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	44MHz
Connectivity	EBI/EMI, I²C, SPI, UART/USART
Peripherals	PLA, PWM, PSM, Temp Sensor, WDT
Number of I/O	30
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 12x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP
Supplier Device Package	64-LQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/aduc7025bstz62-rl

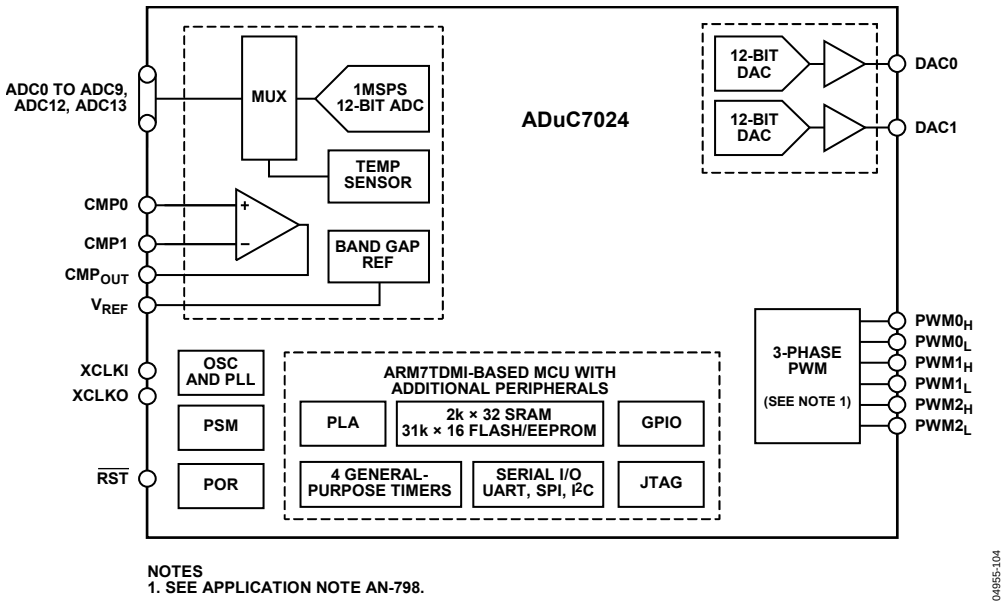


Figure 5.

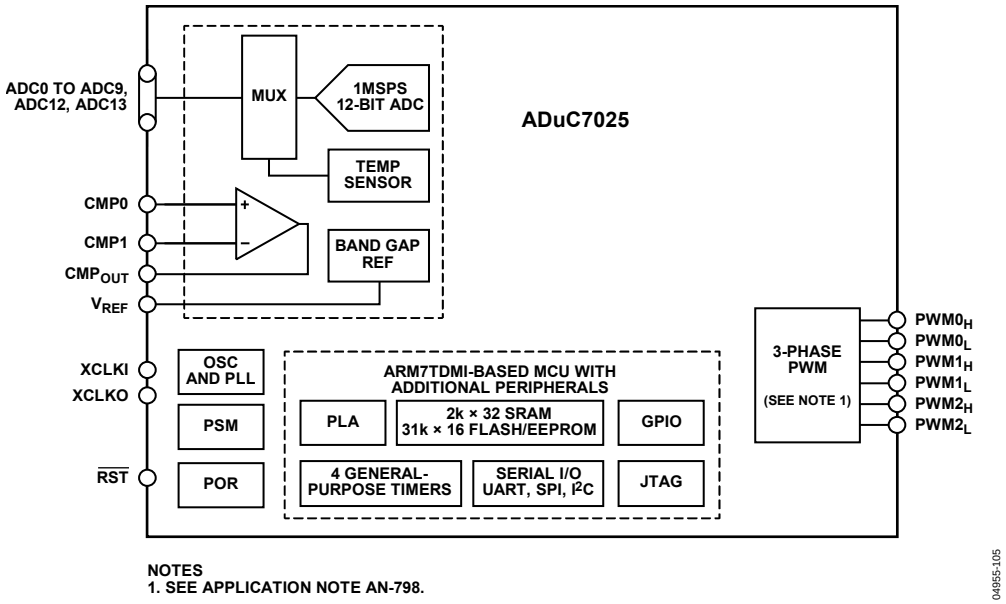


Figure 6.

Table 4. I²C Timing in Fast Mode (400 kHz)

		Slave		Master	
Parameter	**Description**	**Min**	**Max**	**Typ**	**Unit**
t_L	SCL low pulse width[1]	200		1360	ns
t_H	SCL high pulse width[1]	100		1140	ns
t_{SHD}	Start condition hold time	300			ns
t_{DSU}	Data setup time	100		740	ns
t_{DHD}	Data hold time	0		400	ns
t_{RSU}	Setup time for repeated start	100			ns
t_{PSU}	Stop condition setup time	100		400	ns
t_{BUF}	Bus-free time between a stop condition and a start condition	1.3			µs
t_R	Rise time for both SCL and SDA		300	200	ns
t_F	Fall time for both SCL and SDA		300		ns
t_{SUP}	Pulse width of spike suppressed		50		ns

[1] t_{HCLK} depends on the clock divider or CD bits in the POWCON MMR. $t_{HCLK} = t_{UCLK}/2^{CD}$; see Figure 67.

Table 5. I²C Timing in Standard Mode (100 kHz)

		Slave		Master	
Parameter	**Description**	**Min**	**Max**	**Typ**	**Unit**
t_L	SCL low pulse width[1]	4.7			µs
t_H	SCL high pulse width[1]	4.0			ns
t_{SHD}	Start condition hold time	4.0			µs
t_{DSU}	Data setup time	250			ns
t_{DHD}	Data hold time	0	3.45		µs
t_{RSU}	Setup time for repeated start	4.7			µs
t_{PSU}	Stop condition setup time	4.0			µs
t_{BUF}	Bus-free time between a stop condition and a start condition	4.7			µs
t_R	Rise time for both SCL and SDA		1		µs
t_F	Fall time for both SCL and SDA		300		ns

[1] t_{HCLK} depends on the clock divider or CD bits in the POWCON MMR. $t_{HCLK} = t_{UCLK}/2^{CD}$; see Figure 67.

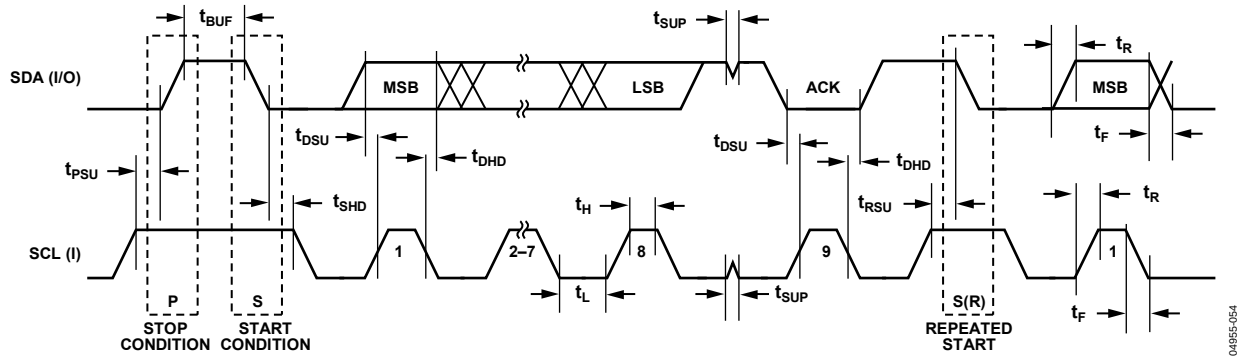
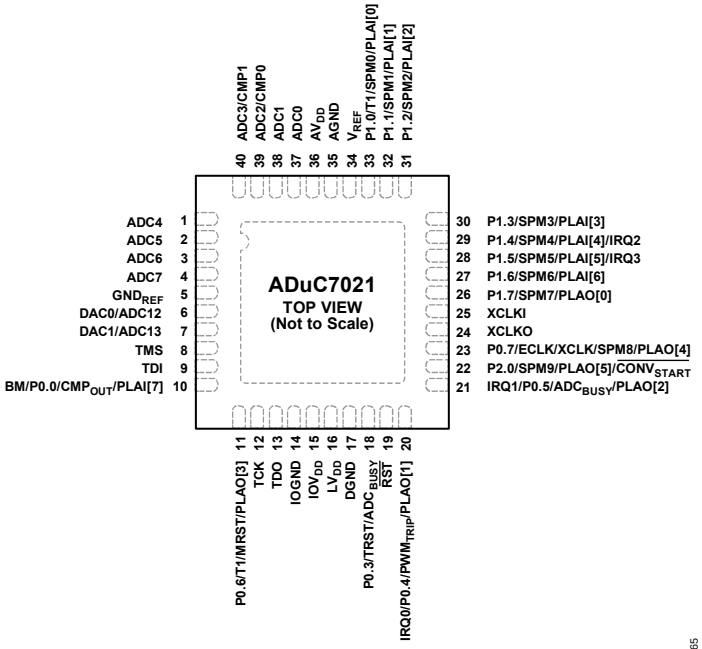


