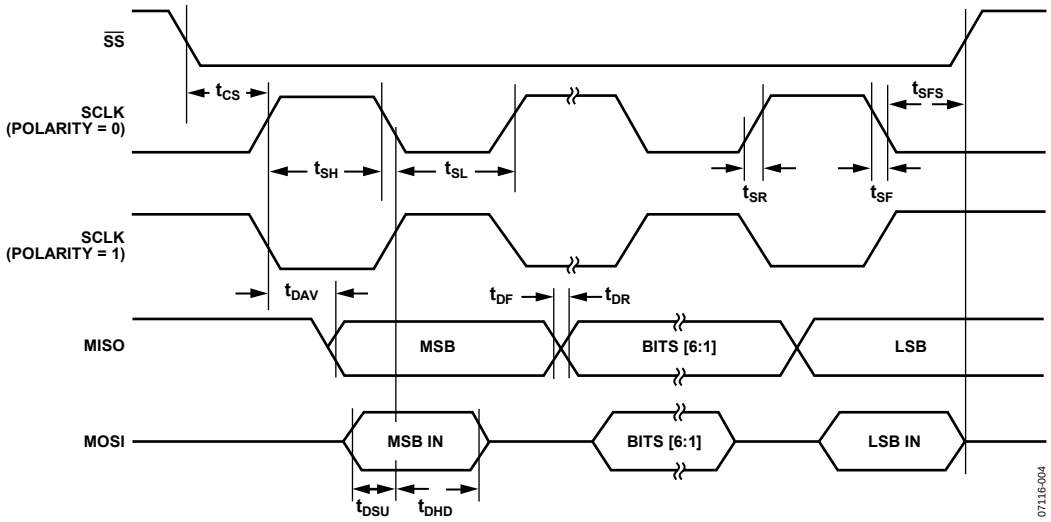

Figure 4. SPI Slave Mode Timing—PHASE Mode = 1

TYPICAL PERFORMANCE CHARACTERISTICS

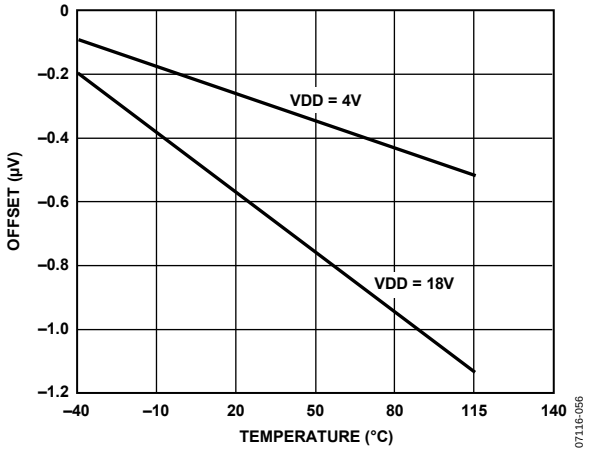

Figure 8. ADC Current Channel Offset vs. Temperature, 10 MHz MCU

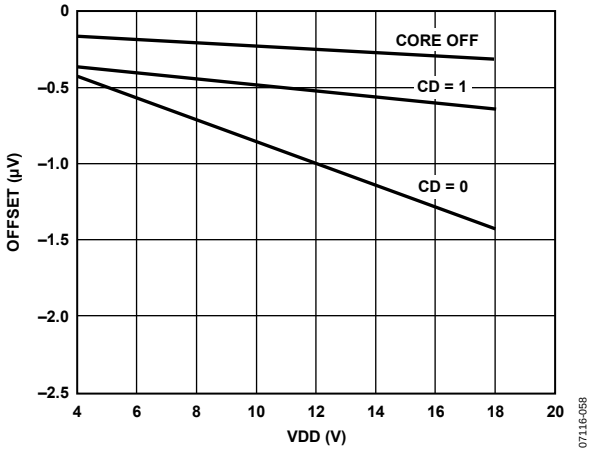

Figure 10. ADC Current Channel Offset vs. Supply @ 25°C

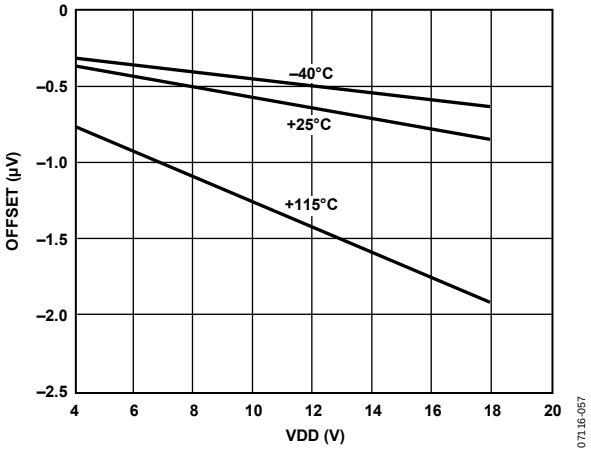

Figure 9. ADC Current Channel Offset vs. VDD (10 MHz, MCU)

FEE0CON Register

Name: FEE0CON

Address: 0xFFFF0E08

Default Value: 0x07

Access: Read/write access

Function: This 8-bit register is written by user code to control the operating modes of the Flash/EE memory controller.

