# MCP3461/2/4

## Two/Four/Eight Channel, 153.6 kSPS, Low-Noise, 16-bit Delta Sigma ADC

### Features

- One/Two/Four Differential or Two/Four/Eight Single-Ended Input Channels
- 16-bit Resolution
- Programmable Data Rate: up to 153.6 kSPS
- Programmable Gain: 0.33X to 64X
- 97.2 dB SINAD, -116 dBc THD, 120 dBc SFDR (Gain = 1X, 4800 SPS)
- Low Temperature Drift:
  - Offset error drift: 4/Gain nV/°C (AZ_MUX = 1)
  - Gain error drift: 0.5 ppm/°C (Gain = 1X)
- Low-Noise: 3.2 µVRMS (Gain = 16x, 9600 sps)
- RMS ENOB: 15.5 bits minimum (All gains, all OSR combinations)
- Wide Input Voltage Range: 0V to $AV_{DD}$
- Differential Voltage Reference Inputs
- Internal Oscillator or External Clock Selection
- Ultra-Low Shutdown Current Consumption (< 2 µA)
- Internal Temperature Sensor
- Burnout Current Sources for Sensor Open/Short Detection
- 16-Bit Digital Offset and Gain Error Calibration Registers
- Internal Conversions Sequencer (SCAN Mode) for Automatic Multiplexing
- Dedicated $\overline{IRQ}$ Pin for Easy Synchronization
- Advanced Security Features:
  - 16-bit CRC for secure SPI communications
  - 16-bit CRC and IRQ for securing configuration
  - Register map lock with 8-bit secure key
  - Monitor controls for system diagnostics
- 20 MHz SPI-Compatible Interface with Mode 0,0 and 1,1
- $AV_{DD}$: 2.7V-3.6V
- $DV_{DD}$: 1.8V-3.6V
- Extended Temperature Range: -40°C to +125°C
- Package: 3 mm x 3 mm UQFN-20

### General Description

The MCP3461/2/4 are 1/2/4-channel, 16-bit, Delta-Sigma Analog-to-Digital Converters (ADCs), with programmable data rate of up to 153.6 kSPS. They offer integrated features, such as internal oscillator, temperature sensor and burnout sensor detection, in order to reduce system component count and total solution cost.

The MCP3461/2/4 ADCs are fully configurable with Oversampling Ratio (OSR) from 32 to 98304 and gain from 1/3X to 64X. These devices include an internal sequencer (SCAN mode) with multiple monitor channels and a 24-bit timer to be able to automatically create conversion loop sequences without needing MCU communications. Advanced security features such as CRC and register map lock, can ensure configuration locking and integrity as well as communication data integrity for secure environments.

These devices come with a 20 MHz SPI-compatible serial interface. Communication is largely simplified with 8-bit commands, including various continuous Read/Write modes and 16/32-bit multiple data formats that can be accessed by the Direct Memory Access (DMA) of an 8-bit, 16-bit or 32-bit MCU.

The MCP3461/2/4 are available in an ultra-small, 3 mm x 3 mm UQFN-20 package and are specified over an extended temperature range from -40°C to +125°C.

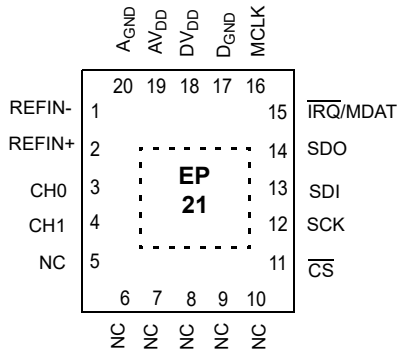
### Applications

- Precision Sensor Transducers and Transmitters: Pressure, Strain, Flow and Force Measurement
- Factory Automation and Process Controls
- Portable Instrumentation
- Temperature Measurements

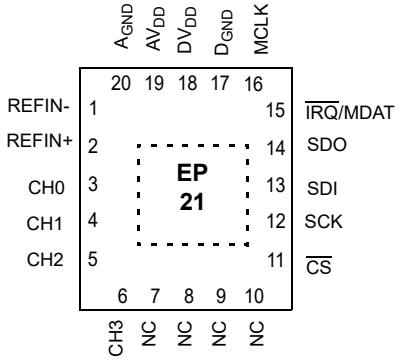
## Package Type

**Package Type for All Devices: 20-Lead UQFN (3 mm x 3 mm)***

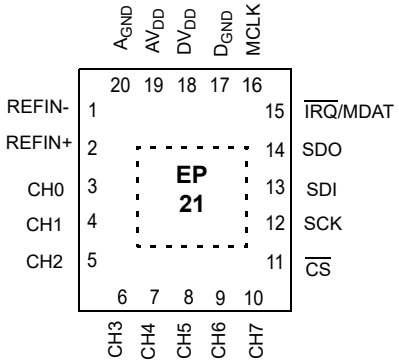
### A. MCP3461: Single Channel Device




### B. MCP3462: Dual Channel Device

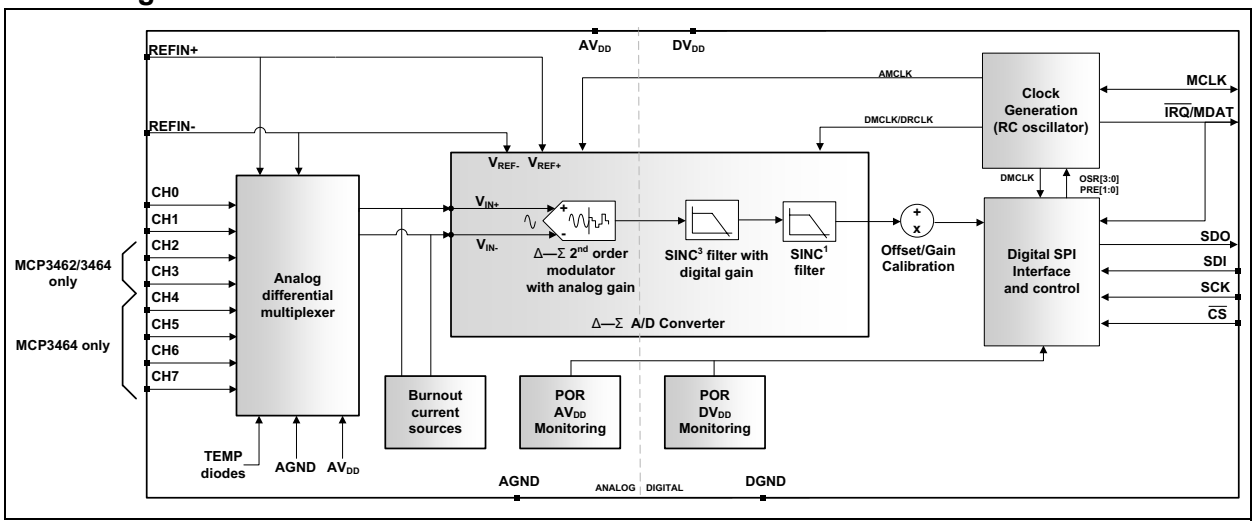



### C. MCP3464: Quad Channel Device




*Includes Exposed Thermal Pad (EP); see Table 3-1.

## Block Diagram of MCP3461/2/4 Devices



# 1.0 ELECTRICAL CHARACTERISTICS

## 1.1 Electrical Specifications

### Absolute Maximum Ratings(†)

$DV_{DD}$, $AV_{DD}$ ..... -0.3 to 4.0V
Digital inputs and outputs w.r.t. $D_{GND}$ ..... -0.3V to $DV_{DD}$ + 0.3V
Analog inputs w.r.t. $A_{GND}$ ..... -0.3V to $AV_{DD}$ + 0.3V
Current at input pins ..... ±5 mA
Current at output and supply pins ..... ±20 mA
Storage temperature ..... -65°C to +150°C
Ambient temp. with power applied ..... -65°C to +125°C
Soldering temperature of leads (10 seconds) ..... +300°C
Maximum Junction Temperature ($T_J$) ..... +150°C
ESD on the analog inputs (HBM) ..... ≥ 6.0 kV
ESD on all other pins (HBM) ..... ≥ 6.0 kV

**† Notice**: Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions, above those indicated in the operational listings of this specification, is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

## ELECTRICAL CHARACTERISTICS

**Electrical Specifications:** Unless otherwise indicated, all parameters apply at $AV_{DD}$ = 2.7V to 3.6V, $DV_{DD}$ = 1.8V to $AV_{DD}$ + 0.1V, MCLK = 4.9152 MHz, $V_{REF}$ = $AV_{DD}$, ADC_MODE[1:0] = 11. All other register map bits to their default conditions. $T_A$ = -40°C to +125°C, $V_{IN}$ = -0.5 dBFS at 50 Hz.

| Parameter | Symbol | Min. | Typ. | Max. | Units | Test Conditions |
|---|---|---|---|---|---|---|
| **Supply Requirements** | | | | | | |
| Operating Voltage, Analog | $AV_{DD}$ | 2.7 | — | 3.6 | V | |
| Operating Voltage, Digital | $DV_{DD}$ | 1.8 | — | $AV_{DD}$ + 0.1 | V | $DV_{DD} \leq 3.6V$ |
| Operating Current, Analog | $AI_{DD}$ | — | 0.56 | 0.81 | mA | BOOST[1:0] = 00, 0.5x |
| | | — | 0.69 | 0.96 | mA | BOOST[1:0] = 01, 0.66x |
| | | — | 0.93 | 1.3 | µA | BOOST[1:0] = 10, 1x |
| | | — | 1.65 | 2.2 | µA | BOOST[1:0] = 11, 2x |
| Operating Current, Digital | $DI_{DD}$ | — | 0.25 | 0.37 | mA | (**Note 8**) |
| Full-Shutdown Current Analog Full-Shutdown Current | $AI_{DDS}$ | — | — | 0.5 | µA | Full-Shutdown mode (CONFIG0 = 0x00) MCLK input in Idle mode. |
| Digital Full-Shutdown Current | $DI_{DDS}$ | — | — | 1.1 | µA | |
| Power-on-Reset Threshold Voltage | $V_{POR\_A}$ | — | 1.75 | — | V | For analog circuits. |
| | $V_{POR\_D}$ | — | 1.2 | — | V | For digital circuits. |
| POR Hysteresis | $V_{POR\_HYS}$ | — | 150 | — | mV | |
| POR Reset Time | $t_{POR}$ | — | 1 | — | µs | |
| **Analog Inputs** | | | | | | |
| Input Voltage at Input Pin | $CH_N$ | $A_{GND}$ - 0.1 | — | $AV_{DD}$ + 0.1 | V | Analog inputs are measured with respect to $A_{GND}$. |
| Differential Input Range | $V_{IN}$ | $-V_{REF}$/GAIN | — | $+V_{REF}$/GAIN | V | |
| Differential Input Impedance (**Note 5**) | $Z_{IN}$ | — | 510 | — | kΩ | GAIN = 0.33x,Proportional to 1/AMCLK |
| | | — | 260 | — | kΩ | GAIN = 1x, Proportional to 1/AMCLK |
| | | — | 150 | — | kΩ | GAIN = 2x, Proportional to 1/AMCLK |
| | | — | 80 | — | kΩ | GAIN = 4x, Proportional to 1/AMCLK |
| | | — | 40 | — | kΩ | GAIN = 8x, Proportional to 1/AMCLK |
| | | — | 20 | — | kΩ | GAIN ≥ 16x, Proportional to 1/AMCLK |
| Analog Input Leakage Current during shut down | $I_{LI\_A}$ | — | ±10 | — | nA | |

**Note 1:** This parameter is ensured by design and not 100% tested.

**2:** This parameter is ensured by characterization and not 100% tested.

**3:** REFIN- should be connected to ground for single-ended measurements.

**4:** Full Scale Range (FSR) = 2 * $V_{REF}$/GAIN.

**5:** This input impedance is due to the internal input sampling capacitor and frequency. This impedance is measured between the two input pins of the selected with the input multiplexer.

**6:** Applies to all analog gains. Offset and gain errors depend on gain settings. See typical performance plots.

**7:** INL is the difference between the endpoints line and the measured code at the center of the quantization band.

**8:** $D_{IDD}$ is measured while no transfer is present on the SPI bus.

## ELECTRICAL CHARACTERISTICS (CONTINUED)

**Electrical Specifications:** Unless otherwise indicated, all parameters apply at $AV_{DD}$ = 2.7V to 3.6V, $DV_{DD}$ = 1.8V to $AV_{DD}$ + 0.1V, MCLK = 4.9152 MHz, $V_{REF}$ = $AV_{DD}$, ADC_MODE[1:0] = 11. All other register map bits to their default conditions. $T_A$ = -40°C to +125°C, $V_{IN}$ = -0.5 dBFS at 50 Hz.

| Parameter | Symbol | Min. | Typ. | Max. | Units | Test Conditions |
|---|---|---|---|---|---|---|
| **External Voltage Reference Input** | | | | | | |
| Reference Voltage Range ($V_{REF}$+ - $V_{REF}$-) | $V_{REF}$ | 0.6 | — | $AV_{DD}$ | V | |
| External NonInverting Input Voltage Reference | $V_{REF+}$ | $V_{REF-}$ + 0.6 | — | $AV_{DD}$ | V | |
| External Inverting Input Voltage Reference | $V_{REF-}$ | $A_{GND}$ | — | $V_{REF+}$ - 0.6 | V | |
| **DC Performance** | | | | | | |
| No Missing Code Resolution | Resolution | 16 | — | — | Bits | (**Note 1**) |
| Offset Error | $V_{OS}$ | -900/GAIN | — | 900/GAIN | µV | AZ MUX = 0 (**Note 6**) |
| | | -(0.05 + 0.8 /GAIN) | — | (0.05 + 0.8/ GAIN) | | AZ MUX = 1 (**Note 2**) |
| Offset Error Temperature Coefficient | $V_{OS\_DRIFT}$ | — | 70/GAIN | 300/GAIN | nV/°C | AZ MUX = 0 (**Note 2**)(**Note 6**) |
| | | — | 4/GAIN | 16/GAIN | | AZ MUX = 1 (**Note 2**)(**Note 6**) |
| Gain Error | $G_E$ | -3 | — | +3 | % | (**Note 6**) |
| Gain Error Temperature Coefficient | $G_{E\_DRIFT}$ | — | 0.5 | 2 | ppm/°C | GAIN: 1X, 2X, 4X (**Note 2**) |
| | | | 1 | 4 | | GAIN: 8X (**Note 2**) |
| | | | 2 | 8 | | GAIN: 0.33x, 16x (**Note 2**) |
| Integral NonLinearity (**Note 7**) | INL | -10 | — | +10 | ppm FSR | GAIN = 0.33 (**Note 2**) |
| | | -7 | — | +7 | | GAIN = 1 (**Note 2**) |
| | | -7 | — | +7 | | GAIN = 2 (**Note 2**) |
| | | 10 | — | +10 | | GAIN = 4 (**Note 2**) |
| | | -20 | — | +20 | | GAIN = 8 (**Note 2**) |
| | | -32 | — | +32 | | GAIN = 16 (**Note 2**) |
| $AV_{DD}$ Power Supply Rejection Ratio | DC PSRR | — | -76 - 20 * LOG (GAIN) | — | dB | $AV_{DD}$ varies from 2.7V to 3.6V, $V_{IN}$ = 0V. |
| $DV_{DD}$ Power Supply Rejection Ratio | | — | -110 | — | dB | $DV_{DD}$ varies from 1.8V to 3.6V, $V_{IN}$ = 0V. |
| DC Common-mode Rejection | DC CMRR | — | -126 | — | dB | $V_{INCOM}$ varies from 0V to $AV_{DD}$. $V_{IN}$ = 0V. |

**Note 1:** This parameter is ensured by design and not 100% tested.

**2:** This parameter is ensured by characterization and not 100% tested.

**3:** REFIN- should be connected to ground for single-ended measurements.

**4:** Full Scale Range (FSR) = 2 * $V_{REF}$/GAIN.

**5:** This input impedance is due to the internal input sampling capacitor and frequency. This impedance is measured between the two input pins of the selected with the input multiplexer.

**6:** Applies to all analog gains. Offset and gain errors depend on gain settings. See typical performance plots.

**7:** INL is the difference between the endpoints line and the measured code at the center of the quantization band.

**8:** $D_{IDD}$ is measured while no transfer is present on the SPI bus.

## ELECTRICAL CHARACTERISTICS (CONTINUED)

**Electrical Specifications:** Unless otherwise indicated, all parameters apply at $AV_{DD}$ = 2.7V to 3.6V, $DV_{DD}$ = 1.8V to $AV_{DD}$ + 0.1V, MCLK = 4.9152 MHz, $V_{REF}$ = $AV_{DD}$, ADC_MODE[1:0] = 11. All other register map bits to their default conditions. $T_A$ = -40°C to +125°C, $V_{IN}$ = -0.5 dBFS at 50 Hz.

| Parameter | Symbol | Min. | Typ. | Max. | Units | Test Conditions |
|---|---|---|---|---|---|---|
| **AC Performance** | | | | | | |
| Signal-to-Noise and Distortion Ratio | SINAD | 96.9 | 97.2 | — | dB | $AV_{DD}$ = $DV_{DD}$ = $V_{REF}$ = 3.3V and TA = +25°C (**Note 2**) |
| Signal-to-Noise Ratio | SNR | 97 | 97.3 | — | dBc | $AV_{DD}$ = $DV_{DD}$ = $V_{REF}$ = 3.3V and TA = +25°C. (**Note 2**) |
| Total Harmonic Distortion | THD | — | -116 | -110 | dB | $AV_{DD}$ = $DV_{DD}$ = $V_{REF}$ = 3.3V and TA = +25°C. Includes the first 10 harmonics. (**Note 2**) |
| Spurious Free Dynamic Range | SFDR | 110 | 120 | — | dBc | $AV_{DD}$ = $DV_{DD}$ = $V_{REF}$ = 3.3V and TA = +25°C (**Note 2**) |
| Input Channel Crosstalk | CTALK | — | -130 | — | dB | $V_{IN}$ = 0V Perturbation = 0 dB at 50 Hz Applies for all perturbation channels and all input channels. |
| AC Power Supply Rejection Ratio | AC PSRR | — | -75 - 20 * LOG (Gain) | — | dB | $V_{IN}$ = 0V, $DV_{DD}$ = 3.3V; $AV_{DD}$ = 3.3V + 0.3 $V_P$ at 50 Hz |
| AC Common-mode Rejection Ratio | AC CMRR | — | -122 | — | dB | $V_{INCOM}$ = 0 dB at 50 Hz $V_{IN}$ = 0V |
| **ADC Timing Parameters** | | | | | | |
| Sampling Frequency | DMCLK | See Table 5-6 | | | MHz | See Figure 4-1. |
| Output Data Rate | DRCLK | See Table 5-6 | | | ksps | See Figure 4-1. |
| Data Conversion Time | $T_{CONV}$ | See Table 5-6 | | | ms | See Figure 4-1. |
| ADC Start-Up Delay | $T_{ADC\_SETUP}$ | — | 256 | — | DMCLK periods | ADC_MODE[1:0] = change from 0X to 1X |
| | | — | 0 | — | DMCLK periods | ADC_MODE[1:0] = change from 10 to 11 |
| Conversion Start Pulse Low Time | $T_{STP}$ | — | 1 | — | DMCLK periods | |
| Scan Mode Time Delays | $T_{DLY\_SCAN}$ | 0 | — | 512 | DMCLK periods | Time delay between sampling channels. |
| | $T_{TIMER\_SCAN}$ | 0 | — | 16777215 | DMCLK periods | Time interval between scan cycles. |
| Data Ready Pulse Low Time | $T_{DRL}$ | | — | OSR-16 | DMCLK periods | See Figure 5-15 |
| Data ready Pulse High Time | $T_{DRH}$ | 16 | — | — | DMCLK periods | See Figure 5-15 |
| Data Transfer Time to $\overline{DR}$ (Data Ready) | $t_{DODR}$ | — | — | 50 | ns | |
| Modulator Output Valid from AMCLK high | $t_{DOMDAT}$ | — | — | 100 | ns | 2.7V ≤ $DV_{DD}$ ≤ 3.6V |
| | | | | 200 | ns | 1.8V ≤ $DV_{DD}$ ≤ 2.7V |

**Note 1:** This parameter is ensured by design and not 100% tested.

**2:** This parameter is ensured by characterization and not 100% tested.

**3:** REFIN- should be connected to ground for single-ended measurements.

**4:** Full Scale Range (FSR) = 2 * $V_{REF}$/GAIN.

**5:** This input impedance is due to the internal input sampling capacitor and frequency. This impedance is measured between the two input pins of the selected with the input multiplexer.

**6:** Applies to all analog gains. Offset and gain errors depend on gain settings. See typical performance plots.

**7:** INL is the difference between the endpoints line and the measured code at the center of the quantization band.

**8:** $D_{IDD}$ is measured while no transfer is present on the SPI bus.

## ELECTRICAL CHARACTERISTICS (CONTINUED)

**Electrical Specifications:** Unless otherwise indicated, all parameters apply at $AV_{DD}$ = 2.7V to 3.6V, $DV_{DD}$ = 1.8V to $AV_{DD}$ + 0.1V, MCLK = 4.9152 MHz, $V_{REF}$ = $AV_{DD}$, ADC_MODE[1:0] = 11. All other register map bits to their default conditions. $T_A$ = -40°C to +125°C, $V_{IN}$ = -0.5 dBFS at 50 Hz.

| Parameter | Symbol | Min. | Typ. | Max. | Units | Test Conditions |
|---|---|---|---|---|---|---|
| **External Master Clock Input (CLK_SEL[1] = 0)** | | | | | | |
| Master Clock, Input Frequency Range | $f_{MCLK\_EXT}$ | 1 | — | 20 | MHz | $DV_{DD} \geq 2.7V$ |
| | | 1 | — | 10 | MHz | $DV_{DD} < 2.7V$ |
| Master Clock Input Duty Cycle | $f_{MCLK\_DUTY}$ | 45 | — | 55 | % | |
| **Internal Clock Oscillator** | | | | | | |
| Internal Master Clock Frequency | $f_{MCLK\_INT}$ | 3.3 | — | 6.6 | MHz | CLK_SEL[1] = 1 |
| Internal Oscillator Start-Up Time | $t_{OSC\_STARTUP}$ | — | 10 | — | µs | CLK_SEL[1] changes from 0 to 1. Time to stabilize the clock frequency to ±1 kHz of the final value |
| Internal Oscillator Current Consumption | $IDD_{OSC}$ | — | 30 | — | µA | Should be added to $DI_{DD}$ when CLK_SEL[1:0] = 1X |
| **Internal Temperature Sensor** | | | | | | |
| Temperature Measurement Accuracy | $T_{ACC}$ | — | ±5 | — | °C | See **Section 5.1.2 "Internal Temperature Sensor"** for accuracy calculation |

**Note 1:** This parameter is ensured by design and not 100% tested.

**2:** This parameter is ensured by characterization and not 100% tested.

**3:** REFIN- should be connected to ground for single-ended measurements.

**4:** Full Scale Range (FSR) = 2 * $V_{REF}$/GAIN.

**5:** This input impedance is due to the internal input sampling capacitor and frequency. This impedance is measured between the two input pins of the selected with the input multiplexer.

**6:** Applies to all analog gains. Offset and gain errors depend on gain settings. See typical performance plots.

**7:** INL is the difference between the endpoints line and the measured code at the center of the quantization band.

**8:** $D_{IDD}$ is measured while no transfer is present on the SPI bus.

## TEMPERATURE CHARACTERISTICS

| **Electrical Specifications:** Unless otherwise specified, all parameters apply for $T_A$ = -40°C to +125°C, $AV_{DD}$ = 2.7V TO 3.6V, $DV_{DD}$ = 1.8V to $AV_{DD}$ + 0.1V, $D_{GND}$ = $A_{GND}$ = 0V. | | | | | | |
|---|---|---|---|---|---|---|
| **Parameters** | **Symbol** | **Min.** | **Typ.** | **Max.** | **Units** | **Conditions** |
| **Temperature Ranges** | | | | | | |
| Specified Temperature Range | $T_A$ | -40 | — | +125 | °C | |
| Operating Temperature Range | $T_A$ | -40 | — | +125 | °C | |
| Storage Temperature Range | $T_A$ | -65 | — | +150 | °C | |
| **Thermal Package Resistance** | | | | | | |
| Thermal Resistance, UQFN-20 | $\theta_{JA}$ | — | 50 | — | °C/W | |

**Note 1:** The internal junction temperature ($T_j$) must not exceed the absolute maximum specification of +150°C

**TABLE 1-1: SPI SERIAL INTERFACE TIMING SPECIFICATIONS FOR $DV_{DD}$ = 2.7V TO 3.6V**

| **Electrical Specifications:** $DV_{DD}$ = 2.7V to 3.6V, $T_A$ = -40°C to +125°C, $C_{LOAD}$ = 30 pF. See Figure 1-1. | | | | | | |
|---|---|---|---|---|---|---|
| **Parameters** | **Symbol** | **Min.** | **Typ.** | **Max.** | **Units** | **Conditions** |
| Serial Clock Frequency | $f_{SCK}$ | — | — | 20 | MHz | |
| $\overline{CS}$ Setup Time | $t_{CSS}$ | 25 | — | — | ns | |
| $\overline{CS}$ Hold Time | $t_{CSH}$ | 50 | — | — | ns | |
| $\overline{CS}$ Disable Time | $t_{CSD}$ | 50 | — | — | ns | |
| Data Setup Time | $t_{SU}$ | 5 | — | — | ns | |
| Data Hold Time | $t_{HD}$ | 10 | — | — | ns | |
| Serial Clock High Time | $t_{HI}$ | 20 | — | — | ns | |
| Serial Clock Low Time | $t_{LO}$ | 20 | — | — | ns | |
| Serial Clock Delay Time | $t_{CLD}$ | 50 | — | — | ns | |
| Serial Clock Enable Time | $t_{CLE}$ | 50 | — | — | ns | |
| Output Valid from SCK Low | $t_{DO}$ | — | — | 25 | ns | |
| Output Hold Time | $t_{HO}$ | 0 | — | — | ns | |
| Output Disable Time | $t_{DIS}$ | — | — | 25 | ns | Measured with 1.5 mA pull-up current source on SDO pin |
| POR IRQ Disable Time | $t_{CSIRQ}$ | — | — | 52 | ns | Measured with 1.5 mA pull-up current source on $\overline{IRQ}$ pin |
| Output Valid from $\overline{CS}$ Low | $t_{CSSDO}$ | — | — | 25 | ns | SDO toggles to logic low at each communication start ($\overline{CS}$ falling edge) |

**TABLE 1-2: SPI SERIAL INTERFACE TIMING SPECIFICATIONS FOR $DV_{DD}$ = 1.8V TO 2.7V (10 MHZ MAXIMUM SCK FREQUENCY)**

| **Electrical Specifications:** $DV_{DD}$ = 1.8V to 2.7V, $T_A$ = -40°C to +125°C, $C_{LOAD}$ = 30 pF. See Figure 1-1. | | | | | | |
|---|---|---|---|---|---|---|
| **Parameters** | **Symbol** | **Min.** | **Typ.** | **Max.** | **Units** | **Conditions** |
| Serial Clock Frequency | $f_{SCK}$ | — | — | 10 | MHz | |
| $\overline{CS}$ Setup Time | $t_{CSS}$ | 50 | — | — | ns | |
| $\overline{CS}$ Hold Time | $t_{CSH}$ | 100 | — | — | ns | |
| $\overline{CS}$ Disable Time | $t_{CSD}$ | 100 | — | — | ns | |
| Data Setup Time | $t_{SU}$ | 10 | — | — | ns | |
| Data Hold Time | $t_{HD}$ | 20 | — | — | ns | |
| Serial Clock High Time | $t_{HI}$ | 40 | — | — | ns | |
| Serial Clock Low Time | $t_{LO}$ | 40 | — | — | ns | |
| Serial Clock Delay Time | $t_{CLD}$ | 100 | — | — | ns | |
| Serial Clock Enable Time | $t_{CLE}$ | 100 | — | — | ns | |
| Output Valid From SCK Low | $t_{DO}$ | — | — | 50 | ns | |
| Output Hold Time | $t_{HO}$ | 0 | — | — | ns | |
| Output Disable Time | $t_{DIS}$ | — | — | 50 | ns | Measured with 1.5 mA pull-up current source on SDO pin |
| POR IRQ Disable Time | $t_{CSIRQ}$ | — | — | 60 | ns | Measured with 1.5 mA pull-up current source on IRQ pin |
| Output Valid From $\overline{CS}$ Low | $t_{CSSDO}$ | — | — | 50 | ns | SDO toggles to logic low at each communication start ($\overline{CS}$ falling edge) |

**TABLE 1-3: DIGITAL I/O DC SPECIFICATIONS**

| **Electrical Specifications:** Unless otherwise indicated, all parameters apply at, $DV_{DD}$ = 1.8V to 3.6V, $T_A$ = -40°C to +125°C. | | | | | | |
|---|---|---|---|---|---|---|
| **Parameters** | **Symbol** | **Min.** | **Typ.** | **Max.** | **Units** | **Conditions** |
| Schmitt Trigger High-Level Input Voltage | $V_{IH}$ | 0.7 * $DV_{DD}$ | — | — | V | |
| Schmitt Trigger Low-Level Input Voltage | $V_{IL}$ | — | — | 0.3 * $DV_{DD}$ | V | |
| Hysteresis Of Schmitt Trigger Inputs | $V_{HYS}$ | — | 200 | — | mV | |
| Low-Level Output Voltage | $V_{OL}$ | — | — | 0.2 * $DV_{DD}$ | V | $I_{OL}$ = +1.5 mA |
| High-Level Output Voltage | $V_{OH}$ | 0.8 * $DV_{DD}$ | — | | V | $I_{OH}$ = -1.5 mA |
| Input Leakage Current | $I_{LI\_D}$ | — | — | 1 | µA | Pins configured as inputs or high-impedance outputs. |

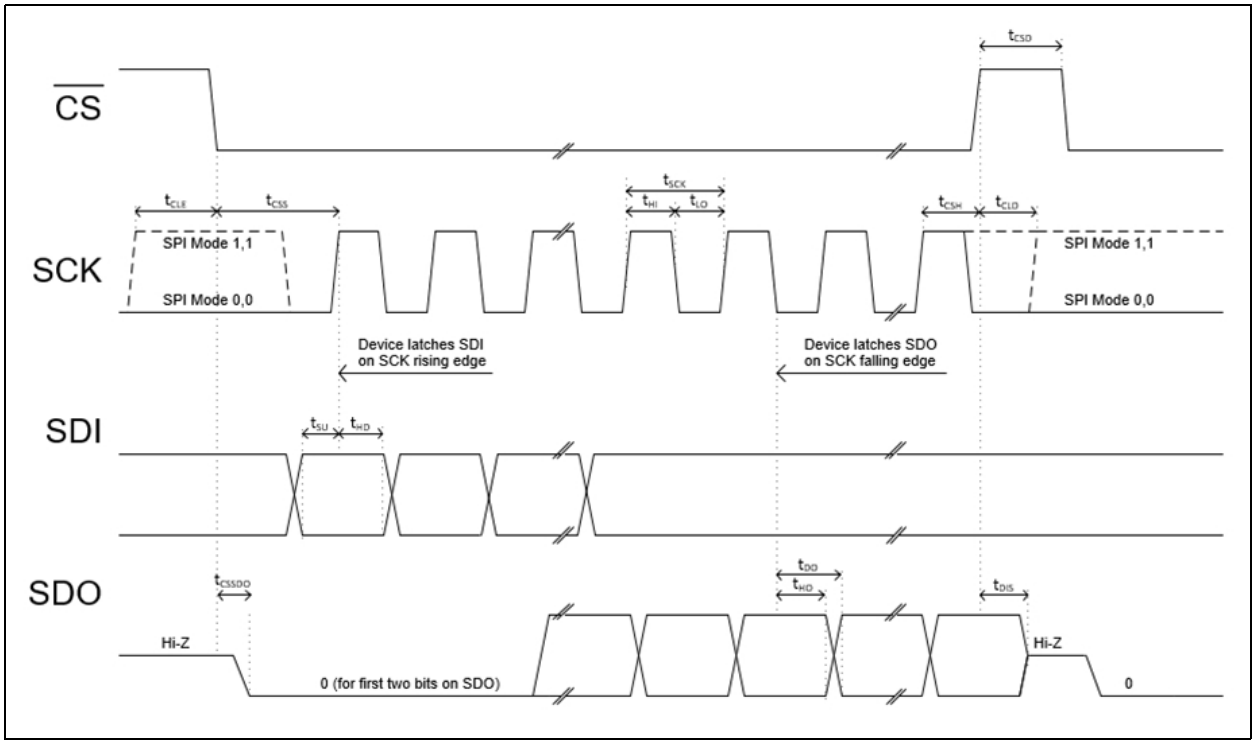



***FIGURE 1-1:*** *Serial Output Timing Diagram.*

**NOTES:**

## 2.0 TYPICAL PERFORMANCE CURVES

**Note:** The graphs and tables provided following this note are a statistical summary based on a limited number of samples and are provided for informational purposes only. The performance characteristics listed herein are not tested or guaranteed. In some graphs or tables, the data presented may be outside the specified operating range (e.g., outside specified power supply range) and therefore outside the warranted range.