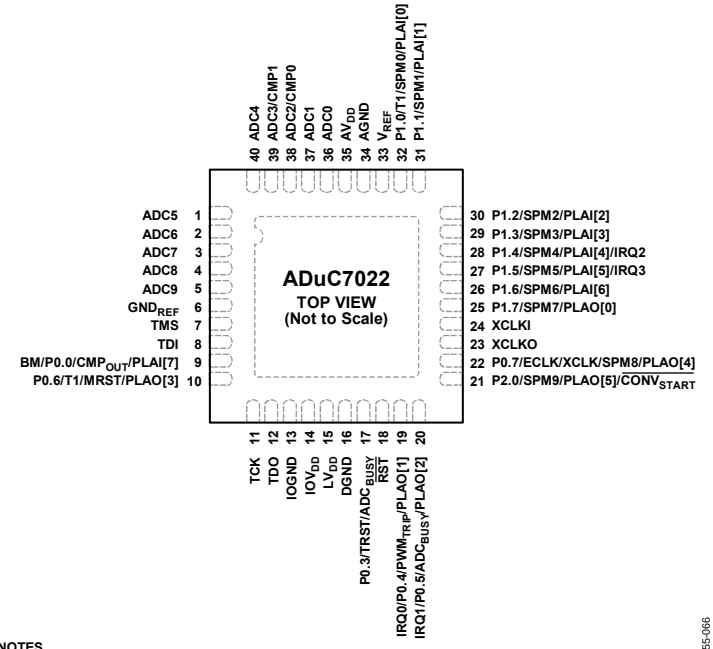
Figure 14. I²C Compatible Interface Timing



NOTES
1. THE EXPOSED PAD MUST BE SOLDERED FOR MECHANICAL PURPOSES AND LEFT UNCONNECTED.

04955-065

Figure 21. 40-Lead LFCSP_WQ Pin Configuration (ADuC7021)

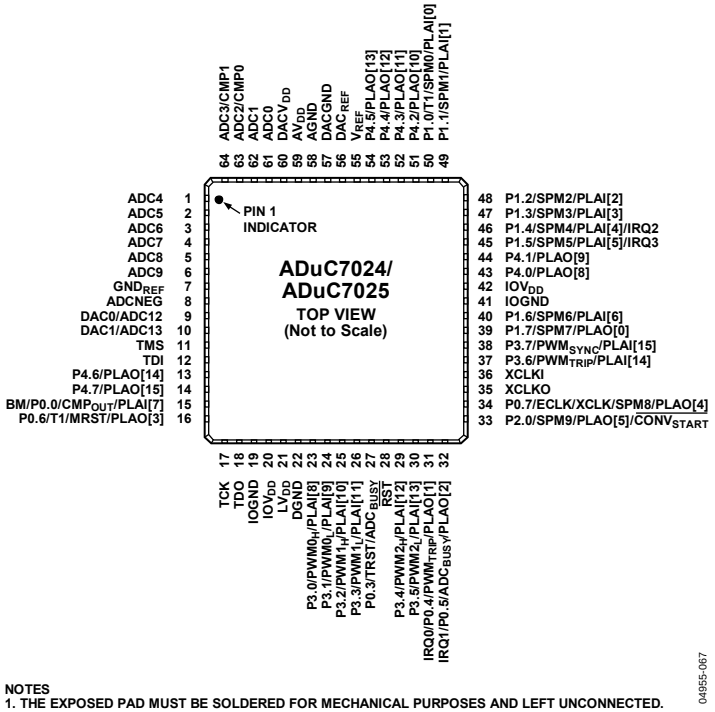


NOTES
1. THE EXPOSED PAD MUST BE SOLDERED FOR MECHANICAL PURPOSES AND LEFT UNCONNECTED.

04955-066

Figure 22. 40-Lead LFCSP_WQ Pin Configuration (ADuC7022)

ADuC7024/ADuC7025



NOTES
1. THE EXPOSED PAD MUST BE SOLDERED FOR MECHANICAL PURPOSES AND LEFT UNCONNECTED.

Figure 23. 64-Lead LFCSP_VQ Pin Configuration (ADuC7024/ADuC7025)

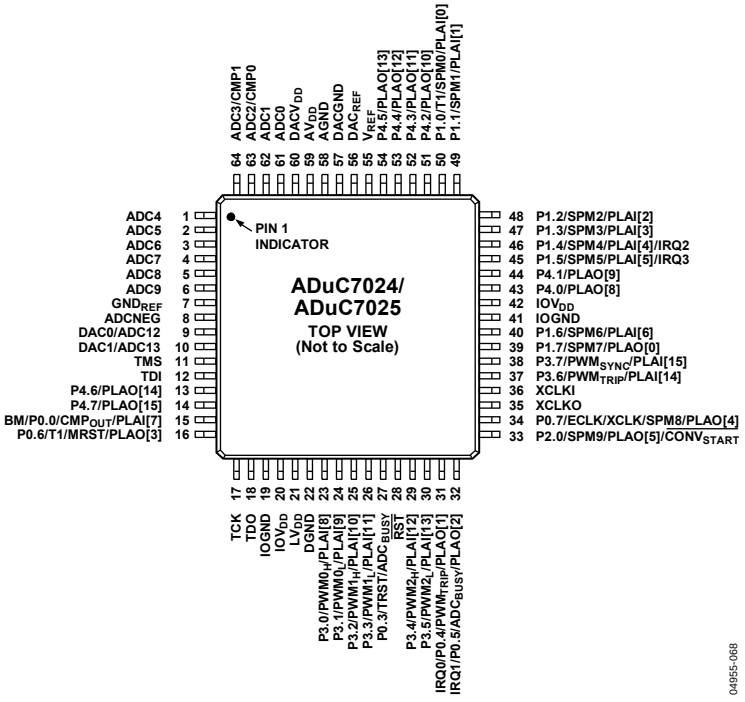


Figure 24. 64-Lead LQFP Pin Configuration (ADuC7024/ADuC7025)

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_{OUT}/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 kΩ 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	TCK	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/$PWM2_L$/$\overline{BHE}$	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_{DD}	3.3 V Supply for GPIO (see Table 78) and Input of the On-Chip Voltage Regulator.
27	LV_{DD}	2.6 V Output of the On-Chip Voltage Regulator. This output must be connected to a 0.47 µF capacitor to DGND only.
28	DGND	Ground for Core Logic.
29	P3.0/AD0/$PWM0_H$/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/$PWM0_L$/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/$PWM1_H$/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/$PWM1_L$/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/$PWM0_H$/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_{BUSY}	General-Purpose Input and Output Port 0.3/JTAG Test Port Input, Test Reset/ADC_{BUSY} Signal Output.
35	P2.5/$PWM0_L$/MS1	General-Purpose Input and Output Port 2.5/PWM Phase 0 Low-Side Output/External Memory Select 1.
36	P2.6/$PWM1_H$/MS2	General-Purpose Input and Output Port 2.6/PWM Phase 1 High-Side Output/External Memory Select 2.
37	$\overline{RST}$	Reset Input, Active Low.
38	P3.4/AD4/$PWM2_H$/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/$PWM2_L$/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_{TRIP}/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_{BUSY}/PLAO[2]/MS2	Multifunction I/O Pin. External Interrupt Request 1, Active High/General-Purpose Input and Output Port 0.5/ADC_{BUSY} Signal Output/Programmable Logic Array Output Element 2/External Memory Select 2.
42	P2.0/SPM9/PLAO[5]/$\overline{CONV_{START}}$	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.