Table 13. Command Codes in FEE0CON

Code	Command	Description
0x00[1]	Reserved	Reserved. This command should not be written by user code.
0x01[1]	Single read	Load FEE0DAT with the 16-bit data indexed by FEE0ADR.
0x02[1]	Single write	Write FEE0DAT at the address pointed by FEE0ADR. This operation takes 50 µs.
0x03[1]	Erase write	Erase the page indexed by FEE0ADR and write FEE0DAT at the location pointed by FEE0ADR. This operation takes 20 ms.
0x04[1]	Single verify	Compare the contents of the location pointed by FEE0ADR to the data in FEE0DAT. The result of the comparison is returned in FEE0STA Bit 1.
0x05[1]	Single erase	Erase the page indexed by FEE0ADR.
0x06[1]	Mass erase	Erase 30 kB of user space. The 2 kB kernel is protected. This operation takes 1.2 sec. To prevent accidental execution, a command sequence is required to execute this instruction; this is described in the Command Sequence for Executing a Mass Erase section.
0x07	Idle	Default command.
0x08	Reserved	Reserved. This command should not be written by user code.
0x09	Reserved	Reserved. This command should not be written by user code.
0x0A	Reserved	Reserved. This command should not be written by user code.
0x0B	Signature	This command results in a 24-bit, LFSR-based signature being generated and loaded into FEE0SIG. If FEE0ADR is less than 0x87800, this command results in a 24-bit, LFSR-based signature of the user code space from the page specified in FEE0ADR upwards, including the kernel, security bits, and Flash/EE key. If FEE0ADR is greater than 0x87800, the kernel and manufacturing data is signed. This operation takes 120 µs.
0x0C	Protect	This command can be run one time only. The value of FEE0PRO is saved and can be removed only with a mass erase (0x06) or with the software protection key.
0x0D	Reserved	Reserved. This command should not be written by user code.
0x0E	Reserved	Reserved. This command should not be written by user code.
0x0F	Ping	No operation, interrupt generated.

[1] The FEE0CON register reads 0x07 immediately after the execution of this command.

CODE EXECUTION TIME FROM SRAM AND FLASH/EE

This section describes SRAM and Flash/EE access times during execution for applications where execution time is critical.

Execution from SRAM

Fetching instructions from SRAM takes one clock cycle because the access time of the SRAM is 2 ns, and a clock cycle is 49 ns minimum. However, when the instruction involves reading or writing data to memory, one extra cycle must be added if the data is in SRAM. If the data is in Flash/EE, two cycles must be added: one cycle to execute the instruction and two cycles to retrieve the 32-bit data from Flash/EE. A control flow instruction, such as a branch instruction, takes one cycle to fetch and two cycles to fill the pipeline with the new instructions.

Execution from Flash/EE

In Thumb mode, where instructions are 16 bits, one cycle is needed to fetch any instruction.

In ARM mode with CD = 0, two cycles are needed to fetch the 32-bit instructions. With CD > 0, no extra cycles are required for the fetch because the Flash/EE memory continues to be clocked at full speed. In addition, some dead time is needed before accessing data for any value of CD bits.

Timing is identical in both modes when executing instructions that involve using the Flash/EE for data memory. If the instruction to be executed is a control flow instruction, an extra cycle is needed to decode the new address of the program counter, and then four cycles are needed to fill the pipeline if CD = 0.

A data processing instruction involving only the core register does not require any extra clock cycles. Data transfer instructions are more complex and are summarized in Table 17.

Table 17. Typical Execution Cycles in ARM/Thumb Mode

Instructions	Fetch Cycles	Dead Time	Data Access
LD	2/1	1	2
LDH	2/1	1	1
LDM/PUSH	2/1	N	2 × N
STR	2/1	1	2 × 50 µs
STRH	2/1	1	50 µs
STRM/POP	2/1	N	2 × N × 50 µs

With 1 < N ≤ 16, N is the number of data to load or store in the multiple load/store instruction.

By default, Flash/EE code execution is suspended during any Flash/EE erase or write cycle. A page (512 bytes) erase cycle takes 20 ms and a write (16 bits) word command takes 50 µs. However, the Flash/EE controller allows erase/write cycles to be aborted if the ARM core receives an enabled interrupt during the current Flash/EE erase/write cycle. The ARM7 can, therefore, immediately service the interrupt and then return to repeat the Flash/EE command. The abort operation typically requires 10 clock cycles. If the abort operation is not feasible, the user can run Flash/EE programming code and the relevant interrupt routines from SRAM to allow the core to immediately service the interrupt.

MEMORY-MAPPED REGISTERS

The memory-mapped register (MMR) space is mapped into the top 4 kB of the MCU memory space and accessed by indirect addressing, loading, and storage commands through the ARM7-banked registers. An outline of the memory-mapped register bank for the ADuC7034 is shown in Figure 16.

The MMR space provides an interface between the CPU and all on-chip peripherals. All registers except the ARM7 core registers (described in the ARM Registers section) reside in the MMR area.

As exhibited in Table 18 to Table 29 in the Complete MMR Listing section, the MMR data widths vary from 1 byte (8 bits) to 4 bytes (32 bits). The ARM7 core can access any of the MMRs (single byte or multiple byte width registers) with a 32-bit read or write access.