**Note:** Unless otherwise indicated, $AV_{DD}$ = 3.3V, $DV_{DD}$ = 3.3V; $T_A$ = +25°C, MCLK = 4.9152 MHz; $V_{IN}$ = -0.5 dBFS at 50 Hz, $V_{REF}$ = $AV_{DD}$; ADC_MODE = 11. All other registers are set to default value. Histogram ticks are centered at their bin center.

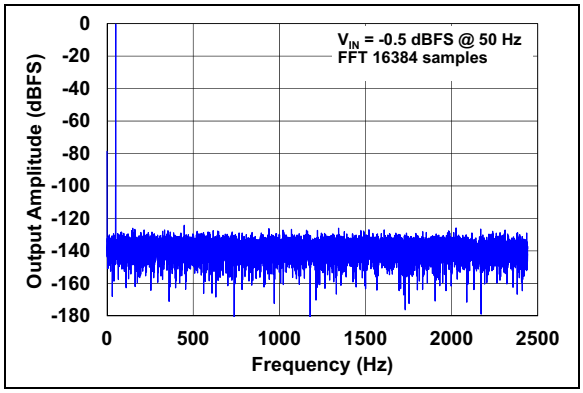



***FIGURE 2-1:*** *Output Spectrum (50 Hz Input).*

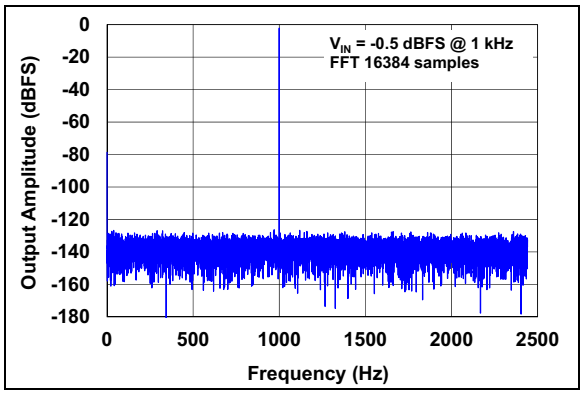



***FIGURE 2-2:*** *Output Spectrum (1 kHz Input).*

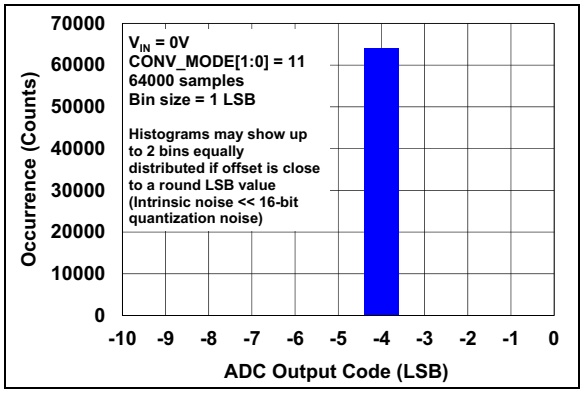



***FIGURE 2-3:*** *Output Noise Histogram.*

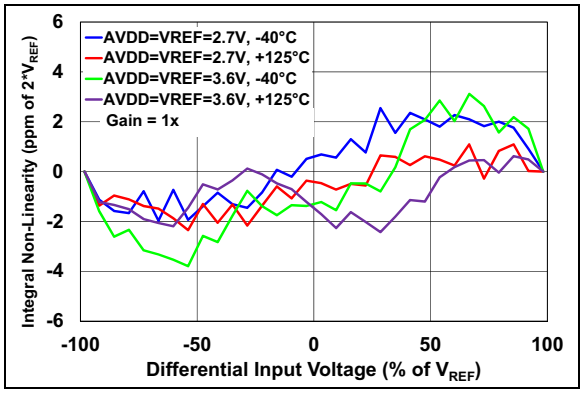



***FIGURE 2-4:*** *INL vs. Input Voltage.*

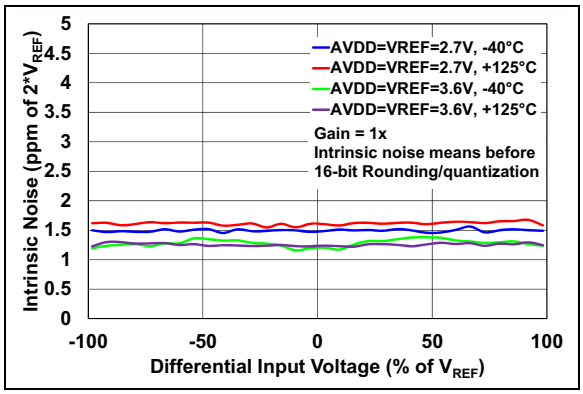



***FIGURE 2-5:*** *Output Noise vs. Input Voltage.*

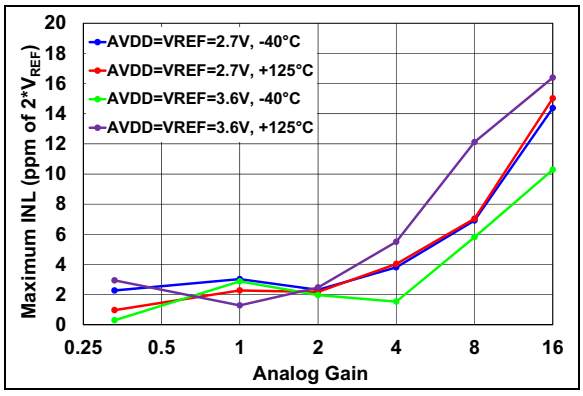



***FIGURE 2-6:*** *Maximum INL vs. Gain.*

**Note:** Unless otherwise indicated, $AV_{DD}$ = 3.3V, $DV_{DD}$ = 3.3V; $T_A$ = +25°C, MCLK = 4.9152 MHz; $V_{IN}$ = -0.5 dBFS at 50 Hz, $V_{REF}$ = $AV_{DD}$; ADC_MODE = 11. All other registers are set to default value. Histogram ticks are centered at their bin center.

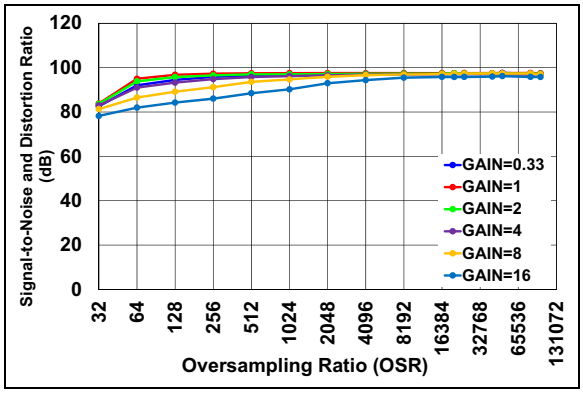



***FIGURE 2-7:*** *SINAD vs. OSR.*

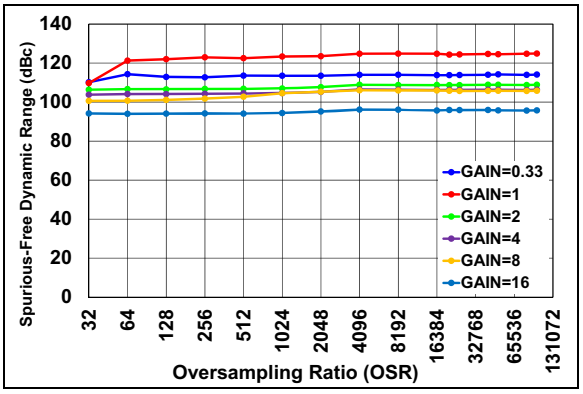



***FIGURE 2-10:*** *SFDR vs. OSR.*

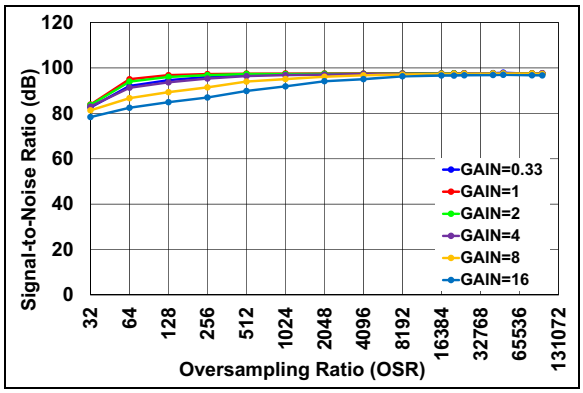



***FIGURE 2-8:*** *SNR vs. OSR.*

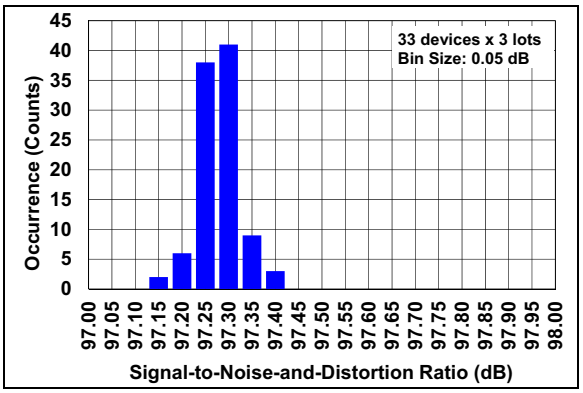



***FIGURE 2-11:*** *SINAD Distribution Histogram.*

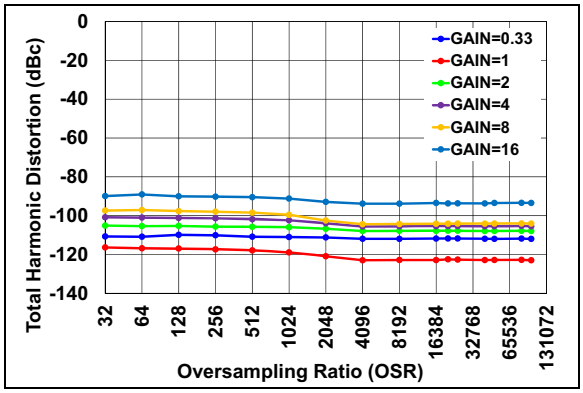



***FIGURE 2-9:*** *THD vs. OSR.*

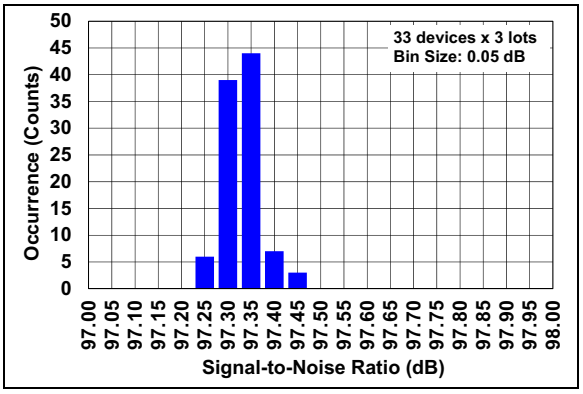



***FIGURE 2-12:*** *SNR Distribution Histogram.*

**Note:** Unless otherwise indicated, $AV_{DD}$ = 3.3V, $DV_{DD}$ = 3.3V; $T_A$ = +25°C, MCLK = 4.9152 MHz; $V_{IN}$ = -0.5 dBFS at 50 Hz, $V_{REF}$ = $AV_{DD}$; ADC_MODE = 11. All other registers are set to default value. Histogram ticks are centered at their bin center.

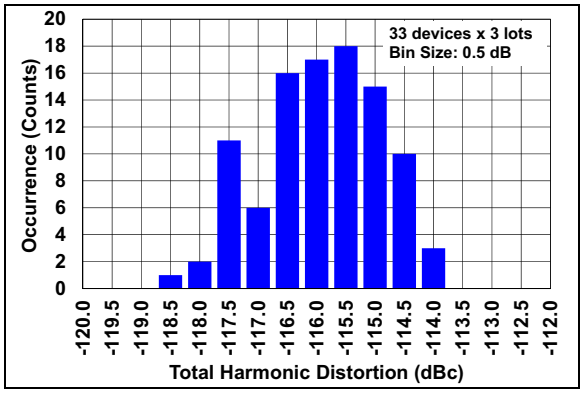



***FIGURE 2-13:*** *THD Distribution Histogram.*

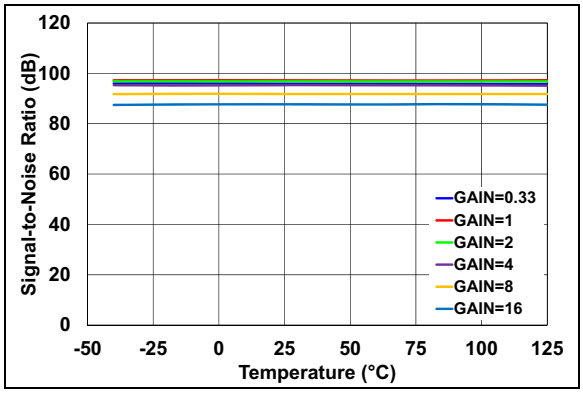



***FIGURE 2-16:*** *SNR vs. Temperature.*

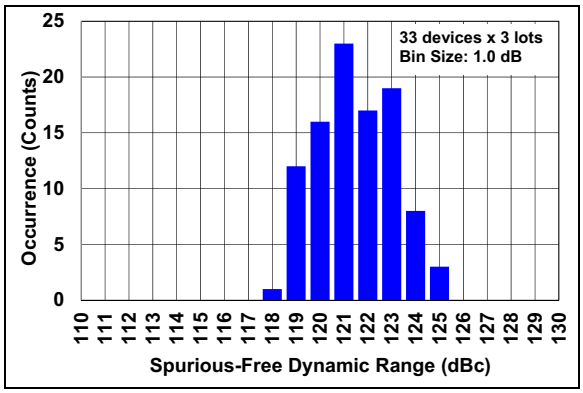



***FIGURE 2-14:*** *SFDR Distribution Histogram.*

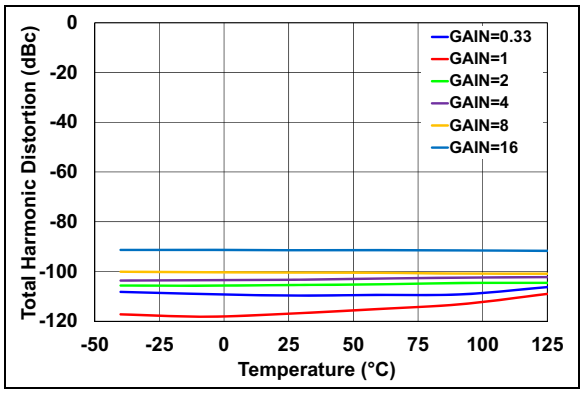



***FIGURE 2-17:*** *THD vs. Temperature.*

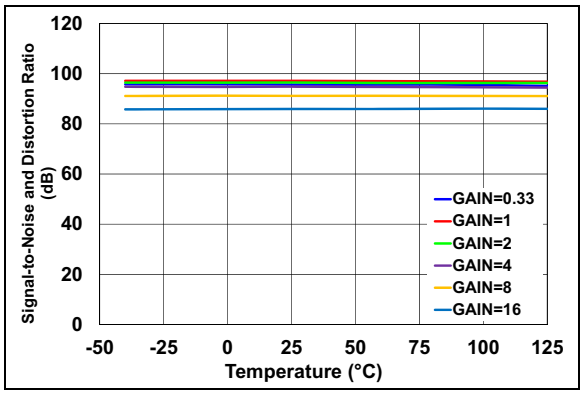



***FIGURE 2-15:*** *SINAD vs. Temperature.*

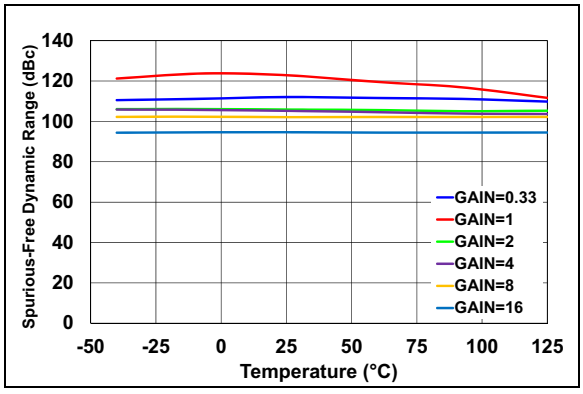



***FIGURE 2-18:*** *SFDR vs Temperature.*

**Note:** Unless otherwise indicated, $AV_{DD}$ = 3.3V, $DV_{DD}$ = 3.3V; $T_A$ = +25°C, MCLK = 4.9152 MHz; $V_{IN}$ = -0.5 dBFS at 50 Hz, $V_{REF}$ = $AV_{DD}$; ADC_MODE = 11. All other registers are set to default value. Histogram ticks are centered at their bin center.

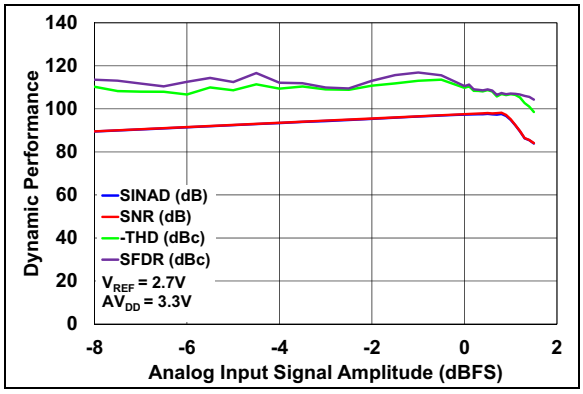



***FIGURE 2-19:*** *SINAD vs. Input Signal Amplitude.*

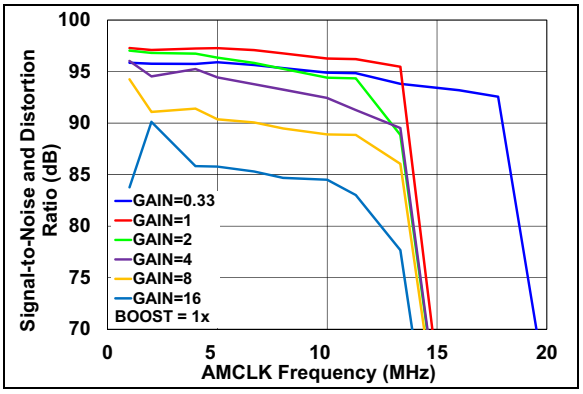



***FIGURE 2-22:*** *SINAD vs. AMCLK (Boost = 1x).*

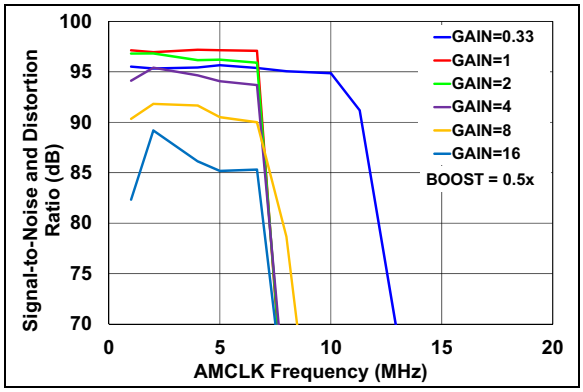



***FIGURE 2-20:*** *SINAD vs. AMCLK (Boost = 0.5x).*

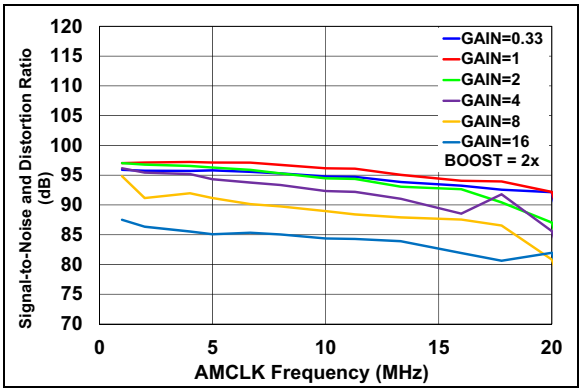



***FIGURE 2-23:*** *SINAD vs. AMCLK (Boost = 2x).*

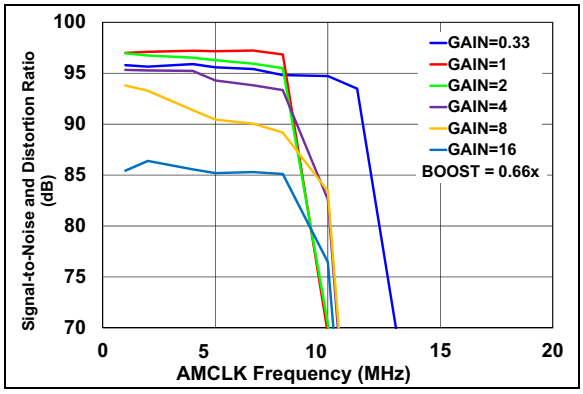



***FIGURE 2-21:*** *SINAD vs. AMCLK (Boost = 0.66x).*

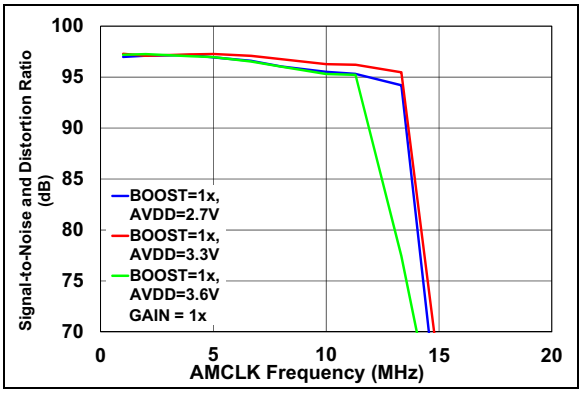



***FIGURE 2-24:*** *SINAD vs. AMCLK vs $AV_{DD}$.*

**Note:** Unless otherwise indicated, $AV_{DD}$ = 3.3V, $DV_{DD}$ = 3.3V; $T_A$ = +25°C, MCLK = 4.9152 MHz; $V_{IN}$ = -0.5 dBFS at 50 Hz, $V_{REF}$ = $AV_{DD}$; ADC_MODE = 11. All other registers are set to default value. Histogram ticks are centered at their bin center.

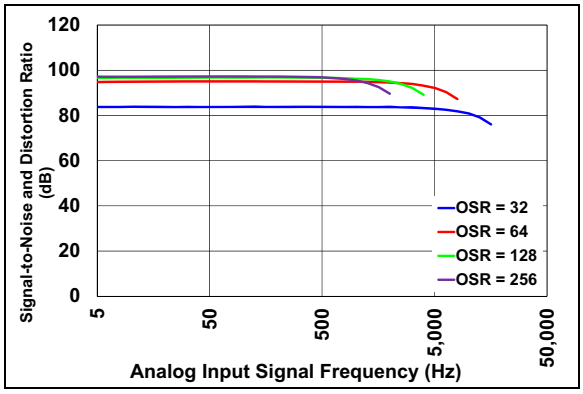



***FIGURE 2-25:*** *SINAD vs. Input Signal Frequency.*

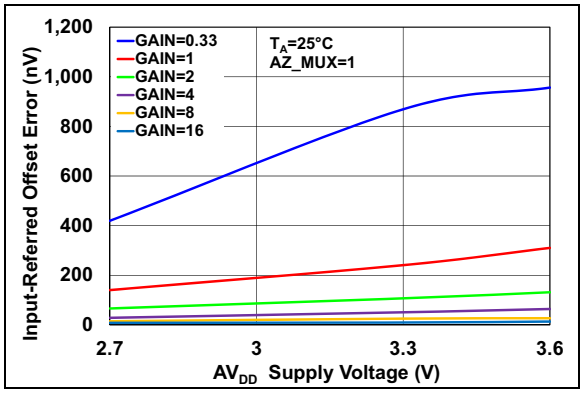



***FIGURE 2-28:*** *Offset Error vs. $AV_{DD}$ (AZ_MUX = 1).*

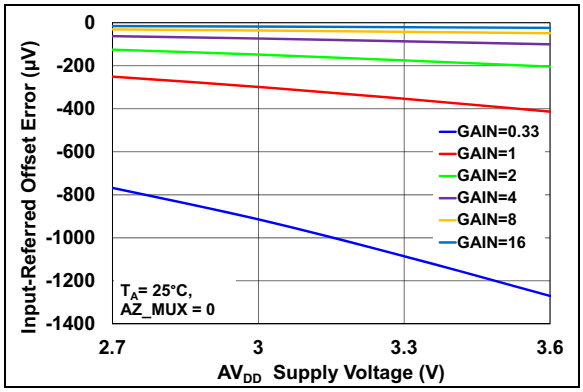



***FIGURE 2-26:*** *Offset Error vs. $AV_{DD}$ (AZ_MUX = 0)*

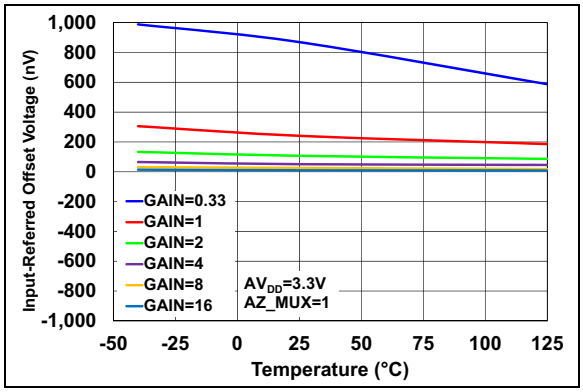



***FIGURE 2-29:*** *Offset Error vs. Temperature (AZ_MUX = 1).*

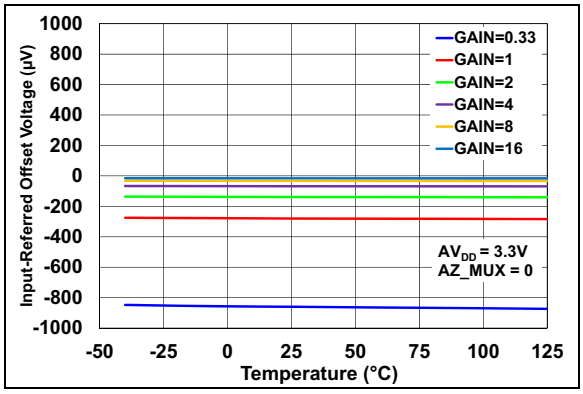



***FIGURE 2-27:*** *Offset Error vs. Temperature (AZ_MUX = 0)*

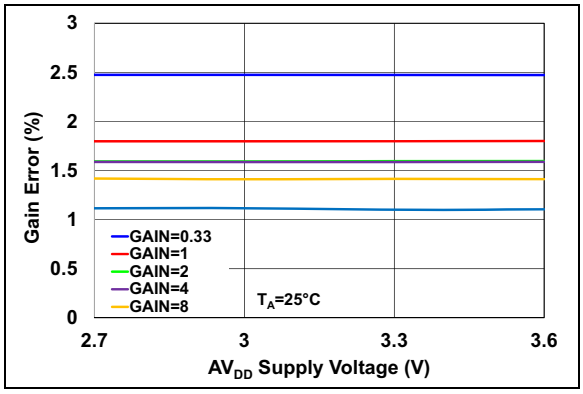



***FIGURE 2-30:*** *Gain Error vs. $AV_{DD}$.*

**Note:** Unless otherwise indicated, $AV_{DD}$ = 3.3V, $DV_{DD}$ = 3.3V; $T_A$ = +25°C, MCLK = 4.9152 MHz; $V_{IN}$ = -0.5 dBFS at 50 Hz, $V_{REF}$ = $AV_{DD}$; ADC_MODE = 11. All other registers are set to default value. Histogram ticks are centered at their bin center.

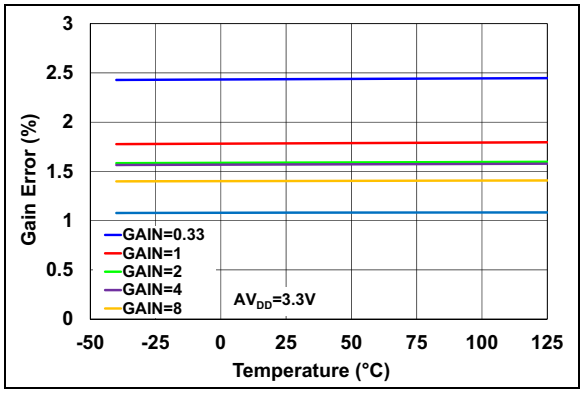



***FIGURE 2-31:*** *Gain Error vs. Temperature.*

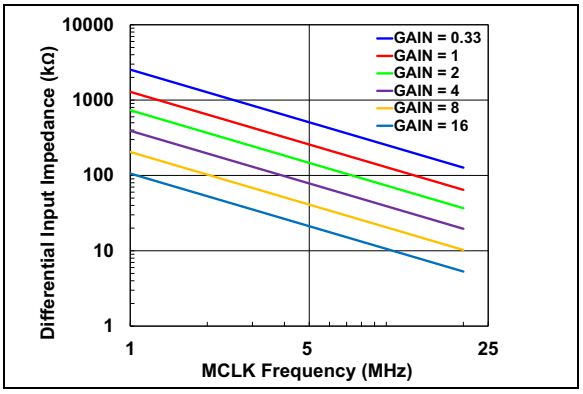



***FIGURE 2-34:*** *Differential Input Impedance vs. MCLK.*

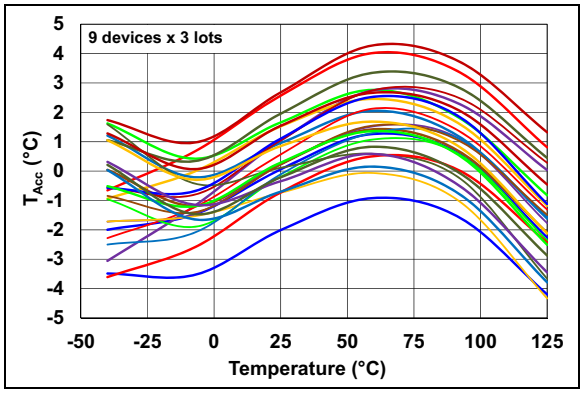



***FIGURE 2-32:*** *Temp Sensor Accuracy vs. Temperature (1st Order Best Fit).*

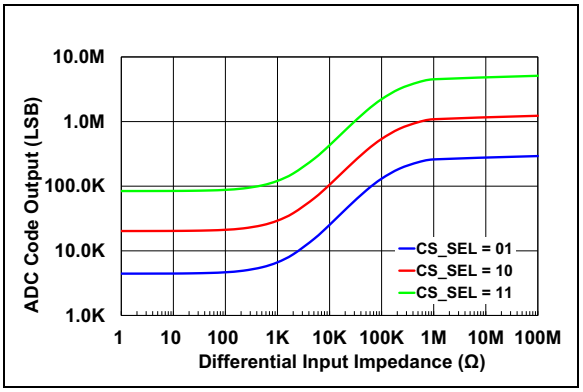



***FIGURE 2-35:*** *ADC Output Code vs. Differential Input Impedance, Burnout Current Sources Enabled.*

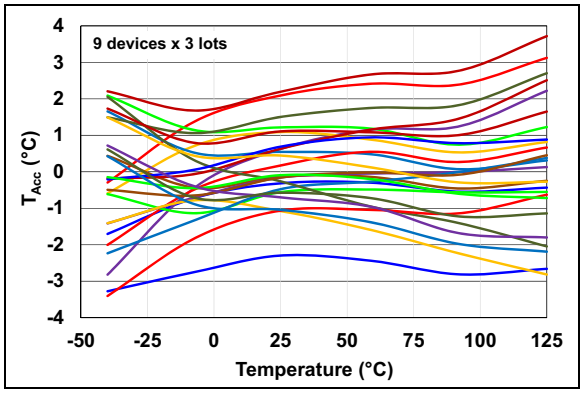



***FIGURE 2-33:*** *Temp Sensor Accuracy vs. Temperature (3rd Order Best Fit).*

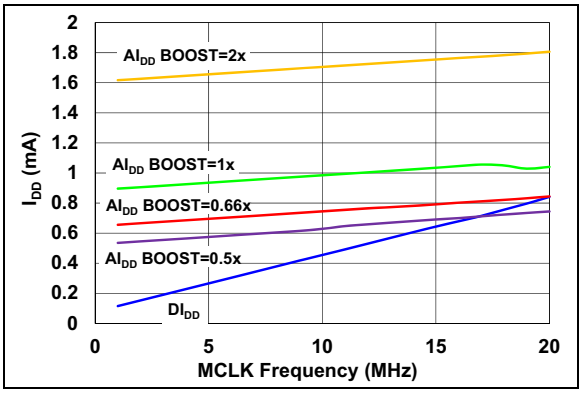



***FIGURE 2-36:*** *Current Consumption vs. MCLK.*

**Note:** Unless otherwise indicated, $AV_{DD}$ = 3.3V, $DV_{DD}$ = 3.3V; $T_A$ = +25°C, MCLK = 4.9152 MHz; $V_{IN}$ = -0.5 dBFS at 50 Hz, $V_{REF}$ = $AV_{DD}$; ADC_MODE = 11. All other registers are set to default value. Histogram ticks are centered at their bin center.

