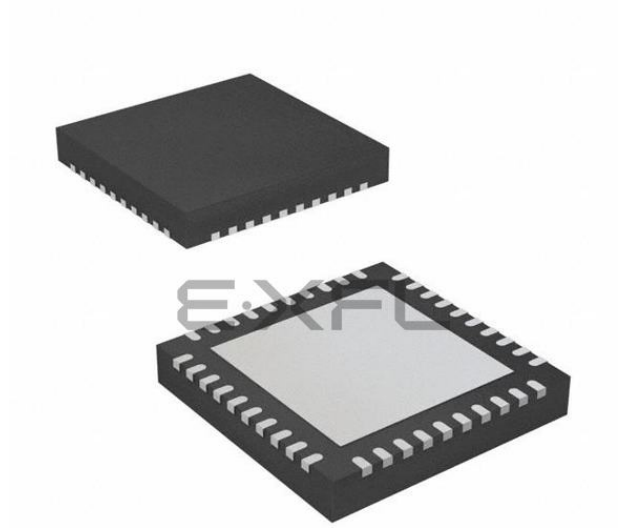


**Welcome to E-XFL.COM**

## 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 | 13 |
| Program Memory Size | 32KB (16K x 16) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 1K x 32 |
| Voltage - Supply (Vcc/Vdd) | 2.7V ~ 3.6V |
| Data Converters | A/D 10x12b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 125°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 40-VFQFN Exposed Pad, CSP |
| Supplier Device Package | 40-LFCSP-VQ (6x6) |
| Purchase URL | https://www.e-xfl.com/product-detail/analog-devices/aduc7022bcpz32 |

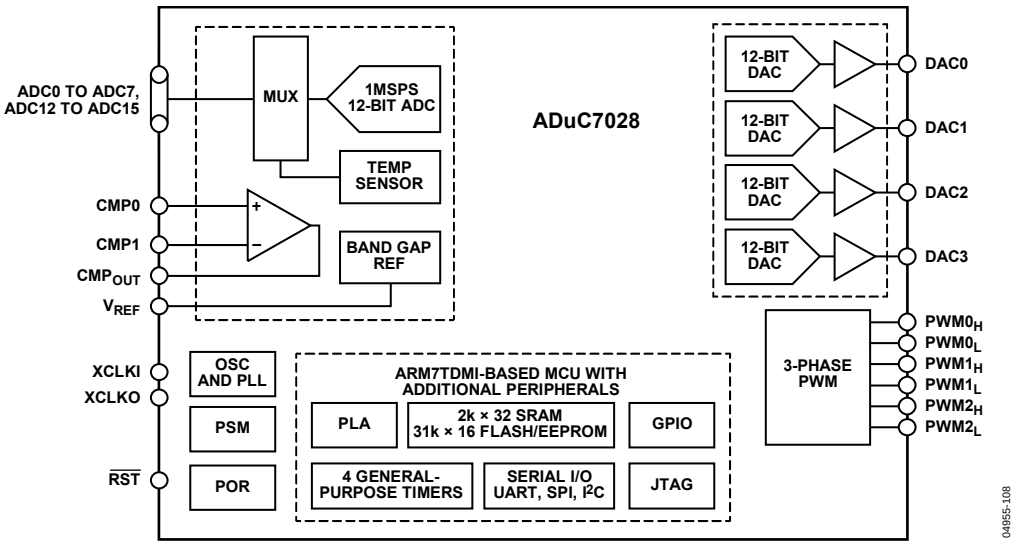



*Figure 9.*

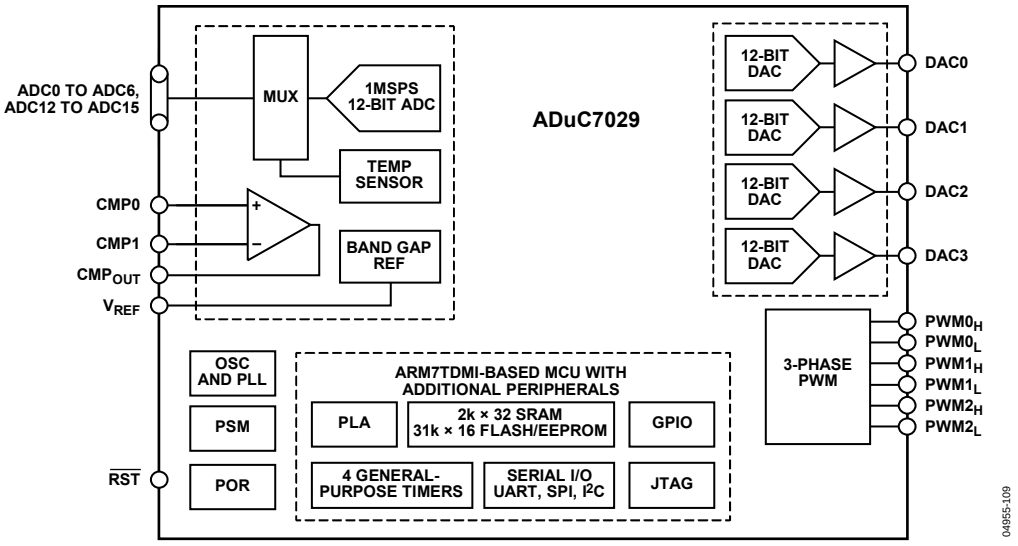



*Figure 10.*

**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.

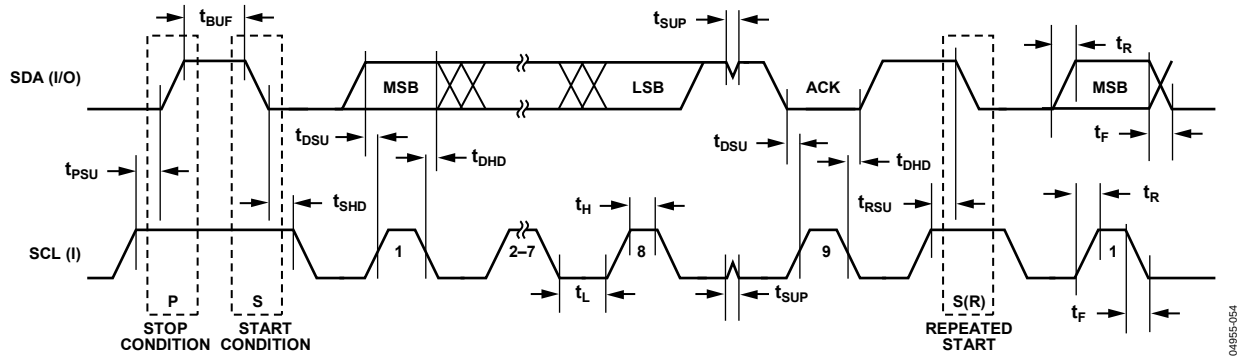



*Figure 14. I²C Compatible Interface Timing*

**Table 6. SPI Master Mode Timing (Phase Mode = 1)**

| Parameter | Description | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| $t_{SL}$ | SCLK low pulse width[1] | | (SPIDIV + 1) × $t_{HCLK}$ | | ns |
| $t_{SH}$ | SCLK high pulse width[1] | | (SPIDIV + 1) × $t_{HCLK}$ | | ns |
| $t_{DAV}$ | Data output valid after SCLK edge | | | 25 | ns |
| $t_{DSU}$ | Data input setup time before SCLK edge[2] | 1 × $t_{UCLK}$ | | | ns |
| $t_{DHD}$ | Data input hold time after SCLK edge[2] | 2 × $t_{UCLK}$ | | | ns |
| $t_{DF}$ | Data output fall time | | 5 | 12.5 | ns |
| $t_{DR}$ | Data output rise time | | 5 | 12.5 | ns |
| $t_{SR}$ | SCLK rise time | | 5 | 12.5 | ns |
| $t_{SF}$ | SCLK fall time | | 5 | 12.5 | ns |

[1] $t_{HCLK}$ depends on the clock divider or CD bits in the POWCONMMR. $t_{HCLK} = t_{UCLK}/2^{CD}$; see Figure 67.
[2] $t_{UCLK}$ = 23.9 ns. It corresponds to the 41.78 MHz internal clock from the PLL before the clock divider; see Figure 67.

