

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 "<u>Embedded -</u> <u>Microcontrollers</u>"

#### Details

E·XFl

Product Status	Active
Core Processor	ARM7®
Core Size	16/32-Bit
Speed	44MHz
Connectivity	EBI/EMI, I <sup>2</sup> C, SPI, UART/USART
Peripherals	PLA, PWM, PSM, Temp Sensor, WDT
Number of I/O	40
Program Memory Size	62KB (31K x16)
Program Memory Type	FLASH
EEPROM Size	
RAM Size	2K x 32
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 12 x12b; D/A 4x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	80-LQFP
Supplier Device Package	80-LQFP (12x12)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/aduc7026bstz62-rl

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

## **GENERAL DESCRIPTION**

The ADuC7019/20/21/22/24/25/26/27/28/29 are fully integrated, 1 MSPS, 12-bit data acquisition systems incorporating high performance multichannel ADCs, 16-bit/32-bit MCUs, and Flash\*/EE memory on a single chip.

The ADC consists of up to 12 single-ended inputs. An additional four inputs are available but are multiplexed with the four DAC output pins. The four DAC outputs are available only on certain models (ADuC7020, ADuC7026, ADuC7028, and ADuC7029). However, in many cases where the DAC outputs are not present, these pins can still be used as additional ADC inputs, giving a maximum of 16 ADC input channels. The ADC can operate in single-ended or differential input mode. The ADC input voltage is 0 V to  $V_{REF}$ . A low drift band gap reference, temperature sensor, and voltage comparator complete the ADC peripheral set.

Depending on the part model, up to four buffered voltage output DACs are available on-chip. The DAC output range is programmable to one of three voltage ranges. The devices operate from an on-chip oscillator and a PLL generating an internal high frequency clock of 41.78 MHz (UCLK). This clock is routed through a programmable clock divider from which the MCU core clock operating frequency is generated. The microcontroller core is an ARM7TDMI\*, 16-bit/32-bit RISC machine, which offers up to 41 MIPS peak performance. Eight kilobytes of SRAM and 62 kilobytes of nonvolatile Flash/EE memory are provided on-chip. The ARM7TDMI core views all memory and registers as a single linear array.

On-chip factory firmware supports in-circuit serial download via the UART or I<sup>2</sup>C serial interface port; nonintrusive emulation is also supported via the JTAG interface. These features are incorporated into a low cost QuickStart<sup>™</sup> development system supporting this MicroConverter<sup>\*</sup> family.

The parts operate from 2.7 V to 3.6 V and are specified over an industrial temperature range of  $-40^{\circ}$ C to  $+125^{\circ}$ C. When operating at 41.78 MHz, the power dissipation is typically 120 mW. The ADuC7019/20/21/22/24/25/26/27/28/29 are available in a variety of memory models and packages (see Ordering Guide).



Figure 2.

### DETAILED BLOCK DIAGRAM



Tuble // of Thinks (Thuse Houe )					
Parameter	Description	Min	Тур	Max	Unit
tsL	SCLK low pulse width <sup>1</sup>		$(SPIDIV + 1) \times t_{HCLK}$		ns
tsн	SCLK high pulse width <sup>1</sup>		$(SPIDIV + 1) \times t_{HCLK}$		ns
t <sub>DAV</sub>	Data output valid after SCLK edge			25	ns
tdosu	Data output setup before SCLK edge			75	ns
<b>t</b> dsu	Data input setup time before SCLK edge <sup>2</sup>	$1 \times t_{UCLK}$			ns
<b>t</b> dhd	Data input hold time after SCLK edge <sup>2</sup>	$2 \times t_{\text{UCLK}}$			ns
t <sub>DF</sub>	Data output fall time		5	12.5	ns
t <sub>DR</sub>	Data output rise time		5	12.5	ns
t <sub>sr</sub>	SCLK rise time		5	12.5	ns
t <sub>SF</sub>	SCLK fall time		5	12.5	ns

### Table 7. SPI Master Mode Timing (Phase Mode = 0)

 $^{1}$  t<sub>HCLK</sub> depends on the clock divider or CD bits in the POWCONMMR. t<sub>HCLK</sub> = t<sub>UCLK</sub>/2<sup>CD</sup>; see Figure 67.

 $^{2}$  t<sub>UCLK</sub> = 23.9 ns. It corresponds to the 41.78 MHz internal clock from the PLL before the clock divider; see Figure 67.



Figure 16. SPI Master Mode Timing (Phase Mode = 0)

Parameter	Description	Min	Тур	Max	Unit
t <sub>cs</sub>	CS to SCLK edge <sup>1</sup>	$(2 \times t_{HCLK}) + (2 \times t_{UCLK})$			ns
t <sub>sL</sub>	SCLK low pulse width <sup>2</sup>		$(SPIDIV + 1) \times t_{HCLK}$		ns
t <sub>sн</sub>	SCLK high pulse width <sup>2</sup>		$(SPIDIV + 1) \times t_{HCLK}$		ns
t <sub>DAV</sub>	Data output valid after SCLK edge			25	ns
<b>t</b> dsu	Data input setup time before SCLK edge <sup>1</sup>	1 × tuclk			ns
t <sub>DHD</sub>	Data input hold time after SCLK edge <sup>1</sup>	$2 \times t_{UCLK}$			ns
t <sub>DF</sub>	Data output fall time		5	12.5	ns
t <sub>DR</sub>	Data output rise time		5	12.5	ns
t <sub>sr</sub>	SCLK rise time		5	12.5	ns
t <sub>sF</sub>	SCLK fall time		5	12.5	ns
tDOCS	Data output valid after CS edge			25	ns
tsfs	CS high after SCLK edge	0			ns

#### Table 9. SPI Slave Mode Timing (Phase Mode = 0)

<sup>1</sup> t<sub>UCLK</sub> = 23.9 ns. It corresponds to the 41.78 MHz internal clock from the PLL before the clock divider; see Figure 67. <sup>2</sup> t<sub>HCLK</sub> depends on the clock divider or CD bits in the POWCONMMR. t<sub>HCLK</sub> = t<sub>UCLK</sub>/2<sup>CD</sup>; see Figure 67.



Figure 18. SPI Slave Mode Timing (Phase Mode = 0)