The resultant read, for example, is aligned per little endian format as described in the ARM Registers section. However, errors result if the ARM7 core tries to access 4-byte (32-bit) MMRs with a 16-bit access. In the case of a 16-bit write access to a 32-bit MMR, the 16 MSBs (the upper 16 bits) are written as 0s. In the case of a 16-bit read access to a 32-bit MMR, only 16 of the MMR bits can be read.

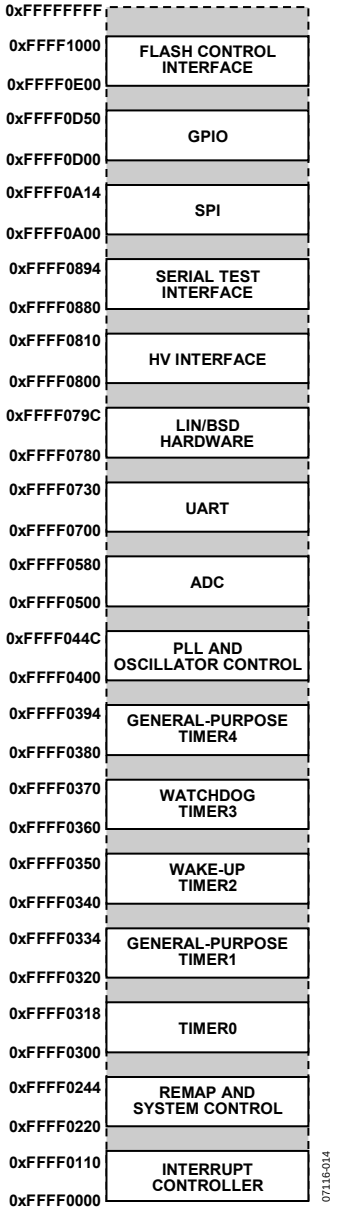

Figure 16. Top-Level MMR Map

16-BIT SIGMA-DELTA ANALOG-TO-DIGITAL CONVERTERS

The ADuC7034 incorporates two independent sigma-delta (Σ-Δ) analog-to-digital converters (ADCs), namely, the current channel ADC (I-ADC) and the voltage/temperature channel ADC (V-/T-ADC). These precision measurement channels integrate on-chip buffering, a programmable gain amplifier, 16-bit Σ-Δ modulators, and digital filtering for precise measurement of current, voltage, and temperature variables in 12 V automotive battery systems.

CURRENT CHANNEL ADC (I-ADC)

The I-ADC converts battery current sensed through an external 100 μΩ shunt resistor. On-chip programmable gain means that the I-ADC can be configured to accommodate battery current levels from ±1 A to ±1500 A.

As shown in Figure 17, the I-ADC employs a Σ-Δ conversion technique to attain 16 bits of no missing codes performance.

The Σ-Δ modulator converts the sampled input signal into a digital pulse train whose duty cycle contains the digital information. A modified sinc3 programmable low-pass filter is then employed to decimate the modulator output data stream to provide a valid 16-bit data conversion result at programmable output rates from 4 Hz to 8 kHz in normal mode and from 1 Hz to 2 kHz in low power mode.

The I-ADC also incorporates counter, comparator, and accumulator logic. This allows the I-ADC result to generate an interrupt after a predefined number of conversions has elapsed or a programmable threshold value has been exceeded. A fast ADC overrange feature is also supported. Once enabled, a 32-bit accumulator automatically sums the 16-bit I-ADC results.

The time to a first valid (fully settled) result on the current channel is three ADC conversion cycles with chop mode disabled and two ADC conversion cycles with chop mode enabled.

Bit	Description
1	Voltage conversion result ready bit. Set by hardware as soon as a valid voltage conversion result is written in the voltage data register (ADC1DAT MMR) if the voltage channel ADC is enabled. It is also set at the end of a calibration. Cleared by reading either ADC1DAT or ADC0DAT.
0	Current conversion result ready bit. Set by hardware as soon as a valid current conversion result is written in the current data register (ADC0DAT MMR) if the current channel ADC is enabled. It is also set at the end of a calibration. Cleared by reading ADC0DAT.

ADC Interrupt Mask Register

Name: ADCMSKI

Address: 0xFFFF0504

Default Value: 0x00

Access: Read/write

Function: This register allows the ADC interrupt sources to be individually enabled. The bit positions in this register are the same as the lower eight bits in the ADCSTA MMR. If a bit is set to 1 by user code, the respective interrupt is enabled. By default, all bits are 0, meaning all ADC interrupt sources are disabled.

ADC Mode Register

Name: ADCMDE

Address: 0xFFFF0508

Default Value: 0x00

Access: Read/write

Function: The ADC mode MMR is an 8-bit register that configures the mode of operation of the ADC subsystem.