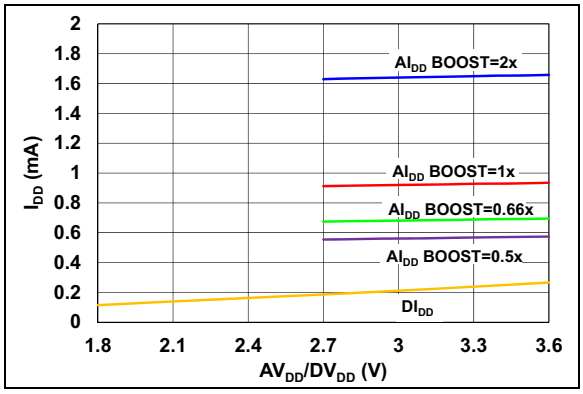



***FIGURE 2-37:*** *Current Consumption vs. $AV_{DD}$ and $DV_{DD}$.*

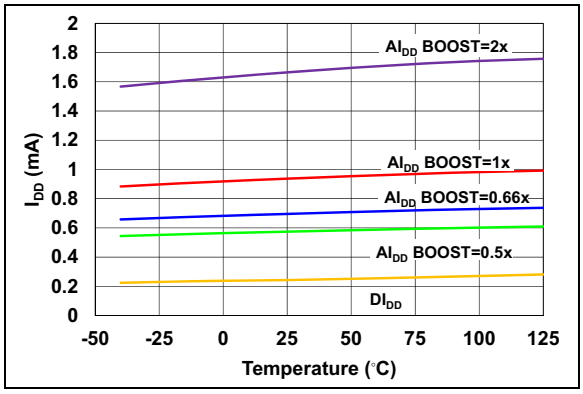



***FIGURE 2-38:*** *Current Consumption vs. Temperature.*

## 2.1 Noise Specifications

Table  and Table 2-1 summarize the noise performance of the MCP3461/2/4 devices. The noise performance is an analog gain function of the ADC (digital gain does not change the noise performance significantly) and the OSR chosen through the user interface. With a higher gain, the input-referred noise is reduced. With a higher OSR setting, the noise is also reduced as the oversampling diminishes both thermal noise and quantization noise induced by the Delta-Sigma modulator loop.

The noise is measured at a room temperature ($T_A$ = +25°C) and increases over temperature. For high OSR settings (> 512), the thermal noise is largely dominant and increases proportionally to the square root of the absolute temperature. The performance on the following tables has been measured with $AV_{DD}$ = $DV_{DD}$ = $V_{REF}$ = 3.3V and with the device placed in Continuous Conversion mode, with the differential input voltage equal to $V_{IN}$ = 0V, default conditions for the register map and MCLK = 4.9152 MHz.

The noise performance is also a function of the measurement duration. For short-duration measurements (low number of consecutive samples), the peak-to-peak noise is usually reduced because the crest factor (ratio between the RMS noise and peak-to-peak noise) is reduced. This is only a consequence of the noise distribution being Gaussian by nature (see Figure 2-3 for noise histogram example and fitting with an ideal Gaussian distribution). The noise specifications have been measured with a sample size of 16384 samples for low OSR values and have been capped to approximately 80 seconds for the 16384 samples leading to a larger duration. The noise specifications are expressed in two different values which lead to the same quantity. It may be more practical to choose one of these representations depending on the desired application.

In Table , the RMS (Root Mean Square) noise is the variance of the ADC output code, expressed in $\mu V_{RMS}$ and input-referred with Equation 5-5. The peak-to-peak noise values are under parentheses. The peak-to-peak noise is the difference of the measurement between the maximum and minimum code observed during the complete time of the measurement (see Equation 5-5).

In Table 2-1, the noise is expressed in ENOB (Effective Number of Bits). The ENOB is a ratio of the full-scale range of the ADC (that depends on $V_{REF}$ and GAIN) and the noise performance of the device. The ENOB can be determined from the RMS or peak-to-peak noise with the following equations:

**EQUATION 2-1:**

$$ENOB_{RMS} = \frac{In\left(\frac{2 \times V_{REF}}{GAIN \times RMS(Noise)}\right)}{In(2)}$$

**EQUATION 2-2:**

$$ENOB_{p-p} = \frac{In\left(\frac{2 \times V_{REF}}{GAIN \times Peak\text{-}to\text{-}Peak\ Noise}\right)}{In(2)}$$

Due to the nature of the noise, the performance detailed in the noise tables can vary significantly from one measurement to another. They present an averaging of the performance over a large distribution of parts over multiple lots. They give the typical expectation of the noise performance but performance can be better or worse if a limited number of measurements is performed. For large GAIN and OSR combinations, if the noise performance is comparable to the quantization step (1 LSB), the performance is limited to 0.5 LSB for the RMS noise and 1 LSB for the peak-to-peak noise (same limits for ENOB values).

These figures correspond to the resolution limit of the device as peak-to-peak noise cannot be better than 1 LSB.

Similarly, if the intrinsic RMS noise of the device is much smaller than 0.5 LSB, it may lead to histogram with either one or two bins depending on the relative position of the input voltage versus the possible quantized outputs of the ADC. If the position is exactly in between two quantization steps, the histogram of output noise will have two bins with exactly 50% occurrence on each. This case gives an RMS noise of a 0.5 LSB value, which is therefore used as a cap of the performance for the sake of clarity and a better representation on the noise tables.

The noise specifications are improved by a ratio of approximately √2 (or 0.5-bit ENOB) when the AZ_MUX setting is enabled. However, the output data rate is significantly reduced (see Figure 5-5 and Table 5-6).

The digital gain added for GAIN = 32x and 64x settings is not significant for the noise performance. Therefore, the noise values can be extracted from the GAIN = 16x columns. ENOB performance is degraded by 1 bit for GAIN = 32x and 2 bits for GAIN = 64x compared to the GAIN = 16x performance.

**NOISE RMS LEVEL VS. GAIN VS. OSR**

| TOTAL OSR | RMS (Peak-to-Peak) Noise (μV) | | | | | |
|---|---|---|---|---|---|---|
| | GAIN = 0.33 | GAIN = 1 | GAIN = 2 | GAIN = 4 | GAIN = 8 | GAIN = 16 |
| 32 | 388.9 (2829.9) | 130.2 (950) | 65.7 (481.7) | 33.2 (240.9) | 17 (125.5) | 8.9 (66.9) |
| 64 | 151.1 (564) | 50.4 (184.6) | 25.2 (102.4) | 12.6 (56.2) | 6.3 (34.8) | 3.4 (22.5) |
| 128 | 151.1 (302.1) | 50.4 (107.4) | 25.2 (57.1) | 12.6 (33.6) | 6.3 (21.4) | 3.2 (14.3) |
| 256 | 151.1 (302.1) | 50.4 (100.7) | 25.2 (50.4) | 12.6 (25.2) | 6.3 (15.9) | 3.2 (10.5) |
| 512 | 151.1 (302.1) | 50.4 (100.7) | 25.2 (50.4) | 12.6 (25.2) | 6.3 (12.6) | 3.2 (6.9) |
| 1024 | 151.1 (302.1) | 50.4 (100.7) | 25.2 (50.4) | 12.6 (25.2) | 6.3 (12.6) | 3.2 (6.3) |
| 2048 | 151.1 (302.1) | 50.4 (100.7) | 25.2 (50.4) | 12.6 (25.2) | 6.3 (12.6) | 3.2 (6.3) |
| 4096 | 151.1 (302.1) | 50.4 (100.7) | 25.2 (50.4) | 12.6 (25.2) | 6.3 (12.6) | 3.2 (6.3) |
| 8192 | 151.1 (302.1) | 50.4 (100.7) | 25.2 (50.4) | 12.6 (25.2) | 6.3 (12.6) | 3.2 (6.3) |
| 16384 | 151.1 (302.1) | 50.4 (100.7) | 25.2 (50.4) | 12.6 (25.2) | 6.3 (12.6) | 3.2 (6.3) |
| 20480 | 151.1 (302.1) | 50.4 (100.7) | 25.2 (50.4) | 12.6 (25.2) | 6.3 (12.6) | 3.2 (6.3) |
| 24576 | 151.1 (302.1) | 50.4 (100.7) | 25.2 (50.4) | 12.6 (25.2) | 6.3 (12.6) | 3.2 (6.3) |
| 40960 | 151.1 (302.1) | 50.4 (100.7) | 25.2 (50.4) | 12.6 (25.2) | 6.3 (12.6) | 3.2 (6.3) |
| 49152 | 151.1 (302.1) | 50.4 (100.7) | 25.2 (50.4) | 12.6 (25.2) | 6.3 (12.6) | 3.2 (6.3) |
| 81920 | 151.1 (302.1) | 50.4 (100.7) | 25.2 (50.4) | 12.6 (25.2) | 6.3 (12.6) | 3.2 (6.3) |
| 98304 | 151.1 (302.1) | 50.4 (100.7) | 25.2 (50.4) | 12.6 (25.2) | 6.3 (12.6) | 3.2 (6.3) |

**TABLE 2-1: EFFECTIVE NUMBER OF BITS VS. GAIN VS. OSR**

| TOTAL OSR | ENOB RMS (Peak-to-Peak) (bits) | | | | | |
|---|---|---|---|---|---|---|
| | GAIN = 0.33 | GAIN = 1 | GAIN = 2 | GAIN = 4 | GAIN = 8 | GAIN = 16 |
| 32 | 15.6 (12.8) | 15.6 (12.8) | 15.6 (12.7) | 15.6 (12.7) | 15.6 (12.7) | 15.5 (12.6) |
| 64 | 17 (15.2) | 17 (15.2) | 17 (15) | 17 (14.9) | 17 (14.5) | 16.9 (14.2) |
| 128 | 17 (16) | 17 (15.9) | 17 (15.9) | 17 (15.6) | 17 (15.3) | 17 (14.9) |
| 256 | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (15.7) | 17 (15.4) |
| 512 | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (15.9) |
| 1024 | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) |
| 2048 | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) |
| 4096 | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) |
| 8192 | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) |
| 16384 | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) |
| 20480 | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) |
| 24576 | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) |
| 40960 | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) |
| 49152 | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) |
| 81920 | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) |
| 98304 | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) | 17 (16) |

**Note:** To calculate NOISE RMS LEVEL and EFFECTIVE NUMBER OF BITS for a given GAIN and data rate, please refer to OSR setting and associated data rate relationship shown in Table 5-6.

**NOTES:**

## 3.0 PIN DESCRIPTION

**TABLE 3-1: MCP3461/2/4 PIN FUNCTION TABLE**

| MCP3461 | MCP3462 | MCP3464 | Symbol | Description |
|---|---|---|---|---|
| UQFN-20 | | | | |
| 1 | | | REFIN- | Inverting reference input pin |
| 2 | | | REFIN+ | Noninverting reference input pin |
| 3 | | | CH0 | Analog input 0 pin |
| 4 | | | CH1 | Analog input 1 pin |
| — | 5 | | CH2 | Analog input 2 pin |
| — | 6 | | CH3 | Analog input 3 pin |
| — | — | 7 | CH4 | Analog input 4 pin |
| — | — | 8 | CH5 | Analog input 5 pin |
| — | — | 9 | CH6 | Analog input 6 pin |
| — | — | 10 | CH7 | Analog input 7 pin |
| 11 | | | $\overline{CS}$ | Serial interface chip select digital input pin |
| 12 | | | SCK | Serial interface digital clock input pin |
| 13 | | | SDI | Serial interface digital data input pin |
| 14 | | | SDO | Serial interface digital data output pin |
| 15 | | | $\overline{IRQ}$/MDAT | Interrupt output pin or Modulator output pin |
| 16 | | | MCLK | Master Clock input or<br>Analog Master Clock output pin |
| 17 | | | $D_{GND}$ | Digital ground pin |
| 18 | | | $DV_{DD}$ | Digital supply voltage pin |
| 19 | | | $AV_{DD}$ | Analog supply voltage pin |
| 20 | | | $A_{GND}$ | Analog ground pin |
| 5, 6, 7, 8, 9, 10 | 7, 8, 9, 10 | — | NC | Not Connected |
| 21 | | | EP | Exposed pad internally connected to $A_{GND}$ |

### 3.1 Differential Reference Voltage Inputs: REFIN+, REFIN-

REFIN+ pin is noninverting differential reference input ($V_{REF+}$).

REFIN- pin is the inverting differential reference input ($V_{REF-}$).

For single-ended reference applications, the REFIN- pin should be directly connected to $A_{GND}$.

The differential reference voltage pins must respect this condition at all times: $0.6V \le V_{REF} \le AV_{DD}$. The differential reference voltage input is given by Equation 3-1:

**EQUATION 3-1:**

$$V_{REF} = V_{REF+} - V_{REF-}$$

For optimal ADC accuracy, appropriate bypass capacitors should always be placed between REFIN+ and $A_{GND}$. Using a 0.1 µF and a 10 µF ceramic capacitors helps with decoupling the reference voltage around the sampling frequency (which would lead to aliasing noise in the base band). These bypass capacitors are not mandatory for correct ADC operation, but removing these capacitors may degrade accuracy of the ADC.

## 3.2 Analog Inputs (CHn): Differential or Single-Ended

The CHn pins are the analog input signal pins for the ADC. Two analog multiplexers are used to connect the CHn pins to the $V_{IN+}/V_{IN-}$ analog inputs of the ADC. Each multiplexer selects independently one input to be connected to an ADC input ($V_{IN+}$ or $V_{IN-}$). Each CHn pin can either be connected to the $V_{IN+}$ or $V_{IN-}$ inputs of the ADC. This multiplexer selection is controlled by either the MUX register in MUX mode or the SCAN register in SCAN mode. See Figure 5-1 for more details on the multiplexer structure.

When the input is selected by the multiplexer, the differential ($V_{IN}$) and common mode voltage ($V_{INCOM}$) at the ADC inputs are defined by Equation 3-2

**EQUATION 3-2:**

$$V_{IN} = V_{IN+} - V_{IN-}$$

$$V_{INCOM} = \frac{V_{IN+} + V_{IN-}}{2}$$

The input signal level is multiplied by the internal programmable analog gain at the front end of the ΔΣ modulator. For single-ended input measurements, the user can select $V_{IN-}$ to be internally connected to $A_{GND}$.

The differential input voltage should not exceed an absolute of $\pm V_{REF}$/GAIN for accurate measurement. If the input is out of range, the converter output code will be saturated or overloaded depending on how the output data format (DATA_FORMAT[1:0]) is selected. See **Section 5.6 "ADC Output Data Format"** for further information on the ADC output coding.

The absolute voltage range on each of the analog signal input pins is from $A_{GND}$ - 0.1V to $V_{DD}$ + 0.1V. Any voltage above or below this range will cause leakage currents through the Electrostatic Discharge (ESD) diodes at the input pins. This ESD current can cause unexpected performance of the device. The Common-mode of the analog inputs should be chosen such that both the differential analog input range and the absolute voltage range on each pin are within the specified operating range defined in **Table "Electrical Characteristics"**.

## 3.3 SPI Serial Interface Communication pins

The SPI interface is compatible with both SPI 0,0 and 1,1 modes.

### 3.3.1 CHIP SELECT ($\overline{CS}$)

This is the SPI chip select pin that enables/disables the SPI serial communication. The $\overline{CS}$ falling edge initiates the serial communication and the rising edge terminates the communication. No communication can take place when this pin is in Logic High state. This input is Schmitt Triggered.

### 3.3.2 SERIAL DATA CLOCK (SCK)

This is the serial clock input pin for SPI communication. This input has Schmitt Trigger structure. The maximum SPI clock speed is 20 MHz. Data are clocked into the device on the rising edge of SCK. Data are clocked out of the device on the falling edge of SCK. The device interface is compatible with both SPI 0,0 and 1,1 modes. SPI modes can be changed when $\overline{CS}$ is in Logic High status.

SCK and MCLK are two different and asynchronous clocks; SCK is only required when a communication happens, while MCLK is continuously required when the part is converting analog inputs.

### 3.3.3 SERIAL DATA OUTPUT PIN (SDO)

This pin is used for the SPI data output (SDO). The SDO data are clocked out on the falling edge of SCK. This pin stays high impedance under the following conditions:

- When $\overline{CS}$ pin is logic high.
- During the whole SPI write or Fast-Command communication period, after the SPI Command byte has been transmitted.
- After the two device address bits in the command have been transmitted, if the device address in the command is not matching internal chip device address.

### 3.3.4 SERIAL DATA INPUT PIN (SDI)

This is the SPI data input pin and it uses Schmitt Trigger structure. When $\overline{CS}$ is logic low, this pin is used to send a command byte just after $\overline{CS}$ falling edge, which can be followed by data words of various lengths. Data are clocked into the device on the rising edge of SCK. Toggling SDI while reading a register has no effect.

## 3.4 $\overline{\text{IRQ}}$/MDAT

This is the digital output pin. This pin can be configured for Interrupt ($\overline{\text{IRQ}}$) or Modulator Data output (MDAT) using the IRQ_MODE[1] bit setting. When IRQ_MODE[1] = 0 (default), this pin can output all four possible interrupts (see **Section 6.8 "Interrupts Description"**). The inactive state of the pin is selectable through the IRQ_MODE[0] bit setting (high-Z or logic high).

When IRQ_MODE[1] = 1, this pin outputs the modulator output synchronously with AMCLK (that can be selected as an output on the MCLK pin). In this mode, the POR and CRC interrupts can still be generated as they are high-level interrupts and will lock the $\overline{\text{IRQ}}$/MDAT pin to logic low until they are cleared.

When the $\overline{\text{IRQ}}$ pin is in Hi-Z mode, an external pull-up resistor must be connected between $DV_{DD}$ and $\overline{\text{IRQ}}$ pin. The device needs to be able to detect a Logic High state when no interrupt occurs in order to function properly (the pad has a Schmitt Trigger input to detect the state of the $\overline{\text{IRQ}}$ pin just like the user is seeing it). The pull-up value can be equal to 100-200 kΩ for a weak pull-up using the typical clock frequency. The pull-up resistor value needs to be chosen in relation with the load capacitance of the $\overline{\text{IRQ}}$ output, the MCLK frequency and the $DV_{DD}$ supply voltage, so that all interrupts can be detected correctly by the SPI master device.

## 3.5 MCLK

This pin is either the MCLK digital input pin for the ADC or the AMCLK digital output pin, depending on the CLK_SEL[1:0] bit settings in the CONFIG0 register.

The typical clock frequency specified is 4.9152 MHz. To optimize the ADC for accuracy and ensure proper operation, AMCLK should be limited to a certain range depending on BOOST and GAIN settings. The higher GAIN settings require higher BOOST settings to maintain high bandwidth, as the input sampling capacitors have a larger value. Figure 2-20 to Figure 2-24 represent the typical accuracy (SINAD) expected with the different combinations of BOOST and GAIN settings and can be used to determine an optimal set for the application depending on the sampling speed (AMCLK) chosen. MCLK can take larger values as long as the prescaler settings (PRE[1:0]) limit AMCLK = MCLK/PRESCALE in the defined range in typical performance curves.

## 3.6 Digital Ground ($D_{GND}$)

$D_{GND}$ is the ground connection to internal digital circuitry. To ensure accuracy and noise cancellation, $D_{GND}$ must be connected to the same ground as $A_{GND}$, preferably with a star connection. If a digital ground plane is available, it is recommended for this pin to be tied to this plane of the PCB. This plane should also reference all other digital circuitry in the system. $D_{GND}$ is not connected internally to $A_{GND}$ and must be connected externally.

## 3.7 Digital Power Supply ($DV_{DD}$)

$DV_{DD}$ is the power supply pin for the digital circuitry within the device. The voltage on this pin must be maintained in the range specified by **Table "Electrical Characteristics"**. For optimal performance, it is recommended to connect appropriate bypass capacitors (typically a 10 µF ceramic in parallel with a 0.1 µF ceramic). $DV_{DD}$ is monitored by the $DV_{DD}$ POR monitoring circuit for the digital section.

## 3.8 Analog Power Supply ($AV_{DD}$)

$AV_{DD}$ is the power supply pin for the analog circuitry within the device. The voltage on this pin must be maintained in the range specified by **Table "Electrical Characteristics"**. For optimal performance, it is recommended to connect appropriate bypass capacitors (typically a 10 µF ceramic in parallel with a 0.1 µF ceramic). $AV_{DD}$ is monitored by the $AV_{DD}$ POR monitoring circuit for the analog section.

## 3.9 Analog Ground ($A_{GND}$)

$A_{GND}$ is the ground connection to internal analog circuitry. To ensure accuracy and noise cancellation, this pin must be connected to the same ground as $D_{GND}$, preferably with a star connection. If an analog ground plane is available, it is recommended that this pin be tied to this plane of the PCB. This plane should also reference all other analog circuitry in the system. $A_{GND}$ is the biasing voltage for the substrate of the die and is not connected internally to $D_{GND}$.

## 3.10 Exposed Pad (EP)

This pad is internally connected to $A_{GND}$. It must be connected to the analog ground of the PCB for optimal accuracy and thermal performance. This pad can also be left floating if necessary.

**NOTES:**

## 4.0 TERMINOLOGY AND FORMULAS

This section defines the terms and formulas used throughout this document. The following terminology is defined:

- MCLK – Master Clock
- AMCLK – Analog Master Clock
- DMCLK – Digital Master Clock
- DRCLK – Data Rate Clock
- OSR – Oversampling Ratio
- Offset Error
- Gain Error
- Integral NonLinearity Error (INL)
- Signal-to-Noise Ratio (SNR)
- Signal-To-Noise And Distortion Ratio (SINAD)
- Total Harmonic Distortion (THD)
- Spurious-Free Dynamic Range (SFDR)
- MCP3461/2/4 Delta-Sigma Architecture
- Power Supply Rejection Ratio (PSRR)
- Common-mode Rejection Ratio (CMRR)
- Digital Pins Output Current Consumption

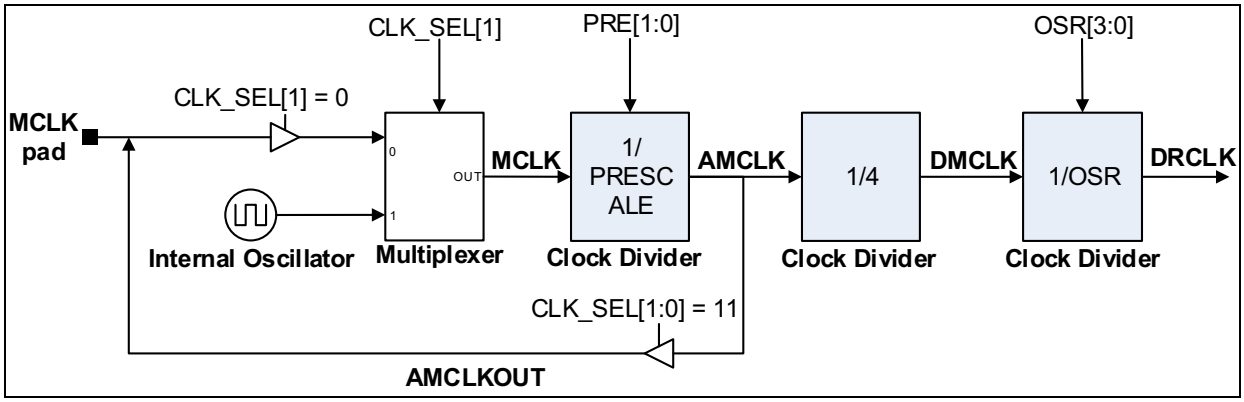



***FIGURE 4-1:*** *System Clock Details.*

### 4.1 MCLK – Master Clock

This is either the master clock frequency at MCLK input pin when an external clock source is selected or the internal clock frequency when an internal clock is selected.

### 4.2 AMCLK – Analog Master Clock

This is the clock frequency that is present on the analog portion of the device, after prescaling has occurred via the PRE[1:0] bits.

**EQUATION 4-1: ANALOG MASTER CLOCK**

$$AMCLK = \frac{MCLK}{Prescale}$$

### 4.3 DMCLK – Digital Master Clock

This is the clock frequency that is present on the digital portion of the device. This is also the sampling frequency, or the rate at which the modulator outputs are refreshed. Each period of this clock corresponds to one sample and one modulator output. See Equation 4-2.

**EQUATION 4-2: DIGITAL MASTER CLOCK**

$$DMCLK = \frac{AMCLK}{4} = \frac{MCLK}{4 \times Prescale}$$

### 4.4 DRCLK – Data Rate Clock

This is the output data rate in Continuous mode, which is the rate at which the ADC outputs new data. Each new data are signaled by a Data Ready pulse on the $\overline{IRQ}$ pin. This data rate depends on the OSR and the prescaler as shown in Equation 4-3.

**EQUATION 4-3: DATA RATE**

$$DRCLK = \frac{DMCLK}{OSR} = \frac{AMCLK}{4 \times OSR} = \frac{MCLK}{4 \times OSR \times Prescale}$$

Since this is the output data rate, and since the decimation filter is a sinc (or Notch) filter, there is a notch in the filter transfer function at each integer multiple of this rate.

## 4.5 OSR – Oversampling Ratio

The ratio of the sampling frequency to the output data rate. OSR = DMCLK/(DRCLK) in Continuous mode. See Table 5-6 for the OSR setting effect on sinc filter parameters.

## 4.6 Offset Error

This is the error induced by the ADC when the inputs are shorted together ($V_{IN}$ = 0V). This error varies based on gain settings, OSR settings and from chip to chip. It can easily be calibrated out by a MCU with a subtraction.

## 4.7 Gain Error

This is the error induced by the ADC on the slope of the transfer function. It is the deviation expressed in percentage compared to the ideal transfer function defined by Equation 5-5. The specification incorporates ADC gain error contributions, but not the $V_{REF}$ contribution. This error varies with GAIN and OSR settings.

The gain error of this device has a low temperature coefficient.

## 4.8 Integral NonLinearity Error (INL)

Integral nonlinearity error is the maximum deviation of an ADC transition point from the corresponding point of an ideal transfer function, with the offset and gain errors removed, or with the end points equal to zero. It is the maximum remaining static error after calibration of offset and gain errors for a DC input signal.

## 4.9 Signal-to-Noise Ratio (SNR)

For this device family, the signal-to-noise ratio is a ratio of the output fundamental signal power to the noise power (not including the harmonics of the signal), when the input is a sine wave at a predetermined frequency. It is measured in dB. Usually, only the maximum signal to noise ratio is specified. The SNR figure depends mainly on the OSR and GAIN settings of the device as well as temperature (due to thermal noise being dominant for high OSR).

**EQUATION 4-4: SIGNAL TO NOISE RATIO**

$$SNR(dB) = 10log\left(\frac{SignalPower}{NoisePower}\right)$$

## 4.10 Signal-To-Noise And Distortion Ratio (SINAD)

Signal-to-noise and distortion ratio is similar to signal-to-noise ratio, with the exception that you must include the harmonics power in the noise power calculation. The SINAD specification depends mainly on the OSR and GAIN settings.

**EQUATION 4-5: SINAD EQUATION**

$$SINAD(dB) = 10log\left(\frac{SignalPower}{Noise + HarmonicsPower}\right)$$

The calculated combination of SNR and THD per Equation 4-6 also yields SINAD:

**EQUATION 4-6: SINAD, THD, AND SNR RELATIONSHIP**

$$SINAD(dB) = 10log\left[10^{\left(\frac{SNR}{10}\right)} + 10^{\left(\frac{THD}{10}\right)}\right]$$

## 4.11 Total Harmonic Distortion (THD)

The total harmonic distortion is the ratio of the output harmonics power to the fundamental signal power for a sine wave input and is defined by Equation 4-7.

**EQUATION 4-7:**

$$THD(dB) = 10log\left(\frac{HarmonicsPower}{FundamentalPower}\right)$$

The THD is usually only measured with respect to the 10 first harmonics. THD is sometimes expressed in percentage (%). For converting the THD from “dB” to “%”, apply the formula in Equation 4-8.

**EQUATION 4-8:**

$$THD(\%) = 100 \times 10^{\frac{THD(dB)}{20}}$$

## 4.12 Spurious-Free Dynamic Range (SFDR)

The ratio between the output power of the fundamental and the highest spur in the frequency spectrum. The spur frequency is not necessarily a harmonic of the fundamental even though it is usually the case. This figure represents the dynamic range of the ADC when a full-scale signal is used at the input. This specification depends mainly on the OSR and GAIN setting.

**EQUATION 4-9:**

$$SFDR(dB) = 10log\left(\frac{FundamentalPower}{HighestSpurPower}\right)$$

## 4.13 MCP3461/2/4 Delta-Sigma Architecture

A Delta-Sigma ADC is an oversampling converter that incorporates a built-in modulator which digitizes the quantity of charge integrated by the modulator loop. The quantizer is the block that performs the Analog-to-Digital Conversion. The quantizer is typically 1-bit or a simple comparator that helps to maintain the linearity performance of the ADC (the DAC structure is in this case inherently linear).

Multi-bit quantizers help to lower the quantization error (the error fed back in the loop can be very large with 1-bit quantizers) without changing the order of the modulator or the OSR that leads to better SNR figures. However, typically, the linearity of such architectures is more difficult to achieve since the DAC is no more simple to realize and its linearity limits the THD of such ADC.

The modulator 5-level quantizer is a Flash ADC composed of 4 comparators arranged with equally spaced thresholds and a thermometer coding. The device also includes proprietary 5-level DAC architecture that is inherently linear for improved THD figures.

## 4.14 Power Supply Rejection Ratio (PSRR)

This is the ratio between a change in the power supply voltage and the ADC output codes. It measures the influence of the power supply voltage on the ADC outputs.

The PSRR specification can be DC (the power supply is taking multiple DC values) or AC (the power supply is a sine wave at a certain frequency with a certain Common-mode). In AC, the amplitude of the sine wave represents the change in the power supply.

**EQUATION 4-10:**

$$PSRR(dB) = 20log\left(\frac{\Delta V_{OUT}}{\Delta AV_{DD}}\right)$$

Where $V_{OUT}$ is the equivalent input voltage that the output code translates to with the ADC transfer function.

## 4.15 Common-mode Rejection Ratio (CMRR)

This is the ratio between a change in the Common-mode input voltage and the change in ADC output codes. It measures the influence of the Common-mode input voltage on the ADC outputs.

The CMRR specification can be DC (Common-mode input voltage is taking multiple DC values) or AC (the Common-mode input voltage is a sine wave at a certain frequency with a certain Common-mode). In AC, the amplitude of the sine wave represents the change in the input Common-mode voltage.. CMRR is defined in Equation 4-11.

**EQUATION 4-11:**

$$CMRR(dB) = 20log\left(\frac{\Delta V_{OUT}}{\Delta V_{INCOM}}\right)$$

Where $V_{INCOM} = (V_{IN+} + V_{IN-})/2$ is the Common-mode input voltage and VOUT is the equivalent input voltage that the output code translates to with the ADC transfer function.

## 4.16 Digital Pins Output Current Consumption

The digital current consumption shown in **Table "Electrical Characteristics"** does not take into account the current consumption generated by the digital output pins and the charge of their capacitive loading. The specification is intended with all output pins left floating and no communication.

In order to estimate the additional current consumption due to the output pins, refer to Equation 4-2. This equation specifies the amount of additional current due to each pin when its output is connected to a $C_{load}$ capacitance, with respect to $D_{GND}$ and submitted to an output signal toggling at an $f_{out}$ frequency.

If a typical 10 MHz SPI frequency is used, with a 30 pF load and $DV_{DD}$ = 3.3V, the SDO output generates an additional maximum current consumption of 500 µA (the maximum toggling frequency of SDO is 5 MHz here since $f_{SCK}$ = 10 MHz and this maximum happens when the ADC output code is a succession of `1`s and `0`s). The $C_{load}$ value includes internal digital output driver capacitance, but this one can generally be neglected with respect to the external loading capacitance.

**EQUATION 4-12:**

$$DIDD_{SPI} = C_{load} \times DV_{DD} \times f_{out}$$

**Where:**

$C_{load}$ = Capacitance on the output pin
$DV_{DD}$ = Digital supply voltage
$f_{out}$ = Output frequency on the output pin

**NOTES:**

# 5.0 DEVICE OVERVIEW

## 5.1 Analog Input Multiplexer

The device includes a fully configurable analog input dual multiplexer that can select which input is connected to each of the two differential input pins ($V_{IN+}/V_{IN-}$) of the Delta-Sigma ADC.

The dual multiplexer is divided into two single-ended multiplexers that are completely independent. Each of these multiplexers include the same possibilities for the input selection so that any required combination of input voltages can be converted by the ADC. The analog multiplexer is composed of parallel low-resistance input switches turned on or off depending on the input channel selection. Their resistance is negligible compared to the input impedance of the ADC (caused by the charge and discharge of the input sampling capacitors on the $V_{IN+}/V_{IN-}$ ADC inputs). The block diagram of the analog multiplexer is shown in Figure 5-1.