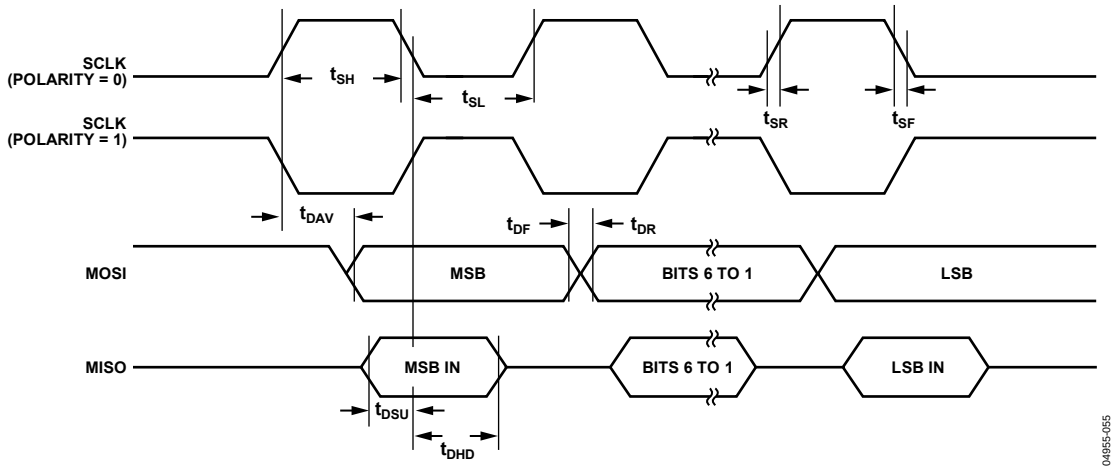



*Figure 15. SPI Master Mode Timing (Phase Mode = 1)*

**Table 12. Pin Function Descriptions (ADuC7024/ADuC7025 64-Lead LFCSP_VQ and 64-Lead LQFP)**

| Pin No. | Mnemonic | Description |
|---|---|---|
| 1 | ADC4 | Single-Ended or Differential Analog Input 4. |
| 2 | ADC5 | Single-Ended or Differential Analog Input 5. |
| 3 | ADC6 | Single-Ended or Differential Analog Input 6. |
| 4 | ADC7 | Single-Ended or Differential Analog Input 7. |
| 5 | ADC8 | Single-Ended or Differential Analog Input 8. |
| 6 | ADC9 | Single-Ended or Differential Analog Input 9. |
| 7 | $GND_{REF}$ | Ground Voltage Reference for the ADC. For optimal performance, the analog power supply should be separated from IOGND and DGND. |
| 8 | 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. |
| 9 | DAC0/ADC12 | DAC0 Voltage Output/Single-Ended or Differential Analog Input 12. DAC outputs are not present on the ADuC7025. |
| 10 | DAC1/ADC13 | DAC1 Voltage Output/Single-Ended or Differential Analog Input 13. DAC outputs are not present on the ADuC7025. |
| 11 | TMS | JTAG Test Port Input, Test Mode Select. Debug and download access. |
| 12 | TDI | JTAG Test Port Input, Test Data In. Debug and download access |
| 13 | P4.6/PLAO[14] | General-Purpose Input and Output Port 4.6/Programmable Logic Array Output Element 14. |
| 14 | P4.7/PLAO[15] | General-Purpose Input and Output Port 4.7/Programmable Logic Array Output Element 15. |
| 15 | BM/P0.0/$CMP_{OUT}$/PLAI[7] | Multifunction I/O Pin. Boot mode. The ADuC7024/ADuC7025 enter 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. |
| 16 | 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. |
| 17 | TCK | JTAG Test Port Input, Test Clock. Debug and download access. |
| 18 | TDO | JTAG Test Port Output, Test Data Out. Debug and download access. |
| 19 | IOGND | Ground for GPIO (see Table 78). Typically connected to DGND. |
| 20 | $IOV_{DD}$ | 3.3 V Supply for GPIO (see Table 78) and Input of the On-Chip Voltage Regulator. |
| 21 | $LV_{DD}$ | 2.6 V Output of the On-Chip Voltage Regulator. This output must be connected to a 0.47 µF capacitor to DGND only. |
| 22 | DGND | Ground for Core Logic. |
| 23 | P3.0/$PWM0_H$/PLAI[8] | General-Purpose Input and Output Port 3.0/PWM Phase 0 High-Side Output/Programmable Logic Array Input Element 8. |
| 24 | P3.1/$PWM0_L$/PLAI[9] | General-Purpose Input and Output Port 3.1/PWM Phase 0 Low-Side Output/Programmable Logic Array Input Element 9. |
| 25 | P3.2/$PWM1_H$/PLAI[10] | General-Purpose Input and Output Port 3.2/PWM Phase 1 High-Side Output/Programmable Logic Array Input Element 10. |
| 26 | P3.3/$PWM1_L$/PLAI[11] | General-Purpose Input and Output Port 3.3/PWM Phase 1 Low-Side Output/Programmable Logic Array Input Element 11. |
| 27 | P0.3/TRST/$ADC_{BUSY}$ | General-Purpose Input and Output Port 0.3/JTAG Test Port Input, Test Reset/$ADC_{BUSY}$ Signal Output. |
| 28 | $\overline{RST}$ | Reset Input, Active Low. |
| 29 | P3.4/$PWM2_H$/PLAI[12] | General-Purpose Input and Output Port 3.4/PWM Phase 2 High-Side Output/Programmable Logic Array Input 12. |
| 30 | P3.5/$PWM2_L$/PLAI[13] | General-Purpose Input and Output Port 3.5/PWM Phase 2 Low-Side Output/Programmable Logic Array Input Element 13. |
| 31 | IRQ0/P0.4/$PWM_{TRIP}$/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. |
| 32 | IRQ1/P0.5/$ADC_{BUSY}$/PLAO[2] | 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. |
| 33 | 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. |
| 34 | 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. |
| 35 | XCLKO | Output from the Crystal Oscillator Inverter. |
| 36 | XCLKI | Input to the Crystal Oscillator Inverter and Input to the Internal Clock Generator Circuits. |

## ADuC7026/ADuC7027

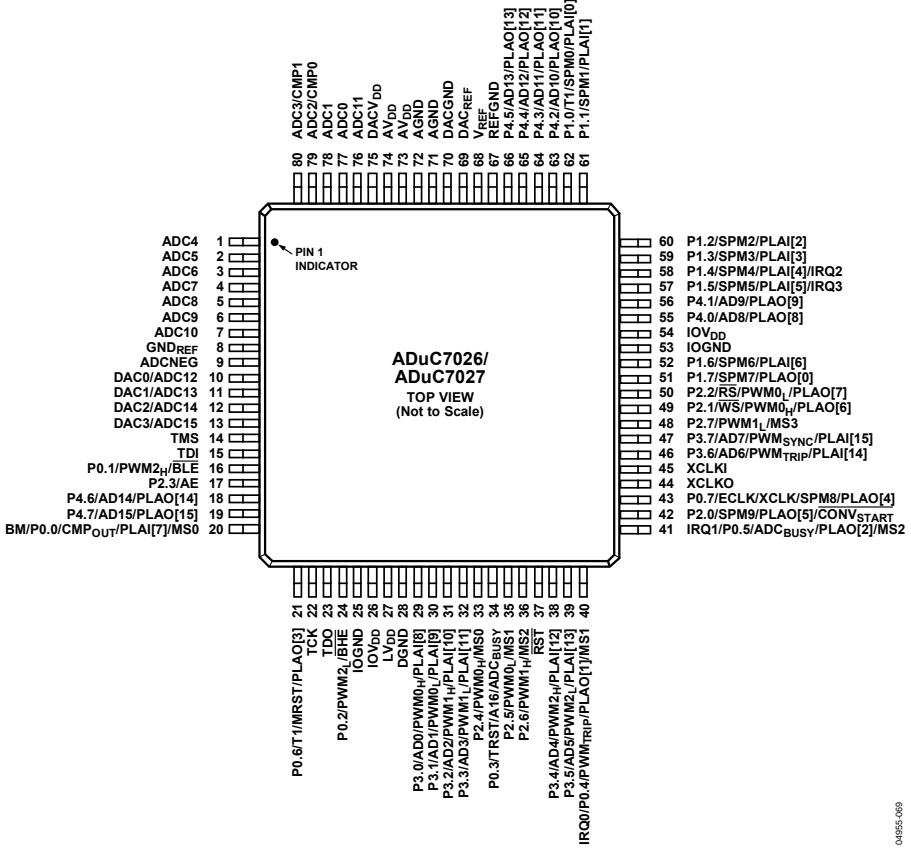



*Figure 25. 80-Lead LQFP Pin Configuration (ADuC7026/ADuC7027)*

**Table 13. Pin Function Descriptions (ADuC7026/ADuC7027)**

| Pin No. | Mnemonic | Description |
|---|---|---|
| 1 | ADC4 | Single-Ended or Differential Analog Input 4. |
| 2 | ADC5 | Single-Ended or Differential Analog Input 5. |
| 3 | ADC6 | Single-Ended or Differential Analog Input 6. |
| 4 | ADC7 | Single-Ended or Differential Analog Input 7. |
| 5 | ADC8 | Single-Ended or Differential Analog Input 8. |
| 6 | ADC9 | Single-Ended or Differential Analog Input 9. |
| 7 | ADC10 | Single-Ended or Differential Analog Input 10. |
| 8 | $GND_{REF}$ | Ground Voltage Reference for the ADC. For optimal performance, the analog power supply should be separated from IOGND and DGND. |
| 9 | 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. |
| 10 | DAC0/ADC12 | DAC0 Voltage Output/Single-Ended or Differential Analog Input 12. DAC outputs are not present on the ADuC7027. |
| 11 | DAC1/ADC13 | DAC1 Voltage Output/Single-Ended or Differential Analog Input 13. DAC outputs are not present on the ADuC7027. |
| 12 | DAC2/ADC14 | DAC2 Voltage Output/Single-Ended or Differential Analog Input 14. DAC outputs are not present on the ADuC7027. |
| 13 | DAC3/ADC15 | DAC3 Voltage Output/Single-Ended or Differential Analog Input 15. DAC outputs are not present on the ADuC7027. |
| 14 | TMS | JTAG Test Port Input, Test Mode Select. Debug and download access. |
| 15 | TDI | JTAG Test Port Input, Test Data In. Debug and download access. |
| 16 | $P0.1/PWM2_H/\overline{BLE}$ | General-Purpose Input and Output Port 0.1/PWM Phase 2 High-Side Output/External Memory Byte Low Enable. |
| 17 | P2.3/AE | General-Purpose Input and Output Port 2.3/External Memory Access Enable. |

| Pin No. | Mnemonic | Description |
|---|---|---|
| E1 | TMS | JTAG Test Port Input, Test Mode Select. Debug and download access. |
| E2 | BM/P0.0/$CMP_{OUT}$/PLAI[7] | Multifunction I/O Pin. Boot mode. The ADuC7029 enters UART download mode if BM is low at reset and executes 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. |
| E3 | DAC2/ADC14 | DAC2 Voltage Output/ADC Input 14. |
| E4 | $IOV_{DD}$ | 3.3 V Supply for GPIO (see Table 78) and Input of the On-Chip Voltage Regulator. |
| E5 | P3.2/$PWM1_H$/PLAI[10] | General-Purpose Input and Output Port 3.2/PWM Phase 1 High-Side Output/Programmable Logic Array Input Element 10. |
| E6 | P3.5/$PWM2_L$/PLAI[13] | General-Purpose Input and Output Port 3.5/PWM Phase 2 Low-Side Output/Programmable Logic Array Input Element 13. |
| E7 | 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. |
| F1 | TDI | JTAG Test Port Input, Test Data In. Debug and download access. |
| F2 | 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. |
| F3 | IOGND | Ground for GPIO (see Table 78). Typically connected to DGND. |
| F4 | P3.1/$PWM0_L$/PLAI[9] | General-Purpose Input and Output Port 3.1/PWM Phase 0 Low-Side Output/Programmable Logic Array Input Element 9. |
| F5 | P3.0/$PWM0_H$/PLAI[8] | General-Purpose Input and Output Port 3.0/PWM Phase 0 High-Side Output/Programmable Logic Array Input Element 8. |
| F6 | $\overline{RST}$ | Reset Input, Active Low. |
| F7 | 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. |
| G1 | TCK | JTAG Test Port Input, Test Clock. Debug and download access. |
| G2 | TDO | JTAG Test Port Output, Test Data Out. Debug and download access. |
| G3 | $LV_{DD}$ | 2.6 V Output of the On-Chip Voltage Regulator. This output must be connected to a 0.47 µF capacitor to DGND only. |
| G4 | DGND | Ground for Core Logic. |
| G5 | P0.3/TRST/$ADC_{BUSY}$ | General-Purpose Input and Output Port 0.3/JTAG Test Port Input, Test Reset/$ADC_{BUSY}$ Signal Output. |
| G6 | IRQ0/P0.4/$PWM_{TRIP}$/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. |
| G7 | IRQ1/P0.5/$ADC_{BUSY}$/PLAO[2] | 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. |

# MEMORY ORGANIZATION

The ADuC7019/20/21/22/24/25/26/27/28/29 incorporate two separate blocks of memory: 8 kB of SRAM and 64 kB of on-chip Flash/EE memory. The 62 kB of on-chip Flash/EE memory is available to the user, and the remaining 2 kB are reserved for the factory-configured boot page. These two blocks are mapped as shown in Figure 45.