Table 35. ADCMDE MMR Bit Designations

Bit	Description
7	Not used. This bit is reserved for future functionality and should be written as 0 by user code.
6	20 kΩ resistor select. 1 = selects the 20 kΩ resistor shown in Figure 20. 0 = selects the direct path to ground shown in Figure 20 (default).
5	Low power mode reference select. 1 = enables the precision voltage reference in either low power mode or low power plus mode. Enabling the precision voltage reference in low power modes requires additional current and therefore results in increased current consumption. 0 = enables the low power voltage reference in either low power mode or low power plus mode (default).
4 to 3	ADC power mode configuration. 00 = ADC normal mode. If enabled, the ADC operates with normal current consumption, yielding optimum electrical performance. 01 = ADC low power mode. If enabled, the I-ADC operates with reduced current consumption. This limitation in current consumption is achieved, but at the expense of degrading ADC noise performance, by fixing the gain to 128 and using the on-chip low power 131 kHz oscillator to directly drive the ADC circuits. 10 = ADC low power plus mode. If this bit is enabled, the ADC operates with reduced current consumption. In this mode, the gain is fixed to 512 and the current consumed is approximately 200 μA more than the ADC low power mode. The additional current consumed also ensures that the ADC noise performance is better than that achieved in ADC low power mode. 11 = not defined.
2 to 0	ADC operation mode configuration. 000 = ADC power-down mode. All ADC circuits, including the internal reference, are powered down. 001 = ADC continuous conversion mode. In this mode, any enabled ADC continuously converts. 010 = ADC single conversion mode. In this mode, any enabled ADC performs a single conversion. The ADC enters idle mode when the single conversion is complete. A single conversion takes two to three ADC clock cycles, depending on the chop mode. 011 = ADC idle mode. In this mode, the ADC is fully powered on but is held in reset. 100 = ADC self-offset calibration. In this mode, an offset calibration is performed on any enabled ADC using an internally generated 0 V. The calibration is carried out at the user programmed ADC settings; therefore, as with a normal single ADC conversion, it takes two to three ADC conversion cycles before a fully settled calibration result is ready. The calibration result is automatically written to the ADCxOF MMR of the respective ADC. The ADC returns to idle mode and the calibration and conversion ready status bits are set at the end of an offset calibration cycle. 101 = ADC self-gain calibration. In this mode, a gain calibration to an internal reference voltage is performed on all enabled ADCs. A gain calibration is a two-stage process and takes twice the time of an offset calibration. The calibration result is automatically written to the ADCxGN MMR of the respective ADC. The ADC returns to idle mode, and the calibration and conversion ready status bits are set at the end of a gain calibration cycle. An ADC self-gain calibration should only be performed on the current channel ADC. Preprogrammed, factory-set calibration coefficients (downloaded automatically from internal Flash/EE) should be used for voltage temperature measurements. If an external NTC is used, an ADC self-calibration should be performed on the temperature channel. 110 = ADC system zero-scale calibration. In this mode, a zero-scale calibration is performed on enabled ADC channels to an external zero-scale voltage driven at the ADC input pins. The calibration is performed at the user programmed ADC settings; therefore, as with a single ADC conversion, three ADC conversion cycles are required before a fully settled calibration result is available. 111 = ADC system full-scale calibration. In this mode, a full-scale calibration is performed on enabled ADC channels to an external full-scale voltage driven at the ADC input pins.

Voltage/Temperature Channel ADC Control Register

Name: ADC1CON

Address: 0xFFFF0510

Default Value: 0x0000

Access: Read/write

Function: The voltage/temperature channel ADC control MMR is a 16-bit register that is used to configure the V-/T-ADC.

Note: If the VBAT attenuator input is selected, the voltage attenuator buffers are automatically enabled.

Table 37. ADC1CON MMR Bit Designations

Bit	Description
15	Voltage/temperature channel ADC enable. Set to 1 by user code to enable the V-/T-ADC. Clearing this bit to 0 powers down the V-/T-ADC.
14 to 13	VTEMP current source enable. 00 = disables current sources. 01 = enables 50 µA current source on VTEMP. 10 = enables 50 µA current source on GND_SW. 11 = enables 50 µA current source on both VTEMP and GND_SW.
12 to 10	Not used. These bits are reserved for future functionality and should not be modified by user code.
9	Voltage/temperature channel ADC output coding. Set to 1 by user code to configure V-/T-ADC output coding as unipolar. Cleared to 0 by user code to configure V-/T-ADC output coding as twos complement.
8	Not used. This bit is reserved for future functionality and should be written as 0 by user code.
7 to 6	Voltage/temperature channel ADC input select. 00 = VBAT attenuator (VBAT/24 and AGND) is selected. The high voltage buffers are enabled automatically in this configuration. The conversion result is written to ADC1DAT. 01 = external temperature inputs (VTEMP and GND_SW) are selected. The conversion result is written to ADC2DAT. 10 = internal sensor is selected. Internal temperature sensor input selected, conversion result written to ADC2DAT. The temperature gradient is 0.33 mV/°C; this is only applicable to the internal temperature sensor. 11 = internal short is selected. Shorted input. The conversion result is written to ADC1DAT.
5 to 4	Voltage/temperature channel ADC reference select. 00 = internal 1.2 V precision reference is selected. 01 = external reference inputs VREF and GND_SW are selected. 10 = external reference inputs divided by 2 (that is, VREF and GND_SW divided by 2) are selected. This allows an external reference up to REG_AVDD. 11 = the reference input REG_AVDD and AGND divided by 2 are selected for the voltage channel. The reference inputs REG_AVDD and GND_SW divided by 2 are selected for the temperature channel.
3 to 0	Not used. These bits are reserved for future functionality and should not be written as 0 by user code.

PLLCON Prewrite Key Register

Name: PLLKEY0

Address: 0xFFFF0410

Access: Write only

Key: 0x000000AA

Function: PLLKEY0 is the PLLCON prewrite key. PLLCON is a keyed register that requires a 32-bit key value to be written before and after PLLCON.