Pin No.	Mnemonic	Description
18	P4.6/AD14/PLAO[14]	General-Purpose Input and Output Port 4.6/External Memory Interface/Programmable Logic Array Output Element 14.
19	P4.7/AD15/PLAO[15]	General-Purpose Input and Output Port 4.7/External Memory Interface/Programmable Logic Array Output Element 15.
20	BM/P0.0/CMP <sub>out</sub> /PLAI[7]/MS0	Multifunction I/O Pin. Boot Mode. The ADuC7026/ADuC7027 enter UART download mode if BM is low at reset and execute code if BM is pulled high at reset through a 1 k $\Omega$ resistor/General-Purpose Input and Output Port 0.0/Voltage Comparator Output/Programmable Logic Array Input Element 7/External Memory Select 0.
21	P0.6/T1/MRST/PLAO[3]	Multifunction Pin, Driven Low After Reset. General-Purpose Output Port 0.6/Timer1 Input/ Power-On Reset Output/Programmable Logic Array Output Element 3.
22	ТСК	JTAG Test Port Input, Test Clock. Debug and download access.
23	TDO	JTAG Test Port Output, Test Data Out. Debug and download access.
24	P0.2/PWM2 <sub>L</sub> /BHE	General-Purpose Input and Output Port 0.2/PWM Phase 2 Low-Side Output/External Memory Byte High Enable.
25	IOGND	Ground for GPIO (see Table 78). Typically connected to DGND.
26	IOV <sub>DD</sub>	3.3 V Supply for GPIO (see Table 78) and Input of the On-Chip Voltage Regulator.
27	LV <sub>DD</sub>	2.6 V Output of the On-Chip Voltage Regulator. This output must be connected to a 0.47 $\mu\text{F}$ capacitor to DGND only.
28	DGND	Ground for Core Logic.
29	P3.0/AD0/PWM0 <sub>H</sub> /PLAI[8]	General-Purpose Input and Output Port 3.0/External Memory Interface/PWM Phase 0 High-Side Output/Programmable Logic Array Input Element 8.
30	P3.1/AD1/PWM0∟/PLAI[9]	General-Purpose Input and Output Port 3.1/External Memory Interface/PWM Phase 0 Low-Side Output/Programmable Logic Array Input Element 9.
31	P3.2/AD2/PWM1 <sub>H</sub> /PLAI[10]	General-Purpose Input and Output Port 3.2/External Memory Interface/PWM Phase 1 High-Side Output/Programmable Logic Array Input Element 10.
32	P3.3/AD3/PWM1L/PLAI[11]	General-Purpose Input and Output Port 3.3/External Memory Interface/PWM Phase 1 Low-Side Output/Programmable Logic Array Input Element 11.
33	P2.4/PWM0 <sub>H</sub> /MS0	General-Purpose Input and Output Port 2.4/PWM Phase 0 High-Side Output/External Memory Select 0.
34	P0.3/TRST/A16/ADC <sub>BUSY</sub>	General-Purpose Input and Output Port 0.3/JTAG Test Port Input, Test Reset/ADC <sub>BUSY</sub> Signal Output.
35	P2.5/PWM0∟/MS1	General-Purpose Input and Output Port 2.5/PWM Phase 0 Low-Side Output/External Memory Select 1.
36	P2.6/PWM1 <sub>H</sub> /MS2	General-Purpose Input and Output Port 2.6/PWM Phase 1 High-Side Output/External Memory Select 2.
37	RST	Reset Input, Active Low.
38	P3.4/AD4/PWM2 <sub>H</sub> /PLAI[12]	General-Purpose Input and Output Port 3.4/External Memory Interface/PWM Phase 2 High-Side Output/Programmable Logic Array Input 12.
39	P3.5/AD5/PWM2L/PLAI[13]	General-Purpose Input and Output Port 3.5/External Memory Interface/PWM Phase 2 Low-Side Output/Programmable Logic Array Input Element 13.
40	IRQ0/P0.4/PWM <sub>TRIP</sub> /PLAO[1]/MS1	Multifunction I/O Pin. External Interrupt Request 0, Active High/General-Purpose Input and Output Port 0.4/PWM Trip External Input/Programmable Logic Array Output Element 1/ External Memory Select 1.
41	IRQ1/P0.5/ADC <sub>BUSY</sub> /PLAO[2]/MS2	Multifunction I/O Pin. External Interrupt Request 1, Active High/General-Purpose Input and Output Port 0.5/ADC <sub>BUSY</sub> Signal Output/Programmable Logic Array Output Element 2/External Memory Select 2.
42	P2.0/SPM9/PLAO[5]/CONV <sub>START</sub>	Serial Port Multiplexed. General-Purpose Input and Output Port 2.0/UART/Programmable Logic Array Output Element 5/Start Conversion Input Signal for ADC.
43	P0.7/ECLK/XCLK/SPM8/PLAO[4]	Serial Port Multiplexed. General-Purpose Input and Output Port 0.7/Output for External Clock
		Signal/Input to the Internal Clock Generator Circuits/UART/Programmable Logic Array Output Element 4.
44	XCLKO	Output from the Crystal Oscillator Inverter.
45	XCLKI	Input to the Crystal Oscillator Inverter and Input to the Internal Clock Generator Circuits.

## ADUC7028



Figure 26. 64-Ball CSP\_BGA Pin Configuration (ADuC7028)

Pin No.	Mnemonic	Description
A1	ADC3/CMP1	Single-Ended or Differential Analog Input 3/Comparator Negative Input.
A2	DACVDD	3.3 V Power Supply for the DACs. Must be connected to AVDD.
A3	AV <sub>DD</sub>	3.3 V Analog Power.
A4	AGND	Analog Ground. Ground reference point for the analog circuitry.
A5	DACGND	Ground for the DAC. Typically connected to AGND.
A6	P4.2/PLAO[10]	General-Purpose Input and Output Port 4.2/Programmable Logic Array Output Element 10.
A7	P1.1/SPM1/PLAI[1]	Serial Port Multiplexed. General-Purpose Input and Output Port 1.1/UART, I2C0/Programmable Logic Array Input Element 1.
A8	P1.2/SPM2/PLAI[2]	Serial Port Multiplexed. General-Purpose Input and Output Port 1.2/UART, I2C1/Programmable Logic Array Input Element 2.
B1	ADC4	Single-Ended or Differential Analog Input 4.
B2	ADC2/CMP0	Single-Ended or Differential Analog Input 2/Comparator Positive Input.
B3	ADC1	Single-Ended or Differential Analog Input 1.
B4	DAC <sub>REF</sub>	External Voltage Reference for the DACs. Range: DACGND to DACVDD.
B5	V <sub>REF</sub>	2.5 V Internal Voltage Reference. Must be connected to a 0.47 $\mu\text{F}$ capacitor when using the internal reference.
B6	P1.0/T1/SPM0/PLAI[0]	Serial Port Multiplexed. General-Purpose Input and Output Port 1.0/Timer1 Input/UART, I2C0/ Programmable Logic Array Input Element 0.
B7	P1.4/SPM4/PLAI[4]/IRQ2	Serial Port Multiplexed. General-Purpose Input and Output Port 1.4/UART, SPI/Programmable Logic Array Input Element 4/External Interrupt Request 2, Active High.
B8	P1.3/SPM3/PLAI[3]	Serial Port Multiplexed. General-Purpose Input and Output Port 1.3/UART, I2C1/Programmable Logic Array Input Element 3.
C1	ADC6	Single-Ended or Differential Analog Input 6.
C2	ADC5	Single-Ended or Differential Analog Input 5.
C3	ADC0	Single-Ended or Differential Analog Input 0.
C4	P4.5/PLAO[13]	General-Purpose Input and Output Port 4.5/Programmable Logic Array Output Element 13.
C5	P4.3/PLAO[11]	General-Purpose Input and Output Port 4.3/Programmable Logic Array Output Element 11.
C6	P4.0/PLAO[8]	General-Purpose Input and Output Port 4.0/Programmable Logic Array Output Element 8.
C7	P4.1/PLAO[9]	General-Purpose Input and Output Port 4.1/Programmable Logic Array Output Element 9.
C8	IOGND	Ground for GPIO (see Table 78). Typically connected to DGND.
D1	ADCNEG	Bias Point or Negative Analog Input of the ADC in Pseudo Differential Mode. Must be connected to the ground of the signal to convert. This bias point must be between 0 V and 1 V.
D2	GND <sub>REF</sub>	Ground Voltage Reference for the ADC. For optimal performance, the analog power supply should be separated from IOGND and DGND.
D3	ADC7	Single-Ended or Differential Analog Input 7.
D4	P4.4/PLAO[12]	General-Purpose Input and Output Port 4.4/Programmable Logic Array Output Element 12.
D5	P3.6/PWM <sub>TRIP</sub> /PLAI[14]	General-Purpose Input and Output Port 3.6/PWM Safety Cutoff/Programmable Logic Array Input Element 14.
D6	P1.7/SPM7/PLAO[0]	Serial Port Multiplexed. General-Purpose Input and Output Port 1.7/UART, SPI/Programmable Logic Array Output Element 0.