Pin No.	Mnemonic	Description
46	P3.6/AD6/PWM_{TRIP}/PLAI[14]	General-Purpose Input and Output Port 3.6/External Memory Interface/PWM Safety Cutoff/ Programmable Logic Array Input Element 14.
47	P3.7/AD7/PWM_{SYNC}/PLAI[15]	General-Purpose Input and Output Port 3.7/External Memory Interface/PWM Synchronization/ Programmable Logic Array Input Element 15.
48	P2.7/$PWM1_L$/MS3	General-Purpose Input and Output Port 2.7/PWM Phase 1 Low-Side Output/External Memory Select 3.
49	P2.1/$\overline{WS}$/$PWM0_H$/PLAO[6]	General-Purpose Input and Output Port 2.1/External Memory Write Strobe/PWM Phase 0 High-Side Output/Programmable Logic Array Output Element 6.
50	P2.2/$\overline{RS}$/$PWM0_L$/PLAO[7]	General-Purpose Input and Output Port 2.2/External Memory Read Strobe/PWM Phase 0 Low-Side Output/Programmable Logic Array Output Element 7.
51	P1.7/SPM7/PLAO[0]	Serial Port Multiplexed. General-Purpose Input and Output Port 1.7/UART, SPI/Programmable Logic Array Output Element 0.
52	P1.6/SPM6/PLAI[6]	Serial Port Multiplexed. General-Purpose Input and Output Port 1.6/UART, SPI/Programmable Logic Array Input Element 6.
53	IOGND	Ground for GPIO (see Table 78). Typically connected to DGND.
54	IOV_{DD}	3.3 V Supply for GPIO (see Table 78) and Input of the On-Chip Voltage Regulator.
55	P4.0/AD8/PLAO[8]	General-Purpose Input and Output Port 4.0/External Memory Interface/Programmable Logic Array Output Element 8.
56	P4.1/AD9/PLAO[9]	General-Purpose Input and Output Port 4.1/External Memory Interface/Programmable Logic Array Output Element 9.
57	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.
58	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.
59	P1.3/SPM3/PLAI[3]	Serial Port Multiplexed. General-Purpose Input and Output Port 1.3/UART, I2C1/Programmable Logic Array Input Element 3.
60	P1.2/SPM2/PLAI[2]	Serial Port Multiplexed. General-Purpose Input and Output Port 1.2/UART, I2C1/Programmable Logic Array Input Element 2.
61	P1.1/SPM1/PLAI[1]	Serial Port Multiplexed. General-Purpose Input and Output Port 1.1/UART, I2C0/Programmable Logic Array Input Element 1.
62	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.
63	P4.2/AD10/PLAO[10]	General-Purpose Input and Output Port 4.2/External Memory Interface/Programmable Logic Array Output Element 10.
64	P4.3/AD11/PLAO[11]	General-Purpose Input and Output Port 4.3/External Memory Interface/Programmable Logic Array Output Element 11.
65	P4.4/AD12/PLAO[12]	General-Purpose Input and Output Port 4.4/External Memory Interface/Programmable Logic Array Output Element 12.
66	P4.5/AD13/PLAO[13]	General-Purpose Input and Output Port 4.5/External Memory Interface/Programmable Logic Array Output Element 13.
67	REFGND	Ground for the Reference. Typically connected to AGND.
68	V_{REF}	2.5 V Internal Voltage Reference. Must be connected to a 0.47 µF capacitor when using the internal reference.
69	DAC_{REF}	External Voltage Reference for the DACs. Range: DACGND to $DACV_{DD}$.
70	DACGND	Ground for the DAC. Typically connected to AGND.
71, 72	AGND	Analog Ground. Ground reference point for the analog circuitry.
73, 74	AV_{DD}	3.3 V Analog Power.
75	$DACV_{DD}$	3.3 V Power Supply for the DACs. Must be connected to AV_{DD}.
76	ADC11	Single-Ended or Differential Analog Input 11.
77	ADC0	Single-Ended or Differential Analog Input 0.
78	ADC1	Single-Ended or Differential Analog Input 1.
79	ADC2/CMP0	Single-Ended or Differential Analog Input 2/Comparator Positive Input.
80	ADC3/CMP1	Single-Ended or Differential Analog Input 3/Comparator Negative Input.

TYPICAL PERFORMANCE CHARACTERISTICS

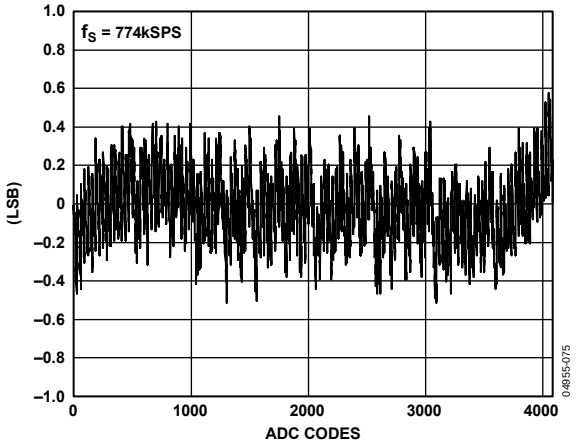


Figure 28. Typical INL Error, f_S = 774 kSPS

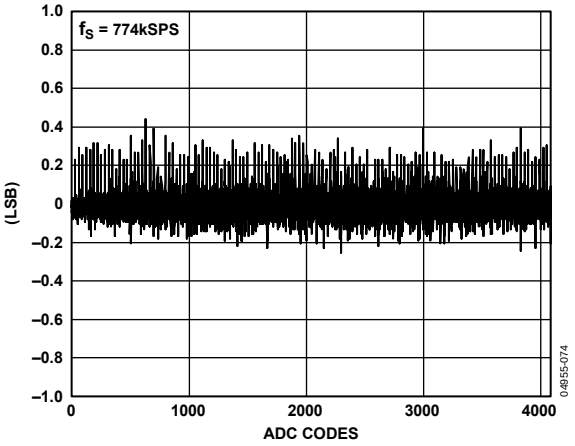


Figure 31. Typical DNL Error, f_S = 774 kSPS

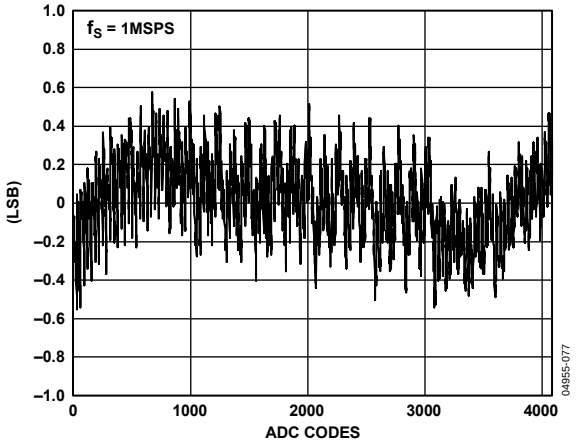


Figure 29. Typical INL Error, f_S = 1 MSPS

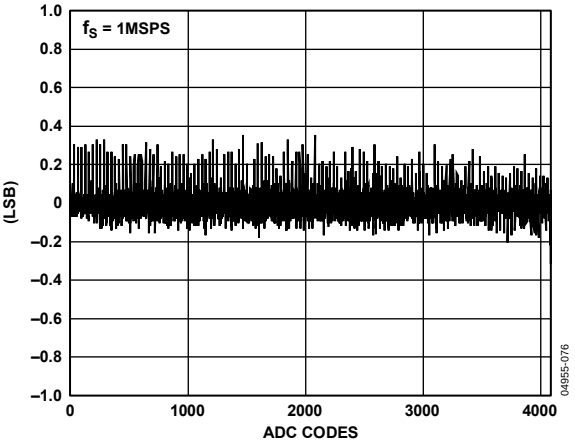


Figure 32. Typical DNL Error, f_S = 1 MSPS

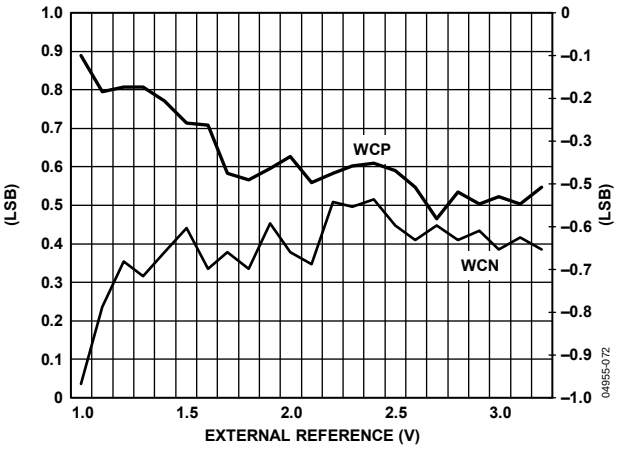


Figure 30. Typical Worst-Case (Positive (WCP) and Negative (WCN)) INL Error vs. V_{REF}, f_S = 774 kSPS

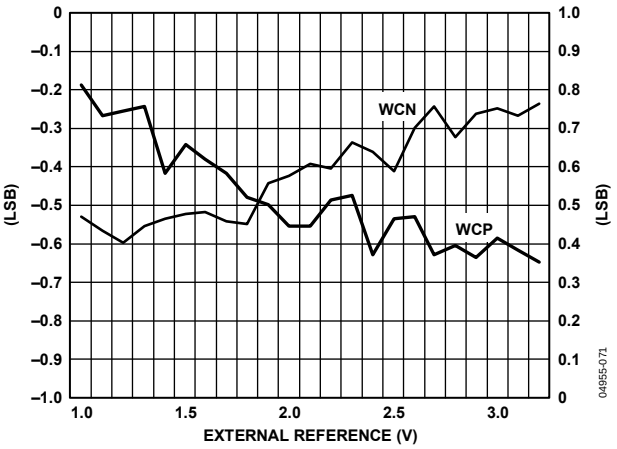


Figure 33. Typical Worst-Case (Positive (WCP)and Negative (WCN)) DNL Error vs. V_{REF}, f_S = 774 kSPS