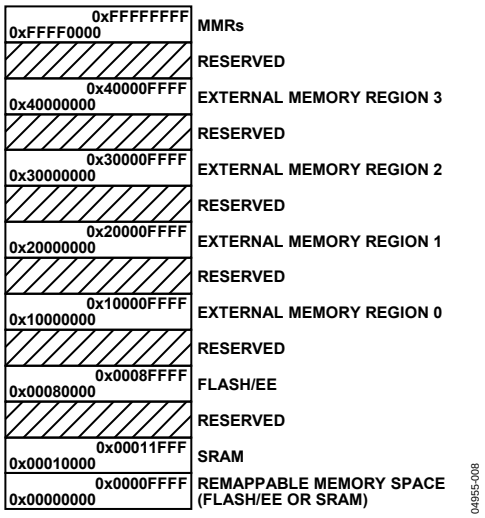



*Figure 45. Physical Memory Map*

Note that by default, after a reset, the Flash/EE memory is mirrored at Address 0x00000000. It is possible to remap the SRAM at Address 0x00000000 by clearing Bit 0 of the REMAP MMR. This remap function is described in more detail in the Flash/EE Memory section.

## MEMORY ACCESS

The ARM7 core sees memory as a linear array of a $2^{32}$ byte location where the different blocks of memory are mapped as outlined in Figure 45.

The ADuC7019/20/21/22/24/25/26/27/28/29 memory organizations are configured in little endian format, which means that the least significant byte is located in the lowest byte address, and the most significant byte is in the highest byte address.

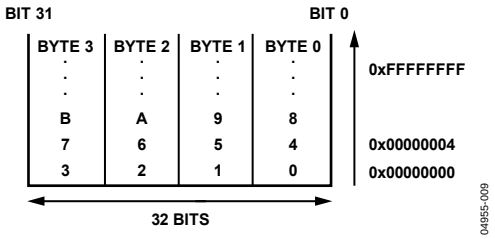



*Figure 46. Little Endian Format*

## FLASH/EE MEMORY

The total 64 kB of Flash/EE memory is organized as 32 k × 16 bits; 31 k × 16 bits is user space and 1 k × 16 bits is reserved for the on-chip kernel. The page size of this Flash/EE memory is 512 bytes.

Sixty-two kilobytes of Flash/EE memory are available to the user as code and nonvolatile data memory. There is no distinction between data and program because ARM code shares the same space. The real width of the Flash/EE memory is 16 bits, which means that in ARM mode (32-bit instruction), two accesses to the Flash/EE are necessary for each instruction fetch. It is therefore recommended to use thumb mode when executing from Flash/EE memory for optimum access speed. The maximum access speed for the Flash/EE memory is 41.78 MHz in thumb mode and 20.89 MHz in full ARM mode. More details about Flash/EE access time are outlined in the Execution Time from SRAM and Flash/EE section.

## SRAM

Eight kilobytes of SRAM are available to the user, organized as 2 k × 32 bits, that is, two words. ARM code can run directly from SRAM at 41.78 MHz, given that the SRAM array is configured as a 32-bit wide memory array. More details about SRAM access time are outlined in the Execution Time from SRAM and Flash/EE section.

## MEMORY MAPPED REGISTERS

The memory mapped register (MMR) space is mapped into the upper two pages of the memory array and accessed by indirect addressing through the ARM7 banked registers.

The MMR space provides an interface between the CPU and all on-chip peripherals. All registers, except the core registers, reside in the MMR area. All shaded locations shown in Figure 47 are unoccupied or reserved locations and should not be accessed by user software. Table 16 shows the full MMR memory map.

The access time for reading from or writing to an MMR depends on the advanced microcontroller bus architecture (AMBA) bus used to access the peripheral. The processor has two AMBA buses: the advanced high performance bus (AHB) used for system modules and the advanced peripheral bus (APB) used for lower performance peripheral. Access to the AHB is one cycle, and access to the APB is two cycles. All peripherals on the ADuC7019/20/21/22/24/25/26/27/28/29 are on the APB except the Flash/EE memory, the GPIOs (see Table 78), and the PWM.

Linearity degradation near ground and $AV_{DD}$ 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-$AV_{DD}$ mode only. In 0-to-$V_{REF}$ or 0-to-$DAC_{REF}$ mode (with $V_{REF} < AV_{DD}$ or $DAC_{REF} < AV_{DD}$), the lower nonlinearity is similar. However, the upper portion of the transfer function follows the ideal line right to the end ($V_{REF}$ in this case, not $AV_{DD}$), showing no signs of endpoint linearity errors.

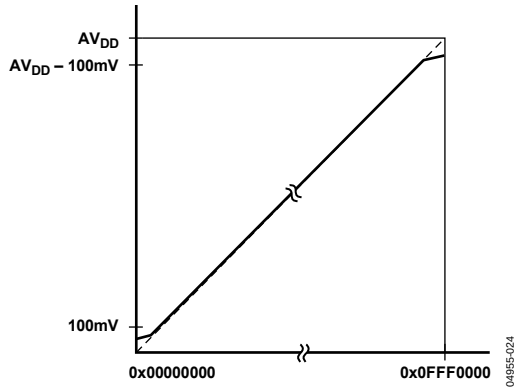



*Figure 64. Endpoint Nonlinearities Due to Amplifier Saturation*

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 kΩ 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 |

**Table 54. PSMCON MMR Bit Descriptions**

| Bit | Name | Description |
|---|---|---|
| 3 | CMP | Comparator bit. This is a read-only bit that directly reflects the state of the comparator. Read 1 indicates that the $IOV_{DD}$ supply is above its selected trip point or that the PSM is in power-down mode. Read 0 indicates that the $IOV_{DD}$ 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 V, 1 = 3.07 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, $CMP_{OUT}$, as shown in Figure 65.

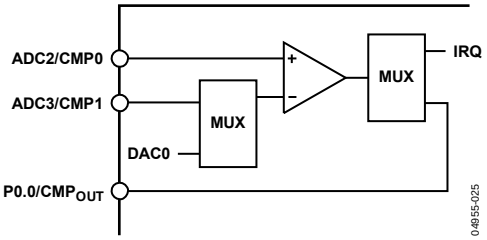



*Figure 65. Comparator*

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.

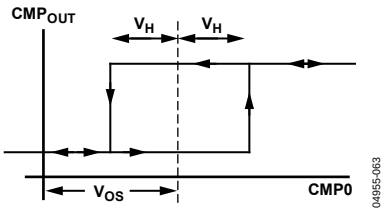



*Figure 66. Comparator Hysteresis Transfer Function*

Input offset voltage ($V_{OS}$) is the difference between the center of the hysteresis range and the ground level. This can either be positive or negative. The hysteresis voltage ($V_H$) is one-half the width of the hysteresis range.

### *Comparator Interface*

The comparator interface consists of a 16-bit MMR, CMPCON, which is described in Table 56.

**Table 55. CMPCON Register**

| Name | Address | Default Value | Access |
|---|---|---|---|
| CMPCON | 0xFFFF0444 | 0x0000 | R/W |

**Table 56. CMPCON MMR Bit Descriptions**

| Bit | Name | Value | Description |
|---|---|---|---|
| 15:11 | | | Reserved. |
| 10 | CMPEN | | Comparator enable bit. Set by user to enable the comparator. Cleared by user to disable the comparator. |
| 9:8 | CMPIN | | Comparator negative input select bits. |
| | | 00 | $AV_{DD}/2$. |
| | | 01 | ADC3 input. |
| | | 10 | DAC0 output. |
| | | 11 | Reserved. |
| 7:6 | CMPOC | | Comparator output configuration bits. |
| | | 00 | Reserved. |
| | | 01 | Reserved. |
| | | 10 | Output on $CMP_{OUT}$. |
| | | 11 | IRQ. |
| 5 | CMPOL | | Comparator output logic state bit. When low, the comparator output is high if the positive input (CMP0) is above the negative input (CMP1). When high, the comparator output is high if the positive input is below the negative input. |
| 4:3 | CMPRES | | Response time. |
| | | 00 | 5 µs response time is typical for large signals (2.5 V differential). 17 µs response time is typical for small signals (0.65 mV differential). |
| | | 11 | 3 µs typical. |
| | | 01/10 | Reserved. |
| 2 | CMPHYST | | Comparator hysteresis bit. Set by user to have a hysteresis of about 7.5 mV. Cleared by user to have no hysteresis. |
| 1 | CMPORI | | Comparator output rising edge interrupt. Set automatically when a rising edge occurs on the monitored voltage (CMP0). Cleared by user by writing a 1 to this bit. |
| 0 | CMPOFI | | Comparator output falling edge interrupt. Set automatically when a falling edge occurs on the monitored voltage (CMP0). Cleared by user. |

## OSCILLATOR AND PLL—POWER CONTROL

### *Clocking System*

Each ADuC7019/20/21/22/24/25/26/27/28/29 integrates a 32.768 kHz ±3% oscillator, a clock divider, and a PLL. The PLL locks onto a multiple (1275) of the internal oscillator or an external 32.768 kHz crystal to provide a stable 41.78 MHz clock (UCLK) for the system. To allow power saving, the core can operate at this frequency, or at binary submultiples of it. The actual core operating frequency, $UCLK/2^{CD}$, is refered to as HCLK. The default core clock is the PLL clock divided by 8 (CD = 3) or 5.22 MHz. The core clock frequency can also come from an external clock on the ECLK pin as described in Figure 67. The core clock can be outputted on ECLK when using an internal oscillator or external crystal.