Pin No.	Mnemonic	Description
D7	P1.6/SPM6/PLAI[6]	Serial Port Multiplexed. General-Purpose Input and Output Port 1.6/UART, SPI/Programmable Logic Array Input Element 6.
D8	IOV <sub>DD</sub>	3.3 V Supply for GPIO (see Table 78) and Input of the On-Chip Voltage Regulator.
E1	DAC3/ADC15	DAC3 Voltage Output/ADC Input 15.
E2	DAC2/ADC14	DAC2 Voltage Output/ADC Input 14.
E3	DAC1/ADC13	DAC1 Voltage Output/ADC Input 13.
E4	P3.0/PWM0 <sub>H</sub> /PLAI[8]	General-Purpose Input and Output Port 3.0/PWM Phase 0 High-Side Output/Programmable Logic Array Input Element 8.
E5	P3.2/PWM1 <sub>H</sub> /PLAI[10]	General-Purpose Input and Output Port 3.2/PWM Phase 1 High-Side Output/Programmable Logic Array Input Element 10.
E6	P1.5/SPM5/PLAI[5]/IRQ3	Serial Port Multiplexed. General-Purpose Input and Output Port 1.5/UART, SPI/Programmable Logic Array Input Element 5/External Interrupt Request 3, Active High.
E7	P3.7/PWM <sub>SYNC</sub> /PLAI[15]	General-Purpose Input and Output Port 3.7/PWM Synchronization/Programmable Logic Array Input Element 15.
E8	XCLKI	Input to the Crystal Oscillator Inverter and Input to the Internal Clock Generator Circuits.
F1	P4.6/PLAO[14]	General-Purpose Input and Output Port 4.6/Programmable Logic Array Output Element 14.
F2	TDI	JTAG Test Port Input, Test Data In. Debug and download access.
F3		DAC0 Voltage Output/ADC Input 12
F4	P3.1/PWM0L/PLAI[9]	General-Purpose Input and Output Port 3.1/PWM Phase 0 Low-Side Output/Programmable
F5	P3.3/PWM1_/PLAI[11]	General-Purpose Input and Output Port 3.3/PWM Phase 1 Low-Side Output/Programmable
F6	RST	Reset Input, Active Low.
F7	P0.7/ECLK/XCLK/SPM8/PLAO[4]	Serial Port Multiplexed. General-Purpose Input and Output Port 0.7/Output for External Clock Signal/Input to the Internal Clock Generator Circuits/UART/Programmable Logic Array Output Element 4.
F8	XCLKO	Output from the Crystal Oscillator Inverter.
G1	BM/P0.0/CMP <sub>out</sub> /PLAI[7]	Multifunction I/O Pin. Boot mode. The ADuC7028 enters UART download mode if BM is low at reset and executes code if BM is pulled high at reset through a 1 k $\Omega$ resistor/General-Purpose Input and Output Port 0.0/Voltage Comparator Output/Programmable Logic Array Input Element 7.
G2	P4.7/PLAO[15]	General-Purpose Input and Output Port 4.7/Programmable Logic Array Output Element 15.
G3	TMS	JTAG Test Port Input, Test Mode Select. Debug and download access.
G4	TDO	JTAG Test Port Output, Test Data Out. Debug and download access.
G5	P0.3/TRST/ADC <sub>BUSY</sub>	General-Purpose Input and Output Port 0.3/JTAG Test Port Input, Test Reset/ADC <sub>BUSY</sub> Signal
66	Ρ3 4/Ρ\ΜΜ2/ΡΙ ΔΙ[12]	Output. General-Purpose Input and Output Port 3 4/PWM Phase 2 High-Side Output/Programmable
60		Logic Array Input 12.
G/	P3.5/PWM2L/PLAI[13]	General-Purpose Input and Output Port 3.5/PWM Phase 2 Low-Side Output/Programmable Logic Array Input Element 13.
G8	P2.0/SPM9/PLAO[5]/CONV <sub>START</sub>	Serial Port Multiplexed. General-Purpose Input and Output Port 2.0/UART/Programmable Logic Array Output Element 5/Start Conversion Input Signal for ADC.
H1	P0.6/T1/MRST/PLAO[3]	Multifunction Pin, Driven Low After Reset. General-Purpose Output Port 0.6/Timer1 Input/ Power-On Reset Output/Programmable Logic Array Output Element 3.
H2	ТСК	JTAG Test Port Input, Test Clock. Debug and download access.
H3	IOGND	Ground for GPIO (see Table 78). Typically connected to DGND.
H4	IOV <sub>DD</sub>	3.3 V Supply for GPIO (see Table 78) and Input of the On-Chip Voltage Regulator.
H5	LV <sub>DD</sub>	2.6 V Output of the On-Chip Voltage Regulator. This output must be connected to a 0.47 $\mu\text{F}$ capacitor to DGND only.
H6	DGND	Ground for Core Logic.
H7	IRQ0/P0.4/PWM <sub>TRIP</sub> /PLAO[1]	Multifunction I/O Pin. External Interrupt Request 0, Active High/General-Purpose Input and Output Port 0.4/PWM Trip External Input/Programmable Logic Array Output Element 1.
H8	IRQ1/P0.5/ADC <sub>BUSY</sub> /PLAO[2]	Multifunction I/O Pin. External Interrupt Request 1, Active High/General-Purpose Input and Output Port 0.5/ADC <sub>BUSY</sub> Signal Output/Programmable Logic Array Output Element 2.





















Figure 39. Current Consumption vs. Temperature @ CD = 0

## **OVERVIEW OF THE ARM7TDMI CORE**

The ARM7° core is a 32-bit reduced instruction set computer (RISC). It uses a single 32-bit bus for instruction and data. The length of the data can be eight bits, 16 bits, or 32 bits. The length of the instruction word is 32 bits.

The ARM7TDMI is an ARM7 core with four additional features.

- T support for the thumb (16-bit) instruction set.
- D support for debug.
- M support for long multiplications.
- I includes the EmbeddedICE module to support embedded system debugging.

## THUMB MODE (T)

An ARM instruction is 32 bits long. The ARM7TDMI processor supports a second instruction set that is compressed into 16 bits, called the thumb instruction set. Faster execution from 16-bit memory and greater code density can usually be achieved by using the thumb instruction set instead of the ARM instruction set, which makes the ARM7TDMI core particularly suitable for embedded applications.

However, the thumb mode has two limitations.

- Thumb code typically requires more instructions for the same job. As a result, ARM code is usually best for maximizing the performance of time-critical code.
- The thumb instruction set does not include some of the instructions needed for exception handling, which automatically switches the core to ARM code for exception handling.

See the ARM7TDMI user guide for details on the core architecture, the programming model, and both the ARM and ARM thumb instruction sets.