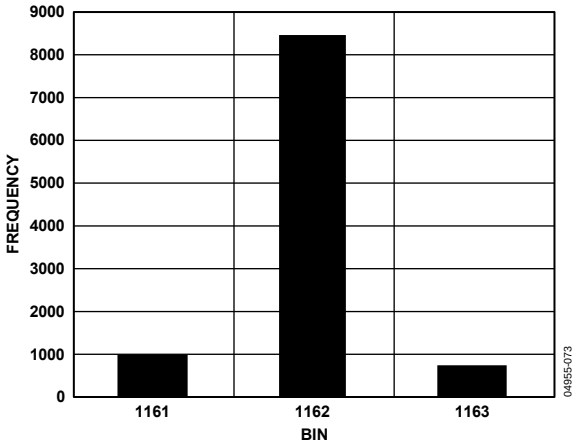


Figure 34. Code Histogram Plot, fs = 774 kSPS, V_{IN} = 0.7 V

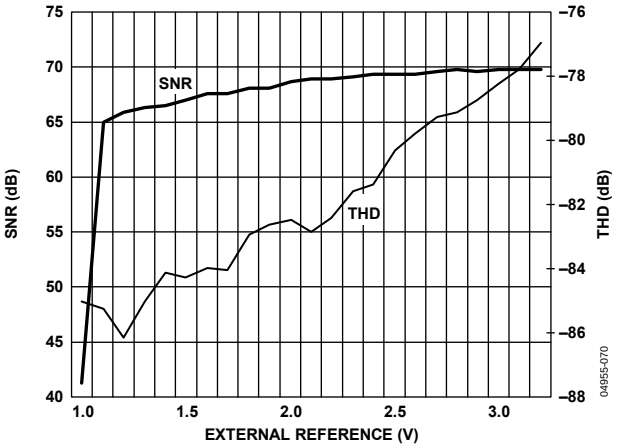


Figure 37. Typical Dynamic Performance vs. V_{REF}

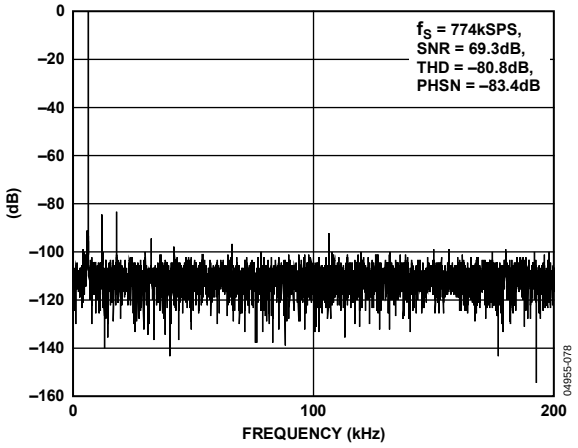


Figure 35. Dynamic Performance, f_S = 774 kSPS

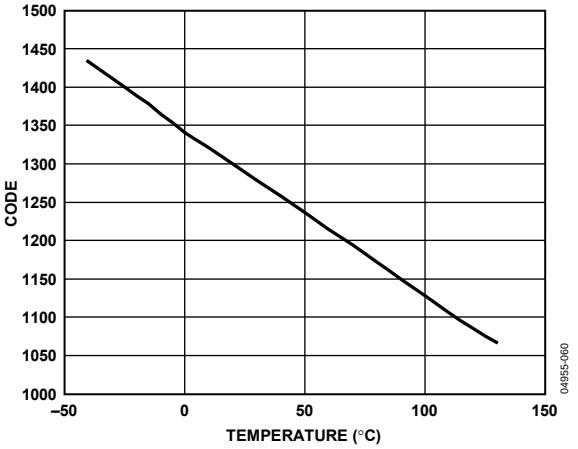


Figure 38. On-Chip Temperature Sensor Voltage Output vs. Temperature

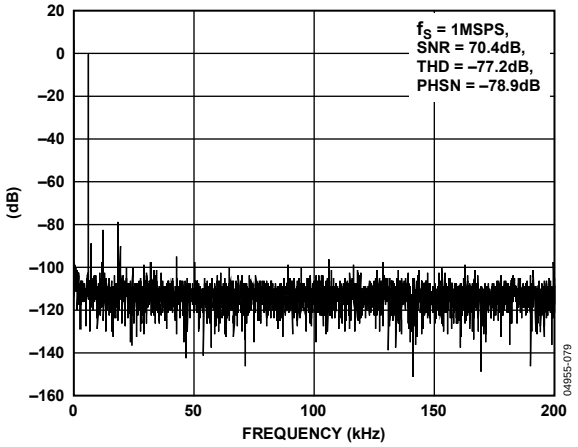


Figure 36. Dynamic Performance, f_S = 1 MSPS

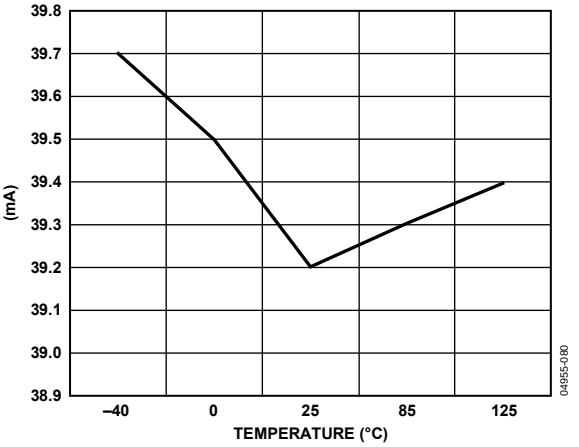


Figure 39. Current Consumption vs. Temperature @ CD = 0

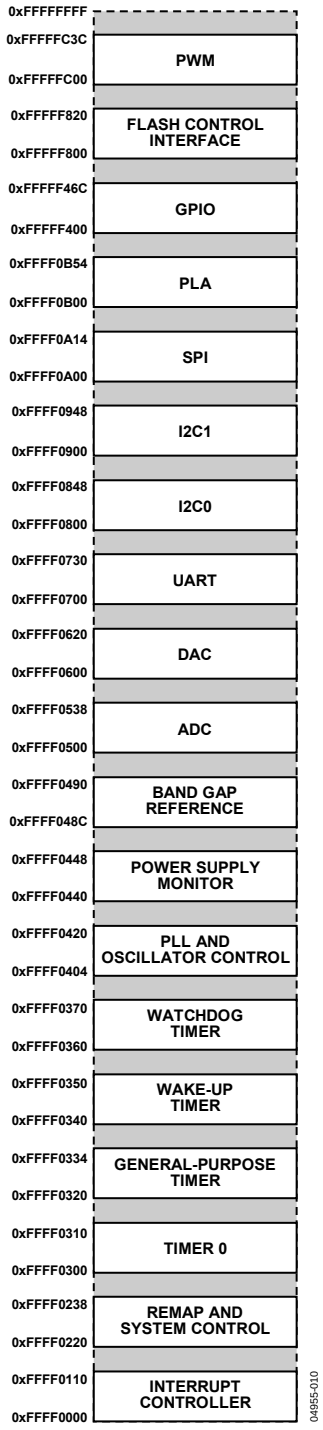


Figure 47. Memory Mapped Registers

Table 16. Complete MMR List

Address	Name	Byte	Access Type	Default Value	Page
IRQ Address Base = 0xFFFF0000					
0x0000	IRQSTA	4	R	0x00000000	83
0x0004	IRQSIG[1]	4	R	0x00XXX000	83
0x0008	IRQEN	4	R/W	0x00000000	83
0x000C	IRQCLR	4	W	0x00000000	83
0x0010	SWICFG	4	W	0x00000000	84
0x0100	FIQSTA	4	R	0x00000000	84
0x0104	FIQSIG[1]	4	R	0x00XXX000	84
0x0108	FIQEN	4	R/W	0x00000000	84
0x010C	FIQCLR	4	W	0x00000000	84

[1] Depends on the level on the external interrupt pins (P0.4, P0.5, P1.4, and P1.5).

System Control Address Base = 0xFFFF0200					
0x0220	REMAP	1	R/W	0xXX[1]	55
0x0230	RSTSTA	1	R/W	0x01	55
0x0234	RSTCLR	1	W	0x00	55

[1] Depends on the model.

Timer Address Base = 0xFFFF0300					
0x0300	T0LD	2	R/W	0x0000	85
0x0304	T0VAL	2	R	0xFFFF	85
0x0308	T0CON	2	R/W	0x0000	85
0x030C	T0CLRI	1	W	0xFF	85
0x0320	T1LD	4	R/W	0x00000000	86
0x0324	T1VAL	4	R	0xFFFFFFFF	86
0x0328	T1CON	2	R/W	0x0000	86
0x032C	T1CLRI	1	W	0xFF	87
0x0330	T1CAP	4	R/W	0x00000000	87
0x0340	T2LD	4	R/W	0x00000000	87
0x0344	T2VAL	4	R	0xFFFFFFFF	87
0x0348	T2CON	2	R/W	0x0000	87
0x034C	T2CLRI	1	W	0xFF	88
0x0360	T3LD	2	R/W	0x0000	88
0x0364	T3VAL	2	R	0xFFFF	88
0x0368	T3CON	2	R/W	0x0000	88
0x036C	T3CLRI	1	W	0x00	89

PLL Base Address = 0xFFFF0400					
0x0404	POWKEY1	2	W	0x0000	60
0x0408	POWCON	2	R/W	0x0003	60
0x040C	POWKEY2	2	W	0x0000	60
0x0410	PLLKEY1	2	W	0x0000	60
0x0414	PLLCON	1	R/W	0x21	60
0x0418	PLLKEY2	2	W	0x0000	60

PSM Address Base = 0xFFFF0440					
0x0440	PSMCON	2	R/W	0x0008	57
0x0444	CMPCON	2	R/W	0x0000	58

Table 18. ADCCON MMR Bit Designations

Bit	Value	Description
15:13		Reserved.
12:10		ADC clock speed.
	000	fADC/1. This divider is provided to obtain 1 MSPS ADC with an external clock <41.78 MHz.
	001	fADC/2 (default value).
	010	fADC/4.
	011	fADC/8.
	100	fADC/16.
	101	fADC/32.
9:8		ADC acquisition time.
	00	Two clocks.
	01	Four clocks.
	10	Eight clocks (default value).
	11	16 clocks.
7		Enable start conversion. Set by the user to start any type of conversion command. Cleared by the user to disable a start conversion (clearing this bit does not stop the ADC when continuously converting).
6		Reserved.
5		ADC power control. Set by the user to place the ADC in normal mode (the ADC must be powered up for at least 5 µs before it converts correctly). Cleared by the user to place the ADC in power-down mode.
4:3		Conversion mode.
	00	Single-ended mode.
	01	Differential mode.
	10	Pseudo differential mode.
	11	Reserved.
2:0		Conversion type.
	000	Enable $\overline{CONV}_{START}$ pin as a conversion input.
	001	Enable Timer1 as a conversion input.
	010	Enable Timer0 as a conversion input.
	011	Single software conversion. Sets to 000 after conversion (note that Bit 7 of ADCCON MMR should be cleared after starting a single software conversion to avoid further conversions triggered by the $\overline{CONV}_{START}$ pin).
	100	Continuous software conversion.
	101	PLA conversion.
	Other	Reserved.