Note that when the ECLK pin is used to output the core clock, the output signal is not buffered and is not suitable for use as a clock source to an external device without an external buffer.

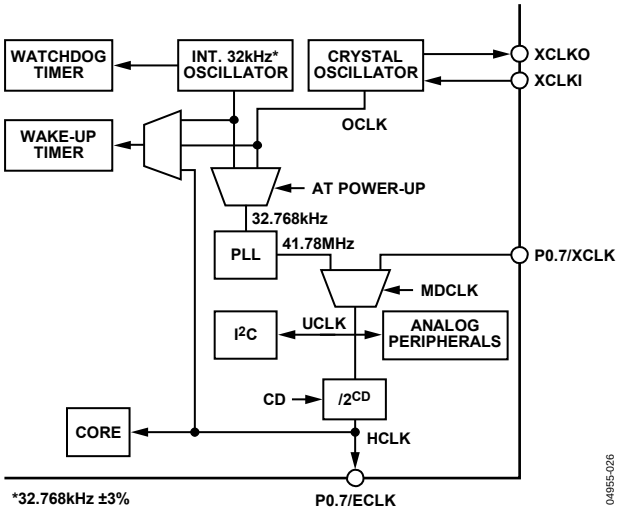



*Figure 67. Clocking System*

The selection of the clock source is in the PLLCON register. By default, the part uses the internal oscillator feeding the PLL.

### External Crystal Selection

To switch to an external crystal, the user must do the following:

1. Enable the Timer2 interrupt and configure it for a timeout period of >120 µs.
2. Follow the write sequence to the PLLCON register, setting the MDCLK bits to 01 and clearing the OSEL bit.
3. Force the part into NAP mode by following the correct write sequence to the POWCON register.

   When the part is interrupted from NAP mode by the Timer2 interrupt source, the clock source has switched to the external clock.

**Example source code**

```
    t2val_old= T2VAL;
     T2LD = 5;
     TCON = 0x480;


  while ((T2VAL == t2val_old) || (T2VAL >
3)) //ensures timer value loaded
  IRQEN = 0x10;
//enable T2 interrupt
  PLLKEY1 = 0xAA;
  PLLCON = 0x01;
  PLLKEY2 = 0x55;

  POWKEY1 = 0x01;
  POWCON = 0x27;
// Set Core into Nap mode
  POWKEY2 = 0xF4;
```

In noisy environments, noise can couple to the external crystal pins, and PLL may lose lock momentarily. A PLL interrupt is provided in the interrupt controller. The core clock is immediately halted, and this interrupt is only serviced when the lock is restored.

In case of crystal loss, the watchdog timer should be used. During initialization, a test on the RSTSTA register can determine if the reset came from the watchdog timer.

### External Clock Selection

To switch to an external clock on P0.7, configure P0.7 in Mode 1. The external clock can be up to 44 MHz, providing the tolerance is 1%.

**Example source code**

```
    t2val_old= T2VAL;
     T2LD = 5;
     TCON = 0x480;


  while ((T2VAL == t2val_old) || (T2VAL
> 3)) //ensures timer value loaded
  IRQEN = 0x10;
//enable T2 interrupt
  PLLKEY1 = 0xAA;
  PLLCON = 0x03; //Select external clock
  PLLKEY2 = 0x55;

  POWKEY1 = 0x01;
  POWCON = 0x27;
// Set Core into Nap mode
  POWKEY2 = 0xF4;
```

### *Power Control System*

A choice of operating modes is available on the ADuC7019/20/21/22/24/25/26/27/28/29. Table 57 describes what part is powered on in the different modes and indicates the power-up time.

Table 58 gives some typical values of the total current consumption (analog + digital supply currents) in the different modes, depending on the clock divider bits. The ADC is turned off. Note that these values also include current consumption of the regulator and other parts on the test board where these values are measured.

**Table 57. Operating Modes[1]**

| Mode | Core | Peripherals | PLL | XTAL/T2/T3 | IRQ0 to IRQ3 | Start-Up/Power-On Time |
|---|---|---|---|---|---|---|
| Active | X | X | X | X | X | 130 ms at CD = 0 |
| Pause | | X | X | X | X | 24 ns at CD = 0; 3 µs at CD = 7 |
| Nap | | | X | X | X | 24 ns at CD = 0; 3 µs at CD = 7 |
| Sleep | | | | X | X | 1.58 ms |
| Stop | | | | | X | 1.7 ms |

[1] X indicates that the part is powered on.

**Table 58. Typical Current Consumption at 25°C in Milliamperes**

| PC[2:0] | Mode | CD = 0 | CD = 1 | CD = 2 | CD = 3 | CD = 4 | CD = 5 | CD = 6 | CD = 7 |
|---|---|---|---|---|---|---|---|---|---|
| 000 | Active | 33.1 | 21.2 | 13.8 | 10 | 8.1 | 7.2 | 6.7 | 6.45 |
| 001 | Pause | 22.7 | 13.3 | 8.5 | 6.1 | 4.9 | 4.3 | 4 | 3.85 |
| 010 | Nap | 3.8 | 3.8 | 3.8 | 3.8 | 3.8 | 3.8 | 3.8 | 3.8 |
| 011 | Sleep | 0.4 | 0.4 | 0.4 | 0.4 | 0.4 | 0.4 | 0.4 | 0.4 |
| 100 | Stop | 0.4 | 0.4 | 0.4 | 0.4 | 0.4 | 0.4 | 0.4 | 0.4 |

The GDCLK value can range from 0 to 255, corresponding to a programmable chopping frequency rate of 40.8 kHz to 10.44 MHz for a 41.78 MHz core frequency. The gate drive features must be programmed before operation of the PWM controller and are typically not changed during normal operation of the PWM controller. Following a reset, all bits of the PWMCFG register are cleared so that high frequency chopping is disabled, by default.

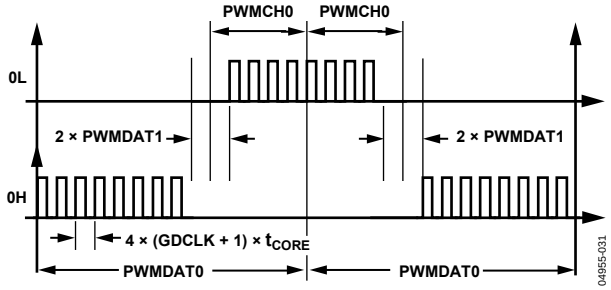



*Figure 72. Typical PWM Signals with High Frequency Gate Chopping Enabled on Both High-Side and Low-Side Switches*

### *PWM Shutdown*

In the event of external fault conditions, it is essential that the PWM system be instantaneously shut down in a safe fashion. A low level on the $PWM_{TRIP}$ pin provides an instantaneous, asynchronous (independent of the MicroConverter core clock) shutdown of the PWM controller. All six PWM outputs are placed in the off state, that is, in low state. In addition, the PWMSYNC pulse is disabled. The $PWM_{TRIP}$ pin has an internal pull-down resistor to disable the PWM if the pin becomes disconnected. The state of the $PWM_{TRIP}$ pin can be read from Bit 3 of the PWMSTA register.

If a PWM shutdown command occurs, a PWMTRIP interrupt is generated, and internal timing of the 3-phase timing unit of the PWM controller is stopped. Following a PWM shutdown, the PWM can be reenabled (in a PWMTRIP interrupt service routine, for example) only by writing to all of the PWMDAT0, PWMCH0, PWMCH1, and PWMCH2 registers. Provided that the external fault is cleared and the PWMTRIP is returned to a high level, the internal timing of the 3-phase timing unit resumes, and new duty-cycle values are latched on the next PWMSYNC boundary.

Note that the PWMTRIP interrupt is available in IRQ only, and the PWMSYNC interrupt is available in FIQ only. Both interrupts share the same bit in the interrupt controller. Therefore, only one of the interrupts can be used at a time. See the Interrupt System section for further details.

### *PWM MMRs Interface*

The PWM block is controlled via the MMRs described in this section.

**Table 66. PWMCON Register**

| Name | Address | Default Value | Access |
|---|---|---|---|
| PWMCON | 0xFFFFFC00 | 0x0000 | R/W |

PWMCON is a control register that enables the PWM and chooses the update rate.

**Table 67. PWMCON MMR Bit Descriptions**

| Bit | Name | Description |
|---|---|---|
| 7:5 | | Reserved. |
| 4 | PWM_SYNCSEL | External sync select. Set to use external sync. Cleared to use internal sync. |
| 3 | PWM_EXTSYNC | External sync select. Set to select external synchronous sync signal. Cleared for asynchronous sync signal. |
| 2 | PWMDBL | Double update mode. Set to 1 by user to enable double update mode. Cleared to 0 by the user to enable single update mode. |
| 1 | PWM_SYNC_EN | PWM synchronization enable. Set by user to enable synchronization. Cleared by user to disable synchronization. |
| 0 | PWMEN | PWM enable bit. Set to 1 by user to enable the PWM. Cleared to 0 by user to disable the PWM. Also cleared automatically with PWMTRIP (PWMSTA MMR). |

**Table 68. PWMSTA Register**

| Name | Address | Default Value | Access |
|---|---|---|---|
| PWMSTA | 0xFFFFFC04 | 0x0000 | R/W |

PWMSTA reflects the status of the PWM.