PLLCON Postwrite Key Register

Name: PLLKEY1

Address: 0xFFFF0418

Access: Write only

Key: 0x00000055

Function: PLLKEY1 is the PLLCON postwrite key. PLLCON is a keyed register that requires a 32-bit key value to be written before and after PLLCON.

PLLCON Register

Name: PLLCON

Address: 0xFFFF0414

Default Value: 0x00

Access: Read/write

Function: This 8-bit register allows user code to dynamically select one of three oscillator sources as the PLL source clock.

Table 44. PLLCON MMR Bit Designations[1]

Bit	Description
7 to 2	Reserved. These bits should be written as 0 by user code.
1 to 0	PLL clock source. 00 = low power 131 kHz oscillator. 01 = precision 131 kHz oscillator. 10 = external 32.768 kHz crystal. 11 = reserved.

[1] If the user code switches MCU clock sources, a dummy MCU cycle should be included after the clock switch is written to PLLCON.

POWCON Prewrite Key Register

Name: POWKEY0

Address: 0xFFFF0404

Access: Write only

Key: 0x00000001

Function: POWKEY0 is the POWCON prewrite key. POWCON is a keyed register that requires a 32-bit key value to be written before and after POWCON.

POWCON Postwrite Key Register

Name: POWKEY1

Address: 0xFFFF040C

Access: Write only

Key: 0x000000F4

Function: POWKEY1 is the POWCON postwrite key. POWCON is a keyed register that requires a 32-bit key value to be written before and after POWCON.

TIMER2—WAKE-UP TIMER

Timer2 is a 32-bit wake-up up/down counter timer with a programmable prescaler. The prescaler is clocked directly from one of four clock sources, namely, the core clock (which is the default selection), the low power 32.768 kHz oscillator, the external 32.768 kHz watch crystal, or the precision 32.768 kHz oscillator. The selected clock source can be scaled by a factor of 1, 16, 256, or 32,768. The wake-up timer continues to run when the core clock is disabled. When the core is operating at 20.48 MHz and at CD = 0 with a prescaler of 1, a minimum resolution of 48.83 ns results.

The counter can be formatted as a plain 32-bit value or as time expressed as hours:minutes:seconds:hundredths.

Timer2 reloads the value from T2LD when Timer2 overflows.

The Timer2 interface consists of four MMRS.

- T2LD and T2VAL are 32-bit registers and hold 32-bit unsigned integers. T2VAL is a read only register.
- T2CLRI is an 8-bit register. Writing any value to this register clears the Timer2 interrupt.
- T2CON is a configuration MMR and is described in Table 54.

Timer2 Load Register

Name: T2LD

Address: 0xFFFF0340

Default Value: 0x00000000

Access: Read/write

Function: T2LD is a 32-bit register that holds the 32-bit value that is loaded into the counter.

Timer2 Clear Register

Name: T2CLRI

Address: 0xFFFF034C

Access: Write only

Function: This 8-bit, write only MMR is written (with any value) by user code to clear the interrupt.

Timer2 Value Register

Name: T2VAL

Address: 0xFFFF0344

Default Value: 0xFFFFFFFF

Access: Read only

Function: T2VAL is a 32-bit register that holds the current value of Timer2.

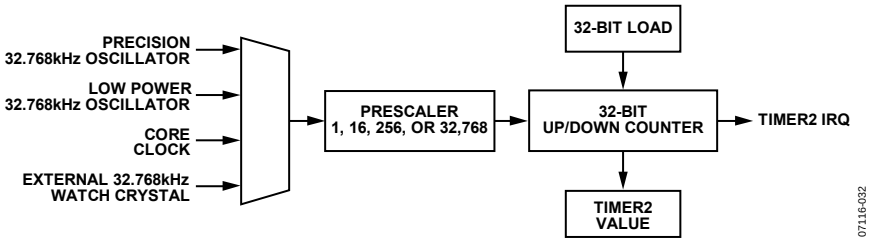

Figure 36. Timer2 Block Diagram

HIGH VOLTAGE PERIPHERAL CONTROL INTERFACE

The ADuC7034 integrates several high voltage circuit functions that are controlled and monitored through a registered interface consisting of two MMRs, namely, HVCON and HVDAT. The HVCON register acts as a command byte interpreter, allowing the microcontroller to indirectly read or write 8-bit data (the value in HVDAT) from or to one of four high voltage status or configuration registers. These high voltage status and configuration registers are not MMRs but are registers commonly referred to as indirect registers, that is, they can only be accessed (as the name suggests) indirectly via the HVCON and HVDAT MMRs.

The physical interface between the HVCON register and the indirect high voltage registers is a 2-wire (data and clock) serial interface based on a 2.56 MHz serial clock. Therefore, there is a finite 10 µs (maximum) latency between the MCU core writing a command into HVCON and the command or data reaching the indirect high voltage registers. There is also a finite 10 µs latency between the MCU core writing a command into HVCON and the indirect register data being read back into the HVDAT register. A busy bit (for example, Bit 0 of the HVCON when read by MCU) can be polled by the MCU to confirm when a read/write command is complete.