## LONG MULTIPLY (M)

The ARM7TDMI instruction set includes four extra instructions that perform 32-bit by 32-bit multiplication with a 64-bit result, and 32-bit by 32-bit multiplication-accumulation (MAC) with a 64-bit result. These results are achieved in fewer cycles than required on a standard ARM7 core.

## EmbeddedICE (I)

EmbeddedICE provides integrated on-chip support for the core. The EmbeddedICE module contains the breakpoint and watchpoint registers that allow code to be halted for debugging purposes. These registers are controlled through the JTAG test port.

When a breakpoint or watchpoint is encountered, the processor halts and enters debug state. Once in a debug state, the processor registers can be inspected as well as the Flash/EE, SRAM, and memory mapped registers.

### **EXCEPTIONS**

ARM supports five types of exceptions and a privileged processing mode for each type. The five types of exceptions are

- Normal interrupt or IRQ, which is provided to service general-purpose interrupt handling of internal and external events.
- Fast interrupt or FIQ, which is provided to service data transfers or communication channels with low latency. FIQ has priority over IRQ.
- Memory abort.
- Attempted execution of an undefined instruction.
- Software interrupt instruction (SWI), which can be used to make a call to an operating system.

Typically, the programmer defines interrupt as IRQ, but for higher priority interrupt, that is, faster response time, the programmer can define interrupt as FIQ.

## **ARM REGISTERS**

ARM7TDMI has a total of 37 registers: 31 general-purpose registers and six status registers. Each operating mode has dedicated banked registers.

When writing user-level programs, 15 general-purpose 32-bit registers (R0 to R14), the program counter (R15), and the current program status register (CPSR) are usable. The remaining registers are used for system-level programming and exception handling only.

When an exception occurs, some of the standard registers are replaced with registers specific to the exception mode. All exception modes have replacement banked registers for the stack pointer (R13) and the link register (R14), as represented in Figure 44. The fast interrupt mode has more registers (R8 to R12) for fast interrupt processing. This means that interrupt processing can begin without the need to save or restore these registers and, thus, save critical time in the interrupt handling process.



## ADC CIRCUIT OVERVIEW

The analog-to-digital converter (ADC) incorporates a fast, multichannel, 12-bit ADC. It can operate from 2.7 V to 3.6 V supplies and is capable of providing a throughput of up to 1 MSPS when the clock source is 41.78 MHz. This block provides the user with a multichannel multiplexer, a differential track-and-hold, an on-chip reference, and an ADC.

The ADC consists of a 12-bit successive approximation converter based around two capacitor DACs. Depending on the input signal configuration, the ADC can operate in one of three modes.

- Fully differential mode, for small and balanced signals
- Single-ended mode, for any single-ended signals
- Pseudo differential mode, for any single-ended signals, taking advantage of the common-mode rejection offered by the pseudo differential input

The converter accepts an analog input range of 0 V to  $V_{\text{REF}}$  when operating in single-ended or pseudo differential mode. In fully differential mode, the input signal must be balanced around a common-mode voltage ( $V_{\text{CM}}$ ) in the 0 V to  $AV_{\text{DD}}$  range with a maximum amplitude of 2  $V_{\text{REF}}$  (see Figure 48).



Figure 48. Examples of Balanced Signals in Fully Differential Mode

A high precision, low drift, factory calibrated, 2.5 V reference is provided on-chip. An external reference can also be connected as described in the Band Gap Reference section.

Single or continuous conversion modes can be initiated in the software. An external CONV<sub>START</sub> pin, an output generated from the on-chip PLA, or a Timer0 or Timer1 overflow can also be used to generate a repetitive trigger for ADC conversions.

A voltage output from an on-chip band gap reference proportional to absolute temperature can also be routed through the front-end ADC multiplexer, effectively an additional ADC channel input. This facilitates an internal temperature sensor channel that measures die temperature to an accuracy of  $\pm 3^{\circ}$ C.

## **TRANSFER FUNCTION**

#### **Pseudo Differential and Single-Ended Modes**

In pseudo differential or single-ended mode, the input range is 0 V to  $V_{\text{REF}}$ . The output coding is straight binary in pseudo differential and single-ended modes with

1 LSB = *FS*/4096, or 2.5 V/4096 = 0.61 mV, or 610  $\mu$ V when *V<sub>REF</sub>* = 2.5 V The ideal code transitions occur midway between successive integer LSB values (that is, 1/2 LSB, 3/2 LSB, 5/2 LSB, ..., FS – 3/2 LSB). The ideal input/output transfer characteristic is shown in Figure 49.



Figure 49. ADC Transfer Function in Pseudo Differential or Single-Ended Mode

### Fully Differential Mode

The amplitude of the differential signal is the difference between the signals applied to the  $V_{\rm IN+}$  and  $V_{\rm IN-}$  input voltage pins (that is,  $V_{\rm IN+}-V_{\rm IN-}$ ). The maximum amplitude of the differential signal is, therefore,  $-V_{\rm REF}$  to  $+V_{\rm REF}$  p-p (that is,  $2\times V_{\rm REF}$ ). This is regardless of the common mode (CM). The common mode is the average of the two signals, for example,  $(V_{\rm IN+}+V_{\rm IN-})/2$ , and is, therefore, the voltage that the two inputs are centered on. This results in the span of each input being CM  $\pm$   $V_{\rm REF}/2$ . This voltage has to be set up externally, and its range varies with  $V_{\rm REF}$  (see the Driving the Analog Inputs section).

The output coding is twos complement in fully differential mode with 1 LSB = 2 V<sub>REF</sub>/4096 or 2 × 2.5 V/4096 = 1.22 mV when V<sub>REF</sub> = 2.5 V. The output result is ±11 bits, but this is shifted by 1 to the right. This allows the result in ADCDAT to be declared as a signed integer when writing C code. The designed code transitions occur midway between successive integer LSB values (that is, 1/2 LSB, 3/2 LSB, 5/2 LSB, ..., FS – 3/2 LSB). The ideal input/output transfer characteristic is shown in Figure 50.



Figure 50. ADC Transfer Function in Differential Mode

#### Table 28. V<sub>CM</sub> Ranges

	-	0		
AV <sub>DD</sub>	VREF	V <sub>CM</sub> Min	<b>V</b> см Мах	Signal Peak-to-Peak
3.3 V	2.5 V	1.25 V	2.05 V	2.5 V
	2.048 V	1.024 V	2.276 V	2.048 V
	1.25 V	0.75 V	2.55 V	1.25 V
3.0 V	2.5 V	1.25 V	1.75 V	2.5 V
	2.048 V	1.024 V	1.976 V	2.048 V
	1.25 V	0.75 V	2.25 V	1.25 V

## CALIBRATION

By default, the factory-set values written to the ADC offset (ADCOF) and gain coefficient registers (ADCGN) yield optimum performance in terms of end-point errors and linearity for standalone operation of the part (see the Specifications section). If system calibration is required, it is possible to modify the default offset and gain coefficients to improve end-point errors, but note that any modification to the factory-set ADCOF and ADCGN values can degrade ADC linearity performance.