Table 19. ADCCP Register

Name	Address	Default Value	Access
ADCCP	0xFFFF0504	0x00	R/W

ADCCP is an ADC positive channel selection register. This MMR is described in Table 20.

Table 20. ADCCP[1] MMR Bit Designation

Bit	Value	Description
7:5		Reserved.
4:0		Positive channel selection bits.
	00000	ADC0.
	00001	ADC1.
	00010	ADC2.
	00011	ADC3.
	00100	ADC4.
	00101	ADC5.
	00110	ADC6.
	00111	ADC7.
	01000	ADC8.
	01001	ADC9.
	01010	ADC10.
	01011	ADC11.
	01100	DAC0/ADC12.
	01101	DAC1/ADC13.
	01110	DAC2/ADC14.
	01111	DAC3/ADC15.
	10000	Temperature sensor.
	10001	AGND (self-diagnostic feature).
	10010	Internal reference (self-diagnostic feature).
	10011	$AV_{DD}/2$.
	Others	Reserved.

[1] ADC and DAC channel availability depends on the part model. See Ordering Guide for details.

Table 21. ADCCN Register

Name	Address	Default Value	Access
ADCCN	0xFFFF0508	0x01	R/W

ADCCN is an ADC negative channel selection register. This MMR is described in Table 22.

Table 22. ADCCN MMR Bit Designation

Bit	Value	Description
7:5		Reserved.
4:0		Negative channel selection bits.
	00000	ADC0.
	00001	ADC1.
	00010	ADC2.
	00011	ADC3.
	00100	ADC4.
	00101	ADC5.
	00110	ADC6.
	00111	ADC7.
	01000	ADC8.
	01001	ADC9.
	01010	ADC10.
	01011	ADC11.
	01100	DAC0/ADC12.
	01101	DAC1/ADC13.
	01110	DAC2/ADC14.
	01111	DAC3/ADC15.
	10000	Internal reference (self-diagnostic feature).
	Others	Reserved.

Table 23. ADCSTA Register

Name	Address	Default Value	Access
ADCSTA	0xFFFF050C	0x00	R

ADCSTA is an ADC status register that indicates when an ADC conversion result is ready. The ADCSTA register contains only one bit, ADCReady (Bit 0), representing the status of the ADC. This bit is set at the end of an ADC conversion, generating an ADC interrupt. It is cleared automatically by reading the ADCDAT MMR. When the ADC is performing a conversion, the status of the ADC can be read externally via the ADC_{BUSY} pin. This pin is high during a conversion. When the conversion is finished, ADC_{BUSY} goes back low. This information can be available on P0.5 (see the General-Purpose Input/Output section) if enabled in the ADCCON register.

Table 24. ADCDAT Register

Name	Address	Default Value	Access
ADCDAT	0xFFFF0510	0x00000000	R

ADCDAT is an ADC data result register. It holds the 12-bit ADC result as shown in Figure 51.

Table 25. ADCRST Register

Name	Address	Default Value	Access
ADCRST	0xFFFF0514	0x00	R/W

ADCRST resets the digital interface of the ADC. Writing any value to this register resets all the ADC registers to their default values.

Table 26. ADCGN Register

Name	Address	Default Value	Access
ADCGN	0xFFFF0530	0x0200	R/W

ADCGN is a 10-bit gain calibration register.

Table 27. ADCOF Register

Name	Address	Default Value	Access
ADCOF	0xFFFF0534	0x0200	R/W

ADCOF is a 10-bit offset calibration register.

CONVERTER OPERATION

The ADC incorporates a successive approximation (SAR) architecture involving a charge-sampled input stage. This architecture can operate in three modes: differential, pseudo differential, and single-ended.

Differential Mode

The ADuC7019/20/21/22/24/25/26/27/28/29 each contain a successive approximation ADC based on two capacitive DACs. Figure 54 and Figure 55 show simplified schematics of the ADC in acquisition and conversion phase, respectively. The ADC comprises control logic, a SAR, and two capacitive DACs. In Figure 54 (the acquisition phase), SW3 is closed and SW1 and SW2 are in Position A. The comparator is held in a balanced condition, and the sampling capacitor arrays acquire the differential signal on the input.

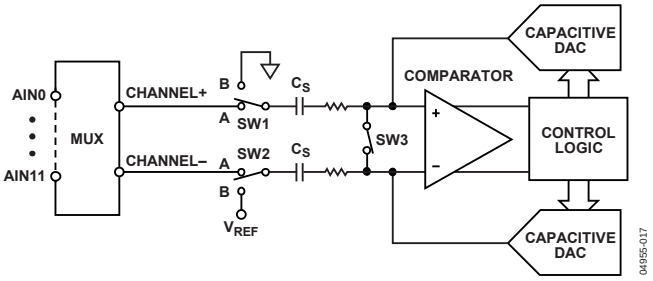


Figure 54. ADC Acquisition Phase

When the ADC starts a conversion, as shown in Figure 55, SW3 opens, and then SW1 and SW2 move to Position B. This causes the comparator to become unbalanced. Both inputs are disconnected once the conversion begins. The control logic and the charge redistribution DACs are used to add and subtract fixed amounts of charge from the sampling capacitor arrays to bring the comparator back into a balanced condition. When the comparator is rebalanced, the conversion is complete. The control logic generates the ADC output code. The output impedances of the sources driving the V_{IN+} and V_{IN-} input voltage pins must be matched; otherwise, the two inputs have different settling times, resulting in errors.

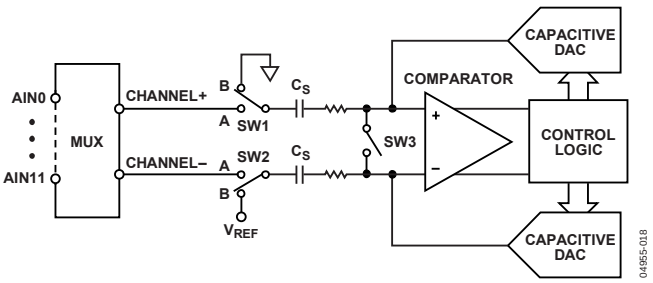


Figure 55. ADC Conversion Phase

Pseudo Differential Mode

In pseudo differential mode, Channel– is linked to the V_{IN-} pin of the ADuC7019/20/21/22/24/25/26/27/28/29. SW2 switches between A (Channel–) and B (V_{REF}). The V_{IN-} pin must be connected to ground or a low voltage. The input signal on V_{IN+} can then vary from V_{IN-} to V_{REF} + V_{IN-}. Note that V_{IN-} must be chosen so that V_{REF} + V_{IN-} does not exceed AV_{DD}.

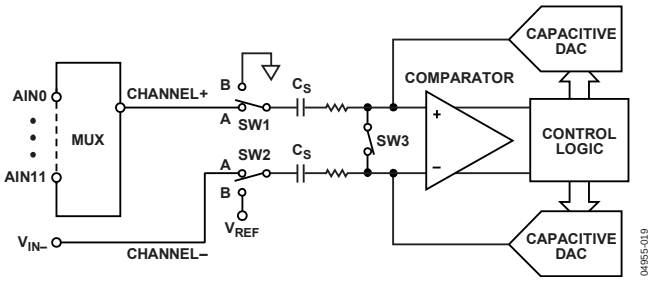


Figure 56. ADC in Pseudo Differential Mode

Single-Ended Mode

In single-ended mode, SW2 is always connected internally to ground. The V_{IN-} pin can be floating. The input signal range on V_{IN+} is 0 V to V_{REF}.

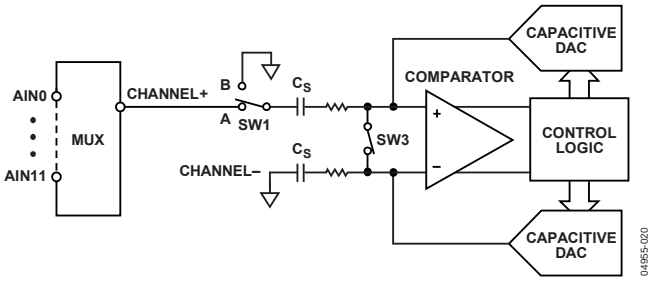


Figure 57. ADC in Single-Ended Mode

Analog Input Structure

Figure 58 shows the equivalent circuit of the analog input structure of the ADC. The four diodes provide ESD protection for the analog inputs. Care must be taken to ensure that the analog input signals never exceed the supply rails by more than 300 mV; exceeding 300 mV causes these diodes to become forward-biased and start conducting into the substrate. These diodes can conduct up to 10 mA without causing irreversible damage to the part.