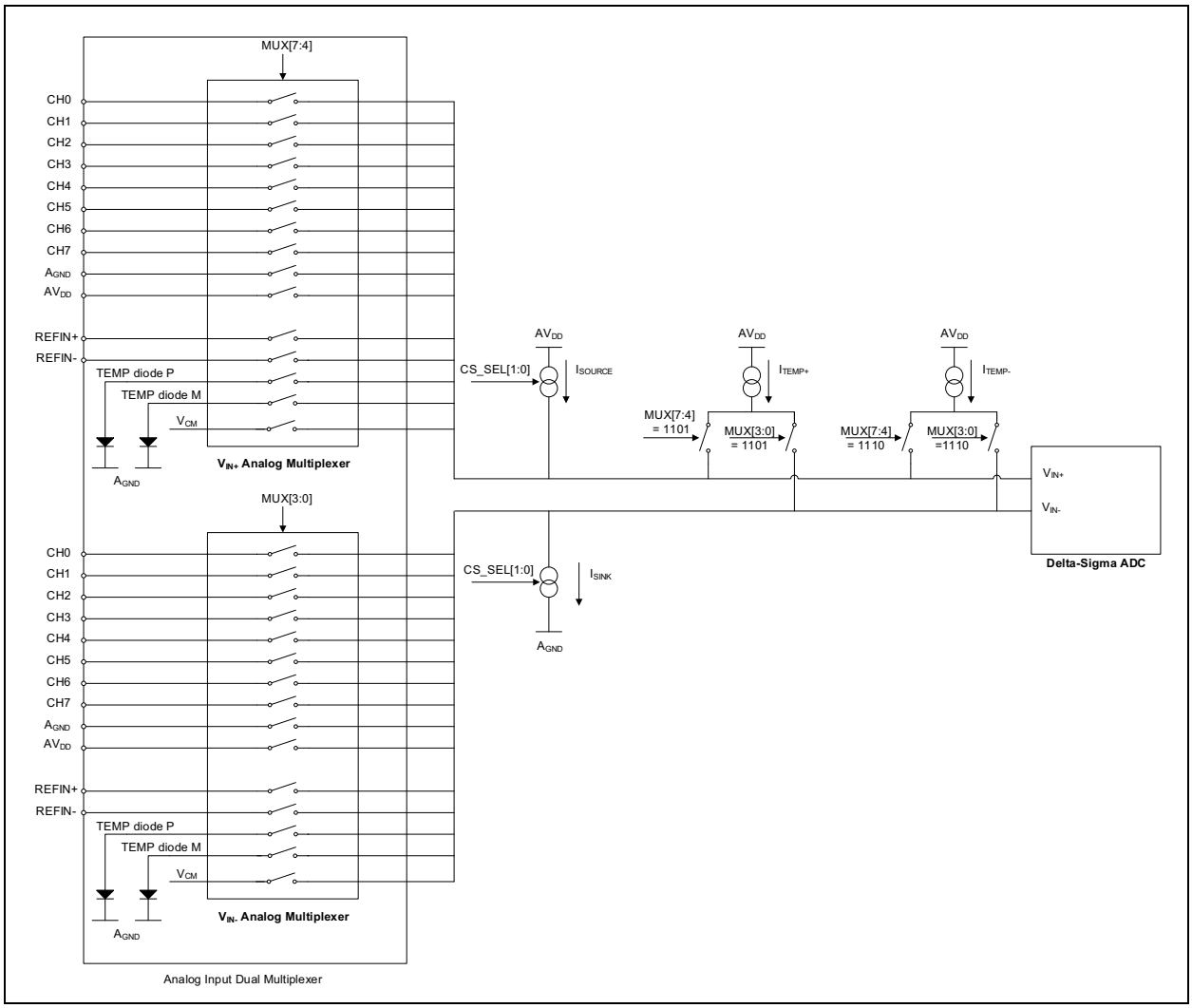



***FIGURE 5-1:*** *Simplified Analog Input Multiplexer Schematic.*

The possible selections are described in Table 5-1 and can be set with the MUX[7:0] register during the MUX mode. The MUX[7:4] bits define the selection for the $V_{IN+}$ (noninverting analog input of the ADC). The MUX[3:0] bits define the selection for the $V_{IN-}$ (inverting analog input of the ADC).

**TABLE 5-1: ANALOG INPUT MUX DECODING TABLE**

| MUX[7:4] ($V_{IN+}$) or MUX[3:0] ($V_{IN-}$) code | Selected Channel | Comment |
|---|---|---|
| 0000 | CH0 | |
| 0001 | CH1 | |
| 0010 | CH2 | Not Connected (NC) for MCP3461 |
| 0011 | CH3 | Not Connected (NC) for MCP3461 |
| 0100 | CH4 | Not Connected (NC) for MCP3461, MCP3462 |
| 0101 | CH5 | Not Connected (NC) for MCP3461, MCP3462 |
| 0110 | CH6 | Not Connected (NC) for MCP3461, MCP3462 |
| 0111 | CH7 | Not Connected (NC) for MCP3461, MCP3462 |
| 1000 | $A_{GND}$ | |
| 1001 | $AV_{DD}$ | |
| 1010 | Reserved | Do not use |
| 1011 | REFIN+ | |
| 1100 | REFIN- | |
| 1101 | TEMP Diode P | |
| 1110 | TEMP Diode M | |
| 1111 | Internal VCM | Internal common-mode voltage for modulator biasing |

During SCAN mode, the two single-ended input multiplexers are automatically set to a certain position depending on the SCAN sequence and on which channel has been selected by the user. The SCAN sequence channels configuration correspond to a certain code in the MUX[7:0] register as defined in Table 5-13.

In order to monitor the digital power supply ($DV_{DD}$) it is necessary to connect externally $DV_{DD}$ to one of the CHn analog inputs since $DV_{DD}$ is not one of the possible selections of the analog multiplexer. A similar setup can be implemented to monitor $D_{GND}$ if $D_{GND}$ is not connected externally to $A_{GND}$.

For MCP3461 and MCP3462 some codes are not available in the selection since the pins are not bonded out on these devices. These codes should then be avoided in the application, as the input they connect to is effectively a high impedance node.

The TEMP diodes P and M are two internal diodes that are biased by a current source and that can be used to perform a temperature measurement. If TEMP diode P is connected to $V_{IN+}$ and TEMP diode M to $V_{IN-}$, then the ADC output code is a function of the temperature using Equation 5-1 (see **Section 5.1.2 "Internal Temperature Sensor"** for more details).

The "VCM" selection measures the internal Common-mode voltage source that biases the Sigma-Delta modulator (this voltage is not provided at any output of the part).

The possible inputs of the analog multiplexer include not only the analog input channels but also REFIN+/- inputs, $AV_{DD}$ and $A_{GND}$ as well as temperature sensor outputs and VCM internal Common-mode. This large selection offers many possibilities for measuring internal or external data resources of the system and can serve as diagnostics purposes to increase the security of the applications. Some monitor channels are already predefined in SCAN mode to further help users to integrate diagnostics to their applications (For example, the analog power supply, or the temperature can be constantly monitored in SCAN mode, see section 5.14.3 for more details of the different resources that can be monitored in SCAN mode).

### 5.1.1 BURN-OUT CURRENT SOURCES FOR SENSOR OPEN/SHORT DETECTION

The ADC inputs $V_{IN-}/V_{IN+}$ features a selectable burnout current source that enables open or short circuit detection, as well as biasing very low current external sensors. The bias current is sourced on the $V_{IN+}$ pin of the ADC (noninverting output of the analog multiplexer) and sunk on the $V_{IN-}$ pin of the ADC (inverting output of the analog multiplexer). Since the same current flows at the $V_{IN+}/V_{IN-}$ pins of the ADC, it can sense the impedance of an externally connected sensor that would be connected between the selected inputs of the multiplexer. When the sensor is in short-circuit, the ADC will convert signals that are close to 0V. When the sensor is an open circuit, the ADC will convert signals that are close to the $AV_{DD}$ voltage.

The current source is an independent peripheral of the ADC. It does not need the ADC to be in Conversion mode to be present. Once enabled, the current source provides current even when the ADC is in Reset or Shutdown modes. The current source can be configured at any time by programming the CS_SEL[1:0] bits in the CONFIG0 register (see Table 5-2).

Since the amount of current selected can be very small, it may be necessary to diminish the MCLK master clock frequency to be able to reach full desired accuracy during conversions (the settling time of the input structure including the sensor can be large if the sensor is very resistive which will limit the bandwidth of the sample and hold input circuit).

The accuracy of the current sources is around ±20% and it is not controlled well internally. However, the mismatch between sink and source is typically around ±1%. This relatively low accuracy on the current is generally sufficient for open/short detection applications.

Figure 2-35 shows how the ADC output code is varying when the burn-out current sources are enabled (with GAIN = 1x) and the input sensor impedance is swept with a large dynamic range. This permits the users to use the ADC as an open/short detection circuit that is practical when manufacturing complex remote sensor systems.

**TABLE 5-2: BURNOUT CURRENT SOURCE SETTINGS**

| CS_SEL[1:0] (source/sink) | Burnout Current Amplitude |
|---|---|
| 00 | 0 µA |
| 01 | 0.9 µA |
| 10 | 3.7 µA |
| 11 | 15 µA |

### 5.1.2 INTERNAL TEMPERATURE SENSOR

The device includes an on-board temperature sensor, that is made of two typical P-N junction diodes biased by fixed current sources (TEMP Diode P and M). The TEMP Diode P has a current density of 4x of the TEMP diode M.

The difference in the current densities of the diodes yields a voltage, which is a function of the absolute temperature.

Once the ADC inputs ($V_{IN+}/V_{IN-}$) are connected to the temperature sensor diodes (MUX[7:0] = 0xDE), the ADC will see a $V_{IN}$ differential input that is the function of the temperature. The transfer function of the temperature sensor can be approximated by a linear equation or a third-order equation for more accuracy.

When the internal temperature sensor is selected for the MUX or SCAN input, the input sink/source current source controlled by CS_SEL[1:0] bits (see **Section 5.1 "Analog Input Multiplexer"**) is disabled internally (even though the CS_SEL[1:0] bits are not modified by the temperature sensor selection). In this case, the input current source is replaced by a specific internal current source that will only be sourced to the diode temperature sensor (see Figure 5-1).

The bias current of the diodes is not calibrated internally and can lead to a relatively large gain and offset error in the transfer function of the temperature sensor. Typical graphs showing the typical error in the temperature measurement are provided in **Section 2.0 "Typical Performance Curves"** (see Figure 2-32 first-order and Figure 2-33 for third-order fitting).

The accuracy can also be optimized by using proper digital gain and offset error calibration schemes.

**EQUATION 5-1: TEMPERATURE SENSOR TRANSFER FUNCTION**

First-order (linear) fitting: GAIN = 1, $V_{REF}$ = 3.3V

$$TEMP(°C) = 0.001581 \times 256 \times ADCDATA(LSB) - 324.27$$

$$V_{IN}(mV) = 0.24871 \times TEMP(°C) + 80.678$$

Third-order fitting: GAIN = 1, $V_{REF}$ = 3.3V

$$TEMP(°C) = 2.71 \times 10^{-14} \times (256 \times ADCDATA(LSB))^3 - 1.8 \times 10^{-8} \times (256 \times ADCDATA(LSB))^2 + 0.0055 \times 256 \times ADCDATA(LSB) - 604.22$$

$$V_{IN}(mV) = 1.579 \times 10^{-6} \times (TEMP(°C))^3 + 0.00012 \times (TEMP(°C))^2 + 0.2558 \times TEMP(°C) + 80.55$$

### 5.1.3 ADC OFFSET CANCELLATION ALGORITHM

The input multiplexer and the ADC include an offset cancellation algorithm that cancels the offset contribution of the ADC. This offset cancellation algorithm is controlled by the bit AZ_MUX in the CONFIG2 register. When AZ_MUX = `0` (default), the offset cancellation algorithm is disabled and the conversions are not affected by this setting. When AZ_MUX = `1`, the algorithm is enabled. When the offset cancellation algorithm is enabled, ADC takes two conversions, one with the differential input as $V_{IN}+/V_{IN}-$, one with $V_{IN+}/V_{IN}-$ inverted. Equation 5-2 calculates the ADC output code. When AZ_MUX = `1`, the conversion time $T_{CONV}$ is multiplied by two, compared to the default case, where AZ_MUX = `0`

**EQUATION 5-2: AZ_MUX CONVERSION RESULT EQUATION**

$$ADC\ Output\ Code\ (AZ\_MUX=1) = \frac{(ADC\ Output\ at\ +V_{IN}) - (ADC\ Output\ at\ -V_{IN})}{2}$$

This technique allows the cancellation of the ADC offset error and the achievement of ultra-low offset without any digital calibration. The resulting offset is the residue of the difference of the two conversions, which is on the same order of magnitude of the noise floor. This offset is effectively canceled at every conversion, so the residual offset error temperature drift is extremely low.

For One-Shot mode, the conversion time is simply multiplied by two. Enabling the AZ_MUX bit is not compatible with the Continuous Conversion mode (because it effectively multiplexes the inputs in between each conversion). If AZ_MUX = `1` and CONV_MODE = `11` (Continuous Conversion mode), the device will reset the digital filter in between each conversion and will therefore have an output data rate of 1/(2 * $T_{CONV}$). The Continuous mode is replaced by a series of One-Shot mode conversions with no delay in between each conversion (see **Section 5.13 "Conversion modes"** and Figure 5-5 for more details about the Conversion modes).

## 5.2 Input Impedance

The ADC inputs ($V_{IN+}$/$V_{IN-}$) are directly tied to the analog multiplexer outputs and are not routed to external pins. The multiplexer input stage contribution to the input impedance is negligible.

The conversion accuracy can be affected by the input signal source impedance when any external circuit is connected to the input pins. The source impedance adds to the internal impedance and directly affects the time required to charge the internal sampling capacitor. Therefore, a large input source impedance connected to the input pins can increase the system performance errors such as offset, gain and integral nonlinearity (INL). Ideally, the input source impedance should be near zero. This can be achievable by using an operational amplifier with a closed-loop output impedance of tens of ohms.

A proper anti-aliasing filter must be placed at the ADC inputs. This will attenuate the frequency contents around DMCLK and keep the desired accuracy over the baseband (DRCLK) of the converter.

This anti-aliasing filter can be a simple first-order RC network with low time constant that will provide a high rejection at DMCLK frequency (see Figure 5.6 for more details). The RC network usually uses small R and large C to avoid additional offset due to IR drop in the signal path. This anti-aliasing filter will induce a small systematic gain error on the AC input signals that can be compensated in the digital section with the digital gain error calibration register (GAINCAL).

## 5.3 ADC Programmable Gain

The gain of the converter is programmable and controlled by the GAIN[2:0] bits in the CONFIG2 register. The ADC programmable gain is divided in two gain stages: one in the analog domain, one in the digital domain as per the Table 5-3.

After the multiplexer, the analog input signals are routed to the Delta-Sigma ADC inputs and are amplified by the analog gain stage (see **Section 5.3.1 "Analog Gain"** for more details). The digital gain stage is placed inside the digital decimation filter (see **Section 5.3.2 "Digital Gain"** for more details).

**TABLE 5-3: SIGMA-DELTA ADC GAIN SETTINGS**

| GAIN[2:0] | | | Total Gain (V/V) | Analog Gain (V/V) | Digital Gain (V/V) | Total Gain (dB) | $V_{IN}$ Range (V) |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0.333 | 0.333 | 1 | -9.5 | ±Min ($AV_{DD}$, 3*$V_{REF}$) |
| 0 | 0 | 1 | 1 | 1 | 1 | 0 | ±$V_{REF}$ |
| 0 | 1 | 0 | 2 | 2 | 1 | 6 | ±$V_{REF}$/2 |
| 0 | 1 | 1 | 4 | 4 | 1 | 12 | ±$V_{REF}$/4 |
| 1 | 0 | 0 | 8 | 8 | 1 | 18 | ±$V_{REF}$/8 |
| 1 | 0 | 1 | 16 | 16 | 1 | 24 | ±$V_{REF}$/16 |
| 1 | 1 | 0 | 32 | 16 | 2 | 30 | ±$V_{REF}$/32 |
| 1 | 1 | 1 | 64 | 16 | 4 | 36 | ±$V_{REF}$/64 |

### 5.3.1 ANALOG GAIN

The gain settings from 0.33x to 16x is done in the analog domain. This analog gain is placed on each ADC differential input. Each doubling of the gain improves the thermal noise due to sampling by approximately 3 dB, which means the lowest noise configuration is obtained when using the highest analog gain. The SNR, however, is degraded since the doubling the gain factor reduces the maximum allowable input signal amplitude by approximately 6 dB.

If the gain is set to 0.33x, the differential input range becomes theoretically ±3 * $V_{REF}$; however, the device does not support input voltages outside of the power supply voltage range. If large reference voltages are used with this gain, the input voltage range will be clipped between $A_{GND}$ and $AV_{DD}$ and therefore the output code span will be limited. This gain is useful when the reference voltage is small and when the input signal voltage is large.

The analog gain stage can be used to amplify very low signals, but the differential input range of the delta-sigma modulator must not be exceeded.

### 5.3.2 DIGITAL GAIN

When the gain setting is chosen from 16x to 64x, the analog gain stays constant at 16x and the additional gain is done in the digital domain, by a simple shift and round of the output code. The digital gain range is 1x to 4x. The output noise is approximately unchanged (outside from the quantization noise that is slightly decreased). The SNR is thus degraded by 6 dB per octave from 16x to 64x setting.

This digital gain is useful to scale-up the signals without using the host device (MCU) operations, but they degrade SNR and resolution (1 bit per octave) and do not significantly improve the noise performance, except for very large OSR settings.

## 5.4 Delta-Sigma Modulator

### 5.4.1 ARCHITECTURE

The Sigma-Delta ADC includes a second-order modulator with a multi-bit DAC architecture. Its 5-level quantizer is a Flash ADC composed of 4 comparators with equally spaced thresholds and a thermometer output coding. The proprietary 5-level architecture ensures minimum quantization noise at the outputs of the modulators without disturbing linearity or inducing additional distortion.

Unlike most multi-bit DAC architectures, the 5-level DAC used in this architecture is inherently linear and therefore does not degrade the ADC linearity and THD performance.

The sampling frequency is DMCLK, therefore, the modulator outputs are refreshed at a DMCLK rate.

Figure 5-2 represents a simplified block diagram of the delta-sigma modulator.

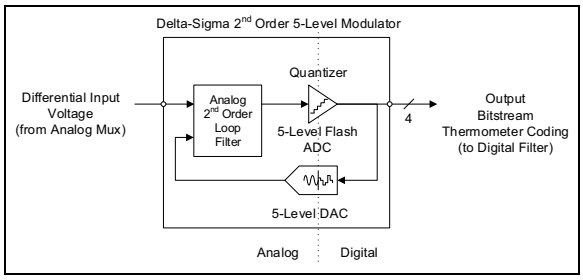



***FIGURE 5-2:*** *Simplified Delta-Sigma ADC Block Diagram.*

### 5.4.2 MODULATOR OUTPUT BLOCK

The modulator output option allows users to apply their own digital filtering on the output bit stream. By setting IRQ_MODE[1] = `1` in IRQ register, the modulator output is available at the IRQ/MDAT pin, at AMCLK rate and through ADCDATA Register (0x0) with DMCLK rate. With this configuration, the digital decimation filter is disabled in order to reduce the current consumption and no data ready interrupt is generated on any of the IRQ mechanisms. The IRQ/MDAT pin is never placed in high-impedance during the Modulator Output mode.

Since the Delta-Sigma modulator has a 5-level output given by the state of four comparators with thermometer coding, the output is represented using four bits, each bit represented the state of the corresponding comparator (see Table 5-4).

The comparator output bits are arranged serially at the AMCLK rate on the IRQ/MDAT output pin (see Figure 5-3).

This 1-bit serial bit stream is considered to be the same one as it is produced by a 1-bit DAC modulator with a sampling frequency of AMCLK. The modulator can either be considered as a 5 level-output at DMCLK rate or as 1-bit output at AMCLK rate. These two representations are inter-changeable. The MDAT outputs can therefore be used in any application that requires 1-bit modulator outputs. This application can be integrated with an external SINC Filter or more advanced decimation filters that are computed in the MCU or DSP device.

When CLK_SEL[1:0] = `11` (internal oscillator with external clock output), the AMCLK clock is present on the MCLK pin. This configuration permits to synchronize correctly the bitstream when the internal oscillator is used as the master clock source.

When CLK_SEL[1:0] = `00`, the modulator outputs are also synchronized with the MCLK input but the ratio between MCLK and AMCLK needs to be taken into account in the user applications to correctly retrieve the desired bitstream.

The default value of the bitstream after a reset or a power-up is `0011`. It is equivalent to a 0V input for the ADC. After each ADC reset and restart (see **Section 5.15 "A/D Conversions Automatic Reset and Restart Feature"**), the bitstream output is also reset and restart and the IRQ/MDAT is kept equal to logic high during the two MCLK clock periods needed for the synchronization. After these two clock periods, the bitstream will be provided on the IRQ/MDAT pin and the first value will be the default value.

**TABLE 5-4: DELTA-SIGMA MODULATOR OUTPUT BITSTREAM CODING**

| Comp[3:0] Code | Modulator Output Code (Decimal) | MDAT Serial Stream | Equivalent $V_{REF}$ Voltage |
|---|---|---|---|
| 1111 | +2 | 1111 | $+V_{REF}$ |
| 0111 | +1 | 0111 | $+V_{REF}/2$ |
| 0011 | 0 | 0011 | 0 |
| 0001 | -1 | 0001 | $-V_{REF}/2$ |
| 0000 | -2 | 0000 | $-V_{REF}$ |

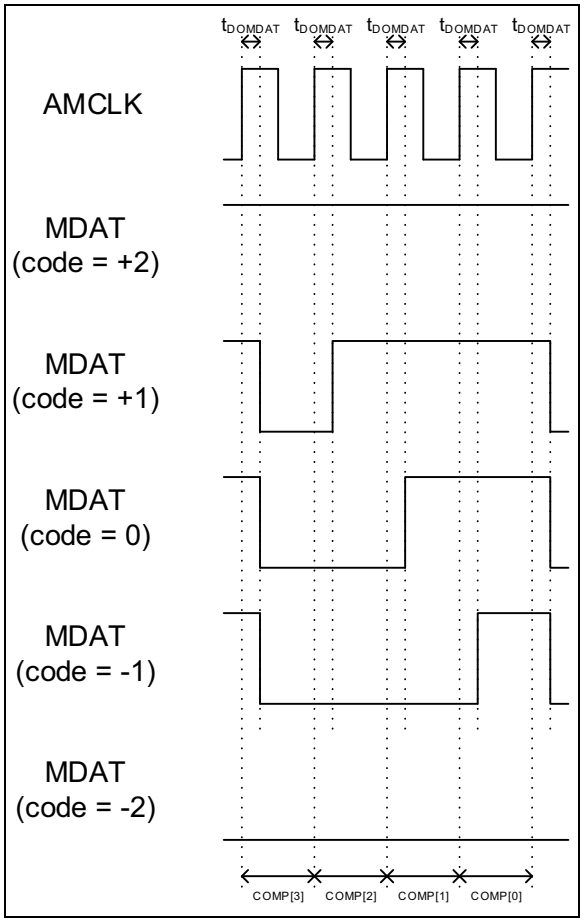



***FIGURE 5-3:*** *MDAT Serial Outputs in Function of the Modulator Output Code.*

### 5.4.3 BOOST MODES

The Delta-Sigma modulator includes a programmable biasing circuit, in order to further adjust the power consumption to the sampling speed applied through the MCLK. This can be programmed through the BOOST[1:0] bits in CONFIG2 register. The different BOOST settings are applied to the whole modulator circuit, including the voltage reference buffers. The settings of the BOOST[1:0] bits are described in Table 5-5.

**TABLE 5-5: BOOST SETTINGS DESCRIPTION**

| BOOST[1:0] | Bias current |
|---|---|
| 00 | x0.5 |
| 01 | x0.66 |
| 10 | x1 (default) |
| 11 | x2 |

The maximum achievable Analog Master Clock speed (AMCLK), the maximum sampling frequency (DMCLK), and the maximum achievable data rate (DRCLK) are highly dependent on the BOOST[1:0] and GAIN[2:0] settings. A higher BOOST setting will allow the circuits' bandwidth to be increased and will allow a higher analog master clock rate that will then increase the baseband of the input signals to be converted. The digital gain (that is enabled at 32x and 64x gains) has no influence on the achievable bandwidth.

A typical dependency of the bandwidth in relation with the GAIN for each BOOST setting combination is shown in Figure 2-20 to Figure 2-23. Typically a larger GAIN setting requires a higher BOOST setting, in order to achieve the same bandwidth performance.

Figure 2-24 shows the behavior of the achievable bandwidth at BOOST = 1x with $AV_{DD}$ corner cases. Since the BOOST settings vary the internal slew rate of the modulator components, using a lower $V_{REF}$ value will improve the bandwidth if low BOOST settings are used and are showing a limited bandwidth behavior.

## 5.5 Digital Decimation Filter

The decimation filter decimates the output bitstream of the modulator to produce 16-bit ADC output data. The decimation filter present in the device is a cascade of two filters: a third-order sinc filter with a decimation ratio of $OSR_3$ (third-order moving average of $3xOSR_3$ values), followed by a first order sinc filter with a decimation ratio of $OSR_1$ (moving average of OSR (third-order moving average of $3xOSR_3$ values) values).

Figure 5-6 represents the decimation filter architecture.

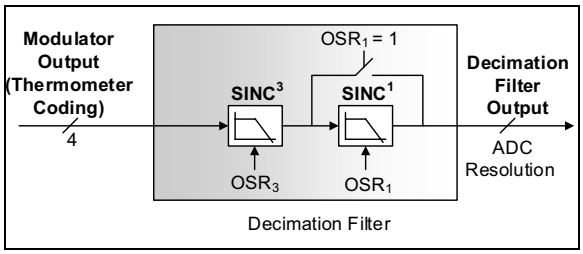



***FIGURE 5-4:*** *Decimation Filter Block Diagram.*

The following equation is the transfer function of the decimation filter:

**EQUATION 5-3: FILTER TRANSFER FUNCTION**

$$H(z) = \frac{\left(1 - z^{-OSR_3}\right)^3}{\left(OSR_3(1 - z^{-1})\right)^3} \times \frac{\left(1 - z^{-OSR_1 \times OSR_3}\right)}{OSR_1\left(1 - z^{-OSR_3}\right)}$$

Where:

$$z = exp\left(\frac{2\pi fj}{DMCLK}\right)$$

The resolution (number of possible output codes expressed in powers of two or in bits) of the digital filter is 16-bit maximum for any OSR ($OSR_3$ x $OSR_1$) and data format choice. The resolution depends only on the OSR through the OSR[3:0] settings in CONFIG1 per the Table 5-6. Once the OSR is chosen, the resolution is fixed and the output code of the ADC is encoded with the data format defined by the DATA_FORMAT[1:0] setting in CONFIG3.

The transfer function of this filter has a unity gain at each multiple of DMCLK. A proper anti-aliasing filter must be placed at the ADC inputs. This will attenuate the frequency contents around each multiple of DMCLK and keep the desired accuracy over the baseband of the converter. This anti-aliasing filter can be a simple first-order RC network with low time constant to provide a high rejection at DMCLK frequency.

The conversion time is a function of the OSR settings with the DMCLK frequency:

**EQUATION 5-4: CONVERSION TIME FOR OSR = OSR3 X OSR1**

$$T_{CONV} = \frac{(3 \times OSR_3 + (OSR_1 - 1) \times OSR_3)}{DMCLK}$$

In One-Shot mode, each conversion is launched individually so the maximum data rate is effectively $1/T_{CONV}$ if each conversion is launched with no delay. The digital filter is reset in between each conversion.

However, due to the nature of the digital filter (which memorizes the sum of the incoming bitstream), the data rate at the filter output can be maximized if the filter is never reset. Because of the internal resampling of the digital filter, the output data rate can be equal to DMCLK/OSR = DRCLK. This is the case in Continuous mode. In this case, the first conversion still happens in the $T_{CONV}$ time, as this is the settling time of the filter. The subsequent conversions are pipelined and give their output at a data rate of DRCLK. The Continuous Conversion mode can optimize the data rate, while consuming the same power as One Shot mode, which is advantageous in applications that require a continuous sampling of the analog inputs. The Continuous mode is not compatible with multiplexing the inputs (see **Section 5.14 "SCAN Mode"** for more details about the Conversion mode settings in MUX and SCAN modes).

Figure 5-5 shows the fundamental difference between One-Shot mode and Continuous mode in a simplified diagram.

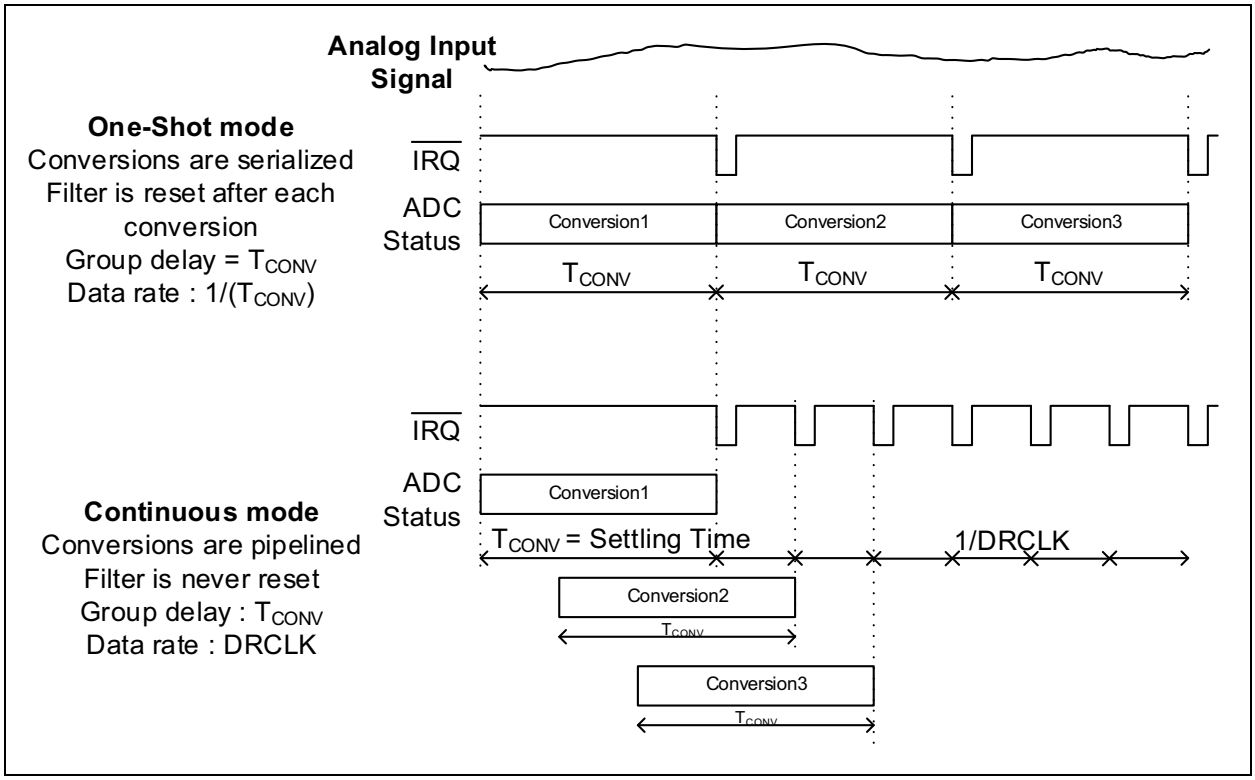



***FIGURE 5-5:*** *One-Shot Mode vs. Continuous Mode.*

Since the converter is effectively doing two conversions when AZ_MUX bit is enabled, the conversion time is equal to 2 x $T_{CONV}$ in this mode. As described in **Section 5.1.3 "ADC Offset cancellation algorithm"**, this selection is not compatible with the Continuous Conversion mode, therefore the output data rate is equal to 1/(2 x $T_{CONV}$) in this mode.

Table 5-6 summarizes the possible filter settings and their associated conversion time $T_{CONV}$, as well as their output data rate (DRCLK) in Continuous mode.

When OSR is larger than 20480, for typical master clock frequency MCLK = 4.9152 MHz, the device includes additional 50/60 Hz rejection by aligning decimation filter notches with multiple of 50 or 60 Hz depending on the OSR setting. The rejection band depends strongly on the master clock accuracy and corresponds to a first order decimation filter rejection rate.

The high OSR settings can be used for applications requiring very low noise and slow data rates.

Figure 5-6 shows the frequency response of the decimation filter with default settings. Figure 5-7 represents the frequency response of the filter with the highest OSR settings and a line rejection at 60 Hz.