**Table 69. PWMSTA MMR Bit Descriptions**

| Bit | Name | Description |
|---|---|---|
| 15:10 | | Reserved. |
| 9 | PWMSYNCINT | PWM sync interrupt bit. Writing a 1 to this bit clears this interrupt. |
| 8 | PWMTRIPINT | PWM trip interrupt bit. Writing a 1 to this bit clears this interrupt. |
| 3 | PWMTRIP | Raw signal from the $PWM_{TRIP}$ pin. |
| 2:1 | | Reserved. |
| 0 | PWMPHASE | PWM phase bit. Set to 1 by the MicroConverter when the timer is counting down (first half). Cleared to 0 by the MicroConverter when the timer is counting up (second half). |

**Table 70. PWMCFG Register**

| Name | Address | Default Value | Access |
|---|---|---|---|
| PWMCFG | 0xFFFFFC10 | 0x0000 | R/W |

PWMCFG is a gate chopping register.

**Table 71. PWMCFG MMR Bit Descriptions**

| Bit | Name | Description |
|---|---|---|
| 15:10 | | Reserved. |
| 9 | CHOPLO | Low-side gate chopping enable bit. |
| 8 | CHOPHI | High-side gate chopping enable bit. |
| 7:0 | GDCLK | PWM gate chopping period (unsigned). |

**Table 72. PWMEN Register**

| Name | Address | Default Value | Access |
|---|---|---|---|
| PWMEN | 0xFFFFFC20 | 0x0000 | R/W |

PWMEN allows enabling of channel outputs and crossover. See its bit definitions in Table 73.

**Table 73. PWMEN MMR Bit Descriptions**

| Bit | Name | Description |
|---|---|---|
| 8 | 0H0L_XOVR | Channel 0 output crossover enable bit. Set to 1 by user to enable Channel 0 output crossover. Cleared to 0 by user to disable Channel 0 output crossover. |
| 7 | 1H1L_XOVR | Channel 1 output crossover enable bit. Set to 1 by user to enable Channel 1 output crossover. Cleared to 0 by user to disable Channel 1 output crossover. |
| 6 | 2H2L_XOVR | Channel 2 output crossover enable bit. Set to 1 by user to enable Channel 2 output crossover. Cleared to 0 by user to disable Channel 2 output crossover. |
| 5 | 0L_EN | 0L output enable bit. Set to 1 by user to disable the 0L output of the PWM. Cleared to 0 by user to enable the 0L output of the PWM. |
| 4 | 0H_EN | 0H output enable bit. Set to 1 by user to disable the 0H output of the PWM. Cleared to 0 by user to enable the 0H output of the PWM. |
| 3 | 1L_EN | 1L output enable bit. Set to 1 by user to disable the 1L output of the PWM. Cleared to 0 by user to enable the 1L output of the PWM. |
| 2 | 1H_EN | 1H Output Enable Bit. Set to 1 by user to disable the 1H output of the PWM. Cleared to 0 by user to enable the 1H output of the PWM. |
| 1 | 2L_EN | 2L output enable bit. Set to 1 by user to disable the 2L output of the PWM. Cleared to 0 by user to enable the 2L output of the PWM. |
| 0 | 2H_EN | 2H output enable bit. Set to 1 by user to disable the 2H output of the PWM. Cleared to 0 by user to enable the 2H output of the PWM. |

**Table 74. PWMDAT0 Register**

| Name | Address | Default Value | Access |
|---|---|---|---|
| PWMDAT0 | 0xFFFFFC08 | 0x0000 | R/W |

PWMDAT0 is an unsigned 16-bit register for switching period.

**Table 75. PWMDAT1 Register**

| Name | Address | Default Value | Access |
|---|---|---|---|
| PWMDAT1 | 0xFFFFFC0C | 0x0000 | R/W |

PWMDAT1 is an unsigned 10-bit register for dead time.

**Table 76. PWMCHx Registers**

| Name | Address | Default Value | Access |
|---|---|---|---|
| PWMCH0 | 0xFFFFFC14 | 0x0000 | R/W |
| PWMCH1 | 0xFFFFFC18 | 0x0000 | R/W |
| PWMCH2 | 0xFFFFFC1C | 0x0000 | R/W |

PWMCH0, PWMCH1, and PWMCH2 are channel duty cycles for the three phases.

**Table 77. PWMDAT2 Register**

| Name | Address | Default Value | Access |
|---|---|---|---|
| PWMDAT2 | 0xFFFFFC24 | 0x0000 | R/W |

PWMDAT2 is an unsigned 10-bit register for PWM sync pulse width.

## GENERAL-PURPOSE INPUT/OUTPUT

The ADuC7019/20/21/22/24/25/26/27/28/29 provide 40 general-purpose, bidirectional I/O (GPIO) pins. All I/O pins are 5 V tolerant, meaning the GPIOs support an input voltage of 5 V.

In general, many of the GPIO pins have multiple functions (see Table 78 for the pin function definitions). By default, the GPIO pins are configured in GPIO mode.

All GPIO pins have an internal pull-up resistor (of about 100 kΩ), and their drive capability is 1.6 mA. Note that a maximum of 20 GPIOs can drive 1.6 mA at the same time. Using the GPxPAR registers, it is possible to enable/disable the pull-up resistors for the following ports: P0.0, P0.4, P0.5, P0.6, P0.7, and the eight GPIOs of P1.

The 40 GPIOs are grouped in five ports, Port 0 to Port 4 (Port x). Each port is controlled by four or five MMRs.

Note that the kernel changes P0.6 from its default configuration at reset (MRST) to GPIO mode. If MRST is used for external circuitry, an external pull-up resistor should be used to ensure that the level on P0.6 does not drop when the kernel switches mode. Otherwise, P0.6 goes low for the reset period. For example, if MRST is required for power-down, it can be reconfigured in GP0CON MMR.

The input level of any GPIO can be read at any time in the GPxDAT MMR, even when the pin is configured in a mode other than GPIO. The PLA input is always active.

When the ADuC7019/20/21/22/24/25/26/27/28/29 part enters a power-saving mode, the GPIO pins retain their state.

**Table 78. GPIO Pin Function Descriptions**

| | | Configuration | | | |
|---|---|---|---|---|---|
| **Port** | **Pin** | **00** | **01** | **10** | **11** |
| 0 | P0.0 | GPIO | CMP | MS0 | PLAI[7] |
| | P0.1 | GPIO | $PWM2_H$ | $\overline{BLE}$ | |
| | P0.2 | GPIO | $PWM2_L$ | $\overline{BHE}$ | |
| | P0.3 | GPIO | TRST | A16 | $ADC_{BUSY}$ |
| | P0.4 | GPIO/IRQ0 | $PWM_{TRIP}$ | MS1 | PLAO[1] |
| | P0.5 | GPIO/IRQ1 | $ADC_{BUSY}$ | MS2 | PLAO[2] |
| | P0.6 | GPIO/T1 | MRST | | PLAO[3] |
| | P0.7 | GPIO | ECLK/XCLK[1] | SIN | PLAO[4] |
| 1 | P1.0 | GPIO/T1 | SIN | SCL0 | PLAI[0] |
| | P1.1 | GPIO | SOUT | SDA0 | PLAI[1] |
| | P1.2 | GPIO | RTS | SCL1 | PLAI[2] |
| | P1.3 | GPIO | CTS | SDA1 | PLAI[3] |
| | P1.4 | GPIO/IRQ2 | RI | SCLK | PLAI[4] |
| | P1.5 | GPIO/IRQ3 | DCD | MISO | PLAI[5] |
| | P1.6 | GPIO | DSR | MOSI | PLAI[6] |
| | P1.7 | GPIO | DTR | $\overline{CS}$ | PLAO[0] |
| 2 | P2.0 | GPIO | $\overline{CONV_{START}}$[2] | SOUT | PLAO[5] |
| | P2.1 | GPIO | $PWM0_H$ | $\overline{WS}$ | PLAO[6] |
| | P2.2 | GPIO | $PWM0_L$ | $\overline{RS}$ | PLAO[7] |
| | P2.3 | GPIO | | AE | |
| | P2.4 | GPIO | $PWM0_H$ | MS0 | |
| | P2.5 | GPIO | $PWM0_L$ | MS1 | |
| | P2.6 | GPIO | $PWM1_H$ | MS2 | |
| | P2.7 | GPIO | $PWM1_L$ | MS3 | |
| 3 | P3.0 | GPIO | $PWM0_H$ | AD0 | PLAI[8] |
| | P3.1 | GPIO | $PWM0_L$ | AD1 | PLAI[9] |
| | P3.2 | GPIO | $PWM1_H$ | AD2 | PLAI[10] |
| | P3.3 | GPIO | $PWM1_L$ | AD3 | PLAI[11] |
| | P3.4 | GPIO | $PWM2_H$ | AD4 | PLAI[12] |
| | P3.5 | GPIO | $PWM2_L$ | AD5 | PLAI[13] |
| | P3.6 | GPIO | $PWM_{TRIP}$ | AD6 | PLAI[14] |
| | P3.7 | GPIO | $PWM_{SYNC}$ | AD7 | PLAI[15] |
| 4 | P4.0 | GPIO | | AD8 | PLAO[8] |
| | P4.1 | GPIO | | AD9 | PLAO[9] |
| | P4.2 | GPIO | | AD10 | PLAO[10] |
| | P4.3 | GPIO | | AD11 | PLAO[11] |
| | P4.4 | GPIO | | AD12 | PLAO[12] |
| | P4.5 | GPIO | | AD13 | PLAO[13] |
| | P4.6 | GPIO | | AD14 | PLAO[14] |
| | P4.7 | GPIO | | AD15 | PLAO[15] |

[1] When configured in Mode 1, P0.7 is ECLK by default, or core clock output. To configure it as a clock input, the MDCLK bits in PLLCON must be set to 11.
[2] The $\overline{CONV_{START}}$ signal is active in all modes of P2.0.

**Table 79. GPxCON Registers**

| Name | Address | Default Value | Access |
|---|---|---|---|
| GP0CON | 0xFFFFF400 | 0x00000000 | R/W |
| GP1CON | 0xFFFFF404 | 0x00000000 | R/W |
| GP2CON | 0xFFFFF408 | 0x00000000 | R/W |
| GP3CON | 0xFFFFF40C | 0x00000000 | R/W |
| GP4CON | 0xFFFFF410 | 0x00000000 | R/W |

GPxCON are the Port x control registers, which select the function of each pin of Port x as described in Table 80.