The C1 capacitors in Figure 58 are typically 4 pF and can be primarily attributed to pin capacitance. The resistors are lumped components made up of the on resistance of the switches. The value of these resistors is typically about 100 Ω. The C2 capacitors are the ADC's sampling capacitors and typically have a capacitance of 16 pF.

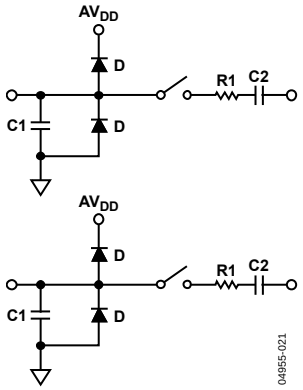


Figure 58. Equivalent Analog Input Circuit Conversion Phase: Switches Open, Track Phase: Switches Closed

For ac applications, removing high frequency components from the analog input signal is recommended by using an RC low-pass filter on the relevant analog input pins. In applications where harmonic distortion and signal-to-noise ratio are critical, the analog input should be driven from a low impedance source. Large source impedances significantly affect the ac performance of the ADC. This can necessitate the use of an input buffer amplifier. The choice of the op amp is a function of the particular application. Figure 59 and Figure 60 give an example of an ADC front end.

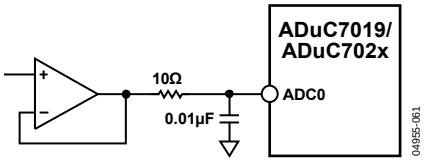


Figure 59. Buffering Single-Ended/Pseudo Differential Input

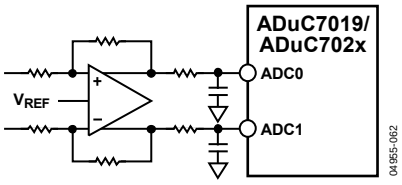


Figure 60. Buffering Differential Inputs

When no amplifier is used to drive the analog input, the source impedance should be limited to values lower than 1 kΩ. The maximum source impedance depends on the amount of total harmonic distortion (THD) that can be tolerated. The THD increases as the source impedance increases and the performance degrades.

DRIVING THE ANALOG INPUTS

Internal or external references can be used for the ADC. In the differential mode of operation, there are restrictions on the common-mode input signal (V_{CM}), which is dependent upon the reference value and supply voltage used to ensure that the signal remains within the supply rails. Table 28 gives some calculated V_{CM} minimum and V_{CM} maximum values.

EXECUTION TIME FROM SRAM AND FLASH/EE

Execution from SRAM

Fetching instructions from SRAM takes one clock cycle; the access time of the SRAM is 2 ns, and a clock cycle is 22 ns minimum. However, if the instruction involves reading or writing data to memory, one extra cycle must be added if the data is in SRAM (or three cycles if the data is in Flash/EE): one cycle to execute the instruction, and two cycles to get the 32-bit data from Flash/EE. A control flow instruction (a branch instruction, for example) takes one cycle to fetch but also takes two cycles to fill the pipeline with the new instructions.

Execution from Flash/EE

Because the Flash/EE width is 16 bits and access time for 16-bit words is 22 ns, execution from Flash/EE cannot be done in one cycle (as can be done from SRAM when the CD Bit = 0). Also, some dead times are needed before accessing data for any value of the CD bit.

In ARM mode, where instructions are 32 bits, two cycles are needed to fetch any instruction when CD = 0. In thumb mode, where instructions are 16 bits, one cycle is needed to fetch any instruction.

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. A data-processing instruction involving only the core register does not require any extra clock cycles. However, if it involves data in Flash/EE, an extra clock cycle is needed to decode the address of the data, and two cycles are needed to get the 32-bit data from Flash/EE. An extra cycle must also be added before fetching another instruction. Data transfer instructions are more complex and are summarized in Table 43.

Table 43. Execution Cycles in ARM/Thumb Mode

Instructions	Fetch Cycles	Dead Time	Data Access	Dead Time
LD[1]	2/1	1	2	1
LDH	2/1	1	1	1
LDM/PUSH	2/1	N[2]	2 × N[2]	N[1]
STR[1]	2/1	1	2 × 20 ns	1
STRH	2/1	1	20 ns	1
STRM/POP	2/1	N[1]	2 × N × 20 ns[1]	N[1]

[1] The SWAP instruction combines an LD and STR instruction with only one fetch, giving a total of eight cycles + 40 ns.
[2] N is the amount of data to load or store in the multiple load/store instruction (1 < N ≤ 16).

RESET AND REMAP

The ARM exception vectors are all situated at the bottom of the memory array, from Address 0x00000000 to Address 0x00000020, as shown in Figure 62.

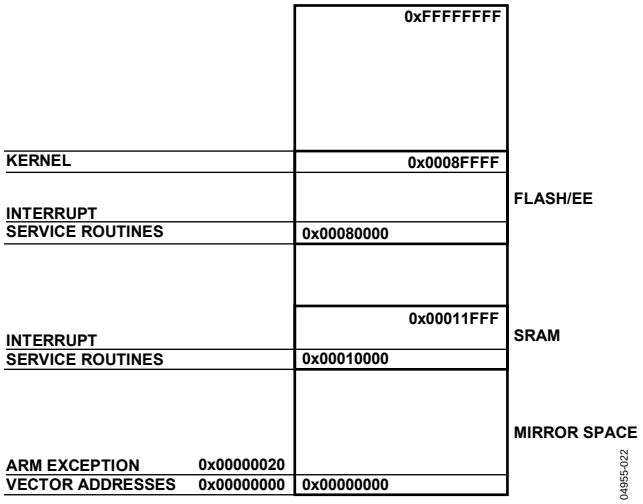


Figure 62. Remap for Exception Execution

By default, and after any reset, the Flash/EE is mirrored at the bottom of the memory array. The remap function allows the programmer to mirror the SRAM at the bottom of the memory array, which facilitates execution of exception routines from SRAM instead of from Flash/EE. This means exceptions are executed twice as fast, being executed in 32-bit ARM mode with 32-bit wide SRAM instead of 16-bit wide Flash/EE memory.

Remap Operation

When a reset occurs on the ADuC7019/20/21/22/24/25/26/27/28/29, execution automatically starts in the factory-programmed, internal configuration code. This kernel is hidden and cannot be accessed by user code. If the part is in normal mode (the BM pin is high), it executes the power-on configuration routine of the kernel and then jumps to the reset vector address, 0x00000000, to execute the user's reset exception routine.

Because the Flash/EE is mirrored at the bottom of the memory array at reset, the reset interrupt routine must always be written in Flash/EE.

The remap is done from Flash/EE by setting Bit 0 of the REMAP register. Caution must be taken to execute this command from Flash/EE, above Address 0x00080020, and not from the bottom of the array because this is replaced by the SRAM.

This operation is reversible. The Flash/EE can be remapped at Address 0x00000000 by clearing Bit 0 of the REMAP MMR. Caution must again be taken to execute the remap function from outside the mirrored area. Any type of reset remaps the Flash/EE memory at the bottom of the array.

DIGITAL PERIPHERALS

3-PHASE PWM

Each ADuC7019/20/21/22/24/25/26/27/28/29 provides a flexible and programmable, 3-phase pulse-width modulation (PWM) waveform generator. It can be programmed to generate the required switching patterns to drive a 3-phase voltage source inverter for ac induction motor control (ACIM). Note that only active high patterns can be produced.

The PWM generator produces three pairs of PWM signals on the six PWM output pins ($PWM0_H$, $PWM0_L$, $PWM1_H$, $PWM1_L$, $PWM2_H$, and $PWM2_L$). The six PWM output signals consist of three high-side drive signals and three low-side drive signals.

The switching frequency and dead time of the generated PWM patterns are programmable using the PWMDAT0 and PWMDAT1 MMRs. In addition, three duty-cycle control registers (PWMCH0, PWMCH1, and PWMCH2) directly control the duty cycles of the three pairs of PWM signals.

Each of the six PWM output signals can be enabled or disabled by separate output enable bits of the PWMEN register. In addition, three control bits of the PWMEN register permit crossover of the two signals of a PWM pair. In crossover mode, the PWM signal destined for the high-side switch is diverted to the complementary low-side output. The signal destined for the low-side switch is diverted to the corresponding high-side output signal.

In many applications, there is a need to provide an isolation barrier in the gate-drive circuits that turn on the inverter power devices. In general, there are two common isolation techniques: optical isolation using optocouplers and transformer isolation using pulse transformers. The PWM controller permits mixing of the output PWM signals with a high frequency chopping signal to permit easy interface to such pulse transformers. The features of this gate-drive chopping mode can be controlled by the PWMCFG register. An 8-bit value within the PWMCFG register directly controls the chopping frequency. High frequency chopping can be independently enabled for the high-side and low-side outputs using separate control bits in the PWMCFG register.

The PWM generator can operate in one of two distinct modes: single update mode or double update mode. In single update mode, the duty cycle values are programmable only once per PWM period so that the resulting PWM patterns are symmetrical about the midpoint of the PWM period. In the double update mode, a second updating of the PWM duty cycle values is implemented at the midpoint of the PWM period.