**TABLE 5-6: OVERSAMPLING RATIO AND SINC FILTER RELATIONSHIP**

| OSR[3:0] | | | | $OSR_3$ | $OSR_1$ | Total OSR | ADC RESOLUTION (BITS NO. MISSING CODES) | Conversion Time ($T_{CONV}$) | Data rate in Continuous Conversion mode | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | Data Rate (Hz) with MCLK = 4.9152 MHz | Fastest Data Rate (Hz) with MCLK = 19.6608 MHz |
| 0 | 0 | 0 | 0 | 32 | 1 | 32 | 16 | 96/DMCLK | 38400 | 153600 |
| 0 | 0 | 0 | 1 | 64 | 1 | 64 | 16 | 192/DMCLK | 19200 | 76800 |
| 0 | 0 | 1 | 0 | 128 | 1 | 128 | 16 | 384/DMCLK | 9600 | 38400 |
| 0 | 0 | 1 | 1 | 256 | 1 | 256 | 16 | 768/DMCLK | 4800 | 19200 |
| 0 | 1 | 0 | 0 | 512 | 1 | 512 | 16 | 1536/DMCLK | 2400 | 9600 |
| 0 | 1 | 0 | 1 | 512 | 2 | 1024 | 16 | 2048/DMCLK | 1200 | 4800 |
| 0 | 1 | 1 | 0 | 512 | 4 | 2048 | 16 | 3072/DMCLK | 600 | 2400 |
| 0 | 1 | 1 | 1 | 512 | 8 | 4096 | 16 | 5120/DMCLK | 300 | 1200 |
| 1 | 0 | 0 | 0 | 512 | 16 | 8192 | 16 | 9216/DMCLK | 150 | 600 |
| 1 | 0 | 0 | 1 | 512 | 32 | 16384 | 16 | 17408/DMCLK | 75 | 300 |
| 1 | 0 | 1 | 0 | 512 | 40 | 20480 | 16 | 21504/DMCLK | 60 | 240 |
| 1 | 0 | 1 | 1 | 512 | 48 | 24576 | 16 | 25600/DMCLK | 50 | 200 |
| 1 | 1 | 0 | 0 | 512 | 80 | 40960 | 16 | 41984/DMCLK | 30 | 120 |
| 1 | 1 | 0 | 1 | 512 | 96 | 49152 | 16 | 50176/DMCLK | 25 | 100 |
| 1 | 1 | 1 | 0 | 512 | 160 | 81920 | 16 | 82944/DMCLK | 15 | 60 |
| 1 | 1 | 1 | 1 | 512 | 192 | 98304 | 16 | 99328/DMCLK | 12.5 | 50 |

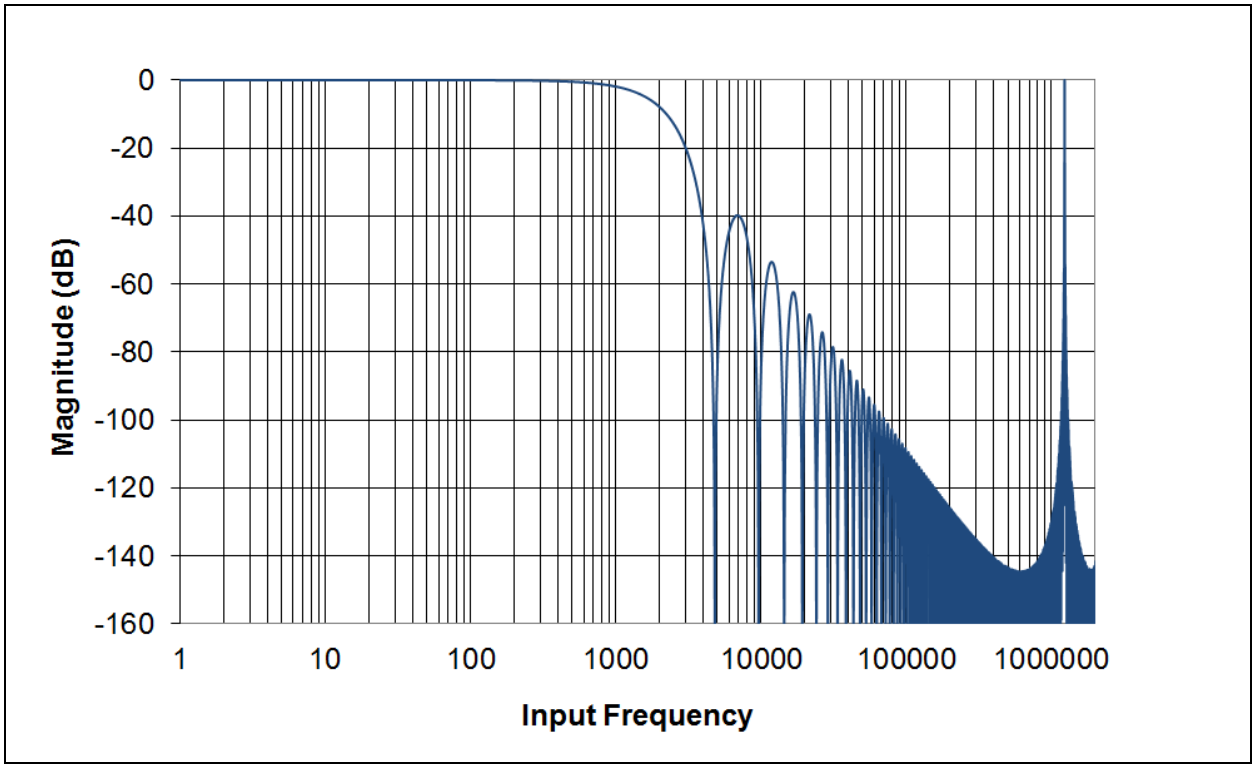



***FIGURE 5-6:*** *Decimation Filter Frequency Response (OSR = 256, PRE = 1:1, MCLK = 4.9152 MHz).*

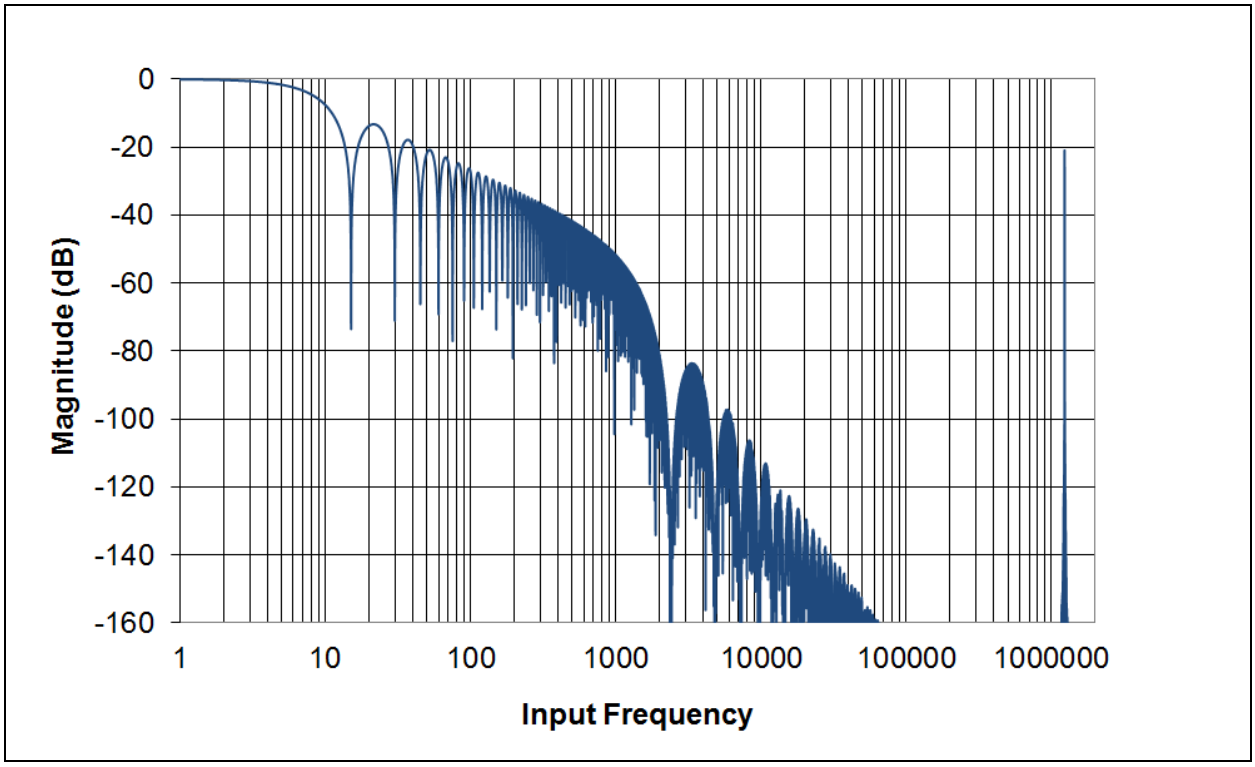



***FIGURE 5-7:*** *Decimation Filter Frequency Response (OSR = 81920, PRE = 1:1, MCLK = 4.9152 MHz)*

## 5.6 ADC Output Data Format

The ADC output data register (ADCDATA) is located at the address 0x0. The default length of the register is 16-bit (15-bit + SIGN).

Output data are calculated in the digital decimation filter with a larger resolution and are rounded to the closest LSB value. The rounding ensures a maximum 1/2 LSB error instead of a simple truncation that ensures a 1 LSB maximum error.

Equation 5-5 calculates ADC output code as a function of the input and reference signals for DC inputs.

**EQUATION 5-5: ADC OUTPUT CODE FOR DC INPUT (DATA FORMAT[1:0] = 00)**

$$ADC\_OUTPUT(LSB) = \left( \frac{V_{IN+} - V_{IN-}}{V_{REF+} - V_{REF-}} \right) \times 32768 \times GAIN$$

For AC sine wave inputs, the decimation filter transfer function (see Equation 5-3) induces an additional gain on the ADC output code which depends on the input frequency (roll-off of the decimation filter). For any inputs, the $V_{IN}+/V_{IN}-$ voltages are averaged out during the whole conversion time, since the ADC is an oversampling converter.

The ADC output format is set by DATA_FORMAT[1:0] bits in CONFIG3 register. These bits define four different possible formats for the ADC data output register: three 32-bit formats and one 16-bit format for MCP3461/2/4.

All possible data formats are described in Figure 5-8

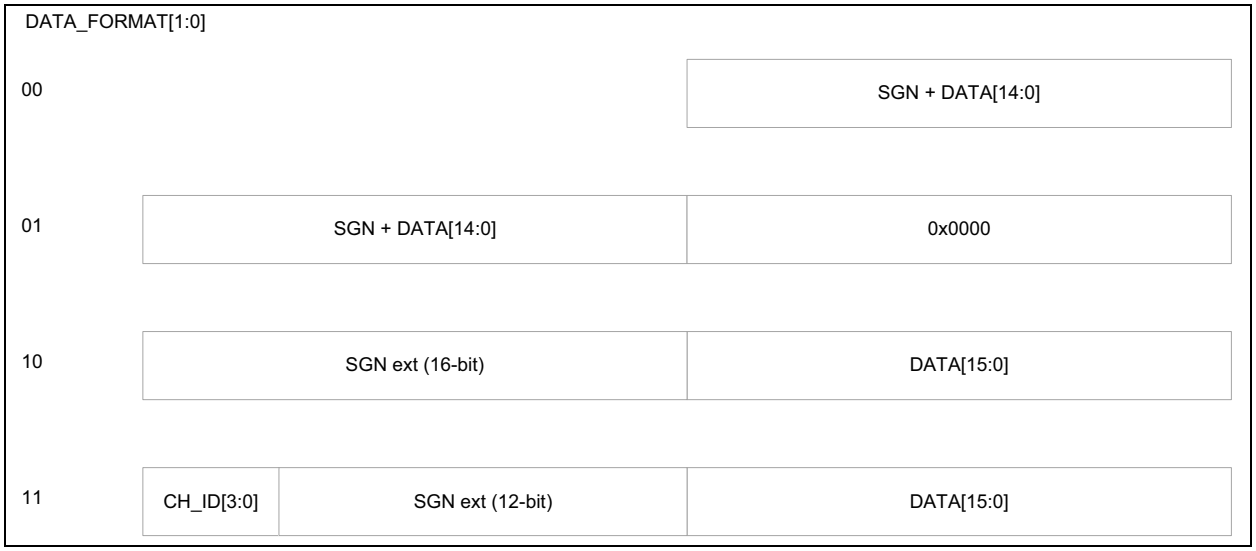



***FIGURE 5-8:*** *ADC Output Format Selection.*

When DATA_FORMAT[1:0] = 0X, the ADC resolution is 16-bit. The ADC output code is represented with MSB-first signed 2's complement coding. With these two data formats, the coding does not allow over-range: the equivalent analog input range is $[-V_{REF}; +V_{REF} - 1\ LSB]$.
When $V_{IN} * Gain > V_{REF} - 1\ LSB$, the 16-bit ADC code (SGN + DATA[14:0]) will saturate and be locked at 0x7FFF.
When $V_{IN} * Gain < -V_{REF}$, the 16-bit ADC code will saturate and be locked at 0x8000. Using these data formats does not permit to correctly evaluate full-scale errors in case of a positive full-scale error.

When DATA_FORMAT[1:0] = 00, the output register shows only the 16-bit value.

When DATA_FORMAT[1:0] = 01, the output register is 32-bit long and the output code is padded with additional zeros on the last two bytes. The output code is left-justified in this case. This format is useful for 32-bit MCU applications.

When DATA_FORMAT[1:0] = 1X, the ADC data are represented on 17 bits. For these two data formats, the output register is 32-bit long. With these two data formats, the coding allows the equivalent analog input range is [-2 * $V_{REF}$; +2 * $V_{REF}$ - 1 LSB]. When $V_{IN}$ * Gain > 2 $V_{REF}$ - 1 LSB, the 17-bit ADC code (SIGN +DATA[15:0]) saturates and locks at 0x0FFFF. When $V_{IN}$ * Gain < -2 $V_{REF}$, the 17-bit ADC code will saturate and be locked at 0x10000. Using these data formats permits to correctly evaluate full-scale errors in case of a positive full-scale error since they allow inputs that can be greater than $V_{REF}$ or less than -$V_{REF}$.

The ADC accuracy is not maintained on the full extended [-2 * $V_{REF}$, +2 * $V_{REF}$ - 1 LSB] range but only on a smaller range that is approximately equal to ±1.05 * $V_{REF}$. This over-range can be useful in high-side measurements and gain error cancellation algorithms. The over range capable formatting on 17 bits is fully compatible with the standard code locked formatting on 16 bits: both coding formats produce the same 16-bit codes for the [-$V_{REF}$; +$V_{REF}$ - 1 LSB] range and the MSB on the 17-bit coding can be considered as a simple sign-bit extension.

When DATA_FORMAT[1:0] = 10, the 17-bit (16-bit plus SGN) value is right-justified. The first two bytes of the 32-bit ADC output code will repeat the sign bit (SGN).

In DATA_FORMAT[1:0] = 11, the output code is similar to the DATA_FORMAT[1:0] = 10. The only difference resides in the four MSBs of the first byte, which are no longer repeats of the sign bit (SGN). They are the channel ID data (CH_ID[3:0]) that is defined in Table 5-13. This CH_ID[3:0] word can be used to verify that the right channel has been converted in SCAN mode and can serve easy data retrieval and logging (see **Section 5.14 "SCAN Mode"** for more details about the SCAN mode). In MUX mode, this 4-bit word is defaulted to '0000' and does not vary with the MUX[7:0] selection. This format is useful for 32-bit MCU applications.

**TABLE 5-7: DATA_FORMAT[1:0] = 0X (16-BIT CODING)**

| Equivalent Input Voltage | ADC Output Code (SGN + DATA[14:0]) | Hexadecimal | Decimal |
|---|---|---|---|
| >$V_{REF}$ - 1 LSB | 0111111111111111 | 0x7FFF | +32767 |
| $V_{REF}$ - 2 LSB | 0111111111111110 | 0x7FFE | +32766 |
| 1 LSB | 0000000000000001 | 0x0001 | +1 |
| 0 | 0000000000000000 | 0x0000 | 0 |
| -1 LSB | 1111111111111111 | 0xFFFF | -1 |
| -$V_{REF}$ + 1 LSB | 1000000000000001 | 0xFFFF | -32767 |
| <-$V_{REF}$ | 1000000000000000 | 0x8000 | -32768 |

**TABLE 5-8: DATA_FORMAT[1:0] = 1X (17-BIT CODING)**

| Equivalent Input Voltage | ADC Output Code (SGN+DATA[15:0]) | Hexadecimal | Decimal |
|---|---|---|---|
| >2 $V_{REF}$ - 1 LSB | 01111111111111111 | 0x0FFFF | +65535 |
| 2 $V_{REF}$ - 2 LSB | 01111111111111110 | 0x0FFFE | +65534 |
| $V_{REF}$ + 1 LSB | 01000000000000001 | 0x08001 | +32769 |
| $V_{REF}$ | 01000000000000000 | 0x08000 | +32768 |
| $V_{REF}$ - 1 LSB | 00111111111111111 | 0x07FFF | +32767 |
| $V_{REF}$ - 2 LSB | 00111111111111110 | 0x07FFE | +32766 |
| 1 LSB | 00000000000000001 | 0x00001 | +1 |
| 0 | 00000000000000000 | 0x00000 | 0 |
| -1 LSB | 11111111111111111 | 0x1FFFF | -1 |
| -$V_{REF}$ + 1 LSB | 11000000000000001 | 0x18001 | -32767 |
| -$V_{REF}$ | 11000000000000000 | 0x18000 | -32768 |
| -$V_{REF}$ - 1 LSB | 10111111111111111 | 0x17FFF | -32769 |
| -2 $V_{REF}$ - 1 LSB | 10000000000000001 | 0x10001 | -65535 |
| <-2 $V_{REF}$ | 10000000000000000 | 0x10000 | -65536 |

## 5.7 Power-on Reset

The analog and digital power supplies are monitored separately by two Power-on-Reset (POR) monitoring circuits at all times except during Full-Shutdown mode (see **Section 5.9 "Full-Shutdown Mode"**).

Each POR circuit has two separate thresholds, one for the rising voltage supply, and one for falling voltage supply. They both include hysteresis (the rising threshold is superior than the falling threshold) so that the device is tolerant to a certain degree of transient noise on each power supply.

If any of the two power supply voltages is below its respective threshold, the POR state is forced internally. In this state, the SPI interface is disabled, no command can be executed by the chip. All registers are cleared and set to their default values.

At power-up, when both power supply voltages are above the rising thresholds, the device powers up and the SPI interface is enabled and can handle communications. Since both thresholds need to be crossed for the power-up, the power-up sequence is not important and any power supply voltage can ramp up first. The detection time for the monitoring circuits ($t_{POR}$) is about 1 µs for relatively fast power-up ramp rates. The normal operation is stopped when any of the falling thresholds of the two POR monitoring circuits is crossed. Figure 5-9 illustrates the power-up and power-down sequences.

If the $\overline{CS}$ pin is kept logic low during a POR state, a logic high pulse is necessary to start the first communication sequence. The $\overline{CS}$ rising edge will reset properly the SPI interface and the falling edge will clear the POR interrupt on the $\overline{IRQ}$ pin (see Figure 6-16).

During Full-Shutdown mode, the power supply voltages are not monitored to be able to reach ultra-low power consumption. The device cannot generate a POR event interrupt in this mode except in cases of extremely low power supply voltages.

The $DV_{DD}$ and $AV_{DD}$ monitoring thresholds are different since their respective voltage ranges are different. The $AV_{DD}$ rising threshold is approximately 1.75V ± 10% and the $DV_{DD}$ is 1.2V ± 10%. The hysteresis is approximately 150 mV (typical). Proper decoupling ceramic capacitors (0.1 µF and 10 µF ceramic) should be placed as close as possible to the power supply pins ($AV_{DD}$, $DV_{DD}$), to provide additional transient immunity.

In order to ensure a proper power-up sequence, the ramp rate of $DV_{DD}$ should not exceed 3V/us when coming out of the POR state.

Additionally, the user needs to lower the $DV_{DD}$ residual voltage as much as possible, close to 0V, when the device is kept for a long time in a POR state (below $DV_{DD}$ POR threshold), in order to ensure a proper power-up sequence. The user can verify if the power-up sequence has been correctly performed by reading the default state of all the registers in the register map, just after powering up the device. If one or more of the registers do not show the proper default setting, a new power-up cycle should be launched to recover from this condition.

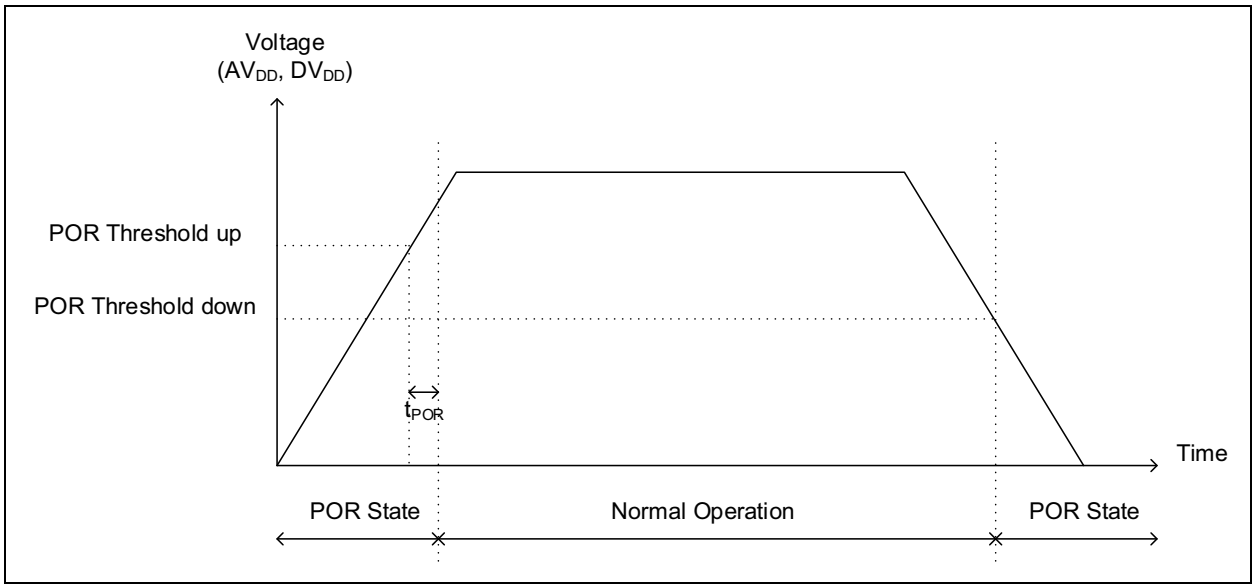



***FIGURE 5-9:*** *Power-On Reset Timing Diagram.*

## 5.8 ADC Operating Modes

The ADC can be placed into three different operating modes: Shutdown, Standby and Conversion. The ADC operating mode is controlled directly by the user using the ADC_MODE[1:0] bits in the CONFIG0 register. The user can directly launch conversions or place the ADC into Shutdown or Standby mode by writing directly to these bits. Additional fast commands are available for each of the three possible states of these bits to allow faster programming in case of time-sensitive applications (see **Section 6.2.4 "Command type bits (CMD[1:0])"**). The different ADC_MODE[1:0] settings available are described in Table 5-9.

The ADC_MODE[1:0] bits do not give an instantaneous representation of the state of the ADC: writing the ADC_MODE[1:0] bits set the desired state of the ADC but this state is only attained after a start-up time depending on the current state of the ADC. See **Section 5.10 "ADC Start-up Timer"** for details about the Start-up timer. Typically, the device starts in Shutdown mode after a POR (ADC_MODE[1:0] = 00 by default). To launch conversions in the desired configuration, the user should program the part in the desired configuration and then set the ADC_MODE[1:0] to 11. In this case the first conversion will start after $T_{ADC\_SETUP}$ = 256 DMCLK periods. This time is necessary for the part to adjust to the new settings programmed and settle to its operating point to convert accurately the input signals.

Internally, the device tracks the current state of the ADC as well as the start-up timer counter to be able to optimize the start up time depending on the desired transitions and internal configurations required and set by the user.

In MUX mode, overwriting the ADC_MODE[1:0] bits to 11 when the ADC is already in conversion, resets and restarts the current conversion immediately. The conversion start pulse will also be regenerated in this case if the EN_STP bit was enabled.

In SCAN mode (see **Section 5.14 "SCAN Mode"**), writing the ADC_MODE[1:0] to 11 starts the conversion SCAN cycle. During the whole cycle, even when the scan TIMER is enabled, reading the ADC_MODE[1:0] bits will give a '11' code output meaning that the SCAN cycle is on-going. Rewriting ADC_MODE[1:0] = 11 during SCAN mode will reset and restart immediately the whole SCAN sequence from the beginning of the sequence. The conversion start pulse will also be regenerated in this case if the EN_STP bit was enabled. The restart of the SCAN sequence may induce a $T_{ADC\_SETUP}$ additional delay if the ADC was effectively in Shutdown mode when the ADC_MODE bits are overwritten (this can happen if the ADC_MODE bits are overwritten during the TIMER delay period where the ADC is placed into shutdown in between two SCAN cycles).

The ADCDATA register is always updated with the last conversion results only. The ADCDATA register cannot provide incomplete conversion results. The A/D conversion needs to be completed to be able to provide a result in the ADCDATA register. Each end of conversion generates a data ready interrupt on all three IRQ mechanisms (see **Section 6.8.1 "Conversion data ready interrupt"**). The ADCDATA register is never cleared when the device transitions from one mode to another. The only way to clear the ADC output register is a POR event or a full Reset Fast command. See **Section 6.2.5 "Fast Commands description"**.

**TABLE 5-9: ADC OPERATING MODES DESCRIPTION**

| ADC_MODE[1:0] | ADC Mode | Description |
|---|---|---|
| 11 | Conversion | The ADC is placed into Conversion mode and consumes the specified current (see **Table "Electrical Characteristics"**). A/D conversions can be reset and restarted immediately once this mode is effectively reached. This mode maybe reached after a maximum of $T_{ADC\_SETUP}$ time depending of the current state of the ADC. |
| 10 | Standby | Conversions are stopped. ADC is placed into Reset but consumes almost as much current as in Conversion mode. A/D conversions can start immediately once this mode is effectively reached. This mode may be reached after a maximum of $T_{ADC\_SETUP}$ time depending of the current state of the ADC. |
| 0X | Shutdown | Conversions are stopped. ADC is placed into shutdown consuming no current. A/D conversions can start only after $T_{ADC\_SETUP}$ start-up time. This mode is effective immediately after programmed. |

## 5.9 Full-Shutdown Mode

The part incorporates an Ultra Low-Power mode called Full-Shutdown mode, where none of the internal circuits consume DC power. This mode is enabled when all device peripherals are placed into shutdown, which means 0x00 on the CONFIG0 register bits.

To enter Full-Shutdown mode by writing CONFIG0 register, the bits CONFIG0[7:6] must be set to '00'. Otherwise, these bits should be set to '11' to ensure correct functioning of the ADC. The Full-Shutdown mode can also be accessed through a specific Fast command (Fast command code = 0x1101).

When this condition happens, the POR monitoring circuits are placed in shutdown, therefore $AV_{DD}$ and $DV_{DD}$ are no more monitored.

The part can still be accessed through SPI interface during this mode and will accept incoming SPI commands. The ADCDATA register is not cleared during Full-Shutdown mode and still holds previous conversion results. The other configuration register settings are not modified or reset due to the entering in Full-Shutdown mode.

When the ADC_MODE[1:0] bits are temporarily set internally to '00' during SCAN mode in between SCAN cycles, the part does not go into Full-Shutdown mode, even if all the other bits in the CONFIG0 register are set to '0'.

The Full-Shutdown mode stops all internal timers and resets them.

The user should place all digital inputs to a static value (logic low or high) in order to optimize power consumption during Full-Shutdown mode. The current consumption specifications during Full-Shutdown mode are intended without any digital pin toggling during the measurement. In this case only leakage current is consumed throughout the device and this current varies exponentially with respect to absolute temperature.

## 5.10 ADC Start-up Timer

The device includes an intelligent start-up timer circuit for the ADC, which ensures before each conversion start that the ADC is properly biased and that internal nodes are properly settled. This timer ensures the proper conditions so that the ADC can convert with its full accuracy for each conversion.

The ADC can operate in three different modes: Shutdown, Standby and Conversion, as described in **Section 5.8 "ADC Operating Modes"**. The start-up timer of the ADC manages the time for the transitions between each mode. These transitions can be instantaneous or can take a maximum of 256 DMCLK periods depending on the type of transition and the current status of the ADC and of the internal start-up timer.

The timer will always try to reduce the transition time from one state to another but will also allow enough time for the internal circuitry to settle to the proper internal operating points.

The transitions from Standby or Conversion mode to Shutdown mode are always immediate. They reset the internal start-up timer to 256 DMCLK periods ($T_{ADC\_SETUP}$).

The transitions from Shutdown to Standby or Conversion mode starts the internal start-up timer, which decrements from 256 to 0. The timer only decrements after a small delay of 2 MCLK periods in case of a transition caused by an SPI command. This small delay is necessary to overcome any possible synchronization issue between the two asynchronous clocks MCLK and SCK. The timer immediately decrements (without the synchronization delay) if the transitions are generated by the internal state machine (for example, when the transitions are generated by the SCAN sequence). Once the timer reaches 0 (when the user has clocked 256 DMCLK periods), the device reaches its internal proper operating points and will either stay in Standby mode (if ADC_MODE[1:0] = 10) or start the Conversion mode (if ADC_MODE[1:0] = 11).

The transition from Standby to Conversion and vice-versa is immediate once the timer has reached 0 (if ADC_MODE[1:0] = 11). If the transition from Standby to Conversion happens and if the timer has not yet reached 0, the timer will continue to decrement to '0' before starting the conversion. The timer cannot decrement faster than 256 DMCLK periods when the ADC transitions from Shutdown mode to Conversion mode (from Shutdown mode, the ADC is allowed 256 DMCLK periods to power-up and settle to its desired operating point before starting conversions). The start-up time has been sized at 256 DMCLK clock periods for the part to be able to settle in all conditions and with all possible clock frequencies as specified. Table 5-10 summarizes the behavior of the internal start-up timer as a function of the ADC_MODE[1:0] settings.

**TABLE 5-10: ADC START-UP TIMER BEHAVIOR AS A FUNCTION OF ADC_MODE[1:0] SETTINGS**

| ADC_MODE[1:0] | ADC State | ADC Start-up Timer Behavior |
|---|---|---|
| 11 | Conversion | ADC start-up timer decrements to 0. Once reached 0, conversion starts. |
| 10 | Standby | ADC start-up timer decrements to 0. Once reached 0, ADC is ready to convert. |
| 0X | Shutdown | ADC Start-up Timer is reset to $T_{ADC\_SETUP}$ = 256. |

Rewriting ADC_MODE[1:0] bits without changing the bit settings does not modify the internal timer and cannot shorten the start-up delay necessary to start accurate conversions. A synchronization delay of 2 MCLK periods to happen after each rewrite if ADC_MODE[1:0] = 1X.

In SCAN mode, when CONV_MODE[1:0] = 11 (Continuous mode), the ADC may be placed into shutdown and restarted in between each SCAN cycle depending on the TIMER[23:0] settings (see **Section 5.14.5 "Delay between each SCAN cycle (TIMER[23:0])"**). If the TIMER register is programmed with a decimal code greater than $T_{ADC\_SETUP}$ = 256, the internal timer will place the part into Shutdown mode automatically at the end of the cycle and will start to transition to the next cycle 256 DMCLK periods before the end of the TIMER delay.

This lowers the power consumed during the TIMER delay as much as possible. If the value of the TIMER delay is less than 256 DMCLK periods, the part will not go into shutdown and stay in standby during the TIMER delay (in this case the power consumed is equivalent to the Conversion mode power consumption).

In order to be sure to catch the start of the conversion in case of complex sequences of transitions, it can be useful to enable the EN_STP bit so that the part will generate a pulse on the $\overline{IRQ}$ pin to indicates a conversion start.

Figure 5-10 shows different cases for the transitions between each mode and shows the internal state of the start-up timer at each step.

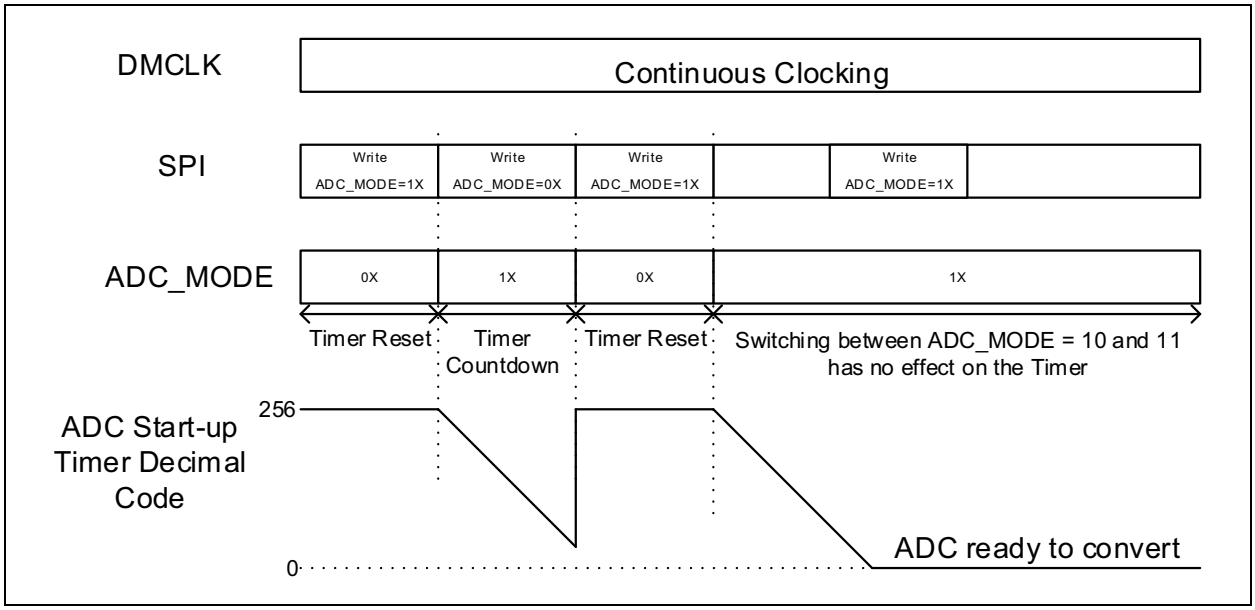



***FIGURE 5-10:*** *ADC Start-Up Timer Timing Diagram.*

## 5.11 Master Clock Selection/Internal Oscillator

The device includes three possible clock modes for the master clock generation. The Master Clock (MCLK) is used by the ADC to perform conversions and is also used by the digital portion to generate the different digital timers. The clock mode selection is made through the CLK_SEL[1:0] bits located in the CONFIG0 register. The possible selections are described in detail in Table 5-12.