**Table 80. GPxCON MMR Bit Descriptions**

| Bit | Description |
|---|---|
| 31:30 | Reserved. |
| 29:28 | Select function of the Px.7 pin. |
| 27:26 | Reserved. |
| 25:24 | Select function of the Px.6 pin. |
| 23:22 | Reserved. |
| 21:20 | Select function of the Px.5 pin. |
| 19:18 | Reserved. |
| 17:16 | Select function of the Px.4 pin. |
| 15:14 | Reserved. |
| 13:12 | Select function of the Px.3 pin. |
| 11:10 | Reserved. |
| 9:8 | Select function of the Px.2 pin. |
| 7:6 | Reserved. |
| 5:4 | Select function of the Px.1 pin. |
| 3:2 | Reserved. |
| 1:0 | Select function of the Px.0 pin. |

**Table 81. GPxPAR Registers**

| Name | Address | Default Value | Access |
|---|---|---|---|
| GP0PAR | 0xFFFFF42C | 0x20000000 | R/W |
| GP1PAR | 0xFFFFF43C | 0x00000000 | R/W |

GPxPAR program the parameters for Port 0 and Port 1. Note that the GPxDAT MMR must always be written after changing the GPxPAR MMR.

**Table 82. GPxPAR MMR Bit Descriptions**

| Bit | Description |
|---|---|
| 31 | Reserved. |
| 30:29 | Drive strength Px.7. |
| 28 | Pull-Up Disable Px.7. |
| 27 | Reserved. |
| 26:25 | Drive strength Px.6. |
| 24 | Pull-Up Disable Px.6. |
| 23 | Reserved. |
| 22:21 | Drive strength Px.5. |
| 20 | Pull-Up Disable Px.5. |
| 19 | Reserved. |
| 18:17 | Drive strength Px.4. |
| 16 | Pull-Up Disable Px.4. |
| 15 | Reserved. |
| 14:13 | Drive strength Px.3. |
| 12 | Pull-Up Disable Px.3. |
| 11 | Reserved. |
| 10:9 | Drive strength Px.2. |
| 8 | Pull-Up Disable Px.2. |
| 7 | Reserved. |
| 6:5 | Drive strength Px.1. |
| 4 | Pull-Up Disable Px.1. |
| 3 | Reserved. |
| 2:1 | Drive strength Px.0. |
| 0 | Pull-Up Disable Px.0. |

**Table 85. GPxDAT Registers**

| Name | Address | Default Value[1] | Access |
|---|---|---|---|
| GP0DAT | 0xFFFFF420 | 0x000000XX | R/W |
| GP1DAT | 0xFFFFF430 | 0x000000XX | R/W |
| GP2DAT | 0xFFFFF440 | 0x000000XX | R/W |
| GP3DAT | 0xFFFFF450 | 0x000000XX | R/W |
| GP4DAT | 0xFFFFF460 | 0x000000XX | R/W |

[1] X = 0, 1, 2, or 3.

GPxDAT are Port x configuration and data registers. They configure the direction of the GPIO pins of Port x, set the output value for the pins configured as output, and store the input value of the pins configured as input.

**Table 86. GPxDAT MMR Bit Descriptions**

| Bit | Description |
|---|---|
| 31:24 | Direction of the data. Set to 1 by user to configure the GPIO pin as an output. Cleared to 0 by user to configure the GPIO pin as an input. |
| 23:16 | Port x data output. |
| 15:8 | Reflect the state of Port x pins at reset (read only). |
| 7:0 | Port x data input (read only). |

**Table 87. GPxSET Registers**

| Name | Address | Default Value[1] | Access |
|---|---|---|---|
| GP0SET | 0xFFFFF424 | 0x000000XX | W |
| GP1SET | 0xFFFFF434 | 0x000000XX | W |
| GP2SET | 0xFFFFF444 | 0x000000XX | W |
| GP3SET | 0xFFFFF454 | 0x000000XX | W |
| GP4SET | 0xFFFFF464 | 0x000000XX | W |

[1] X = 0, 1, 2, or 3.

GPxSET are data set Port x registers.

**Table 88. GPxSET MMR Bit Descriptions**

| Bit | Description |
|---|---|
| 31:24 | Reserved. |
| 23:16 | Data Port x set bit. Set to 1 by user to set bit on Port x; also sets the corresponding bit in the GPxDAT MMR. Cleared to 0 by user; does not affect the data out. |
| 15:0 | Reserved. |

**Table 89. GPxCLR Registers**

| Name | Address | Default Value[1] | Access |
|---|---|---|---|
| GP0CLR | 0xFFFFF428 | 0x000000XX | W |
| GP1CLR | 0xFFFFF438 | 0x000000XX | W |
| GP2CLR | 0xFFFFF448 | 0x000000XX | W |
| GP3CLR | 0xFFFFF458 | 0x000000XX | W |
| GP4CLR | 0xFFFFF468 | 0x000000XX | W |

[1] X = 0, 1, 2, or 3.

GPxCLR are data clear Port x registers.

**Table 90. GPxCLR MMR Bit Descriptions**

| Bit | Description |
|---|---|
| 31:24 | Reserved. |
| 23:16 | Data Port x clear bit. Set to 1 by user to clear bit on Port x; also clears the corresponding bit in the GPxDAT MMR. Cleared to 0 by user; does not affect the data out. |
| 15:0 | Reserved. |

## SERIAL PORT MUX

The serial port mux multiplexes the serial port peripherals (an SPI, UART, and two $I^2C$s) and the programmable logic array (PLA) to a set of 10 GPIO pins. Each pin must be configured to one of its specific I/O functions as described in Table 91.

**Table 91. SPM Configuration**

| SPMMUX | GPIO (00) | UART (01) | UART/$I^2C$/SPI (10) | PLA (11) |
|---|---|---|---|---|
| SPM0 | P1.0 | SIN | I2C0SCL | PLAI[0] |
| SPM1 | P1.1 | SOUT | I2C0SDA | PLAI[1] |
| SPM2 | P1.2 | RTS | I2C1SCL | PLAI[2] |
| SPM3 | P1.3 | CTS | I2C1SDA | PLAI[3] |
| SPM4 | P1.4 | RI | SCLK | PLAI[4] |
| SPM5 | P1.5 | DCD | MISO | PLAI[5] |
| SPM6 | P1.6 | DSR | MOSI | PLAI[6] |
| SPM7 | P1.7 | DTR | $\overline{CS}$ | PLAO[0] |
| SPM8 | P0.7 | ECLK/XCLK | SIN | PLAO[4] |
| SPM9 | P2.0 | CONV | SOUT | PLAO[5] |

Table 91 also details the mode for each of the SPMMUX pins. This configuration must be done via the GP0CON, GP1CON, and GP2CON MMRs. By default, these 10 pins are configured as GPIOs.

## UART SERIAL INTERFACE

The UART peripheral is a full-duplex, universal, asynchronous receiver/transmitter. It is fully compatible with the 16,450 serial port standard. The UART performs serial-to-parallel conversions on data characters received from a peripheral device or modem, and parallel-to-serial conversions on data characters received from the CPU. The UART includes a fractional divider for baud rate generation and has a network addressable mode. The UART function is made available on the 10 pins of the ADuC7019/20/21/22/24/25/26/27/28/29 (see Table 92).

**Table 92. UART Signal Description**

| Pin | Signal | Description |
|---|---|---|
| SPM0 (Mode 1) | SIN | Serial receive data. |
| SPM1 (Mode 1) | SOUT | Serial transmit data. |
| SPM2 (Mode 1) | RTS | Request to send. |
| SPM3 (Mode 1) | CTS | Clear to send. |
| SPM4 (Mode 1) | RI | Ring indicator. |
| SPM5 (Mode 1) | DCD | Data carrier detect. |
| SPM6 (Mode 1) | DSR | Data set ready. |
| SPM7 (Mode 1) | DTR | Data terminal ready. |
| SPM8 (Mode 2) | SIN | Serial receive data. |
| SPM9 (Mode 2) | SOUT | Serial transmit data. |

**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 automatically 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.

### *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 master-based 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 116. COMIID1 MMR Bit Descriptions**

| Bit 3:1 Status Bits | Bit 0 NINT | Priority | Definition | Clearing Operation |
|---|---|---|---|---|
| 000 | 1 | | No interrupt | |
| 110 | 0 | 2 | Matching network address | Read COMRX |
| 101 | 0 | 3 | Address transmitted, buffer empty | Write data to COMTX or read COMIID0 |
| 011 | 0 | 1 | Receive line status interrupt | Read COMSTA0 |
| 010 | 0 | 2 | Receive buffer full interrupt | Read COMRX |
| 001 | 0 | 3 | Transmit buffer empty interrupt | Write data to COMTX or read COMIID0 |
| 000 | 0 | 4 | Modem status interrupt | Read COMSTA1 |

Note that to receive a network address interrupt, the slave must ensure that Bit 0 of COMIEN0 (enable receive buffer full interrupt) is set to 1.

**Table 117. COMADR Register**

| Name | Address | Default Value | Access |
|---|---|---|---|
| COMADR | 0xFFFF0728 | 0xAA | R/W |

COMADR is an 8-bit, read/write network address register that holds the address checked for by the network addressable UART. Upon receiving this address, the device interrupts the processor and/or sets the appropriate status bit in COMIID1.

## SERIAL PERIPHERAL INTERFACE

The ADuC7019/20/21/22/24/25/26/27/28/29 integrate a complete hardware serial peripheral interface (SPI) on-chip. SPI is an industry standard, synchronous serial interface that allows eight bits of data to be synchronously transmitted and simultaneously received, that is, full duplex up to a maximum bit rate of 3.48 Mb, as shown in Table 118. The SPI interface is not operational with core clock divider (CD) bits. POWCON[2:0] = 6 or 7 in master mode.

The SPI port can be configured for master or slave operation. and typically consists of four pins: MISO (P1.5), MOSI (P1.6), SCLK (P1.4), and $\overline{CS}$ (P1.7).

On the transmit side, the SPITX register (and a TX shift register outside it) loads data onto the transmit pin (in slave mode, MISO; in master mode, MOSI). The transmit status bit, Bit 0, in SPISTA indicates whether there is valid data in the SPITX register.

Similarly, the receive data path consists of the SPIRX register (and an RX shift register). SPISTA, Bit 3 indicates whether there is valid data in the SPIRX register. If valid data in the SPIRX register is overwritten or if valid data in the RX shift register is discarded, SPISTA, Bit 5 (the overflow bit) is set.