In double update mode, it is also possible to produce asymmetrical PWM patterns that produce lower harmonic distortion in 3-phase PWM inverters. This technique permits closed-loop controllers to change the average voltage applied to the machine windings at a faster rate. As a result, faster closed-loop bandwidths are achieved. The operating mode of the PWM block is selected by a control bit in the PWMCON register. In single update mode, an internal synchronization pulse, PWMSYNC, is produced at the start of each PWM period. In double update mode, an additional PWMSYNC pulse is produced at the midpoint of each PWM period.

The PWM block can also provide an internal synchronization pulse on the PWM_{SYNC} pin that is synchronized to the PWM switching frequency. In single update mode, a pulse is produced at the start of each PWM period. In double update mode, an additional pulse is produced at the mid-point of each PWM period. The width of the pulse is programmable through the PWMDAT2 register. The PWM block can also accept an external synchronization pulse on the PWM_{SYNC} pin. The selection of external synchronization or internal synchronization is in the PWMCON register. The SYNC input timing can be synchronized to the internal peripheral clock, which is selected in the PWMCON register. If the external synchronization pulse from the chip pin is asynchronous to the internal peripheral clock (typical case), the external PWMSYNC is considered asynchronous and should be synchronized. The synchronization logic adds latency and jitter from the external pulse to the actual PWM outputs. The size of the pulse on the PWM_{SYNC} pin must be greater than two core clock periods.

The PWM signals produced by the ADuC7019/20/21/22/24/25/26/27/28/29 can be shut off via a dedicated asynchronous PWM shutdown pin, PWM_{TRIP}. When brought low, PWM_{TRIP} instantaneously places all six PWM outputs in the off state (high). This hardware shutdown mechanism is asynchronous so that the associated PWM disable circuitry does not go through any clocked logic. This ensures correct PWM shutdown even in the event of a core clock loss.

Status information about the PWM system is available to the user in the PWMSTA register. In particular, the state of the PWM_{TRIP} pin is available, as well as a status bit that indicates whether operation is in the first half or the second half of the PWM period.

40-Pin Package Devices

On the 40-pin package devices, the PWM outputs are not directly accessible, as described in the General-Purpose Input/Output section. One channel can be brought out on a GPIO (see Table 78) via the PLA as shown in the following example:

```
PWMCON = 0x1;           // enables PWM o/p
PWMDAT0 = 0x055F;       // PWM switching freq
// Configure Port Pins
GP4CON = 0x300;         // P4.2 as PLA output
GP3CON = 0x1;           // P3.0 configured as
                        // output of PWM0
                        //(internally)
// PWM0 onto P4.2
PLAELM8 = 0x0035;       // P3.0 (PWM output)
                        // input of element 8
PLAELM10 = 0x0059;      // PWM from element 8
```

Table 129. I2C0SSTA MMR Bit Descriptions

Bit	Value	Description
31:15		Reserved. These bits should be written as 0.
14		Start decode bit. Set by hardware if the device receives a valid start plus matching address. Cleared by an I²C stop condition or an I²C general call reset.
13		Repeated start decode bit. Set by hardware if the device receives a valid repeated start and matching address. Cleared by an I²C stop condition, a read of the I2CSSTA register, or an I²C general call reset.
12:11		ID decode bits.
	00	Received Address Matched ID Register 0.
	01	Received Address Matched ID Register 1.
	10	Received Address Matched ID Register 2.
	11	Received Address Matched ID Register 3.
10		Stop after start and matching address interrupt. Set by hardware if the slave device receives an I²C stop condition after a previous I²C start condition and matching address. Cleared by a read of the I2C0SSTA register.
9:8		General call ID.
	00	No general call.
	01	General call reset and program address.
	10	General call program address.
	11	General call matching alternative ID.
7		General call interrupt. Set if the slave device receives a general call of any type. Cleared by setting Bit 8 of the I2CxCFG register. If it is a general call reset, all registers are at their default values. If it is a hardware general call, the Rx FIFO holds the second byte of the general call. This is similar to the I2C0ALT register (unless it is a general call to reprogram the device address). For more details, see the I²C bus specification, Version 2.1, January 2000.
6		Slave busy. Set automatically if the slave is busy. Cleared automatically.
5		No ACK. Set if master asking for data and no data is available. Cleared automatically by reading the I2C0SSTA register.
4		Slave receive FIFO overflow. Set automatically if the slave receive FIFO is overflowing. Cleared automatically by reading the I2C0SSTA register.
3		Slave receive IRQ. Set after receiving data. Cleared automatically by reading the I2C0SRX register or flushing the FIFO.
2		Slave transmit IRQ. Set at the end of a transmission. Cleared automatically by writing to the I2C0STX register.
1		Slave transmit FIFO underflow. Set automatically if the slave transmit FIFO is underflowing. Cleared automatically by writing to the I2C0SSTA register.
0		Slave transmit FIFO not full. Set automatically if the slave transmit FIFO is not full. Cleared automatically by writing twice to the I2C0STX register.

Table 130. I2CxSRX Registers

Name	Address	Default Value	Access
I2C0SRX	0xFFFF0808	0x00	R
I2C1SRX	0xFFFF0908	0x00	R

I2CxSRX are receive registers for the slave channel.

Table 131. I2CxSTX Registers

Name	Address	Default Value	Access
I2C0STX	0xFFFF080C	0x00	W
I2C1STX	0xFFFF090C	0x00	W

I2CxSTX are transmit registers for the slave channel.

Table 132. I2CxMRX Registers

Name	Address	Default Value	Access
I2C0MRX	0xFFFF0810	0x00	R
I2C1MRX	0xFFFF0910	0x00	R

I2CxMRX are receive registers for the master channel.

Table 133. I2CxMTX Registers

Name	Address	Default Value	Access
I2C0MTX	0xFFFF0814	0x00	W
I2C1MTX	0xFFFF0914	0x00	W

I2CxMTX are transmit registers for the master channel.

Table 134. I2CxCNT Registers

Name	Address	Default Value	Access
I2C0CNT	0xFFFF0818	0x00	R/W
I2C1CNT	0xFFFF0918	0x00	R/W

I2CxCNT are 3-bit, master receive, data count registers. If a master read transfer sequence is initiated, the I2CxCNT registers denote the number of bytes (−1) to be read from the slave device. By default, this counter is 0, which corresponds to the one byte expected.

Table 135. I2CxADR Registers

Name	Address	Default Value	Access
I2C0ADR	0xFFFF081C	0x00	R/W
I2C1ADR	0xFFFF091C	0x00	R/W

I2CxADR are master address byte registers. The I2CxADR value is the device address that the master wants to communicate with. It automatically transmits at the start of a master transfer sequence if there is no valid data in the I2CxMTX register when the master enable bit is set.

Table 136. I2CxBYTE Registers

Name	Address	Default Value	Access
I2C0BYTE	0xFFFF0824	0x00	R/W
I2C1BYTE	0xFFFF0924	0x00	R/W

I2CxBYTE are broadcast byte registers. Data written to these registers does not go through the TxFIFO. This data is transmitted at the start of a transfer sequence before the address. After the byte is transmitted and acknowledged, the I²C expects another byte written in I2CxBYTE or an address written to the address register.

Timer1 (General-Purpose Timer)

Timer1 is a general-purpose, 32-bit timer (count down or count up) with a programmable prescaler. The source can be the 32 kHz external crystal, the core clock frequency, or an external GPIO (P1.0 or P0.6). The maximum frequency of the clock input is 44 Mhz). This source can be scaled by a factor of 1, 16, 256, or 32,768.

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

Timer1 has a capture register (T1CAP) that can be triggered by a selected IRQ source initial assertion. This feature can be used to determine the assertion of an event more accurately than the precision allowed by the RTOS timer when the IRQ is serviced.

Timer1 can be used to start ADC conversions as shown in the block diagram in Figure 78.

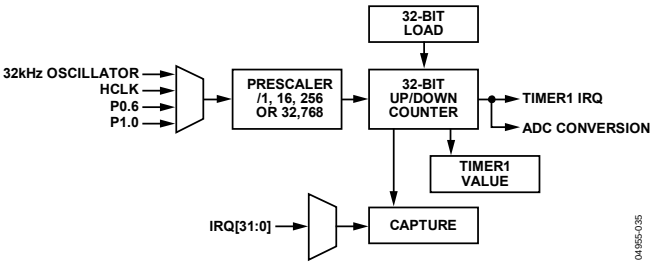


Figure 78. Timer1 Block Diagram

The Timer1 interface consists of five MMRs: T1LD, T1VAL, T1CON, T1CLRI, and T1CAP.

Table 177. T1LD Register

Name	Address	Default Value	Access
T1LD	0xFFFF0320	0x00000000	R/W

T1LD is a 32-bit load register.

Table 178. T1VAL Register

Name	Address	Default Value	Access
T1VAL	0xFFFF0324	0xFFFFFFFF	R

T1VAL is a 32-bit read-only register that represents the current state of the counter.

Table 179. T1CON Register

Name	Address	Default Value	Access
T1CON	0xFFFF0328	0x0000	R/W

T1CON is the configuration MMR described in Table 180.