For system offset error correction, the ADC channel input stage must be tied to AGND. A continuous software ADC conversion loop must be implemented by modifying the value in ADCOF until the ADC result (ADCDAT) reads Code 0 to Code 1. If the ADCDAT value is greater than 1, ADCOF should be decremented until ADCDAT reads 0 to 1. Offset error correction is done digitally and has a resolution of 0.25 LSB and a range of  $\pm 3.125\%$  of V<sub>REF</sub>.

For system gain error correction, the ADC channel input stage must be tied to  $V_{REF}$ . A continuous software ADC conversion loop must be implemented to modify the value in ADCGN until the ADC result (ADCDAT) reads Code 4094 to Code 4095. If the ADCDAT value is less than 4094, ADCGN should be incremented until ADCDAT reads 4094 to 4095. Similar to the offset calibration, the gain calibration resolution is 0.25 LSB with a range of  $\pm 3\%$  of V<sub>REF</sub>.

## **TEMPERATURE SENSOR**

The ADuC7019/20/21/22/24/25/26/27/28/29 provide voltage output from on-chip band gap references proportional to absolute temperature. This voltage output can also be routed through the front-end ADC multiplexer (effectively an additional ADC channel input) facilitating an internal temperature sensor channel, measuring die temperature to an accuracy of  $\pm 3^{\circ}$ C.

The following is an example routine showing how to use the internal temperature sensor:

```
int main(void)
{
float a = 0;
   short b;
   ADCCON = 0x20; // power-on the ADC
   delay(2000);
```

```
ADCCP = 0x10; // Select Temperature
Sensor as an // input to the ADC
     REFCON = 0x01; // connect internal 2.5V
reference // to Vref pin
     ADCCON = 0xE4; // continuous conversion
     while(1)
     {
             while (!ADCSTA){};
     // wait for end of conversion
             b = (ADCDAT >> 16);
     // To calculate temperature in °C, use
the formula:
             a = 0x525 - b;
     // ((Temperature = 0x525 - Sensor
Voltage) / 1.3)
             a /= 1.3;
             b = floor(a);
             printf("Temperature: %d
oC\n",b);
     }
     return 0;
}
```

## **BAND GAP REFERENCE**

Each ADuC7019/20/21/22/24/25/26/27/28/29 provides an onchip band gap reference of 2.5 V, which can be used for the ADC and DAC. This internal reference also appears on the V<sub>REF</sub> pin. When using the internal reference, a 0.47  $\mu$ F capacitor must be connected from the external V<sub>REF</sub> pin to AGND to ensure stability and fast response during ADC conversions. This reference can also be connected to an external pin (V<sub>REF</sub>) and used as a reference for other circuits in the system. An external buffer is required because of the low drive capability of the V<sub>REF</sub> output. A programmable option also allows an external reference input on the V<sub>REF</sub> pin. Note that it is not possible to disable the internal reference. Therefore, the external reference source must be capable of overdriving the internal reference source.

Table 29. REFCON Register

Name	Address	Default Value	Access
REFCON	0xFFFF048C	0x00	R/W

The band gap reference interface consists of an 8-bit MMR REFCON, described in Table 30.

Table 30. REFCON MMR Bit Designations

Bit	Description
7:1	Reserved.
0	Internal reference output enable. Set by user to connect the internal 2.5 V reference to the V <sub>REF</sub> pin. The reference can be used for an external component but must be buffered. Cleared by user to disconnect the reference from the V <sub>REF</sub> pin.

Linearity degradation near ground and AV<sub>DD</sub> is caused by saturation of the output amplifier, and a general representation of its effects (neglecting offset and gain error) is illustrated in Figure 64. The dotted line in Figure 64 indicates the ideal transfer function, and the solid line represents what the transfer function may look like with endpoint nonlinearities due to saturation of the output amplifier. Note that Figure 64 represents a transfer function in 0-to-AV<sub>DD</sub> mode only. In 0-to-V<sub>REF</sub> or 0-to-DAC<sub>REF</sub> mode (with V<sub>REF</sub> < AV<sub>DD</sub> or DAC<sub>REF</sub> < AV<sub>DD</sub>), the lower nonlinearity is similar. However, the upper portion of the transfer function follows the ideal line right to the end (V<sub>REF</sub> in this case, not AV<sub>DD</sub>), showing no signs of endpoint linearity errors.



Figure 64. Endpoint Nonlinearities Due to Amplifier Saturation

The endpoint nonlinearities conceptually illustrated in Figure 64 get worse as a function of output loading. Most of the ADuC7019/20/21/22/24/25/26/27/28/29 data sheet specifications assume a 5 k $\Omega$  resistive load to ground at the DAC output. As the output is forced to source or sink more current, the nonlinear regions at the top or bottom (respectively) of Figure 64 become larger. With larger current demands, this can significantly limit output voltage swing.

## **POWER SUPPLY MONITOR**

The power supply monitor regulates the  $IOV_{DD}$  supply on the ADuC7019/20/21/22/24/25/26/27/28/29. It indicates when the  $IOV_{DD}$  supply pin drops below one of two supply trip points. The monitor function is controlled via the PSMCON register. If enabled in the IRQEN or FIQEN register, the monitor interrupts the core using the PSMI bit in the PSMCON MMR. This bit is immediately cleared after CMP goes high.

This monitor function allows the user to save working registers to avoid possible data loss due to low supply or brown-out conditions. It also ensures that normal code execution does not resume until a safe supply level is established.

#### Table 53. PSMCON Register

Name	Address	Default Value	Access
PSMCON	0xFFFF0440	0x0008	R/W

# ADuC7019/20/21/22/24/25/26/27/28/29

#### Table 54. PSMCON MMR Bit Descriptions

Bit	Name	Description
3	СМР	Comparator bit. This is a read-only bit that directly reflects the state of the comparator. Read 1 indicates that the IOV <sub>DD</sub> supply is above its selected trip point or that the PSM is in power-down mode. Read 0 indicates that the IOV <sub>DD</sub> supply is below its selected trip point. This bit should be set before leaving the interrupt service routine.
2	TP	Trip point selection bit. $0 = 2.79 \text{ V}$ , $1 = 3.07 \text{ V}$ .
1	PSMEN	Power supply monitor enable bit. Set to 1 to enable the power supply monitor circuit. Cleared to 0 to disable the power supply monitor circuit.
0	PSMI	Power supply monitor interrupt bit. This bit is set high by the MicroConverter after CMP goes low, indicating low I/O supply. The PSMI bit can be used to interrupt the processor. After CMP returns high, the PSMI bit can be cleared by writing a 1 to this location. A 0 write has no effect. There is no timeout delay; PSMI can be immediately cleared after CMP goes high.

## COMPARATOR

The ADuC7019/20/21/22/24/25/26/27/28/29 integrate voltage comparators. The positive input is multiplexed with ADC2, and the negative input has two options: ADC3 and DAC0. The output of the comparator can be configured to generate a system interrupt, be routed directly to the programmable logic array, start an ADC conversion, or be on an external pin, CMP<sub>OUT</sub>, as shown in Figure 65.



Note that because the ADuC7022, ADuC7025, and ADu7027 parts do not support a DAC0 output, it is not possible to use DAC0 as a comparator input on these parts.

## Hysteresis

Figure 66 shows how the input offset voltage and hysteresis terms are defined.