### *MISO (Master In, Slave Out) Pin*

The MISO pin is configured as an input line in master mode and an output line in slave mode. The MISO line on the master (data in) should be connected to the MISO line in the slave device (data out). The data is transferred as byte wide (8-bit) serial data, MSB first.

### *MOSI (Master Out, Slave In) Pin*

The MOSI pin is configured as an output line in master mode and an input line in slave mode. The MOSI line on the master (data out) should be connected to the MOSI line in the slave device (data in). The data is transferred as byte wide (8-bit) serial data, MSB first.

### *SCLK (Serial Clock I/O) Pin*

The master serial clock (SCLK) is used to synchronize the data being transmitted and received through the MOSI SCLK period. Therefore, a byte is transmitted/received after eight SCLK periods. The SCLK pin is configured as an output in master mode and as an input in slave mode.

In master mode, the polarity and phase of the clock are controlled by the SPICON register, and the bit rate is defined in the SPIDIV register as follows:

$$f_{SERIAL\,CLOCK} = \frac{f_{UCLK}}{2\times(1+SPIDIV)}$$

The maximum speed of the SPI clock is dependent on the clock divider bits and is summarized in Table 118.

**Table 118. SPI Speed vs. Clock Divider Bits in Master Mode**

| CD Bits | 0 | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|---|
| SPIDIV in Hex | 0x05 | 0x0B | 0x17 | 0x2F | 0x5F | 0xBF |
| SPI dpeed in MHz | 3.482 | 1.741 | 0.870 | 0.435 | 0.218 | 0.109 |

In slave mode, the SPICON register must be configured with the phase and polarity of the expected input clock. The slave accepts data from an external master up to 10.4 Mb at CD = 0. The formula to determine the maximum speed is as follows:

$$f_{SERIAL\,CLOCK} = \frac{f_{HCLK}}{4}$$

In both master and slave modes, data is transmitted on one edge of the SCL signal and sampled on the other. Therefore, it is important that the polarity and phase be configured the same for the master and slave devices.

### *Chip Select ($\overline{CS}$ Input) Pin*

In SPI slave mode, a transfer is initiated by the assertion of $\overline{CS}$, which is an active low input signal. The SPI port then transmits and receives 8-bit data until the transfer is concluded by deassertion of $\overline{CS}$. In slave mode, $\overline{CS}$ is always an input.

## I²C-COMPATIBLE INTERFACES

The ADuC7019/20/21/22/24/25/26/27/28/29 support two licensed I²C interfaces. The I²C interfaces are both implemented as a hard-ware master and a full slave interface. Because the two I²C inter-faces are identical, this data sheet describes only I2C0 in detail. Note that the two masters and one of the slaves have individual interrupts (see the Interrupt System section).

Note that when configured as an I²C master device, the ADuC7019/20/21/22/24/25/26/27/28/29 cannot generate a repeated start condition.

The two GPIO pins used for data transfer, SDAx and SCLx, are configured in a wired-AND format that allows arbitration in a multimaster system. These pins require external pull-up resistors. Typical pull-up values are 10 kΩ.

The I²C bus peripheral address in the I²C bus system is programmed by the user. This ID can be modified any time a transfer is not in progress. The user can configure the interface to respond to four slave addresses.

The transfer sequence of an I²C system consists of a master device initiating a transfer by generating a start condition while the bus is idle. The master transmits the slave device address and the direction of the data transfer during the initial address transfer. If the master does not lose arbitration and the slave acknowledges, the data transfer is initiated. This continues until the master issues a stop condition and the bus becomes idle.

The I²C peripheral can be configured only as a master or slave at any given time. The same I²C channel cannot simultaneously support master and slave modes.

### *Serial Clock Generation*

The I²C master in the system generates the serial clock for a transfer. The master channel can be configured to operate in fast mode (400 kHz) or standard mode (100 kHz).

The bit rate is defined in the I2C0DIV MMR as follows:

$$f_{SERIAL\ CLOCK} = \frac{f_{UCLK}}{(2 + DIVH) + (2 + DIVL)}$$

where:
$f_{UCLK}$ = clock before the clock divider.
$DIVH$ = the high period of the clock.
$DIVL$ = the low period of the clock.

Thus, for 100 kHz operation,

$DIVH = DIVL$ = 0xCF

and for 400 kHz,

$DIVH$ = 0x28, $DIVL$ = 0x3C

The I2CxDIV registers correspond to DIVH:DIVL.

### *Slave Addresses*

The registers I2C0ID0, I2C0ID1, I2C0ID2, and I2C0ID3 contain the device IDs. The device compares the four I2C0IDx registers to the address byte. To be correctly addressed, the seven MSBs of either ID register must be identical to that of the seven MSBs of the first received address byte. The LSB of the ID registers (the transfer direction bit) is ignored in the process of address recognition.

### *I²C Registers*

The I²C peripheral interface consists of 18 MMRs, which are discussed in this section.

**Table 126. I2CxMSTA Registers**

| Name | Address | Default Value | Access |
|---|---|---|---|
| I2C0MSTA | 0xFFFF0800 | 0x00 | R/W |
| I2C1MSTA | 0xFFFF0900 | 0x00 | R/W |

I2CxMSTA are status registers for the master channel.

**Table 127. I2C0MSTA MMR Bit Descriptions**

| Bit | Access Type | Description |
|---|---|---|
| 7 | R/W | Master transmit FIFO flush. Set by user to flush the master Tx FIFO. Cleared automatically after the master Tx FIFO is flushed. This bit also flushes the slave receive FIFO. |
| 6 | R | Master busy. Set automatically if the master is busy. Cleared automatically. |
| 5 | R | Arbitration loss. Set in multimaster mode if another master has the bus. Cleared when the bus becomes available. |
| 4 | R | No ACK. Set automatically if there is no acknowledge of the address by the slave device. Cleared automatically by reading the I2C0MSTA register. |
| 3 | R | Master receive IRQ. Set after receiving data. Cleared automatically by reading the I2C0MRX register. |
| 2 | R | Master transmit IRQ. Set at the end of a transmission. Cleared automatically by writing to the I2C0MTX register. |
| 1 | R | Master transmit FIFO underflow. Set automatically if the master transmit FIFO is underflowing. Cleared automatically by writing to the I2C0MTX register. |
| 0 | R | Master TX FIFO not full. Set automatically if the slave transmit FIFO is not full. Cleared automatically by writing twice to the I2C0STX register. |

**Table 128. I2CxSSTA Registers**

| Name | Address | Default Value | Access |
|---|---|---|---|
| I2C0SSTA | 0xFFFF0804 | 0x01 | R |
| I2C1SSTA | 0xFFFF0904 | 0x01 | R |

I2CxSSTA are status registers for the slave channel.

**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**

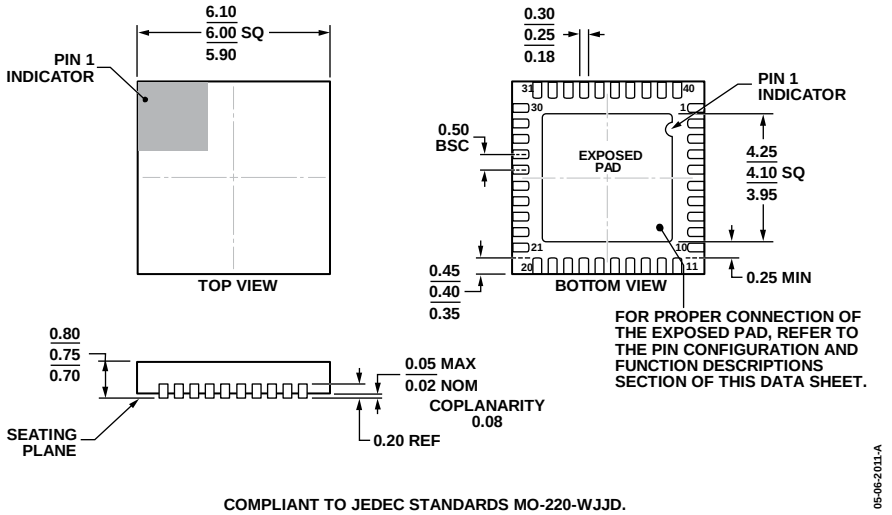
| Name | Address | Default Value | Access |
|---|---|---|---|
| I2C0FSTA | 0xFFFF084C | 0x0000 | R/W |
| I2C1FSTA | 0xFFFF094C | 0x0000 | R/W |

I2CxFSTA are FIFO status registers.

**Table 144. I2C0FSTA MMR Bit Descriptions**

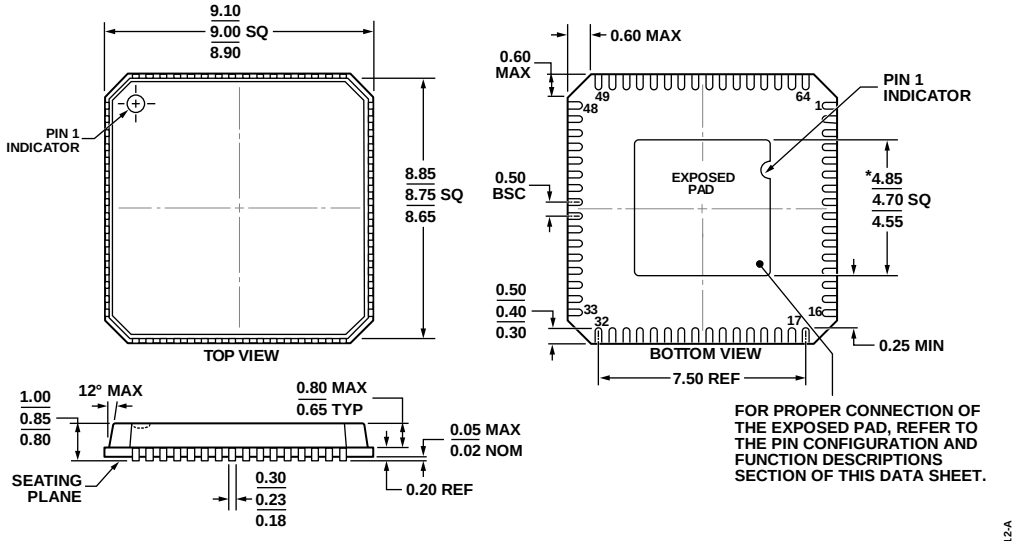
| Bit | Access Type | Value | Description |
|---|---|---|---|
| 15:10 | | | Reserved. |
| 9 | R/W | | Master transmit FIFO flush. Set by the user to flush the master Tx FIFO. Cleared automatically when the master Tx FIFO is flushed. This bit also flushes the slave receive FIFO. |
| 8 | R/W | | Slave 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. |

# OUTLINE DIMENSIONS




COMPLIANT TO JEDEC STANDARDS MO-220-WJJD.