The Master Clock is not propagated in the chip when the chip is placed into the Full-Shutdown mode (see **Section 5.9 "Full-Shutdown Mode"**). Any change to the CLK_SEL bits creates a reset and restart for the currently running conversions and a restart of the ADC setup timer. Each reset and restart will reset all internal phases to their default values and can lead to a possible temporary duty cycle change at the clock output pin.

**TABLE 5-11: CLOCK SELECTION BITS**

| CLK_SEL[1:0] | Clock Mode | MCLK Pin |
|---|---|---|
| 00 or 01 | External clock | MCLK digital input |
| 10 | Internal RC Oscillator no clock output | High-Z |
| 11 | Internal RC Oscillator with clock output | AMCLK digital output |

### 5.11.1 EXTERNAL MASTER CLOCK MODE (CLK_SEL[1:0]=0X)

The External Clock mode is used to input the MCLK clock necessary for the ADC conversions and can accept duty cycles with a large range since the clock is redivided internally to generate the different internal phases.

The external clock can be provided on the MCLK pin for the MCP3461/2/4 devices.

### 5.11.2 INTERNAL OSCILLATOR

The device includes an internal RC-type oscillator powered by the digital power supply ($DV_{DD}/D_{GND}$).The frequency of this internal oscillator ranges from 3.3 to 6.6 MHz. The oscillator is not trimmed in production, therefore, the precision of the center frequency is approximately ±30% from chip-to-chip. The duty cycle of the internal oscillator is centered around 50% and varies very slightly from chip-to-chip. The internal oscillator has no Reset feature and is always running once selected.

### 5.11.3 INTERNAL MASTER CLOCK MODES (CLK_SEL[1:0]=1X)

When CLK_SEL[1] = 1, the internal oscillator is selected and the master clock is generated internally. The internal oscillator has no Reset feature and is always running once selected. The master clock generation is independent of the ADC, as the clock can still be generated even if the ADC is in Shutdown mode. The internal oscillator is only disabled when CLK_SEL[1:0] = 0X. The clock can be distributed to the dedicated output pin depending on the CLK_SEL[0] bit. When the clock output is selected (CLK_SEL[0] = 1), the AMCLK clock derived from the MCLK (AMCLK=MCLK/PRESCALE) is available on the output pin. The AMCLK output can serve as the clock pin to synchronize either the modulator output or other MCP3461/2/4 devices that would be configured with CLK_SEL[1:0] = 00 or 01.

The AMCLK output is available on the MCLK clock output pin as soon as the write command (CLK_SEL[1:0] = 11) is finished.

## 5.12 Digital System Offset And Gain Calibrations

The MCP3461/2/4 devices include a digital calibration feature for offset and gain errors. The calibration scheme for offset error consists of the addition of a fixed offset value to the ADC output code (ADCDATA at address 0x0). The offset value added (OFFSETCAL) is determined in the OFFSETCAL register (address 0x9). The calibration scheme for gain error consists of the multiplication of a fixed gain value to the ADC output code. The gain value (GAINCAL) multiplied is determined in the GAINCAL register (address 0xA).

The digital offset and gain calibration schemes are enabled or disabled via the EN_OFFCAL and EN_GAINCAL control bits of the CONFIG3 register. When both calibration control bits are enabled (EN_OFFCAL = EN_GAINCAL = 1), the ADCDATA register contents are modified with the digital offset and gain calibration schemes as described in Equation 5-6. When a calibration enable bit is off, its corresponding register becomes a don't care register and the corresponding calibration is not performed.

**EQUATION 5-6: ADCDATA OUTPUT AFTER DIGITAL GAIN AND OFFSET ERROR CALIBRATION**

**ADCDATA (post-cal)**
**= (ADCDATA (pre-cal) + OFFSETCAL) * GAINCAL**

The calculations are performed internally with proper management of overloading, so that the overload detection is done on the output result only and not on the intermediate results. A sufficient number of

additional overload bits are maintained and propagated internally to overcome all possible overload and/or overload recovery situations.

For example, if ADCDATA (pre-cal) + OFFSETCAL is out of bounds but (ADCDATA (pre-cal) + OFFSETCAL) * GAINCAL is still in the right range (possible with 0 < GAINCAL < 1), then the result is not saturated.

### 5.12.1 DIGITAL OFFSET ERROR CALIBRATION

The offset calibration register (OFFSETCAL, address 0x9) is a signed MSB-first 2s complement coding 24-bit register that holds the digital offset calibration value OFFSETCAL. The OFFSETCAL equivalent input voltage value is calculated with Equation 5-7.

**EQUATION 5-7: OFFSETCAL CALIBRATION VALUE (EQUIVALENT INPUT VOLTAGE)**

**OFFSETCAL (V) =**
**$V_{REF}$ * (OFFSETCAL[23:8])/(32768 * GAIN)**

For the MCP3461/2/4 devices, the offset calibration is done by adding bit-by-bit the OFFSETCAL[23:8] calibration value to the ADCDATA code. The last byte of the OFFSETCAL register (OFFSETCAL[7:0]) is ignored and internally reset to 0x00 during the calibration, therefore the addition just takes in account the OFFSETCAL[23:8] bits and is done bit-by-bit with the ADC output code.

The offset calibration value range in equivalent voltage is [$-V_{REF}$/GAIN; ($+V_{REF}$ - 1LSB)/GAIN] which permits to cancel any possible offset in the ADC but also in the system. The offset calibration is realized with a simple 16-bit signed adder and is instantaneous (no pipeline delay). Enabling the offset calibration will affect the next conversion result: the conversion result already held in the ADCDATA output register (0x0) is not modified when the EN_OFFCAL is set to '1', but the next one will take in account the offset calibration. Changing the OFFSETCAL register to a new value will not affect the current ADCDATA value but the next one (after a data ready interrupt) will take in account the new OFFSETCAL value. Figure 5-11 shows the different cases and their implication on the ADCDATA register as well on the $\overline{IRQ}$ output.

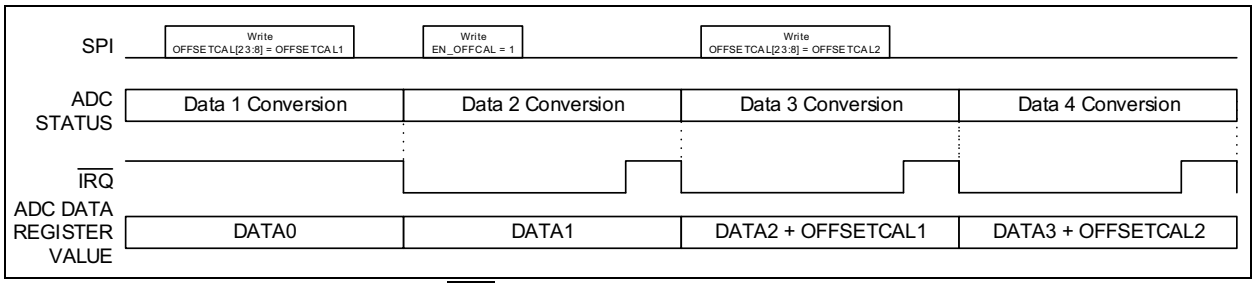



***FIGURE 5-11:*** *ADC Output and $\overline{IRQ}$ Behavior with Digital Offset Calibration Enabled.*

### 5.12.2 DIGITAL GAIN ERROR CALIBRATION

The gain error calibration register (GAINCAL, address 0xA) is a unsigned 24-bit register that holds the digital gain error calibration value GAINCAL. The GAINCAL multiplier is calculated with Equation 5-8.

**EQUATION 5-8: GAINCAL CALIBRATION VALUE (MULTIPLIER VALUE)**

**GAINCAL (V/V) = (GAINCAL[23:8] unsigned decimal code) / 32768**

For the MCP3461/2/4 devices, the gain error calibration is done by multiplying the GAINCAL value to the ADC output code. The last byte of the GAINCAL register (GAINCAL[7:0]) is ignored and internally reset to 0x00 during the calibration, therefore the multiplication just takes in account the GAINCAL[23:8] bits. The gain error calibration value range in equivalent voltage is [0; $2\text{-}2^{-15}$] which permits to cancel any possible gain error in the ADC but also in the system. The gain error calibration is realized with a simple add-and-shift circuit clocked on DMCLK and is inducing a pipeline delay of $T_{GCAL}$ = 15 DMCLK periods. This pipeline delay acts as a delay on the data ready interrupt position that is shifted by $T_{GCAL}$ = 15 DMCLK periods. During this delay, the converter can process the next conversion, the delay does not shift the next conversion and does not change the conversion time $T_{CONV}$. Enabling the gain error calibration will affect the next conversion result: the conversion result already held in the ADCDATA output register (0x0) is not modified when the EN_GAINCAL is set to 1, but the next one will take in account the offset calibration. Changing the GAINCAL register to a new value will not affect the current ADCDATA value but the next one (after a data ready interrupt) will take into account the new GAINCAL value. Figure 5-12 details the different cases and their associated effects to the ADCDATA register and the $\overline{IRQ}$ output.

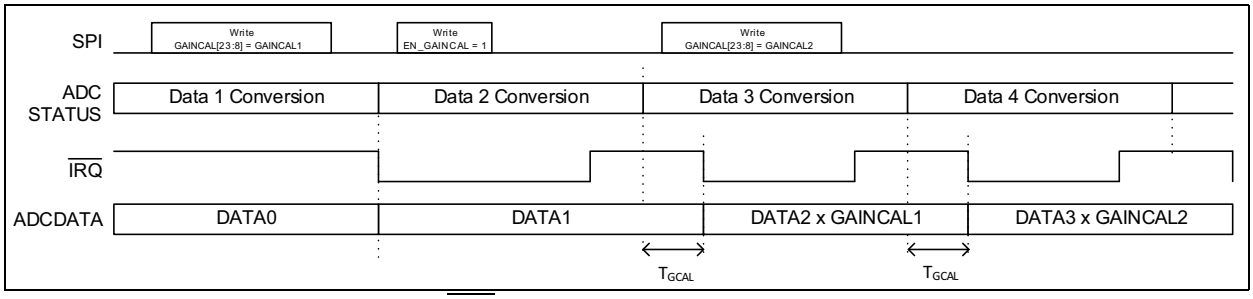



***FIGURE 5-12:*** *ADC Output and $\overline{IRQ}$ Behavior with Digital Gain Error Calibration Enabled.*

## 5.13 Conversion modes

The ADC includes several Conversion modes that can be selected through the CONV_MODE[1:0] bits located in the CONFIG3 register. The behavior of the ADC with respect to these bits depends on whether the ADC is in MUX or SCAN mode. Table 5-12 summarizes the possible configurations.

**TABLE 5-12: ADC CONVERSION MODES IN MUX OR SCAN MODES**

| CONV_MODE[1:0] | ADC Behavior (MUX mode) | ADC Behavior (SCAN Mode) | ADC_MODE[1:0] bit settings |
|---|---|---|---|
| 0X | Performs a One-shot conversion and returns automatically to Shutdown mode. | Performs one complete SCAN cycle and returns automatically to Shutdown mode. | Returns to 0X after one conversion (MUX mode) or one SCAN cycle (SCAN mode). |
| 10 | Performs a One-shot conversion and returns automatically to Standby mode. | Performs one complete SCAN cycle and returns automatically to Standby mode. | Returns to 10 after one conversion (MUX mode) or one SCAN cycle (SCAN mode). |
| 11 | Performs continuous conversions. | Performs continuously SCAN cycles with TIMER[23:0] delay between each cycle. | Stays at 11 continuously. |

### 5.13.1 CONVERSION MODES IN MUX MODE

In MUX mode, the user can choose between One-shot and Continuous Conversions.

A One-shot Conversion is a single conversion and takes a certain conversion time $T_{CONV}$ (or 2 x $T_{CONV}$ when AZ_MUX = 1, see **Section 5.1.3 "ADC Offset cancellation algorithm"**). Once this conversion is performed, the part returns automatically to a standby or shutdown state depending on the CONV_MODE[1:0] bit settings. The Conversion mode determined by the CONV_MODE[1:0] bits settings will also affect the state of the ADC_MODE[1:0], as described in Table 5-12.

The conversion can be preceded by a start-up time that depends on the state of the ADC (see **Section 5.10 "ADC Start-up Timer"**). In One-Shot mode, the ADC data has to be completely read with the SPI interface for the interrupt to be cleared on the $\overline{IRQ}$ pin (the $\overline{IRQ}$ pin cannot be automatically cleared like in the Continuous Conversion mode).

This mode is recommended for low-power, low-bandwidth applications, requiring once in a while one A/D conversion.

In the Continuous Conversion mode, the ADC is never placed in Standby or Shutdown mode and converts continuously without any internal Reset. In this mode, the output data rate of the ADC is defined by DRCLK (see Figure 5-5). The digital decimation filter induces a pipeline or group delay of $T_{CONV}$ for the first data ready and is structured to give a continuous stream of data at the DRCLK rate after this first data ready (the internal registers of the filter are never reset in this mode thus the decimation filter acts as a moving average). Each data ready interrupt corresponds to a valid and complete conversion that has processed through the digital filter (the digital filter has no latency in this respect). This mode permits to have a much faster data

rate than the one-shot mode and is, therefore, recommended for higher bandwidth applications. The pipeline delay should be carefully determined and adapted to the user needs especially in closed-loop low-latency applications. This mode is recommended for applications requiring continuous sampling/averaging of the input signals. If AZ_MUX = 1, the Continuous Conversion mode is replaced by a series of subsequent One-shot mode conversions, with a reset between each conversion. This makes the group delay equal to 2 * $T_{CONV}$ and data rate equal to 1/(2 * $T_{CONV}$).

Figure 5-13 and Figure 5-14 detail one-shot and Continuous Conversion modes for MUX mode.

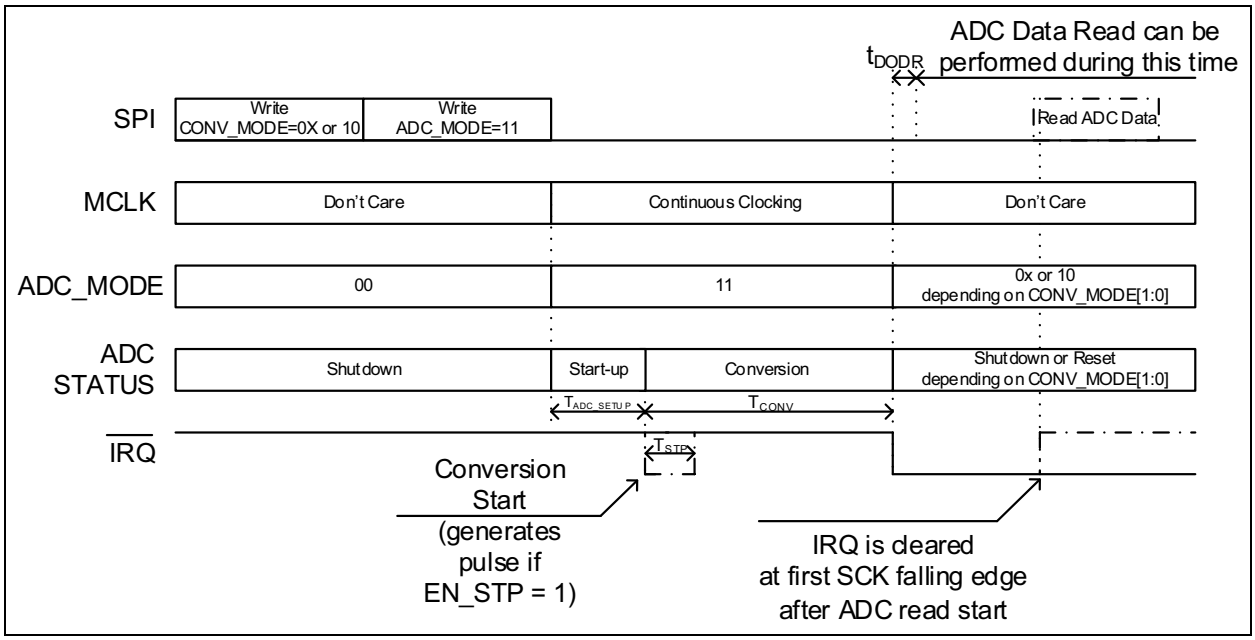



***FIGURE 5-13:*** *MUX One-Shot Conversion Mode Timing Diagram.*

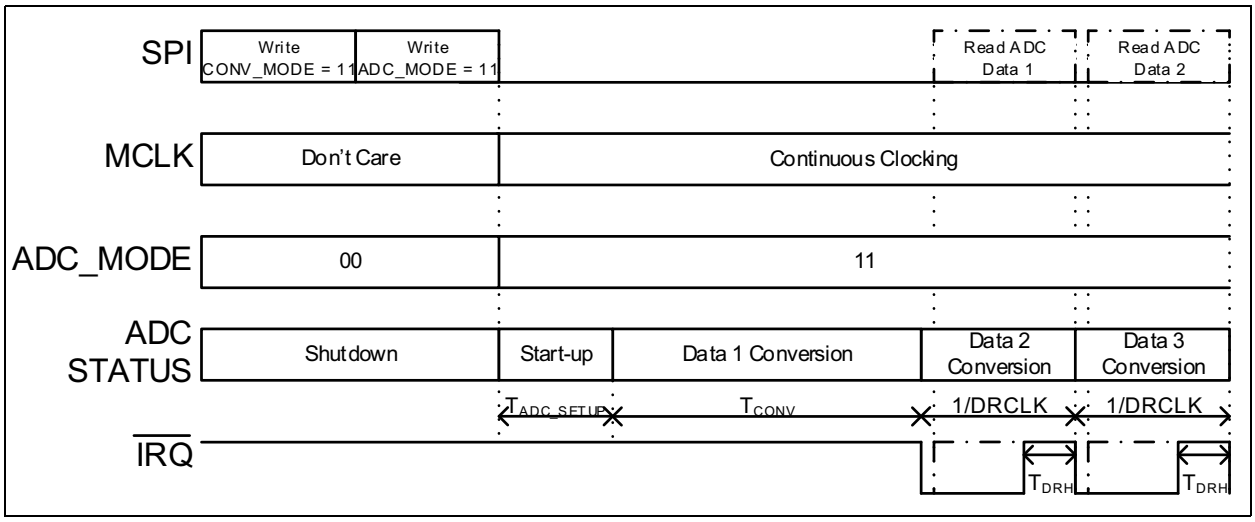



***FIGURE 5-14:*** *MUX Continuous Conversion Mode Timing Diagram.*

### 5.13.2 CONVERSION MODES IN SCAN MODE

In SCAN mode, the device takes one conversion per channel and multiplexes the input to the next channel in the SCAN sequence. Therefore, all conversions are One-Shot mode conversions, regardless of what CONV_MODE[1:0] bits are set to. Each conversion takes the same time $T_{CONV}$ (or 2 x $T_{CONV}$ when AZ_MUX = 1, see **Section 5.1.3 "ADC Offset cancellation algorithm"**) to be performed. If CONV_MODE[1:0]= 00, 01 or 10, the SCAN cycle is executed once and then the ADC is placed into standby or shutdown. If CONV_MODE[1:0] = 11, the ADC runs in a SCAN cycle mode continuously with a TIMER[23:0] delay between each cycle.

Writing the CONV_MODE[1:0] bits with the SPI interface within a conversion does not create an internal Reset. It is recommended not to wait for the end of conversion to change the CONV_MODE[1:0] to the desired value but to change to the desired value just after a data ready to avoid possible glitches. Figure 5-15 and Figure 5-16 respectively detail the ADC timing behavior in One-Shot and Continuous Conversion modes when configured for SCAN mode with N channels chosen among 16 SCAN possibilities.

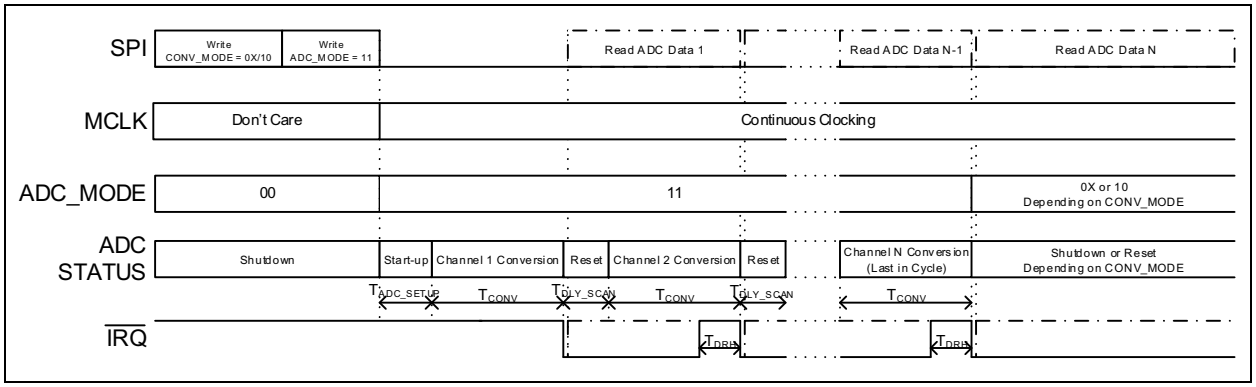



***FIGURE 5-15:*** *SCAN One-Shot Conversion Mode Timing Diagram.*

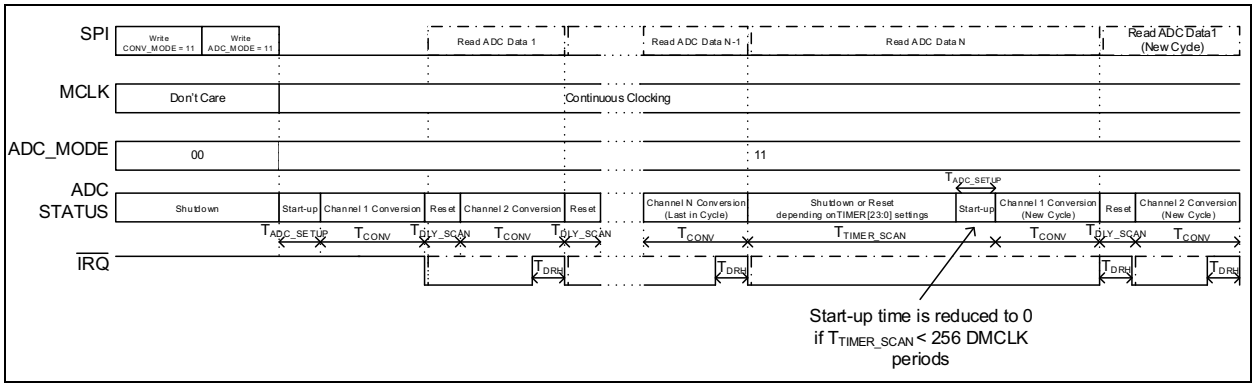



***FIGURE 5-16:*** *SCAN Continuous Conversion Mode Timing Diagram.*

## 5.14 SCAN Mode

### 5.14.1 SCAN MODE PRINCIPLE

In SCAN mode, the device converts sequentially and automatically a list of predefined differential inputs (also referred as input channels) in a defined order. After this series of conversions, the ADC can be placed in standby or Shutdown mode or can wait a certain time in order to perform the same sequence of conversions periodically.

This mode is useful for applications that require constant monitoring of defined channels or internal resources (like $AV_{DD}$ or REFIN+/REFIN-) and allow minimal and simplified communication.

When in SCAN mode, the MUX register (address: 0x6) becomes a Don't Care register.

SCAN mode includes a configurable delay between each SCAN cycle, as well as a configurable delay between each conversion within a SCAN cycle.

Each conversion within the SCAN cycle leads to a data ready interrupt and to an update of the ADCDATA register as soon as the current conversion is finished. The device does not include additional memory to retain all SCAN cycle A/D conversion results. Therefore, each result has to be read when it is available and before it is overwritten by the next conversion result.

### 5.14.2 SCAN MODE ENABLE AND SCAN CHANNEL SELECTION

The ADC is by default in MUX mode at power-up. The ADC enters in the SCAN mode as soon as one of the SCAN[15:0] bits in the SCAN register is set to '1'. MUX mode and SCAN mode cannot be enabled at the same time. When SCAN[15:0] = 0x0000, SCAN mode is disabled and the part returns to MUX mode where the input channel selection is defined by MUX[7:0] bits.

The SCAN cycle conversions are effectively started as soon as ADC_MODE[1:0] bits are programmed through the SPI interface to '11' (direct write or fast command ADC reset and restart).

After the ADC_MODE[1:0] bits have been set to '11', they keep the same value until SCAN mode is completed or aborted.

Each of the SCAN[15:0] bits defines a possible input channel for the SCAN cycle which corresponds to a certain selection of the analog multiplexer input channel and possibly a certain predefined gain of the ADC. The SCAN cycle processes and converts each channel that has been enabled (SCAN[n] = 1) with a defined order of priority from MSB to LSB (SCAN[15] to SCAN[0]). The list of channels with their corresponding inputs is defined in Table 5-13.

When using DATA_FORMAT[1:0] = 11, each channel conversion result in the SCAN sequence can be identified with a channel ID (CH_ID[3:0]) code that will appear in the 4 MSBs of the ADCDATA Register output value (**Section 5.6 "ADC Output Data Format"**). The channel ID permits to retrieve which channel the output data came from. Table 5-13 shows each possible channel ID value and its associated channel.

**TABLE 5-13: ADC CHANNEL SELECTION**

| SCAN[n] bit | Channel Name | Channel ID | MUX[7:0] corresponding setting | Specific ADC gain |
|---|---|---|---|---|
| 15 | OFFSET | 1111 | 0x88 | None |
| 14 | VCM | 1110 | 0xF8 | 1x |
| 13 | $AV_{DD}$ | 1101 | 0x98 | 0.33x |
| 12 | TEMP | 1100 | 0xDE | 1x |
| 11 | Differential Channel D (CH6-CH7) | 1011 | 0x67 | None |
| 10 | Differential Channel C (CH4-CH5) | 1010 | 0x45 | None |
| 9 | Differential Channel B (CH2-CH3) | 1001 | 0x23 | None |
| 8 | Differential Channel A (CH0-CH1) | 1000 | 0x01 | None |
| 7 | Single Ended Channel CH7 | 0111 | 0x78 | None |
| 6 | Single Ended Channel CH6 | 0110 | 0x68 | None |
| 5 | Single Ended Channel CH5 | 0101 | 0x58 | None |
| 4 | Single Ended Channel CH4 | 0100 | 0x48 | None |
| 3 | Single Ended Channel CH3 | 0011 | 0x38 | None |
| 2 | Single Ended Channel CH2 | 0010 | 0x28 | None |
| 1 | Single Ended Channel CH1 | 0001 | 0x18 | None |
| 0 | Single Ended Channel CH0 | 0000 | 0x08 | None |

**Note 1:** SCAN[11:10] and SCAN[7:4] are not available for MCP3462. Writing these bits has no effect.

**2:** SCAN[11:9] and SCAN[7:2] are not available for MCP3461. Writing these bits has no effect.

### 5.14.3 SCAN MODE INTERNAL RESOURCE CHANNELS

#### 5.14.3.1 Analog Supply Voltage Reading ($AV_{DD}$)

During the conversion that reads $AV_{DD}$ in the SCAN mode, the multiplexer selection becomes 0x98 ($AV_{DD}$-$A_{GND}$) which is equal to the analog power supply voltage. Since $AV_{DD}$ is the highest voltage available in the chip, when reading $AV_{DD}$ in SCAN mode, the gain of the ADC is automatically set to 0.33x, which maximizes the input full-scale range, regardless of the GAIN[2:0] settings. This temporary internal configuration does not change the register settings, but just for the gain of the device during this conversion.

With this fixed 0.33x gain, the ADC can measure the maximum specified analog supply voltage ($AV_{DD}$ = 3.6V) with a reference voltage as low as 1.2V.

#### 5.14.3.2 Temperature Reading (TEMP)

During the conversion that reads TEMP in the SCAN mode, the multiplexer selection becomes 0xDE which enables the two temperature diode sensors at each input of the ADC. During the temperature reading, the gain of the ADC is automatically set to 1X regardless of the GAIN[2:0] settings. This temporary internal configuration does not change the register setting, but just for the gain of the device during this conversion.

#### 5.14.3.3 Offset Reading (OFFSET)

During the conversion that reads OFFSET in the SCAN mode, the differential MUX output is shorted to $A_{GND}$ (internally). The offset reading varies from part to part over $AV_{DD}$ and temperature. The reading of this offset value can be used for the device offset calibration or tracking of the offset value in applications.

There is no automatic offset calibration in the device, so the user has to manually write the opposite signed value of the offset measured into the OFFSETCAL register to effectively cancel the offset on the subsequent outputs.

#### 5.14.3.4 VCM Reading (VCM)

During the conversion that reads VCM, the device monitors the internal common-mode voltage of the device, in order to ensure proper operation.

The VCM voltage of the device should be located at 1.2V ± 2%, to ensure proper accuracy. In this setting, the internal multiplexer setting becomes 0xF8h (VCM - $A_{GND}$). In order to properly measure VCM, the voltage reference at the inputs needs to be larger than 1.2V.

During the VCM reading, the gain of the ADC is set to 1x regardless of the GAIN[2:0] settings. This temporary internal configuration does not change the register setting, but just the gain of the device during this conversion.

The VCM reading is susceptible to the gain error and offset error of the ADC which should be calibrated out to obtain a precise internal common-mode measurement.

### 5.14.4 DELAY BETWEEN EACH CONVERSION WITHIN A SCAN CYCLE (DLY[2:0])

While the ADC and multiplexer are optimized to switch from one channel to another instantaneously, it may not be the case of an application that may require additional settling time to overcome the transition. The device can insert an additional delay between each conversion of the SCAN cycle.

The delay value is controlled by the DLY[2:0] bits located in the SCAN Register (SCAN[23:20]) with Table 5-14.

**TABLE 5-14: DELAY BETWEEN CONVERSIONS WITH A SCAN CYCLE**

| DLY[2:0] | Delay Value (DMCLK periods) |
|---|---|
| 111 | 512 |
| 110 | 256 |
| 101 | 128 |
| 100 | 64 |
| 011 | 32 |
| 010 | 16 |
| 001 | 8 |
| 000 | 0 |

The delay is only added in between two conversions of the same SCAN cycle. There is no delay added at the end or the beginning of each SCAN cycle, as a result of the DLY[2:0] bit settings.

During this delay, the ADC is internally kept in Standby mode (ADC_MODE[1:0] = 10 internally, but ADC_MODE[1:0] is always read as '11' through the SPI interface).

The analog multiplexer switches to the next selected input at the end of each conversion which means at the beginning of the added delay so that the application can have additional time to settle properly.

### 5.14.5 DELAY BETWEEN EACH SCAN CYCLE (TIMER[23:0])

During Continuous mode, SCAN cycles are processed continuously one after another separated by a time delay ($T_{TIMER\_SCAN}$), which is defined by the TIMER (address 0x8) register value. During this delay, the ADC is automatically placed into a power-saving mode (Standby or Shutdown). The $T_{TIMER\_SCAN}$ delay offers better power efficiency for applications that run a SCAN sequence periodically. Since the delay can be very long, it allows to have synchronous applications with very slow update rates without having to use an external timer. The TIMER register defines the time $T_{TIMER\_SCAN}$ between each cycle with a 24-bit

unsigned value going from 0 to 16777215 DMCLK periods. Table 5-15 details the TIMER possible values with respect to the TIMER[23:0] code.

**TABLE 5-15: TIMER DELAY VALUE IN BETWEEN EACH SCAN CYCLE**

| TIMER[23:0] | $T_{TIMER\_SCAN}$ Delay Value (DMCLK periods) |
|---|---|
| 111111111111111111111111 | 16777215 |
| 111111111111111111111110 | 16777214 |
| 100000000000000000000000 | 8388608 |
| 000000000000000000000001 | 1 |
| 000000000000000000000000 | 0 |

The internal TIMER counter will decrement from the $T_{TIMER\_SCAN}$ value to 0 and launch the new SCAN cycle.

If the $T_{TIMER\_SCAN}$ value is greater than $T_{ADC\_SETUP}$ (256 DMCLK periods), the device will be placed in Shutdown mode (ADC_MODE internally is set to '00') at each end of a SCAN cycle. When the internal TIMER counter reaches 256, the device will startup the ADC during a $T_{ADC\_SETUP}$ time to be ready to convert when the internal counter reaches 0.

If the $T_{TIMER\_SCAN}$ value is less than $T_{ADC\_SETUP}$, the part will be placed in Standby mode between each SCAN cycle (ADC_MODE internally set to '10').