The following high voltage circuit functions are controlled and monitored via this interface. Figure 40 shows the top-level architecture of the high voltage interface and the following related circuits:

- Precision oscillator
- Wake-up (WU) pin functionality
- Power supply monitor (PSM)
- Low voltage flag (LVF)
- LIN operating modes
- STI diagnostics
- High voltage diagnostics
- High voltage attenuator-buffer circuit
- High voltage (HV) temperature monitor

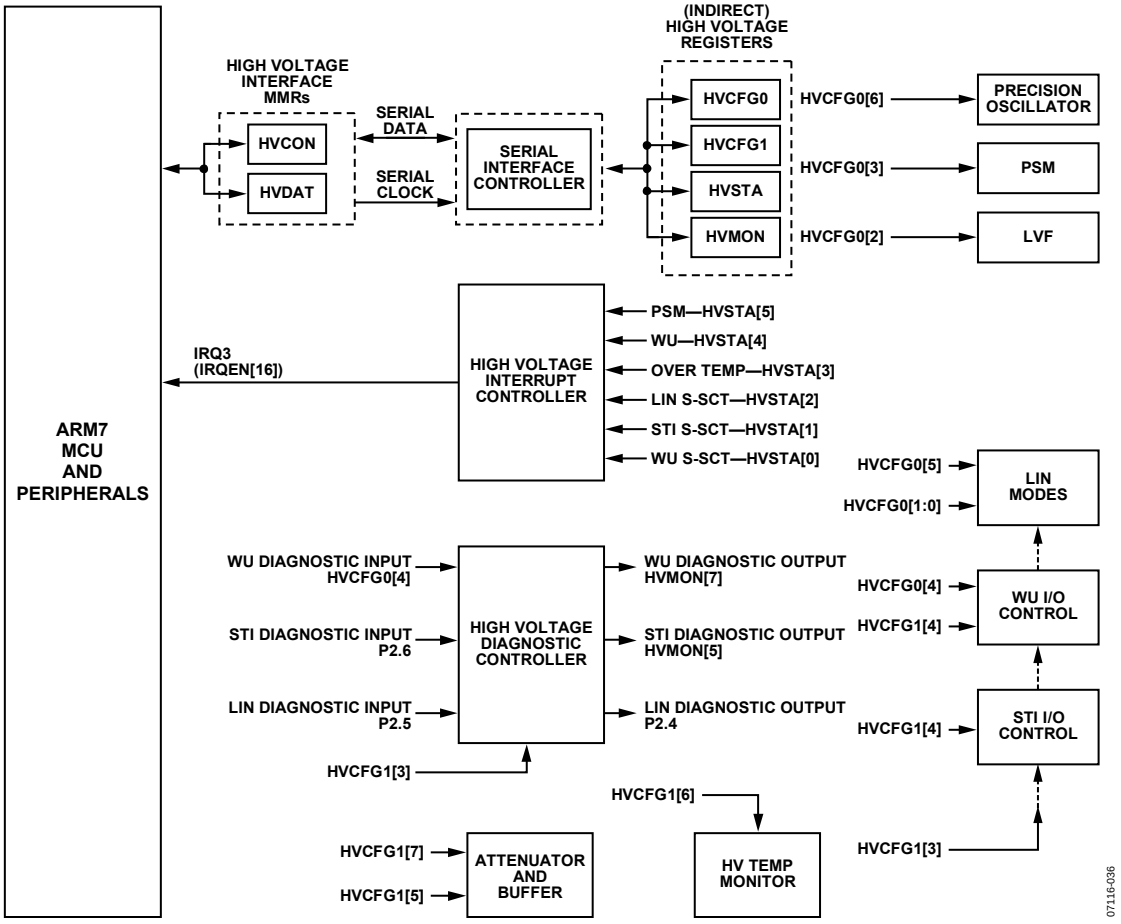

Figure 40. High Voltage Interface, Top-Level Block Diagram

UART SERIAL INTERFACE

The ADuC7034 features a 16,450-compatible UART. The UART is a full-duplex, universal, asynchronous receiver/transmitter. A UART performs serial-to-parallel conversion on data characters received from a peripheral device and performs parallel-to-serial conversion on data characters received from the ARM7TDMI. The UART features a fractional divider that facilitates high accuracy baud rate generation and a network addressable mode. The UART functionality is available on the GPIO_5/RxD and GPIO_6/TxD pins of the ADuC7034.

The serial communication adopts an asynchronous protocol that supports various word lengths, stop bits, and parity generation options selectable in the configuration register.

BAUD RATE GENERATION

The ADuC7034 features two methods of generating the UART baud rate: normal 450 UART baud rate generation and ADuC7034 fractional divider baud rate generation.

Normal 450 UART Baud Rate Generation

The baud rate is a divided version of the core clock using the values in the COMDIV0 and COMDIV1 MMRs (each is a 16-bit value, DL). The standard baud rate generator formula is

$$Baud\ Rate = \frac{20.48\ \text{MHz}}{2^{CD} \times 16 \times 2 \times DL} \quad (1)$$

Table 78 lists common baud rate values.

Table 78. Baud Rate Using the Standard Baud Rate Generator

Baud Rate (bps)	CD	DL	Actual Baud Rate	% Error
9600	0	0x43	9552	0.50%
19,200	0	0x21	19,394	1.01%
115,200	0	0x6	106,667	7.41%
9600	3	0x8	10,000	4.17%
19,200	3	0x4	20,000	4.17%
115,200	3	0x1	80,000	30.56%

ADuC7034 Fractional Divider Baud Rate Generation

The fractional divider combined with the normal baud rate generator allows the generation of accurate, high speed baud rates.