*Figure 96. 40-Lead Lead Frame Chip Scale Package [LFCSP_WQ]*
*6 × 6 mm Body, Very Very Thin Quad*
*(CP-40-9)*
*Dimensions shown in millimeters*




*COMPLIANT TO JEDEC STANDARDS MO-220-VMMD-4
EXCEPT FOR EXPOSED PAD DIMENSION

*Figure 97. 64-Lead Lead Frame Chip Scale Package [LFCSP_VQ]*
*9 mm × 9 mm Body, Very Thin Quad*
*(CP-64-1)*
*Dimensions shown in millimeters*

## ORDERING GUIDE

| Model[1, 2] | ADC Channels[3] | DAC Channels | FLASH/ RAM | GPIO | Down-loader | Temperature Range | Package Description | Package Option | Ordering Quantity |
|---|---|---|---|---|---|---|---|---|---|
| ADuC7019BCPZ62I | 5 | 3 | 62 kB/8 kB | 14 | I²C | −40°C to +125°C | 40-Lead LFCSP_WQ | CP-40-9 | |
| ADuC7019BCPZ62I-RL | 5 | 3 | 62 kB/8 kB | 14 | I²C | −40°C to +125°C | 40-Lead LFCSP_WQ | CP-40-9 | 2,500 |
| ADuC7019BCPZ62IRL7 | 5 | 3 | 62 kB/8 kB | 14 | I²C | −40°C to +125°C | 40-Lead LFCSP_WQ | CP-40-9 | 750 |
| ADuC7020BCPZ62 | 5 | 4 | 62 kB/8 kB | 14 | UART | −40°C to +125°C | 40-Lead LFCSP_WQ | CP-40-9 | |
| ADuC7020BCPZ62-RL7 | 5 | 4 | 62 kB/8 kB | 14 | UART | −40°C to +125°C | 40-Lead LFCSP_WQ | CP-40-9 | 750 |
| ADuC7020BCPZ62I | 5 | 4 | 62 kB/8 kB | 14 | I²C | −40°C to +125°C | 40-Lead LFCSP_WQ | CP-40-9 | |
| ADuC7020BCPZ62I-RL | 5 | 4 | 62 kB/8 kB | 14 | I²C | −40°C to +125°C | 40-Lead LFCSP_WQ | CP-40-9 | 2,500 |
| ADuC7020BCPZ62IRL7 | 5 | 4 | 62 kB/8 kB | 14 | I²C | −40°C to +125°C | 40-Lead LFCSP_WQ | CP-40-9 | 750 |
| ADuC7021BCPZ62 | 8 | 2 | 62 kB/8 kB | 13 | UART | −40°C to +125°C | 40-Lead LFCSP_WQ | CP-40-9 | |
| ADuC7021BCPZ62-RL | 8 | 2 | 62 kB/8 kB | 13 | UART | −40°C to +125°C | 40-Lead LFCSP_WQ | CP-40-9 | 2,500 |
| ADuC7021BCPZ62-RL7 | 8 | 2 | 62 kB/8 kB | 13 | UART | −40°C to +125°C | 40-Lead LFCSP_WQ | CP-40-9 | 750 |
| ADuC7021BCPZ62I | 8 | 2 | 62 kB/8 kB | 13 | I²C | −40°C to +125°C | 40-Lead LFCSP_WQ | CP-40-9 | |
| ADuC7021BCPZ62I-RL | 8 | 2 | 62 kB/8 kB | 13 | I²C | −40°C to +125°C | 40-Lead LFCSP_WQ | CP-40-9 | 2,500 |
| ADuC7021BCPZ32 | 8 | 2 | 32 kB/4 kB | 13 | UART | −40°C to +125°C | 40-Lead LFCSP_WQ | CP-40-9 | |
| ADuC7021BCPZ32-RL7 | 8 | 2 | 32 kB/4 kB | 13 | UART | −40°C to +125°C | 40-Lead LFCSP_WQ | CP-40-9 | 750 |
| ADuC7022BCPZ62 | 10 | | 62 kB/8 kB | 13 | UART | −40°C to +125°C | 40-Lead LFCSP_WQ | CP-40-9 | |
| ADuC7022BCPZ62-RL7 | 10 | | 62 kB/8 kB | 13 | UART | −40°C to +125°C | 40-Lead LFCSP_WQ | CP-40-9 | 750 |
| ADuC7022BCPZ32 | 10 | | 32 kB/4 kB | 13 | UART | −40°C to +125°C | 40-Lead LFCSP_WQ | CP-40-9 | |
| ADuC7022BCPZ32-RL | 10 | | 32 kB/4 kB | 13 | UART | −40°C to +125°C | 40-Lead LFCSP_WQ | CP-40-9 | 2,500 |
| ADuC7024BCPZ62 | 10 | 2 | 62 kB/8 kB | 30 | UART | −40°C to +125°C | 64-Lead LFCSP_VQ | CP-64-1 | |
| ADuC7024BCPZ62-RL7 | 10 | 2 | 62 kB/8 kB | 30 | UART | −40°C to +125°C | 64-Lead LFCSP_VQ | CP-64-1 | 750 |
| ADuC7024BCPZ62I | 10 | 2 | 62 kB/8 kB | 30 | I²C | −40°C to +125°C | 64-Lead LFCSP_VQ | CP-64-1 | |
| ADuC7024BCPZ62I-RL | 10 | 2 | 62 kB/8 kB | 30 | I²C | −40°C to +125°C | 64-Lead LFCSP_VQ | CP-64-1 | 2,500 |
| ADuC7024BSTZ62 | 10 | 2 | 62 kB/8 kB | 30 | UART | −40°C to +125°C | 64-Lead LQFP | ST-64-2 | |
| ADuC7024BSTZ62-RL | 10 | 2 | 62 kB/8 kB | 30 | UART | −40°C to +125°C | 64-Lead LQFP | ST-64-2 | 1,000 |
| ADuC7025BCPZ62 | 12 | | 62 kB/8 kB | 30 | UART | −40°C to +125°C | 64-Lead LFCSP_VQ | CP-64-1 | |
| ADuC7025BCPZ62-RL | 12 | | 62 kB/8 kB | 30 | UART | −40°C to +125°C | 64-Lead LFCSP_VQ | CP-64-1 | 2,500 |
| ADuC7025BCPZ32 | 12 | | 32 kB/4 kB | 30 | UART | −40°C to +125°C | 64-Lead LFCSP_VQ | CP-64-1 | |
| ADuC7025BCPZ32-RL | 12 | | 32 kB/4 kB | 30 | UART | −40°C to +125°C | 64-Lead LFCSP_VQ | CP-64-1 | 2,500 |
| ADuC7025BSTZ62 | 12 | | 62 kB/8 kB | 30 | UART | −40°C to +125°C | 64-Lead LQFP | ST-64-2 | |
| ADuC7025BSTZ62-RL | 12 | | 62 kB/8 kB | 30 | UART | −40°C to +125°C | 64-Lead LQFP | ST-64-2 | 1,000 |
| ADuC7026BSTZ62 | 12 | 4 | 62 kB/8 kB | 40 | UART | −40°C to +125°C | 80-Lead LQFP | ST-80-1 | |
| ADuC7026BSTZ62-RL | 12 | 4 | 62 kB/8 kB | 40 | UART | −40°C to +125°C | 80-Lead LQFP | ST-80-1 | 1,000 |
| ADuC7026BSTZ62I | 12 | 4 | 62 kB/8 kB | 40 | I²C | −40°C to +125°C | 80-Lead LQFP | ST-80-1 | |
| ADuC7026BSTZ62I-RL | 12 | 4 | 62 kB/8 kB | 40 | I²C | −40°C to +125°C | 80-Lead LQFP | ST-80-1 | 1,000 |
| ADuC7027BSTZ62 | 16 | | 62 kB/8 kB | 40 | UART | −40°C to +125°C | 80-Lead LQFP | ST-80-1 | |
| ADuC7027BSTZ62-RL | 16 | | 62 kB/8 kB | 40 | UART | −40°C to +125°C | 80-Lead LQFP | ST-80-1 | 1,000 |
| ADuC7027BSTZ62I | 16 | | 62 kB/8 kB | 40 | I²C | −40°C to +125°C | 80-Lead LQFP | ST-80-1 | |
| ADuC7027BSTZ62I-RL | 16 | | 62 kB/8 kB | 40 | I²C | −40°C to +125°C | 80-Lead LQFP | ST-80-1 | 1,000 |
| ADuC7028BBCZ62 | 8 | 4 | 62 kB/8 kB | 30 | UART | −40°C to +125°C | 64-Ball CSP_BGA | BC-64-4 | |
| ADuC7028BBCZ62-RL | 8 | 4 | 62 kB/8 kB | 30 | UART | −40°C to +125°C | 64-Ball CSP_BGA | BC-64-4 | 2,500 |
| ADuC7029BBCZ62 | 7 | 4 | 62 kB/8 kB | 22 | UART | −40°C to +125°C | 49-Ball CSP_BGA | BC-49-1 | |
| ADuC7029BBCZ62-RL | 7 | 4 | 62 kB/8 kB | 22 | UART | −40°C to +125°C | 49-Ball CSP_BGA | BC-49-1 | 4,000 |
| ADuC7029BBCZ62I | 7 | 4 | 62 kB/8 kB | 22 | I²C | −40°C to +125°C | 49-Ball CSP_BGA | BC-49-1 | |
| ADuC7029BBCZ62I-RL | 7 | 4 | 62 kB/8 kB | 22 | I²C | −40°C to +125°C | 49-Ball CSP_BGA | BC-49-1 | 4,000 |