ADC_MODE[1:0] bits in the CONFIG0 can only be read as 11 by the SPI interface during the whole SCAN cycle and in between SCAN cycles.

## 5.15 A/D Conversions Automatic Reset and Restart Feature

When the A/D conversions are running, the user can change the device configuration through the SPI interface by writing any register. Some register settings directly impact the conversion results and would lead to invalid ADC data if they were changed within a conversion. The device incorporates an automatic reset and restart feature for the A/D conversions to avoid these invalid data to be generated. Some register writes with the SPI interface during a conversion will automatically reset and restart the A/D conversion with the new settings.

The automatic reset and restart feature behavior depends on the register bits that are written by the SPI interface.

### 5.15.1 REGISTER BITS MODIFICATIONS NOT CAUSING RESET/RESTART

The first group of bits does not generate any reset and restart. This group is composed of all the unused and all read-only bits and of some digital settings like CONV_MODE[1:0], DATA_FORMAT[1:0], CRC_FORMAT, EN_CRCCOM, IRQ_MODE[0], EN_FASTCMD, EN_STP and LOCK[7:0] bits.

### 5.15.2 REGISTER BITS MODIFICATIONS CAUSING IMMEDIATE RESET/RESTART

The second group of bits generates immediate reset and restart. The reset is immediate, the restart is only valid after a period of 2 MCLK periods (necessary to handle the reset and ensures that the restart is synchronous with the master clock). This group is composed of settings that do not induce an analog operating point change. This group includes ADC_MODE[1:0], PRE[1:0], OSR[3:0], GAIN[2:0], AZ_MUX, EN_OFFCAL, EN_GAINCAL, IRQ_MODE[1:0], MUX[7:0] and DLY[2:0] bits.

The EN_OFFCAL, EN_GAINCAL and IRQ_MODE[1:0] bits generate the reset and restart only if they are changed to a new value. An overwrite of the same value has no effect. In SCAN mode, the reset and restart feature will just restart the current conversion for this group of bits, the SCAN cycle is not modified and not restarted. The MUX[7:0] bits can be changed within SCAN mode without generating a reset and restart since this register is a don't care during SCAN mode The DLY[2:0] bits can be changed during the MUX mode without generating a reset and restart, since these bits are Don't Care during the MUX mode. The OFFSETCAL[23:0] and GAINCAL[23:0] will only generate a reset and restart when written if their corresponding enable bit (EN_OFFCAL, EN_GAINCAL) is enabled.

The ADC_MODE[1:0] bits generate an immediate reset and restart but only if they are overwritten with '11' (in any other case, the conversions are stopped). Depending on the part being in MUX or SCAN mode, the reset and restart feature resets either the conversion or the complete SCAN cycle.

### 5.15.3 REGISTER BITS MODIFICATIONS CAUSING DELAYED RESET/RESTART

A third group of bits will generate a reset and restart that induces a new start-up delay ($T_{ADC\_SETUP}$) so that the internal analog operating points can be settled with the new settings before the new conversion is started. The reset is immediate, the start-up timer is only restarted after a period of 2 MCLK periods (necessary to handle the reset and ensures that the restart is synchronous with the master clock). Overall, the delay from reset to actual restart of the conversion with the new settings is then 2 MCLK plus $T_{ADC\_SETUP}$. This group includes CONFIG0[7:6], CLK_SEL[1:0], CS_SEL[1:0], BOOST[1:0] and the RESERVED address registers (0xB and 0xC). The CS_SEL[1:0], CLK_SEL[1:0] and BOOST[1:0] will induce a start-up timer delay only if they are changed to a new value. If they are overwritten with the same value, they will generate an immediate reset and restart. In SCAN mode, the reset and restart feature will just restart the current conversion for this group of bits, the SCAN cycle is not modified and not restarted.

This third group of bits will induce a start-up timer delay even when ADC_MODE[1:0] = 10 or if the ADC is in Standby mode.

During the reset and restart sequence, the reset is immediate and resets the internal phases to the original state, which can lead to a discontinuity in the clock output frequency if the AMCLK clock output is enabled. The restart is synchronous with the AMCLK generation and is effective only after a two MCLK periods. The restart will also generate a conversion start pulse (only after the two MCLK periods or the 2 MCLK + $T_{ADC\_SETUP}$ necessary for the restart) if enabled for the user to be able to align the system with the exact start of the new conversion.

In MUX mode, the TIMER and SCAN registers do not generate a reset and restart when written, except if the SCAN register is modified to effectively enter in SCAN mode. In this case, the MUX mode is superseded by the SCAN mode immediately.

In SCAN mode, a write access of the SCAN register during or between conversions within the SCAN cycle will create a reset and restart of the whole SCAN sequence. Within the same conditions, a write access on the TIMER register will not create a reset and restart of the whole SCAN sequence. However during the $T_{TIMER\_SCAN}$ delay between each SCAN cycle, a write on the SCAN register will not generate a reset and restart of the whole sequence. Within the same conditions, a write on the TIMER register will generate a reset and restart of the whole sequence.

Depending on the phase between AMCLK and the SPI commands, the 2 MCLK delay can become a 4 MCLK delay to ensure the proper synchronization of the device. If very precise synchronization is required, it is recommended to either not change dynamically the register configurations (i.e. not during conversions), or to use the EN_STP = 1 setting so that the start of the conversions can be clearly determined.

# 6.0 SPI SERIAL INTERFACE AND DEVICE OPERATION

## 6.1 OVERVIEW

The MCP3461/2/4 devices use SPI interface for reading and writing the internal registers. The SPI interface includes a four-wire ($\overline{CS}$, SCK, SDI, SDO) serial SPI interface that is compatible with SPI modes 0,0 and 1,1. Data are clocked out of the device on the falling edge of SCK and data are clocked into the device on the rising edge of SCK. In these modes, the SCK clock can idle either high (1,1) or low (0,0). The digital interface is asynchronous with the MCLK clock that controls the ADC sampling and digital filtering. All digital input pins are Schmitt-Triggered to avoid system noise perturbations on the communications. The SPI interface is maintained in reset state during POR.

Each SPI communication starts with a $\overline{CS}$ falling edge and stops with the $\overline{CS}$ rising edge. Each SPI communication is independent. When $\overline{CS}$ is logic-high, SDO is in high-impedance, the transitions on SCK and SDI have no effect. Changing from SPI mode 1,1 to an SPI mode 0,0 and vice-versa is possible and must be done while the $\overline{CS}$ pin is logic high. Any $\overline{CS}$ rising edge clears the communication and resets the SPI digital interface. See Figure 1-1 for the SPI timing details.

The MCP3461/2/4 digital interface is capable of handling various continuous read and write modes, which allows to perform the ADC data streaming or full register map writing within only one communication (and therefore with only one unique command byte). It also includes single-byte Fast commands that allow faster access to common and useful configurations. The device does not include a master Reset pin, but it includes an SPI fast command to be able to fully reset the part at any time and place it back in a default configuration.

The device family also includes advanced security features to secure communication and alert user's of unwanted Write commands which change the desired configuration. To secure the entire configuration of the device, the device includes an 8-bit lock code (LOCK[7:0]), which blocks all write commands to the full register map if the value of the lock code is not equal to a defined password (0xA5). The user can protect its configuration by changing the LOCK[7:0] value to 0x00 after full programming, so that any unwanted write command will not result to a change in the configuration. Each SPI read communication can be secured through a selectable CRC-16 checksum provided on SDO pin at the end of every communication sequence. This checksum computation is compatible with the DMA CRC hardware of the PIC24® and PIC32® MCUs, as well as many other MCU references, resulting in no additional overhead for the added security.

Once the part is locked (write-protected) an additional checksum calculation is also running continuously in background to ensure the integrity of the full register map. All writable registers of the register map are processed through a CRC-16 calculation engine and give a CRC-16 checksum that depends on the configuration. This checksum is readable from the CRC register and updated at all times when MCLK is running. If a change in this checksum happens, a CRC interrupt generates a flag to warn the user that the configuration has been corrupted.

The MCP3461/2/4 devices also include additional digital signal pins such as a dedicated $\overline{IRQ}$ interrupt output pin and a master clock input/output pin (MCLK) which help for easier synchronization and faster interrupt handling, facilitating the implementation of the device in many different applications.

## 6.2 SPI COMMUNICATION STRUCTURE

The MCP3461/2/4 interface has a simple communication structure. Every communication starts with a $\overline{CS}$ falling edge and stops with a $\overline{CS}$ rising edge.

After the communication start, the communication is always started by the COMMAND byte (8 bits) clocking on the SDI input. The COMMAND byte defines the command that will be executed by the digital interface. It includes the device address, the register address bits and the command type bits.

The COMMAND byte is typically followed by data bytes clocked on SDI if the command type is a write, and on SDO if the command type is a read. The COMMAND byte can also define a Fast command, in which case, it is not followed by any other byte. The following subsections detail the COMMAND byte structure and all possible commands.

During the COMMAND byte clocking on SDI, a STATUS byte is also propagated on the SDO output to enable easy polling of the device status. During this time the interface is full-duplex, but the part can still be used by MCUs handling only half-duplex communications if the STATUS byte is ignored.

### 6.2.1 COMMAND BYTE STRUCTURE

The Command byte fully defines the command that is executed by the part. This byte is divided into three parts: the device address bits (CMD[7:6]), the Command address bits (CMD[5:2]) and the command type bits (CMD[1:0]). A representation of this command byte is available in the Figure 6-1.

| CMD[7] | CMD[6] | CMD[5] | CMD[4] | CMD[3] | CMD[2] | CMD[1] | CMD[0] |
|---|---|---|---|---|---|---|---|
| Device Address Bits | | Register Address / Fast Command bits | | | | Command Type Bits | |

***FIGURE 6-1:*** *COMMAND Byte.*

### 6.2.2 DEVICE ADDRESS BITS (CMD[7:6])

The SPI interface of the MCP3461/2/4 devices is addressable which means that multiple devices can communicate on the same SPI bus with only one chip select line for all devices. Each device communication starts by a $\overline{CS}$ falling edge followed by the clocking of the device address (CMD[7:6]). Each device contains an internal device address which the device can respond to.

This device address is coded on two bits, so four possible addresses are available. The address is hard-coded within the device and should be determined at the ordering of the device. The device address is part of the device markings, to avoid potential confusion (see **Section 9.1 "Package Marking Information"**). When the CMD[7:6] bits match the device address, the communication proceeds and the part will execute the commands defined in the control byte and its subsequent data bytes.

When the CMD[7:6] bits do not correspond to the device address hard-coded in the device, the command is ignored. In this case, the SDO output will become high-impedance which prevents bus contention errors when multiple devices are connected on the same SPI bus (see Figure 6-3). The user has to exit from this communication through a $\overline{CS}$ rising edge to be able to launch another command.

### 6.2.3 COMMAND ADDRESS BITS (CMD[5:2])

The COMMAND byte contains four address bits (CMD[5:2]) that can serve two purposes. In case of a register write or read access, they define at which register address the first read/write is performed. In case of a fast command, they determine which Fast command is executed by the device. In case of a Write command on a read-only register, the command is not executed and the communication should be aborted ($\overline{CS}$ rising edge) to place another command. All registers can be read, there is no undefined address in the register map.

### 6.2.4 COMMAND TYPE BITS (CMD[1:0])

The last two bits of the COMMAND byte define the command type. These bits are an extension of the typical read/write bits present in most SPI communication protocols. The two bits define four possible command types: Incremental Write, Incremental Read, Static Read and Fast command. Changing command type within the same communication (while $\overline{CS}$ is logic-low) is not possible. The communication has to be stopped ($\overline{CS}$ rising edge) and restarted ($\overline{CS}$ falling edge) to change its command type. The list of possible commands, their type and their possible command addresses are described in Table 6-1.

**TABLE 6-1: COMMAND TYPES DESCRIPTION TABLE**

| CMD[5:2] | CMD[1:0] | Command Description |
|---|---|---|
| 0xxx | 00 | Don't Care |
| 100x | 00 | Don't Care |
| 1010 | 00 | ADC Conversion Start/Restart fast command (Overwrites ADC_MODE[1:0] = 11) |
| 1011 | 00 | ADC Standby mode fast command (Overwrites ADC_MODE[1:0] = 10) |
| 1100 | 00 | ADC Shutdown mode fast command (Overwrites ADC_MODE[1:0] = 00) |
| 1101 | 00 | Full-Shutdown mode fast command (Overwrites CONFIG0[7:0] = 0x00) |
| 1110 | 00 | Device Full Reset fast command (resets whole register map to default value) |
| 1111 | 00 | Don't Care |
| ADDR | 01 | Static read of register address ADDR |
| ADDR | 10 | Incremental write starting at register address ADDR |
| ADDR | 11 | Incremental read starting at register address ADDR |

### 6.2.5 FAST COMMANDS DESCRIPTION

There are five possible fast commands available in the MCP3461/2/4 devices. For each command, only the COMMAND byte has to be provided on the SPI port and the command will be executed right after the COMMAND byte has been clocked. The Fast command codes are detailed in Table 6-1. All undefined command address codes for fast commands will be ignored and will have no effect. SDO will stay in high-impedance after the COMMAND byte for a fast command until a $\overline{CS}$ rising edge is provided. The fast commands can be enabled or disabled by placing the EN_FASTCMD bit in the IRQ register to 1 (default). Disabling fast commands can increase the security of the device because it can avoid unwanted fast commands to be executed which can be useful in harsh environments.

The ADC Start/Restart command (command address: 1010) overwrites the ADC_MODE[1:0] bits to '11' creating a conversion start (or a restart if the conversion was already running).

The ADC Standby mode command (command address: 1011) overwrites the ADC_MODE[1:0] bits to '10' and is therefore placing the ADC in Standby mode.

The ADC Shutdown mode command (command address: 1100) overwrites the ADC_MODE[1:0] bits to '00' and is therefore placing the ADC in Shutdown mode.

The Full-Shutdown mode command (command address: 1101) is overwriting the CONFIG0 register to 0x00h which places the device in full Shutdown mode. (see **Section 5.9 "Full-Shutdown Mode"** for a full description of this mode).

The Full Reset command (command address: 1110) resets the whole device and places the whole register map into its default state condition, including the non-writable registers. The only difference with a POR event is that the $\overline{POR\_STATUS}$ bit in the IRQ register is set to 1 after a full reset and is reset to 0 after a POR event. The full reset command is the only way with POR to clear the ADC data output register to its default value.

### 6.2.6 DEVICE ADDRESS AND STATUS BYTE DURING CONTROL BYTE

During the clocking of the command byte on the SDI pin, the SDO pin displays a STATUS byte to help the user to retrieve quickly interrupt status information.

The STATUS byte permits to do fast polling of the different interrupts without having to read the IRQ register. However it requires a MCU that can communicate in full-duplex mode (SDI and SDO are clocked at the same time). For MCUs that are only half-duplex and for devices that do not incorporate a separate $\overline{IRQ}$ pin, or for applications that do not connect the existing $\overline{IRQ}$ pin, the polling of the IRQ status can still be done by reading the IRQ register continuously.

The STATUS byte structure is described in Figure 6-2.

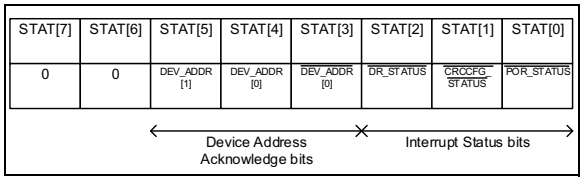



***FIGURE 6-2:*** *STATUS Byte*

The first two bits are always equal to 0 and SDO toggles to 0 as soon as a $\overline{CS}$ pin falling edge is performed. This allows the application of multiple devices with different device addresses sharing one common SPI bus and avoiding bus contention during STATUS byte clocking.

The next three bits of the STATUS byte give a confirmation (acknowledge) of the hard-coded device address. If the device address of the command byte and the internal device address of the chip match, these three bits will be transmitted and they are equal to:

- STAT[5:4] = DEV_ADDR[1:0]
- STAT[3] = $\overline{DEV\_ADDR}$[0]

The STAT[3] bit permits the user to distinguish the SDO output from a High-Impedance state (device address not matched) as the bits STAT[4] and STAT[3] are complementary and will induce a toggle on the SDO output.

If the two device address bits are not matched with the internally hard-coded device address bits, SDO is maintained in a High-Impedance state during the rest of the communication and the command is ignored. This behavior avoids potential bus contention errors if multiple devices with different device addresses are sharing the same SPI bus, as after the transmission of the first two bits only one device is responding to the command (all other devices with non-matching device addresses have SDO kept in high-impedance). In this case, the user needs to abort the communication (CS rising edge) to be able to perform another command.

The three LSBs of the STATUS byte are the three interrupt status bits:

- STAT[2] = $\overline{DR\_STATUS}$ ADC (data ready interrupt status)
- STAT[1] = $\overline{CRCCFG\_STATUS}$ (CRC checksum error on the register map interrupt status)
- STAT[0] = $\overline{POR\_STATUS}$ (POR interrupt status).

These three interrupt status bits are independent of the two other interrupt mechanisms ($\overline{IRQ}$ pin and IRQ register) and are cleared each time the STATUS byte is fully clocked. This enables the polling on the STATUS byte as a possible interrupt management solution without requiring to connect the $\overline{IRQ}$ pin in the system. All the status bits value are latched together just after the device address has been correctly recognized by the chip. Any interrupt happening after the two first STATUS bits have been clocked out will appear on the STATUS byte of the next communication sequence.

Figure 6-3 represents the beginning of each communication with both COMMAND and STATUS bytes depicted. After the STATUS byte is propagated, the SDO pin will be placed in High-Impedance for fast commands or Write commands and will transfer data bytes for Read commands as long as the $\overline{CS}$ pin stays logic-low.

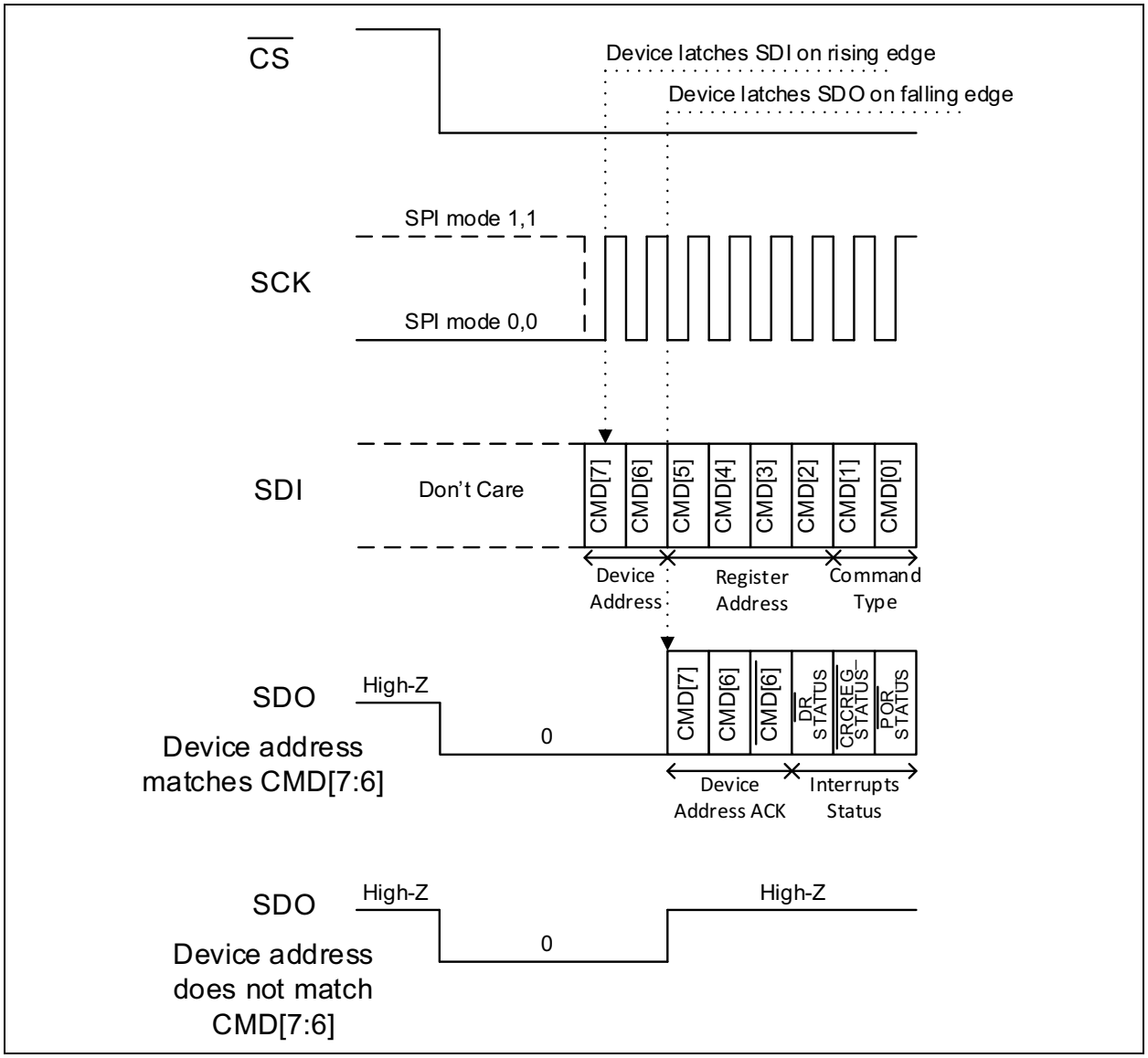



***FIGURE 6-3:*** *SPI Communication Start (COMMAND on SDI and STATUS on SDO) in Cases of a Device Address Match and Not Matched.*

## 6.3 WRITING TO THE DEVICE

When the command type is “Incremental Write” (CMD[1:0] = 10), the device enters Write mode and starts writing the first data byte to the address given in the CMD[5:2] bits.

After the STATUS byte has been transferred, SDO is always in a high-impedance state during an incremental write communication. Writing to a read-only address (such as addresses 0x0 or 0xF) has no effect and does not increment the Address Pointer. In this case, the user needs to stop the communication and restart a communication with a COMMAND byte pointing to a writable address (0x1 to 0xD).

Each register is effectively written after receiving the last bit for the register (SCK last rising edge). Any $\overline{CS}$ rising edge during a write communication aborts the current writing. In this case, the register being written will not be updated and will keep its old value.

The registers may need 8, 16 or 24 bits to be effectively written depending on their address (see Table 8-1). After each register is written, the Address Pointer is automatically incremented as long as $\overline{CS}$ stays logic-low. Attempted data writes to read-only registers will result in the data byte being written to the next sequential writable register/address in the register map. When the Address Pointer reaches to 0xD, the next register to write is the register 0x1 (see Figure 6-4 for a graphical representation of the address looping). The incremental write feature can be used to fully configure the part, by using a unique communication which can save time in the application. This unique communication can end at address 0xD so that the user can also lock the configuration when written, providing additional security in the application (see **Section 6.6 “Locking/Unlocking Register Map Write Access”**).

Internal registers located at addresses 0xB, 0xC and 0xE should be kept to their default state at all times for proper operation. These are reserved registers and should not be modified.

Figure 6-5 and Figure 6-6 show an example of a write communication in detail with a single register incremental write communication.

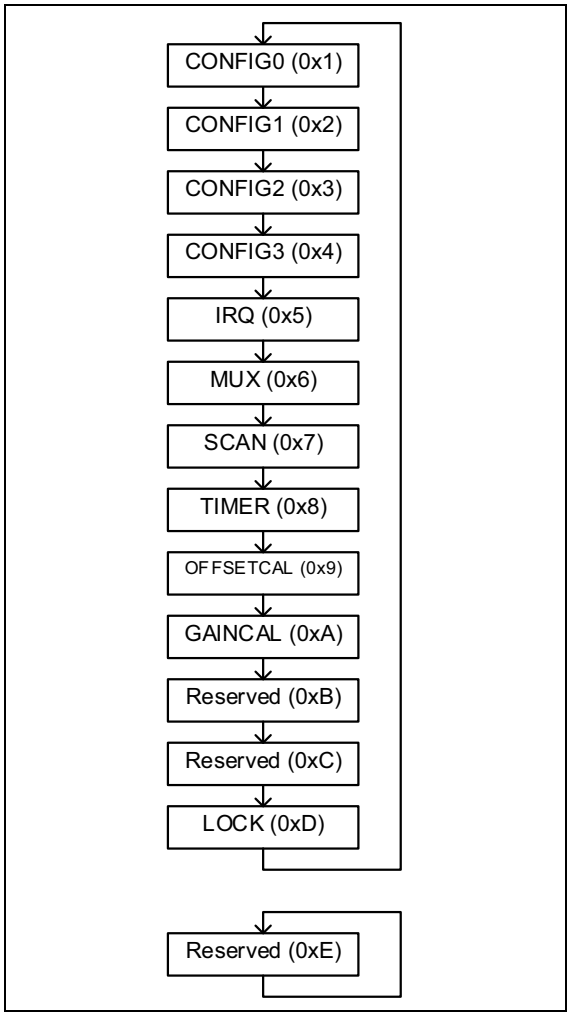



***FIGURE 6-4:*** *Incremental Write Loop.*

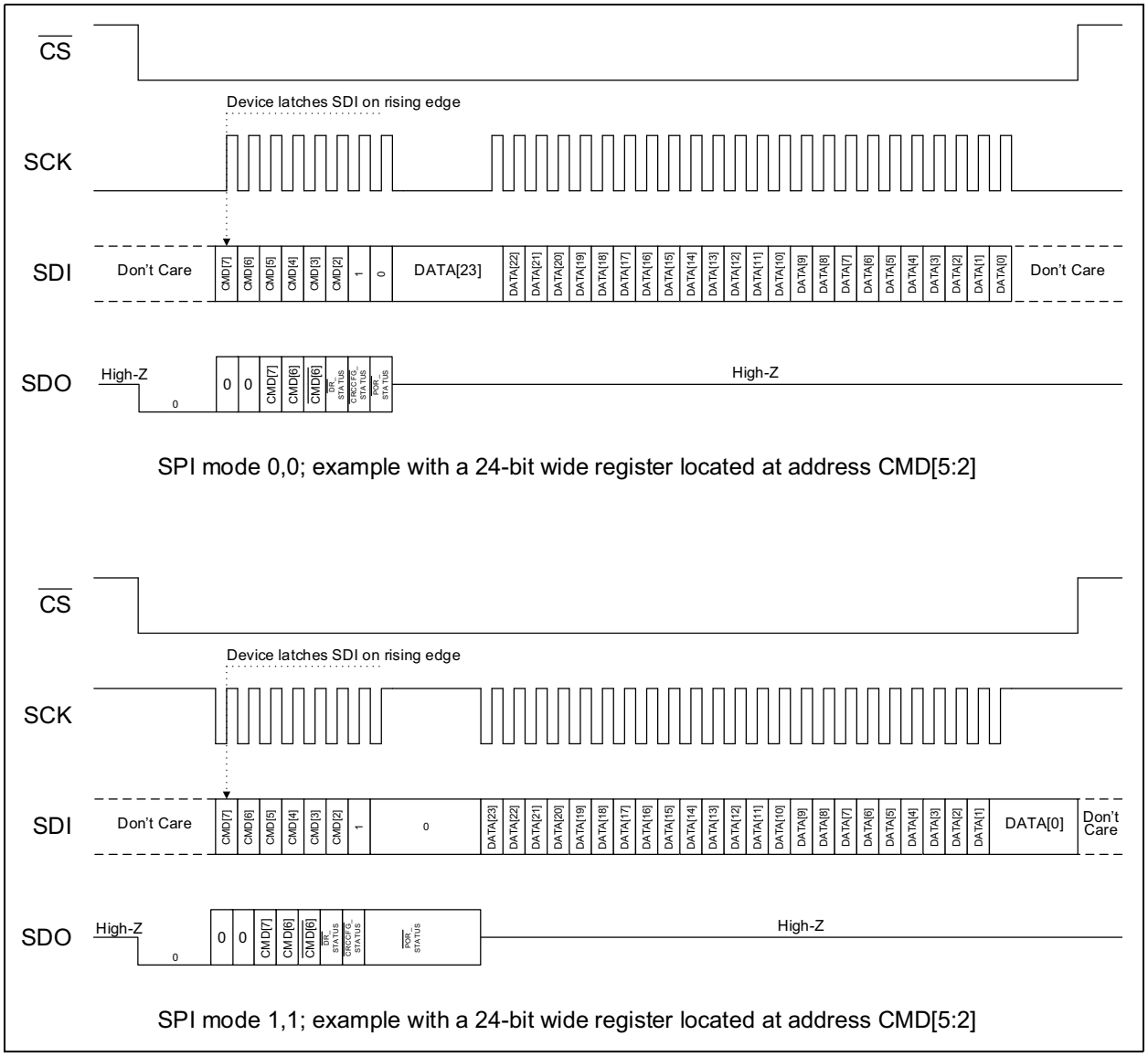



***FIGURE 6-5:*** *Single Register Write Communication (CMD[1:0] = 10) Timing Diagram.*

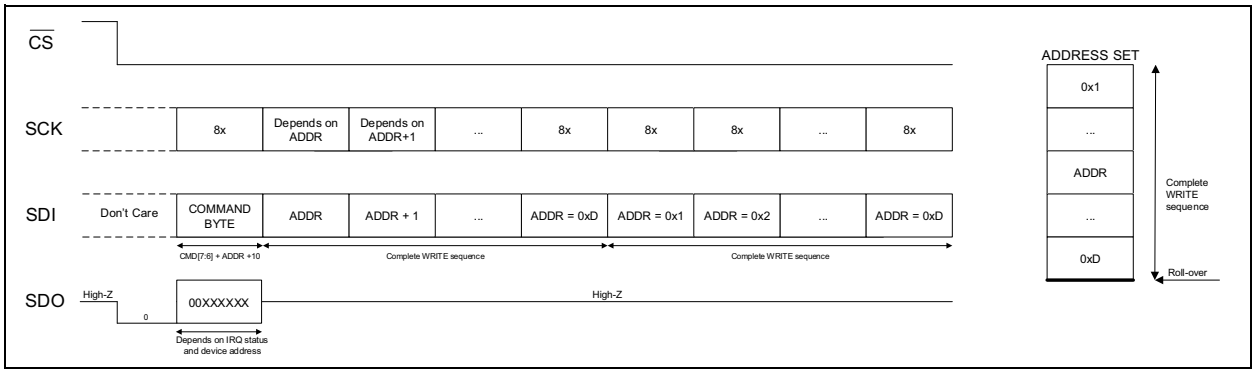



***FIGURE 6-6:*** *Multiple Register Write Within One Communication Using Incremental Write Feature.*

## 6.4 READING FROM THE DEVICE

When the COMMAND bit CMD[0] is equal to 1, the command is a read communication. After the STATUS byte has been transferred, the first register to be read on the SDO pin is the one which address is defined by the command address bits (CMD[5:2]).

Any $\overline{CS}$ rising edge during a read communication aborts the current reading.

The registers may need 4, 8, 16, 24 or 32 bits to be fully read depending on their address (see Table 8-1).

If the CMD[1:0] bits are equal to '11', the command type is incremental read. In this case, after each register is read, the address pointer is automatically incremented as long as $\overline{CS}$ stays logic-low. The following data bytes are read from the next address sequentially defined in the register map. When the Address Pointer reaches 0xF (last register in the register map for reading), the next register to read is the register 0x0. See Figure 6-7 for a graphical representation of the address looping.

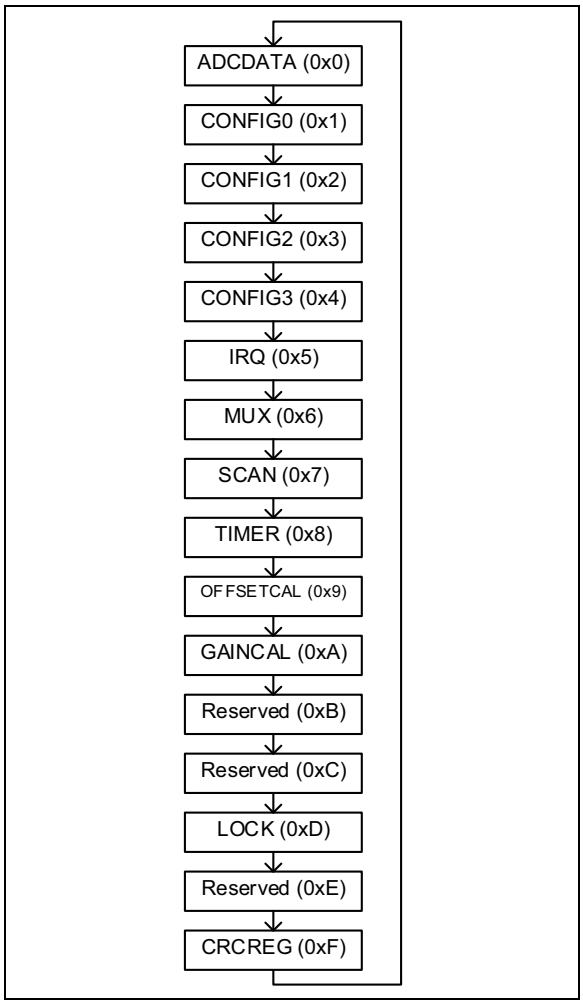



***FIGURE 6-7:*** *Incremental Read Loop.*

If the CMD[1:0] bits are equal to '01', the command type is static read. In this case the register address defined in the COMMAND byte is read continuously. The Address Pointer is not incremented automatically. Continuously clocking SCK while $\overline{CS}$ stays logic-low will continuously read the same register. Reading another register is only possible by aborting the current communication sequence by raising $\overline{CS}$ and issuing another command.