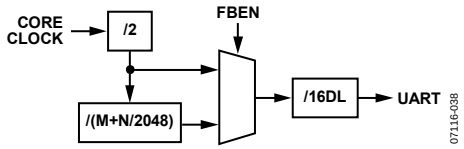

Figure 42. Fractional Divider Baud Rate Generation

Calculation of the baud rate using a fractional divider is as follows:

$$Baud\ Rate = \frac{20.48\ \text{MHz}}{2^{CD} \times 16 \times DL \times 2 \times (M + \frac{N}{2048})} \quad (2)$$

$$M + \frac{N}{2048} = \frac{20.48\ \text{MHz}}{Baud\ Rate \times 2^{CD} \times 16 \times DL \times 2}$$

where:
CD is the clock divider.
DL is the divisor latch.
M is the integer part of the divisor; a fractional divider divides an input by a nonwhole number M.N.
N is the fractional part of the divisor; a fractional divider divides an input by a nonwhole number M.N.

Table 79 lists common baud rate values.

Table 79. Baud Rate Using the Fractional Baud Rate Generator

Baud Rate (bps)	CD	DL	M	N	Actual Baud Rate	% Error
9600	0	0x42	1	21	9598.55	0.015%
19,200	0	0x21	1	21	19,197.09	0.015%
115,200	0	0x5	1	228	115,177.51	0.0195%

UART REGISTER DEFINITION

The UART interface consists of the following nine registers:

- COMTX: 8-bit transmit register
- COMRX: 8-bit receive register
- COMDIV0: divisor latch (low byte)
- COMDIV1: divisor latch (high byte)
- COMCON0: line control register
- COMSTA0: line status register
- COMIEN0: interrupt enable register
- COMIID0: interrupt identification register
- COMDIV2: 16-bit fractional baud divider register

COMTX, COMRX, and COMDIV0 share the same address location. COMTX and COMRX can be accessed when Bit 7 in the COMCON0 register is cleared, and COMDIV0 can be accessed when Bit 7 of COMCON0 is set.

UART TX Register

Name: COMTX

Address: 0xFFFF0700

Access: Write only

Function: Write to this 8-bit register to transmit data using the UART.

UART RX Register

Name: COMRX

Address: 0xFFFF0700

Default Value: 0x00

Access: Read only

Function: This 8-bit register is read from to receive data transmitted using the UART.

UART Divisor Latch Register 0

Name: COMDIV0

Address: 0xFFFF0700

Default Value: 0x00

Access: Read/write

Function: This 8-bit register contains the LSB of the divisor latch that controls the baud rate at which the UART operates.

UART Divisor Latch Register 1

Name: COMDIV1

Address: 0xFFFF0704

Default Value: 0x00

Access: Read/write

Function: This 8-bit register contains the MSB of the divisor latch that controls the baud rate at which the UART operates.

UART Control Register 0

Name: COMCON0

Address: 0xFFFF070C

Default Value: 0x00

Access: Read/write

Function: This 8-bit register controls the operation of the UART in conjunction with COMCON1.

UART Interrupt Enable Register 0

Name: COMIEN0

Address: 0xFFFF0704

Default Value: 0x00

Access: Read/write

Function: The 8-bit register enables and disables the individual UART interrupt sources.

Table 83. COMIEN0 MMR Bit Designations

Bit	Name	Description
7 to 4	N/A	Reserved. Not used.
3	N/A	Reserved. This bit should be written as 0.
2	ELSI	RxD status interrupt enable bit. Set by the user to enable generation of an interrupt if any of the COMSTA0[3:1] register bits are set. Cleared by the user.
1	ETBEI	Enable transmit buffer empty interrupt. Set by the user to enable an interrupt when the buffer is empty during a transmission, that is, when COMSTA[5] is set. Cleared by the user.
0	ERBFI	Enable receive buffer full interrupt. Set by the user to enable an interrupt when the buffer is full during a reception. Cleared by the user.

UART Interrupt Identification Register 0

Name: COMIID0

Address: 0xFFFF0708

Default Value: 0x01

Access: Read only

Function: This 8-bit register reflects the source of the UART interrupt.

Table 84. COMIID0 MMR Bit Designations

Bits[2:1] Status Bits	Bit 0 NINT	Priority	Description	Clearing Operation
00	1	N/A	No interrupt	N/A
11	0	1	Receive line status interrupt	Read COMSTA0
10	0	2	Receive buffer full interrupt	Read COMRX
01	0	3	Transmit buffer empty interrupt	Write data to COMTX or read COMIID0
00	0	4	Modem status interrupt	Read COMSTA1 register

UART Fractional Divider Register

Name: COMDIV2

Address: 0xFFFF072C

Default Value: 0x0000

Access: Read/write

Function: This 16-bit register controls the operation of the fractional divider for the ADuC7034.

Table 85. COMDIV2 MMR Bit Designations

Bit	Name	Description
15	FBEN	Fractional baud rate generator enable bit. Set by the user to enable the fractional baud rate generator. Cleared by the user to generate the baud rate using the standard 450 UART baud rate generator.
14 to 13		Reserved.
12 to 11	FBM[1:0]	Fractional Divider M. If FBM = 0, M = 4. See Equation 2 for the calculation of the baud rate using the M fractional divider and Table 79 for common baud rate values.
10 to 0	FBN[10:0]	Fractional Divider N. See Equation 2 for the calculation of the baud rate using the N fractional divider and Table 79 for common baud rate values.