Table 180. T1CON MMR Bit Descriptions

Bit	Value	Description
31:18		Reserved.
17		Event select bit. Set by user to enable time capture of an event. Cleared by user to disable time capture of an event.
16:12		Event select range, 0 to 31. These events are as described in Table 160. All events are offset by two; that is, Event 2 in Table 160 becomes Event 0 for the purposes of Timer1.
11:9		Clock select.
	000	Core clock (HCLK).
	001	External 32.768 kHz crystal.
	010	P1.0 rising edge triggered.
	011	P0.6 rising edge triggered.
8		Count up. Set by user for Timer1 to count up. Cleared by user for Timer1 to count down by default.
7		Timer1 enable bit. Set by user to enable Timer1. Cleared by user to disable Timer1 by default.
6		Timer1 mode. Set by user to operate in periodic mode. Cleared by user to operate in free-running mode. Default mode.
5:4		Format.
	00	Binary.
	01	Reserved.
	10	Hr: min: sec: hundredths (23 hours to 0 hour).
	11	Hr: min: sec: hundredths (255 hours to 0 hour).
3:0		Prescale.
	0000	Source Clock/1.
	0100	Source Clock/16.
	1000	Source Clock/256.
	1111	Source Clock/32,768.

Table 181. T1CLRI Register

Name	Address	Default Value	Access
T1CLRI	0xFFFF032C	0xFF	W

T1CLRI is an 8-bit register. Writing any value to this register clears the Timer1 interrupt.

Table 182. T1CAP Register

Name	Address	Default Value	Access
T1CAP	0xFFFF0330	0x00000000	R/W

T1CAP is a 32-bit register. It holds the value contained in T1VAL when a particular event occurs. This event must be selected in T1CON.

Timer2 (Wake-Up Timer)

Timer2 is a 32-bit wake-up timer (count down or count up) with a programmable prescaler. The source can be the 32 kHz external crystal, the core clock frequency, or the internal 32 kHz oscillator. The 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.

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

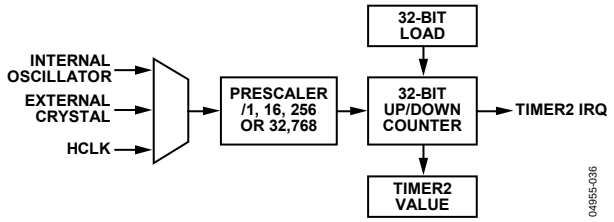


Figure 79. Timer2 Block Diagram

The Timer2 interface consists of four MMRs: T2LD, T2VAL, T2CON, and T2CLRI.

Table 183. T2LD Register

Name	Address	Default Value	Access
T2LD	0xFFFF0340	0x00000000	R/W

T2LD is a 32-bit register load register.

Table 184. T2VAL Register

Name	Address	Default Value	Access
T2VAL	0xFFFF0344	0xFFFFFFFF	R

T2VAL is a 32-bit read-only register that represents the current state of the counter.

Table 185. T2CON Register

Name	Address	Default Value	Access
T2CON	0xFFFF0348	0x0000	R/W

T2CON is the configuration MMR described in Table 186.

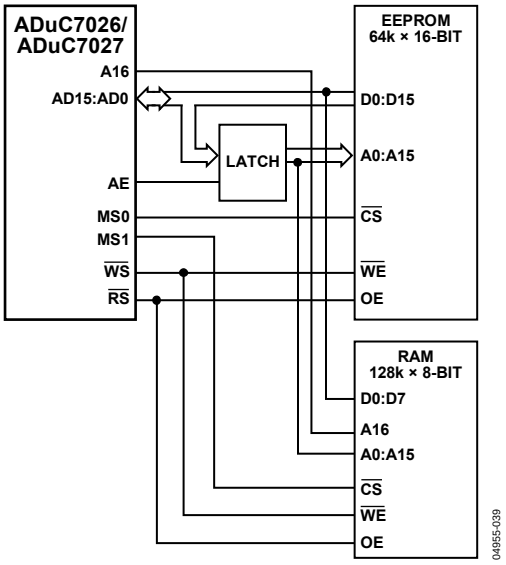


Figure 82. Interfacing to External EEPROM/RAM

Table 195. XMCFG Register

Name	Address	Default Value	Access
XMCFG	0xFFFFF000	0x00	R/W

XMCFG is set to 1 to enable external memory access. This must be set to 1 before any port pins function as external memory access pins. The port pins must also be individually enabled via the GPxCON MMR.

Table 196. XMxCON Registers

Name	Address	Default Value	Access
XM0CON	0xFFFFF010	0x00	R/W
XM1CON	0xFFFFF014	0x00	R/W
XM2CON	0xFFFFF018	0x00	R/W
XM3CON	0xFFFFF01C	0x00	R/W

XMxCON are the control registers for each memory region. They allow the enabling/disabling of a memory region and control the data bus width of the memory region.

Table 197. XMxCON MMR Bit Descriptions

Bit	Description
1	Selects data bus width. Set by user to select a 16-bit data bus. Cleared by user to select an 8-bit data bus.
0	Enables memory region. Set by user to enable the memory region. Cleared by user to disable the memory region.

Table 198. XMxPAR Registers

Name	Address	Default Value	Access
XM0PAR	0xFFFFF020	0x70FF	R/W
XM1PAR	0xFFFFF024	0x70FF	R/W
XM2PAR	0xFFFFF028	0x70FF	R/W
XM3PAR	0xFFFFF02C	0x70FF	R/W

XMxPAR are registers that define the protocol used for accessing the external memory for each memory region.

Table 199. XMxPAR MMR Bit Descriptions

Bit	Description
15	Enable byte write strobe. This bit is used only for two, 8-bit memory devices sharing the same memory region. Set by the user to gate the A0 output with the $\overline{WS}$ output. This allows byte write capability without using $\overline{BHE}$ and $\overline{BLE}$ signals. Cleared by user to use $\overline{BHE}$ and $\overline{BLE}$ signals.
14:12	Number of wait states on the address latch enable STROBE.
11	Reserved.
10	Extra address hold time. Set by user to disable extra hold time. Cleared by user to enable one clock cycle of hold on the address in read and write.
9	Extra bus transition time on read. Set by user to disable extra bus transition time. Cleared by user to enable one extra clock before and after the read strobe ($\overline{RS}$).
8	Extra bus transition time on write. Set by user to disable extra bus transition time. Cleared by user to enable one extra clock before and after the write strobe ($\overline{WS}$).
7:4	Number of write wait states. Select the number of wait states added to the length of the $\overline{WS}$ pulse. 0x0 is 1 clock; 0xF is 16 clock cycles (default value).
3:0	Number of read wait states. Select the number of wait states added to the length of the $\overline{RS}$ pulse. 0x0 is 1 clock; 0xF is 16 clock cycles (default value).

Figure 83, Figure 84, Figure 85, and Figure 86 show the timing for a read cycle, a read cycle with address hold and bus turn cycles, a write cycle with address and write hold cycles, and a write cycle with wait sates, respectively.

DEVELOPMENT TOOLS

PC-BASED TOOLS

Four types of development systems are available for the ADuC7019/20/21/22/24/25/26/27/28/29 family.

- The ADuC7026 QuickStart Plus is intended for new users who want to have a comprehensive hardware development environment. Because the ADuC7026 contains the superset of functions available on the ADuC7019/20/21/22/24/25/26/27/28/29, it is suitable for users who wish to develop on any of the parts in this family. All parts are fully code compatible.
- The ADuC7019, ADuC7024, and ADuC7026 QuickStart systems are intended for users who already have an emulator.

These systems consist of the following PC-based (Windows® compatible) hardware and software development tools.

Hardware

- ADuC7019/20/21/22/24/25/26/27/28/29 evaluation board
- Serial port programming cable
- RDI-compliant JTAG emulator (included in the ADuC7026 QuickStart Plus only)

Software

- Integrated development environment, incorporating assembler, compiler, and nonintrusive JTAG-based debugger
- Serial downloader software
- Example code

Miscellaneous

CD-ROM documentation

IN-CIRCUIT SERIAL DOWNLOADER

The serial downloader is a Windows application that allows the user to serially download an assembled program to the on-chip program Flash/EE memory via the serial port on a standard PC.

The UART-based serial downloader is included in all the development systems and is usable with the ADuC7019/20/21/22/24/25/26/27/28/29 parts that do not contain the I suffix in the Ordering Guide.

An I^2C based serial downloader and a USB-to-I^2C adaptor board, USB-EA-CONVZ, are also available at www.analog.com. The I^2C-based serial downloader is only usable with the part models containing the I suffix (see Ordering Guide).

Model[1, 2]	ADC Channels[3]	DAC Channels	FLASH/ RAM	GPIO	Down-loader	Temperature Range	Package Description	Package Option	Ordering Quantity
EVAL-ADuC7020MKZ							ADuC7020 MiniKit		
EVAL-ADuC7020QSZ							ADuC7020 QuickStart Development System		
EVAL-ADuC7020QSPZ							ADuC7020 QuickStart Development System		
EVAL-ADuC7024QSZ							ADuC7024 QuickStart Development System		
EVAL-ADuC7026QSZ							ADuC7026 QuickStar Development System		
EVAL-ADuC7026QSPZ							ADuC7026 QuickStart Plus Development System		
EVAL-ADuC7028QSZ							ADuC7028 QuickStart Development System		
EVAL-ADUC7029QSZ							ADuC7029 QuickStart Development System		

[1] Z = RoHS Compliant Part.
[2] Models ADuC7026 and ADuC7027 include an external memory interface.
[3] One of the ADC channels is internally buffered for ADuC7019 models.

I²C refers to a communications protocol originally developed by Phillips Semiconductors (now NXP Semiconductors).