Figure 66. Comparator Hysteresis Transfer Function

## **Output Control Unit**

The operation of the output control unit is controlled by the 9-bit read/write PWMEN register. This register controls two distinct features of the output control unit that are directly useful in the control of electronic counter measures (ECM) or binary decimal counter measures (BDCM). The PWMEN register contains three crossover bits, one for each pair of PWM outputs. Setting Bit 8 of the PWMEN register enables the crossover mode for the 0H/0L pair of PWM signals, setting Bit 7 enables crossover on the 1H/1L pair of PWM signals, and setting Bit 6 enables crossover on the 2H/2L pair of PWM signals. If crossover mode is enabled for any pair of PWM signals, the high-side PWM signal from the timing unit (0H, for example) is diverted to the associated low-side output of the output control unit so that the signal ultimately appears at the PWM0<sub>L</sub> pin. Of course, the corresponding low-side output of the timing unit is also diverted to the complementary high-side output of the output control unit so that the signal appears at the PWM0<sub>H</sub> pin. Following a reset, the three crossover bits are cleared, and the crossover mode is disabled on all three pairs of PWM signals. The PWMEN register also contains six bits (Bit 0 to Bit 5) that can be used to individually enable or disable each of the six PWM outputs. If the associated bit of the PWMEN register is set, the corresponding PWM output is disabled regardless of the corresponding value of the duty cycle register. This PWM output signal remains in the off state as long as the corresponding enable/disable bit of the PWMEN register is set. The implementation of this output enable function is implemented after the crossover function.

Following a reset, all six enable bits of the PWMEN register are cleared, and all PWM outputs are enabled by default. In a manner identical to the duty cycle registers, the PWMEN is latched on the rising edge of the PWMSYNC signal. As a result, changes to this register become effective only at the start of each PWM cycle in single update mode. In double update mode, the PWMEN register can also be updated at the midpoint of the PWM cycle.

In the control of an ECM, only two inverter legs are switched at any time, and often the high-side device in one leg must be switched on at the same time as the low-side driver in a second leg. Therefore, by programming identical duty cycle values for two PWM channels (for example, PWMCH0 = PWMCH1) and setting Bit 7 of the PWMEN register to cross over the 1H/1L pair of PWM signals, it is possible to turn on the high-side switch of Phase A and the low-side switch of Phase B at the same time. In the control of ECM, it is usual for the third inverter leg (Phase C in this example) to be disabled for a number of PWM cycles. This function is implemented by disabling both the 2H and 2L PWM outputs by setting Bit 0 and Bit 1 of the PWMEN register.

This situation is illustrated in Figure 71, where it can be seen that both the 0H and 1L signals are identical because PWMCH0 = PWMCH1 and the crossover bit for Phase B is set.

# ADuC7019/20/21/22/24/25/26/27/28/29



In addition, the other four signals (0L, 1H, 2H, and 2L) have been disabled by setting the appropriate enable/disable bits of the PWMEN register. In Figure 71, the appropriate value for the PWMEN register is 0x00A7. In normal ECM operation, each inverter leg is disabled for certain periods of time to change the PWMEN register based on the position of the rotor shaft (motor commutation).

#### **Gate Drive Unit**

The gate drive unit of the PWM controller adds features that simplify the design of isolated gate-drive circuits for PWM inverters. If a transformer-coupled, power device, gate-drive amplifier is used, the active PWM signal must be chopped at a high frequency. The 16-bit read/write PWMCFG register programs this high frequency chopping mode. The chopped active PWM signals can be required for the high-side drivers only, the low-side drivers only, or both the high-side and lowside switches. Therefore, independent control of this mode for both high-side and low-side switches is included with two separate control bits in the PWMCFG register.

Typical PWM output signals with high frequency chopping enabled on both high-side and low-side signals are shown in Figure 72. Chopping of the high-side PWM outputs (0H, 1H, and 2H) is enabled by setting Bit 8 of the PWMCFG register. Chopping of the low-side PWM outputs (0L, 1L, and 2L) is enabled by setting Bit 9 of the PWMCFG register. The high chopping frequency is controlled by the 8-bit word (GDCLK) placed in Bit 0 to Bit 7 of the PWMCFG register. The period of this high frequency carrier is

 $t_{CHOP} = (4 \times (GDCLK + 1)) \times t_{CORE}$ 

The chopping frequency is, therefore, an integral subdivision of the MicroConverter core frequency

 $f_{CHOP} = f_{CORE}/(4 \times (GDCLK + 1))$ 

# ADuC7019/20/21/22/24/25/26/27/28/29



### Table 83. GPIO Drive Strength Control Bits Descriptions

The drive strength bits can be written to one time only after reset. More writing to related bits has no effect on changing drive strength. The GPIO drive strength and pull-up disable is not always adjustable for the GPIO port. Some control bits cannot be changed (see Table 84).

Bit	GPOPAR	GP1PAR
31	Reserved	Reserved
30 to 29	R/W	R/W
28	R/W	R/W
27	Reserved	Reserved
26 to 25	R/W	R/W
24	R/W	R/W
23	Reserved	Reserved
22 to 21	R/W	R (b00)
20	R/W	R/W
19	Reserved	Reserved
18 to 17	R (b00)	R (b00)
16	R/W	R/W
15	Reserved	Reserved
14 to 13	R (b00)	R (b00)
12	R/W	R/W
11	Reserved	Reserved
10 to 9	R (b00)	R (b00)
8	R/W	R/W
7	Reserved	Reserved
6 to 5	R (b00)	R (b00)
4	R/W	R/W
3	Reserved	Reserved
2 to 1	R (b00)	R (b00)
0	R/W	R/W

#### Table 84. GPxPAR Control Bits Access Descriptions

#### Table 108. COMSTA1 Register

Name	Address	Default Value	Access
COMSTA1	0xFFFF0718	0x00	R

COMSTA1 is a modem status register.

#### Table 109. COMSTA1 MMR Bit Descriptions

Bit	Name	Description
7	DCD	Data carrier detect.
6	RI	Ring indicator.
5	DSR	Data set ready.
4	CTS	Clear to send.
3	DDCD	Delta DCD. Set automatically if DCD changed state since last COMSTA1 read. Cleared automati-cally by reading COMSTA1.
2	TERI	Trailing edge RI. Set if RI changed from 0 to 1 since COMSTA1 was last read. Cleared automatically by reading COMSTA1.
1	DDSR	Delta DSR. Set automatically if DSR changed state since COMSTA1 was last read. Cleared automatically by reading COMSTA1.
0	DCTS	Delta CTS. Set automatically if CTS changed state since COMSTA1 was last read. Cleared automatically by reading COMSTA1.

#### Table 110. COMSCR Register

Name	Address	Default Value	Access
COMSCR	0xFFFF071C	0x00	R/W

COMSCR is an 8-bit scratch register used for temporary storage. It is also used in network addressable UART mode.

#### Table 111. COMDIV2 Register

Name	Address	Default Value	Access
COMDIV2	0xFFFF072C	0x0000	R/W

COMDIV2 is a 16-bit fractional baud divide register.

#### Table 112. COMDIV2 MMR Bit Descriptions

Bit	Name	Description
15	FBEN	Fractional baud rate generator enable bit. Set by user to enable the fractional baud rate generator. Cleared by user to generate baud rate using the standard 450 UART baud rate generator.
14:13		Reserved.
12:11	FBM[1:0]	M if FBM = 0, M = 4 (see the Fractional Divider section).
10:0	FBN[10:0]	N (see the Fractional Divider section).