SPI Receive Register

Name: SPIRX

Address: 0xFFFF0A04

Default Value: 0x00

Access: Read only

Function: This 8-bit MMR contains the data received using the serial peripheral interface.

SPI Transmit Register

Name: SPITX

Address: 0xFFFF0A08

Access: Write only

Function: Write to this 8-bit MMR to transmit data using the serial peripheral interface.

SPI Divider Register

Name: SPIDIV

Address: 0xFFFF0A0C

Default Value: 0x1B

Access: Read/write

Function: The 8-bit MMR represents the frequency at which the serial peripheral interface is operating. For more information on the calculation of the baud rate, refer to Equation 3.

LIN Hardware Synchronization Status Register

Name: LHSSTA

Address: 0xFFFF0780

Default Value: 0x00

Access: Read only

Function: The LHS status register is an 8-bit register whose bits reflect the current operating status of the LIN interface.

Table 91. LHSSTA MMR Bit Designations

Bit	Description
7	Reserved. These read only bits are reserved for future use.
6	Rising edge detected (BSD mode only). Set to 1 by hardware to indicate a rising edge has been detected on the BSD bus. Cleared to 0 after user code reads the LHSSTA MMR.
5	LHS reset complete flag. Set to 1 by hardware to indicate a LHS reset command has completed successfully. Cleared to 0, after user code reads the LHSSTA MMR.
4	Break field error. Set to 1 by hardware and generates an LHS interrupt (IRQEN[7]) when the 12-bit break timer (LHSVAL1) register overflows to indicate the LIN bus has stayed low too long, thus suggesting a possible LIN bus error. Cleared to 0 after user code reads the LHSSTA MMR.
3	LHS compare interrupt. Set to 1 by hardware when the value in LHSVAL0 (LIN synchronization bit timer) equals the value in the LHSCMP register. Cleared to 0 after user code reads the LHSSTA MMR.
2	Stop condition interrupt. Set to 1 by hardware when a stop condition is detected. Cleared to 0 after user code reads LHSSTA MMR.
1	Start condition interrupt. Set to 1 by hardware when a start condition is detected. Cleared to 0 after user code reads LHSSTA MMR.
0	Break timer compare interrupt. Set to 1 by hardware when a valid LIN break condition is detected. A LIN break condition is generated when the LIN break timer value reaches the break timer compare value (see LHSVAL1 in the LIN Hardware Break Timer1 Register section for more information). Cleared to 0 after user code reads the LHSSTA MMR.

OUTLINE DIMENSIONS

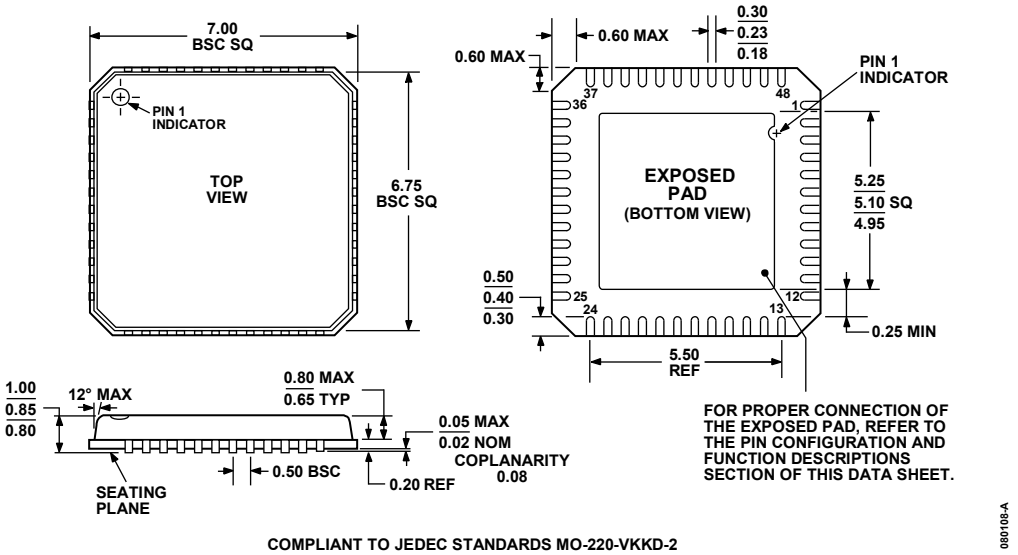

Figure 60. 48-Lead Lead Frame Chip Scale Package [LFCSP_VQ]
7 mm × 7 mm Body, Very Thin Quad
(CP-48-1)
Dimensions shown in millimeters

ORDERING GUIDE

Model[1]	Temperature Range	Package Description	Package Option
ADuC7034BCPZ	−40°C to +115°C	48-Lead Lead Frame Chip Scale Package [LFCSP_VQ]	CP-48-1
ADuC7034BCPZ-RL	−40°C to +115°C	48-Lead Lead Frame Chip Scale Package [LFCSP_VQ]	CP-48-1

[1] Z = RoHS Compliant Part.