#### Network Addressable UART Mode

This mode connects the MicroConverter to a 256-node serial network, either as a hardware single master or via software in a multimaster network. Bit 7 (ENAM) of the COMIEN1 register must be set to enable UART in network addressable mode (see Table 114). Note that there is no parity check in this mode.

## ADuC7019/20/21/22/24/25/26/27/28/29

#### Network Addressable UART Register Definitions

Four additional registers, COMIEN0, COMIEN1, COMIID1, and COMADR are used in network addressable UART mode only.

In network address mode, the least significant bit of the COMIEN1 register is the transmitted network address control bit. If set to 1, the device is transmitting an address. If cleared to 0, the device is transmitting data. For example, the following masterbased code transmits the slave's address followed by the data:

COMIEN1 = 0xE7; //Setting ENAM, E9BT, E9BR, ETD, NABP COMTX = 0xA0; // Slave address is 0xA0 while(!(0x020==(COMSTA0 & 0x020))){} // wait for adr tx to finish. COMIEN1 = 0xE6; // Clear NAB bit to indicate Data is coming COMTX = 0x55; // Tx data to slave: 0x55

#### Table 113. COMIEN1 Register

Name	Address	Default Value	Access
COMIEN1	0xFFFF0720	0x04	R/W

COMIEN1 is an 8-bit network enable register.

#### Table 114. COMIEN1 MMR Bit Descriptions

Bit	Name	Description
7	ENAM	Network address mode enable bit. Set by user to enable network address mode. Cleared by user to disable network address mode.
6	E9BT	9-bit transmit enable bit. Set by user to enable 9-bit transmit. ENAM must be set. Cleared by user to disable 9-bit transmit.
5	E9BR	9-bit receive enable bit. Set by user to enable 9-bit receive. ENAM must be set. Cleared by user to disable 9-bit receive.
4	ENI	Network interrupt enable bit.
3	E9BD	Word length. Set for 9-bit data. E9BT has to be cleared. Cleared for 8-bit data.
2	ETD	Transmitter pin driver enable bit. Set by user to enable SOUT pin as an output in slave mode or multimaster mode. Cleared by user; SOUT is three-state.
1	NABP	Network address bit. Interrupt polarity bit.
0	NAB	Network address bit (if NABP = 1). Set by user to transmit the slave address. Cleared by user to transmit data.

#### Table 115. COMIID1 Register

Name	Address	Default Value	Access
COMIID1	0xFFFF0724	0x01	R

COMIID1 is an 8-bit network interrupt register. Bit 7 to Bit 4 are reserved (see Table 116).

## Table 140. I2CxDIV Registers

Name	Address	Default Value	Access	
I2C0DIV	0xFFFF0830	0x1F1F	R/W	
I2C1DIV	0xFFFF0930	0x1F1F	R/W	

I2CxDIV are the clock divider registers.

#### Table 141. I2CxIDx Registers

Name	Address	Default Value	Access
I2C0ID0	0xFFFF0838	0x00	R/W
I2C0ID1	0xFFFF083C	0x00	R/W
I2C0ID2	0xFFFF0840	0x00	R/W
I2C0ID3	0xFFFF0844	0x00	R/W
I2C1ID0	0xFFFF0938	0x00	R/W
I2C1ID1	0xFFFF093C	0x00	R/W
I2C1ID2	0xFFFF0940	0x00	R/W
I2C1ID3	0xFFFF0944	0x00	R/W

I2CxID0, I2CxID1, I2CxID2, and I2CxID3 are slave address device ID registers of I2Cx.

#### Table 142. I2CxCCNT Registers

Name	Address	Default Value	Access
I2C0CCNT	0xFFFF0848	0x01	R/W
I2C1CCNT	0xFFFF0948	0x01	R/W

I2CxCCNT are 8-bit start/stop generation counters. They hold off SDA low for start and stop conditions.

#### Table 143. I2CxFSTA Registers

Name	Address	Default Value	Access
I2C0FSTA	0xFFFF084C	0x0000	R/W
I2C1FSTA	0xFFFF094C	0x0000	R/W

I2CxFSTA are FIFO status registers.

Table 144.	I2C0I	FSTA M	MR Bit Descriptions	

	Access		
Bit	Туре	Value	Description
15:10			Reserved.
9	R/W		Master transmit FIFO flush. Set by the user to flush the master Tx FIFO. Cleared automatically when the master Tx FIFO is flushed. This bit also flushes the slave receive FIFO.
8	R/W		Slave transmit FIFO flush. Set by the user to flush the slave Tx FIFO. Cleared automatically after the slave Tx FIFO is flushed.
7:6	R		Master Rx FIFO status bits.
		00	FIFO empty.
		01	Byte written to FIFO.
		10	One byte in FIFO.
		11	FIFO full.
5:4	R		Master Tx FIFO status bits.
		00	FIFO empty.
		01	Byte written to FIFO.
		10	One byte in FIFO.
		11	FIFO full.
3:2	R		Slave Rx FIFO status bits.
		00	FIFO empty.
		01	Byte written to FIFO.
		10	One byte in FIFO.
		11	FIFO full.
1:0	R		Slave Tx FIFO status bits.
		00	FIFO empty.
		01	Byte written to FIFO.
		10	One byte in FIFO.
		11	FIFO full.

### Table 148. PLACLK Register

Name	Address	Default Value	Access
PLACLK	0xFFFF0B40	0x00	R/W

PLACLK is the clock selection for the flip-flops of Block 0 and Block 1. Note that the maximum frequency when using the GPIO pins as the clock input for the PLA blocks is 44 MHz.

### Table 149. PLACLK MMR Bit Descriptions

Bit	Value	Description
7		Reserved.
6:4		Block 1 clock source selection.
	000	GPIO clock on P0.5.
	001	GPIO clock on P0.0.
	010	GPIO clock on P0.7.
	011	HCLK.
	100	OCLK (32.768 kHz) external crystal only.
	101	Timer1 overflow.
	Other	Reserved.
3		Reserved.
2:0		Block 0 clock source selection.
	000	GPIO clock on P0.5.
	001	GPIO clock on P0.0.
	010	GPIO clock on P0.7.
	011	HCLK.
	100	OCLK (32.768 kHz) external crystal only.
	101	Timer1 overflow.
	Other	Reserved.

#### Table 150. PLAIRQ Register

Name	Address	Default Value	Access
PLAIRQ	0xFFFF0B44	0x0000000	R/W

PLAIRQ enables IRQ0 and/or IRQ1 and selects the source of the IRQ.

#### Table 151. PLAIRQ MMR Bit Descriptions

Bit	Value	Description		
15:13		Reserved.		
12		PLA IRQ1 enable bit. Set by user to enable IRQ1 output from PLA. Cleared by user to disable IRQ1 output from PLA.		
11:8		PLA IRQ1 source.		
	0000	PLA Element 0.		
	0001	PLA Element 1.		
	1111	PLA Element 15.		
7:5		Reserved.		
4		PLA IRQ0 enable bit. Set by user to enable IRQ0 output from PLA. Cleared by user to disable IRQ0 output from PLA.		
3:0		PLA IRQ0 source.		
	0000	PLA Element 0.		
	0001	PLA Element 1.		
	1111	PLA Element 15.		

### Table 152. Feedback Configuration

Bit	Value	PLAELMO	PLAELM1 to PLAELM7	PLAELM8	PLAELM9 to PLAELM15
10:9	00	Element 15	Element 0	Element 7	Element 8
	01	Element 2	Element 2	Element 10	Element 10
	10	Element 4	Element 4	Element 12	Element 12
	11	Element 6	Element 6	Element 14	Element 14
8:7	00	Element 1	Element 1	Element 9	Element 9
	01	Element 3	Element 3	Element 11	Element 11
	10	Element 5	Element 5	Element 13	Element 13
	11	Element 7	Element 7	Element 15	Element 15

## PROCESSOR REFERENCE PERIPHERALS INTERRUPT SYSTEM

There are 23 interrupt sources on the ADuC7019/20/21/22/ 24/25/26/27/28/29 that are controlled by the interrupt controller. Most interrupts are generated from the on-chip peripherals, such as ADC and UART. Four additional interrupt sources are generated from external interrupt request pins, IRQ0, IRQ1, IRQ2, and IRQ3. The ARM7TDMI CPU core only recognizes interrupts as one of two types: a normal interrupt request IRQ or a fast interrupt request FIQ. All the interrupts can be masked separately.

The control and configuration of the interrupt system are managed through nine interrupt-related registers, four dedicated to IRQ, and four dedicated to FIQ. An additional MMR is used to select the programmed interrupt source. The bits in each IRQ and FIQ register (except for Bit 23) represent the same interrupt source as described in Table 160.

#### Table 160. IRQ/FIQ MMRs Bit Description

Bit	Description
0	All interrupts OR'ed (FIQ only)
1	SWI
2	Timer0
3	Timer1
4	Wake-up timer (Timer2)
5	Watchdog timer (Timer3)
6	Flash control
7	ADC channel
8	PLL lock
9	I2C0 slave
10	I2C0 master
11	I2C1 master
12	SPI slave
13	SPI master
14	UART
15	External IRQ0
16	Comparator
17	PSM
18	External IRQ1
19	PLA IRQ0
20	PLA IRQ1
21	External IRQ2
22	External IRQ3
23	PWM trip (IRQ only)/PWM sync (FIQ only)

## IRQ

The interrupt request (IRQ) is the exception signal to enter the IRQ mode of the processor. It is used to service general-purpose interrupt handling of internal and external events.

The four 32-bit registers dedicated to IRQ are IRQSTA, IRQSIG, IRQEN, and IRQCLR.

#### Table 161. IRQSTA Register

Name	Address	Default Value	Access
IRQSTA	0xFFFF0000	0x0000000	R

IRQSTA (read-only register) provides the current-enabled IRQ source status. When set to 1, that source should generate an active IRQ request to the ARM7TDMI core. There is no priority encoder or interrupt vector generation. This function is implemented in software in a common interrupt handler routine. All 32 bits are logically ORed to create the IRQ signal to the ARM7TDMI core.

#### Table 162. IRQSIG Register

Name	Address	Default Value	Access
IRQSIG	0xFFFF0004	0x00XXX0001	R

<sup>1</sup>X indicates an undefined value.

IRQSIG reflects the status of the different IRQ sources. If a peripheral generates an IRQ signal, the corresponding bit in the IRQSIG is set; otherwise, it is cleared. The IRQSIG bits are cleared when the interrupt in the particular peripheral is cleared. All IRQ sources can be masked in the IRQEN MMR. IRQSIG is read only.

#### Table 163. IRQEN Register

Name	Address	Default Value	Access
IRQEN	0xFFFF0008	0x0000000	R/W

IRQEN provides the value of the current enable mask. When each bit is set to 1, the source request is enabled to create an IRQ exception. When each bit is set to 0, the source request is disabled or masked, which does not create an IRQ exception.

Note that to clear an already enabled interrupt source, the user must set the appropriate bit in the IRQCLR register. Clearing an interrupt's IRQEN bit does not disable the interrupt.

#### Table 164. IRQCLR Register

Name	Address	Default Value	Access
IRQCLR	0xFFFF000C	0x0000000	W

IRQCLR (write-only register) clears the IRQEN register in order to mask an interrupt source. Each bit set to 1 clears the corresponding bit in the IRQEN register without affecting the remaining bits. The pair of registers, IRQEN and IRQCLR, independently manipulates the enable mask without requiring an atomic read-modify-write.

In normal mode, an IRQ is generated each time the value of the counter reaches zero when counting down. It is also generated each time the counter value reaches full scale when counting up. An IRQ can be cleared by writing any value to clear the register of that particular timer (TxCLRI).

When using an asynchronous clock-to-clock timer, the interrupt in the timer block may take more time to clear than the time it takes for the code in the interrupt routine to execute. Ensure that the interrupt signal is cleared before leaving the interrupt service routine. This can be done by checking the IRQSTA MMR.

### Hour:Minute:Second:1/128 Format

To use the timer in hour:minute:second:hundredths format, select the 32,768 kHz clock and prescaler of 256. The hundredths field does not represent milliseconds but 1/128 of a second (256/32,768). The bits representing the hour, minute, and second are not consecutive in the register. This arrangement applies to TxLD and TxVAL when using the hour:minute:second:hundredths format as set in TxCON[5:4]. See Table 171 for additional details.

#### Table 171. Hour:Minnute:Second:Hundredths Format

Bit	Value	Description
31:24	0 to 23 or 0 to 255	Hours
23:22	0	Reserved
21:16	0 to 59	Minutes
15:14	0	Reserved
13.8	0 to 59	Seconds
7	0	Reserved
6:0	0 to 127	1/128 second

## Timer0 (RTOS Timer)

Timer0 is a general-purpose, 16-bit timer (count down) with a programmable prescaler (see Figure 77). The prescaler source is the core clock frequency (HCLK) and can be scaled by factors of 1, 16, or 256.

Timer0 can be used to start ADC conversions as shown in the block diagram in Figure 77.



#### Figure 77. Timer0 Block Diagram

# ADuC7019/20/21/22/24/25/26/27/28/29

The Timer0 interface consists of four MMRs: T0LD, T0VAL, T0CON, and T0CLRI.

#### Table 172. T0LD Register

Name	Address	Default Value	Access
TOLD	0xFFFF0300	0x0000	R/W

T0LD is a 16-bit load register.

#### Table 173. TOVAL Register

Name	Address	Default Value	Access
TOVAL	0xFFFF0304	0xFFFF	R

TOVAL is a 16-bit read-only register representing the current state of the counter.

#### Table 174. TOCON Register

Name	Address	Default Value	Access
T0CON	0xFFFF0308	0x0000	R/W

T0CON is the configuration MMR described in Table 175.

#### Table 175. TOCON MMR Bit Descriptions

Bit	Value	Description
15:8		Reserved.
7		Timer0 enable bit. Set by user to enable Timer0. Cleared by user to disable Timer0 by default.
6		Timer0 mode. Set by user to operate in periodic mode. Cleared by user to operate in free-running mode. Default mode.
5:4		Reserved.
3:2		Prescale.
	00	Core Clock/1. Default value.
	01	Core Clock/16.
	10	Core Clock/256.
	11	Undefined. Equivalent to 00.
1:0		Reserved.

#### Table 176. T0CLRI Register

Name	Address	Default Value Access	
TOCLRI	0xFFFF030C	0xFF	W

TOCLRI is an 8-bit register. Writing any value to this register clears the interrupt.

# ADuC7019/20/21/22/24/25/26/27/28/29



Dimensions shown in millimeters