In both static and incremental modes, the registers will be updated after each register read is fully performed. If the value of the register changes internally during the read, it will only be updated after the end of the read. The value of each register is latched in the SDO output shift register at the first rising edge of SCK of each individual register reading. Figure 6-8 shows the details, bit by bit, of a single register read communication. Figure 6-9 shows the examples of Static and Incremental Read Communications.

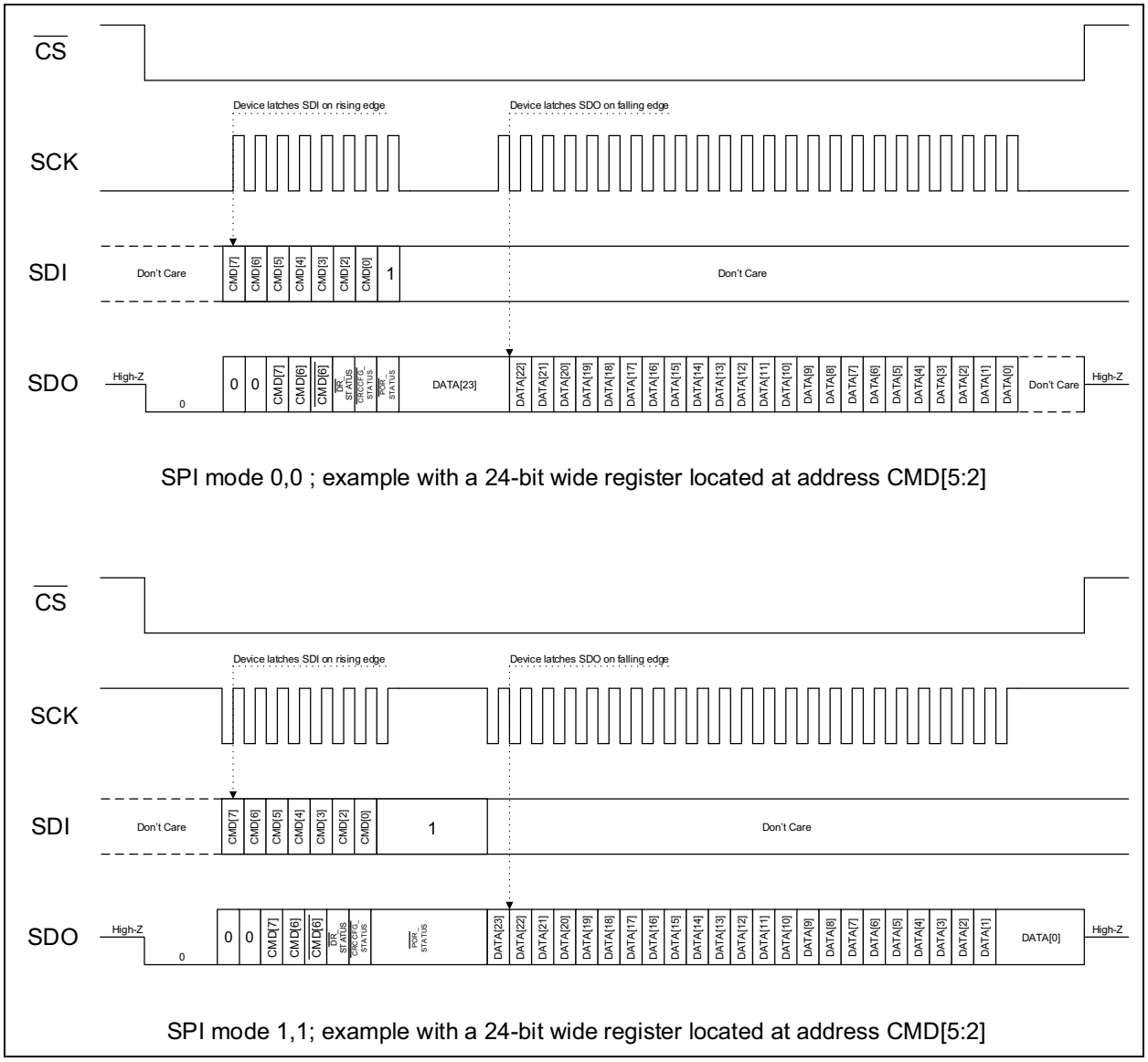



***FIGURE 6-8:*** *Single Register Read SPI Communication.*

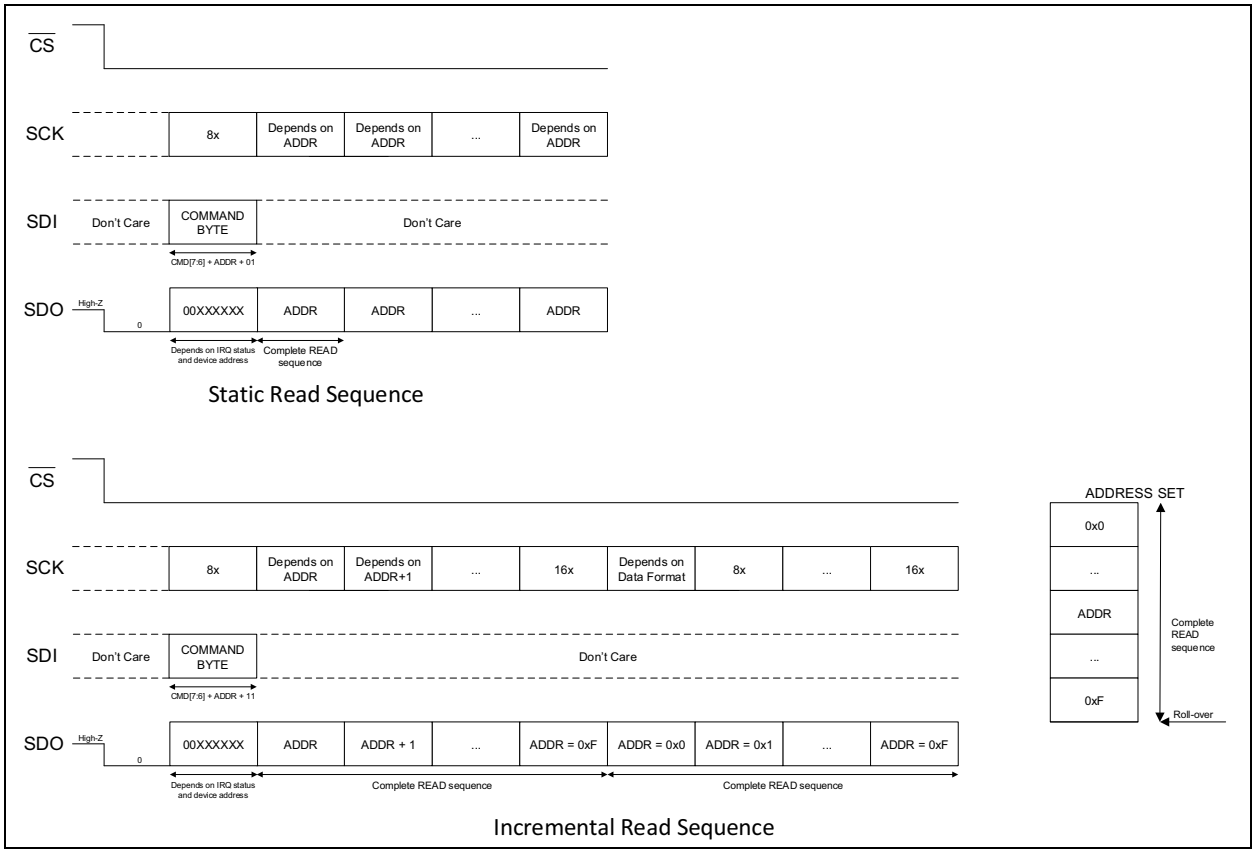



***FIGURE 6-9:*** *Static and Incremental Read SPI Communications.*

If the COMMAND byte defines a static read of the ADCDATA register (address 0x0), the ADC data will be present on SDO and will be updated continuously at each read. In this case, when a data ready interrupt happens within a read, the data are not corrupted and will be updated to a new value after the old value has been completely read. The ADC register contains a double buffer that prevents data from being corrupted while reading it. The part will be able to stream output data continuously with no additional command if the communication is not stopped with a $\overline{CS}$ rising edge. Figure 6-10 represents the continuous streaming of incoming ADCDATA through the SPI port with both SPI modes 0,0 and 1,1.

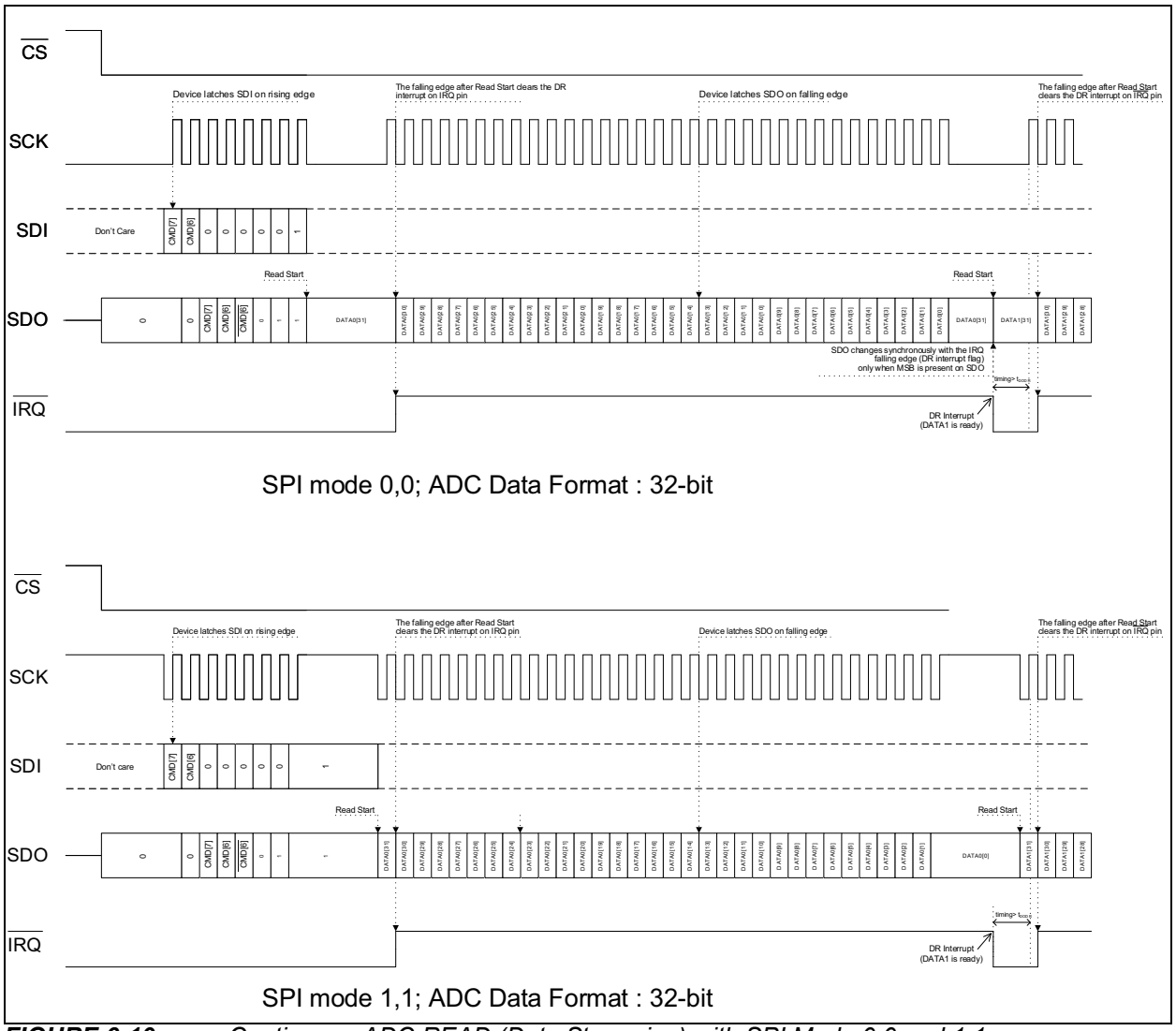



***FIGURE 6-10:*** *Continuous ADC READ (Data Streaming) with SPI Mode 0,0 and 1,1.*

For continuous reading of ADCDATA in SPI mode (0,0) once the data has been completely read after a data ready interrupt, the SDO pin will take the MSB value of the previous data at the end of the reading (falling edge of the last SCK clock). If SCK stays idle at logic-low (by definition of mode 0,0), the SDO pin will be updated at the falling edge of the next data ready pulse (synchronously with the $\overline{IRQ}$ pin falling edge with an output timing of $t_{DODR}$) with the new MSB of the data corresponding to the data ready pulse. This mechanism allows the device to continuously read ADC data outputs seamlessly, even in SPI mode (0,0).

In SPI mode (1,1), the SDO pin stays in the last state (LSB of previous data) after a complete reading which also allows seamless continuous Read mode.

An ADC output data can only be properly read after a $t_{DODR}$ time after the data ready interrupt is coming on the IRQ pin. The $t_{DODR}$ timing is shorter than the time necessary to input a command on the SDI pin which ensures proper reading in the case a new read command is triggered by the data ready interrupt. In case of continuous reading (with CS pin kept logic-low), the $t_{DODR}$ timing must be carefully taken care of by the MCU, but in general, the interrupt service time is longer

than the $t_{DODR}$ timing. Retrieving a data ready interrupt by reading the STATUS byte or reading IRQ register automatically ensures that the $t_{DODR}$ timing is respected.

## 6.5 Securing Read Communications Through CRC-16 Checksum

Since some applications can generate or receive large EMI/EMC interferences and large transient spikes, it is helpful to secure SPI communications as much as possible to maintain data integrity and desired configurations during the lifetime of the application.

The communication data on the SDO pin can be secured through the insertion of a Cyclic Redundancy Check (CRC) checksum at the end of each read sequence. The CRC checksum on communications can be enabled or disabled through the EN_CRCCOM bit in the CONFIG3 register. The CRC message ensures the integrity of the read sequence bits transmitted on the SDO pin.

The CRC checksum in the MCP3461/2/4 devices uses the 16-bit CRC-16 ANSI polynomial as defined in the IEEE 802.3 standard:

- $x^{16} + x^{15} + x^2 + 1$

This polynomial can also be noted as 0x8005. CRC-16 detects all single and double-bit errors, all errors with an odd number of bits, all burst errors of length 16 bits or less, and most errors for longer bursts. This allows an excellent coverage of the SPI communication errors that can happen in the system, and heavily reduces the risk of a miscommunication, even under noisy environments.

When enabled, the CRC checksum (CRCCOM[15:0]) is propagated on SDO after each read communication sequence. In case of a Static Read command, the checksum is propagated after each register read. In case of an incremental read command, the checksum is propagated after the last register read in the register map (address 0xF). Figure 6-12 and Figure 6-13 show typical read communications in Static and Incremental Read modes when the EN_CRCCOM bit is enabled. Since the STATUS byte is propagated on SDO, it is part of the first message, and therefore, it is included in the calculation of the first checksum. For subsequent checksum calculations, the message only contains the registers that are effectively read between two checksums.

The CRC-16 format displayed on the SDO pin depends on the CRC_FORMAT bit in the CONFIG3 register (see Figure 6-11). It can be either 16-bit or 32-bit format, to be compatible with both 16-bit and 32-bit MCUs. The CRCCOM[15:0] bits calculated by the device are not dependent on the format (the device always calculates only a 16-bit only CRC checksum).

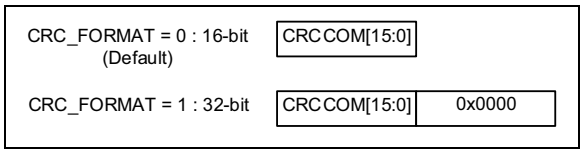



***FIGURE 6-11:*** *CRC Format Table for Read Communications.*

The CRC calculation computed by the device is fully compatible with CRC hardware contained in the Direct Memory Access (DMA) of the PIC24 and PIC32 MCU product lines. The CRC message that should be considered in the PIC® device DMA is the concatenation of the read sequence and its associated checksum. When the DMA CRC hardware computes this extended message, the resulted checksum should be 0x0000. Any other result indicates that a miscommunication has happened and that the current communication sequence should be stopped and restarted.

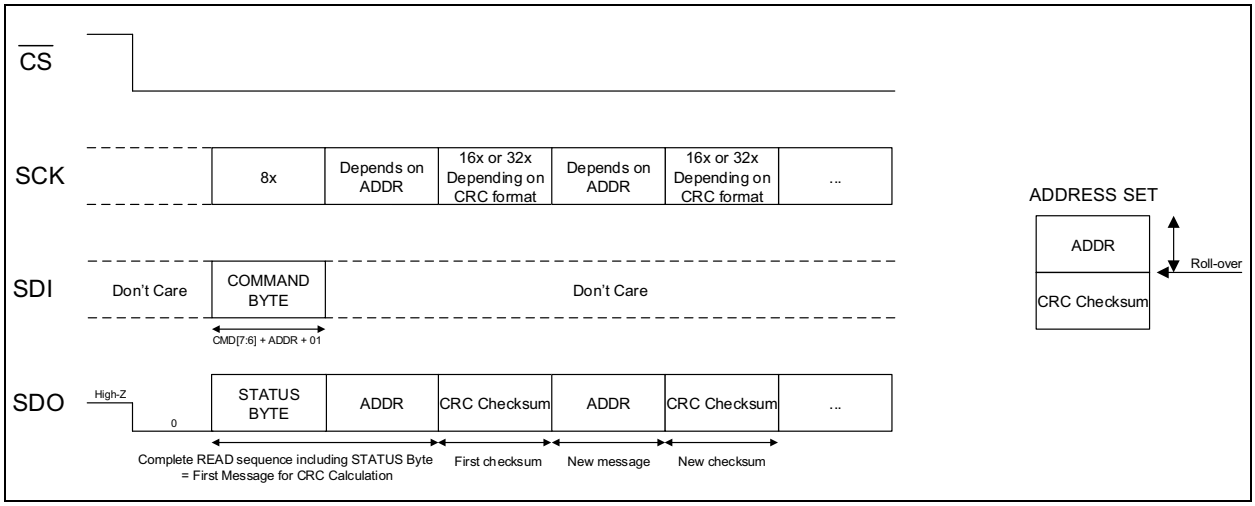



***FIGURE 6-12:*** *SPI Static Read with Communication CRC Enabled.*

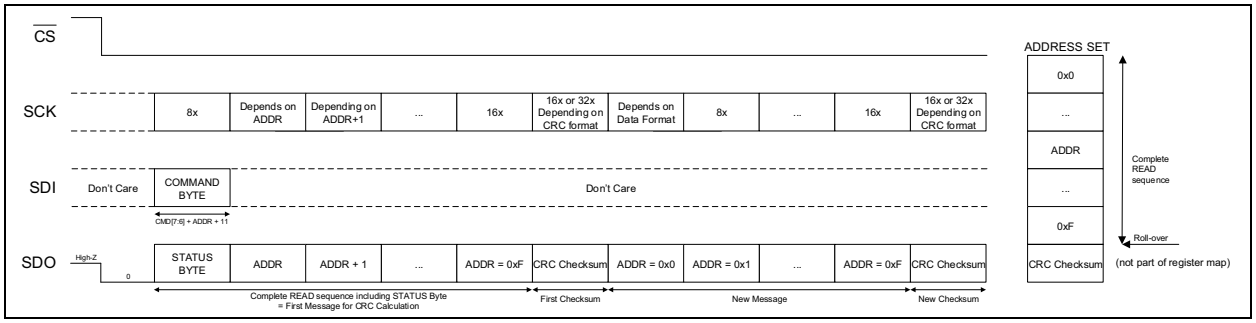



***FIGURE 6-13:*** *SPI Incremental Read with Communication CRC Enabled.*

## 6.6 Locking/Unlocking Register Map Write Access

The MCP3461/2/4 digital interface includes an advanced security feature that permits locking or unlocking the register map write access. This feature prevents the miscommunication that can corrupt the desired configuration of the device, especially an SPI read becoming an SPI write because of the noisy environment.

The last register address of the incremental write loop (0xD: LOCK) contains the LOCK[7:0] bits. If these bits are equal to the password value (0xA5), the register map write access is not locked. Any write can take place and the communications are not protected. The devices are by default after POR in an unlocked state (LOCK[7:0]=0xA5).

When the LOCK[7:0] bits are not equal to 0xA5, the register map write access is locked. The register map, and therefore the full device configuration, is write- protected. Any write to an address other than 0xD will yield no result. All the register addresses, except the address 0xD, become read-only. In this case, if the user wants to change the configuration, the LOCK[7:0] bits have to be reprogrammed back to 0xA5 before sending the desired write command.

The LOCK[7:0] bits are located in the last register of the Incremental write address loop, so the user can program the whole register map, starting from 0x1 to 0xD within one continuous write sequence and then lock the configuration at the end of the sequence by writing all zeros (for example) in the address 0xD.

## 6.7 Detecting Configuration Change Through CRC-16 Checksum On Register Map and its Associated Interrupt Flag

In order to prevent internal corruption and to provide additional security on the register map configuration, the MCP3461/2/4 devices include an automatic and continuous CRC checksum calculation on the full register map Configuration bits. This calculation is not the same as the communication CRC checksum described in **Section 6.5 "Securing Read Communications Through CRC-16 Checksum"**.

This calculation takes the contents of the register map from address 0x1 to 0xE and produces a checksum which is held in the CRCCFG[15:0] bits located in the CRCCFG register (address 0xF). The CRC checksum for the register map uses the 16-bit CRC-16 ANSI polynomial as defined in the IEEE 802.3 standard: $x^{16}+x^{15}+x^{2}+1$.

Since this feature is intended for protecting the configuration of the device, this calculation is run continuously only when the register map is locked (LOCK[7:0] different than 0xA5, see **Section 6.6 "Locking/Unlocking Register Map Write Access"**). If the register map is unlocked (for example after POR), the CRCCFG[15:0] bits are cleared and no CRC is calculated.

The $\overline{\text{DR\_STATUS}}$, $\overline{\text{CRCCFG\_STATUS}}$ and $\overline{\text{POR\_STATUS}}$ bits are set to '1' (default) and the CRCCFG[15:0] bits are set to '0' (default) for this calculation, as they could vary and lead to unwanted CRC errors.

After the $\overline{\text{DR\_STATUS}}$, $\overline{\text{CRCCFG\_STATUS}}$ and $\overline{\text{POR\_STATUS}}$ bits are cleared (with a read on IRQ register), the CRC checksum on the register map can be verified by reading all registers in an incremental read sequence and by using the CRC communication: at the second incremental read loop, the checksum provided by the communication CRC should be equal to all zeros if the checksum on the register map is correct.

The checksum will be calculated for the first time in 11 DMCLK periods. This first value will then be the reference checksum value and will be latched internally, until an unlocking of the register map

happens. The checksum will then be calculated continuously every 11 DMCLK periods and checked against the reference checksum. If the checksum is different than the reference, an interrupt flag will be generated on the CRCCFG_STATUS bit within the STATUS byte on SDO, on the CRCCFG_STATUS bit in the IRQ register and on the IRQ output pin. The interrupt flag is maintained on all three mechanisms until the register map write access is unlocked.

When the part write access is unlocked, the interrupt on IRQ pin will clear immediately and the two other interrupt mechanisms will be cleared when the interrupt has been read (read STATUS byte or read IRQ register). The CRC interrupt can happen even if the IRQ pin is configured as the MDAT modulator output. In this case, the interrupt stays present and forces a logic-low output on this pin as long as the LOCK[7:0] register is locked (LOCK[7:0] <> 0xA5).

At power-up, the interrupt is not present and the register map is unlocked. As soon as the user finishes writing its configuration, the user needs to lock the register map (writing 0x00 for example in the LOCK bits) to be able to use the interrupt flag and to calculate the checksum of the register map.

## 6.8 Interrupts Description

The MCP3461/2/4 incorporate multiple interrupt mechanisms to be able to synchronize the device with a MCU and to warn against external perturbations. There are four events that can generate interrupt flags:

- conversion start
- data ready
- Power-on-Reset
- CRC error on the register map configuration

Additionally, there are three independent interrupt mechanisms that allow the devices to be implemented in many different applications and many different configurations. A summary of the different mechanisms is available on the Table 6-2 (Interrupt description summary table).

**TABLE 6-2: INTERRUPT DESCRIPTION SUMMARY TABLE**

| Interrupt Flag Type | Description | Clearing Procedure |
|---|---|---|
| STATUS Byte | Three status bits (DR_STATUS, CRCCFG_STATUS, POR_STATUS) are latched together after device address detection and are clocked out during each command on the SDO STATUS byte. | Cleared when STATUS byte clocking is finished (on the last SCK falling edge). |
| IRQ register status bits | IRQ register Status bits can be read when reading the address 0x5 (IRQ register). IRQ latching happens at the beginning of the IRQ register reading. | Cleared when the IRQ register reading is finished (on last SCK falling edge). |
| IRQ pin state | • When IRQ_MODE[1] = 0: The IRQ pin can be asserted to logic-low by any of the interrupts.<br>• When IRQ_MODE[1] = 1: Only POR and CRC interrupts can assert the IRQ pin to logic-low. | • Conversion Start interrupt is cleared automatically at the beginning of a new conversion cycle after a $T_{STP}$ timing.<br>• DR interrupt is cleared by the first SCK falling edge of an ADC read or automatically 16 DMCLK before a new data ready in Continuous Conversion mode or in SCAN mode.<br>• POR interrupt is cleared on the first CS falling edge when both $AV_{DD}$ and $DV_{DD}$ monitoring circuits are detecting their power supply to be over their respective thresholds.<br>• CRCCFG interrupt is cleared when the device is unlocked (Writing 0xA5 to LOCK register) or when a fast command ADC start/restart conversion is performed. |

### 6.8.1 CONVERSION DATA READY INTERRUPT

The data ready interrupt happens when a new conversion is ready to be read on the ADCDATA register. This event happens synchronously with DMCLK and at each end of conversion. This interrupt is implemented with three different and independent mechanisms: STATUS byte on SDO, IRQ register status bit, and $\overline{IRQ}$ pin state.

1. STATUS byte on SDO

When the interrupt happens, on the next STATUS byte transmitted on SDO, the $\overline{DR\_STATUS}$ bit is logic low. Once the STATUS byte has been transmitted, the $\overline{DR\_STATUS}$ bit appears as '1' until a new interrupt will be present. If between two STATUS byte transmissions, the interrupt happens once again, the $\overline{DR\_STATUS}$ bit on SDO will appear as '0' on the second reading.

2. IRQ register Status bit

When the interrupt happens, the bit $\overline{DR\_STATUS}$ in the IRQ register will be set to '0'. Once the IRQ register has been fully read, this $\overline{DR\_STATUS}$ bit is reset again to 1.

If between two readings of the IRQ register, the interrupt happens once again, the IRQ register Status bit will appear as equal to '0' on the second reading.

3. $\overline{IRQ}$ pin state

The interrupt generates an $\overline{IRQ}$ pin falling edge (transition to logic low) as soon as it happens.

The data ready interrupt is cleared by the first event of the following two events:

- First falling edge of SCK during an ADC output data register read
- 16 DMCLK clock periods before current conversion ends

If the user does not read the ADCDATA register in time in Continuous Conversion mode or in SCAN mode, the $\overline{IRQ}$ pin will automatically reset to its inactive state 16 DMCLK prior to the new data ready interrupt. This feature is designed to avoid the case in which the $\overline{IRQ}$ pin output would always be logic low if the reading of the ADC data were not performed. The user can determine exactly when to expect new data and can respect the $t_{DODR}$ timing in all cases to ensure proper reading of the ADC data. See Figure 6-14 for more details.

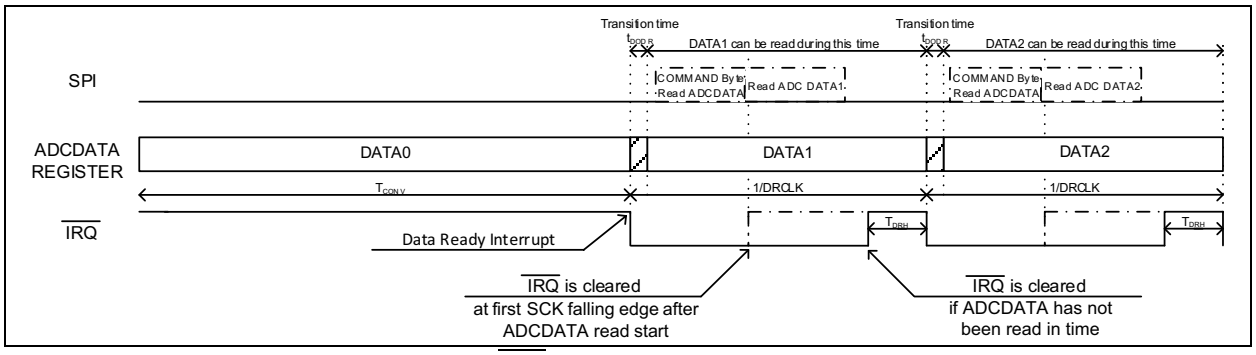



***FIGURE 6-14:*** *Data Ready Interrupt $\overline{IRQ}$ pin Timing Diagram.*

### 6.8.2 CONVERSION CYCLE START INTERRUPT

This interrupt is the only one that is selectable and the only one not present in the STATUS Byte on SDO and IRQ register. It is only available on the $\overline{IRQ}$ pin. The user can enable or disable this output using

- <EN_STP> = 1: Conversion start Interrupt output is enabled (default).
- <EN_STP> = 0: Conversion start Interrupt output is disabled

This interrupt marks the beginning of a conversion cycle. In case of a one-shot mode or continuous mode conversion in MUX mode, it marks the start of the sampling in the first conversion (happening after the ADC start-up delay of 256 DMCLK periods). In case of a SCAN mode, it marks the start of the sampling in the first conversion of the first SCAN mode cycle. The host MCU can utilize this interrupt to synchronize the start of the ADC conversion and manage synchronous events together with the conversion process. See Figure 6-15 for more details.

This interrupt output generates a falling edge on $\overline{IRQ}$ pin and is cleared automatically after a short time $T_{STP}$.

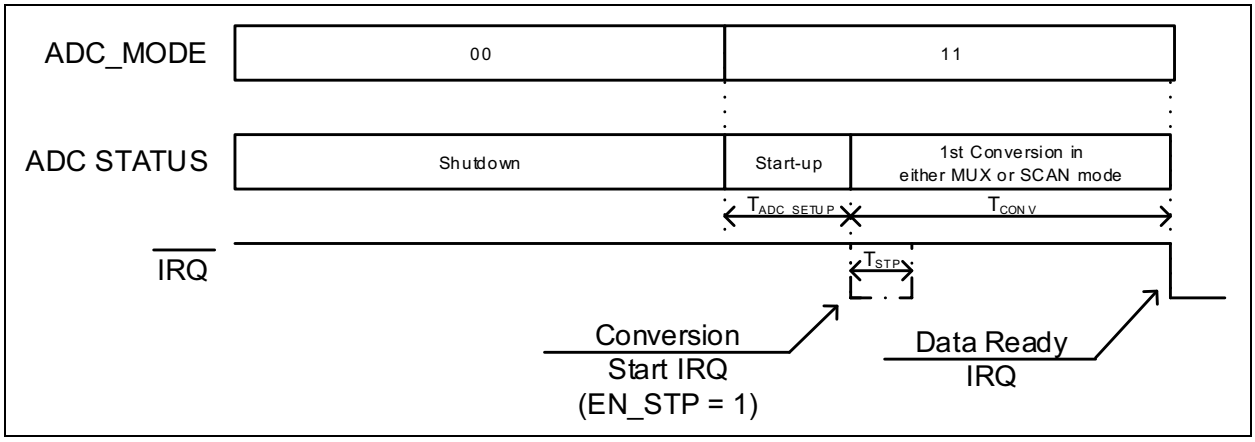



***FIGURE 6-15:*** *Conversion Start IRQ Timing Diagram.*

### 6.8.3 POR INTERRUPT

The POR interrupt provides information to the user that either a POR event has happened previously or if the part is in a POR state when the $\overline{IRQ}$ pin is used.

This interrupt is implemented with three different and independent mechanisms: STATUS byte on SDO, IRQ register Status bit, and $\overline{IRQ}$ pin state.

#### 6.8.3.1 STATUS Byte on SDO

When the device powers up, on the first STATUS byte transmitted on SDO (first communication), the $\overline{POR\_STATUS}$ is logic low. Once the STATUS byte has been transmitted, the $\overline{POR\_STATUS}$ bit appears as 1 until the part is powered down. If between two STATUS byte transmissions, a POR event happens once again, and if the part is properly re-powered up, the $\overline{POR\_STATUS}$ bit on SDO will appear as equal to 0 on the latter reading. This mechanism can only work when the power supplies are back above the POR thresholds on the analog and digital cores as retrieving data from the SPI port is not possible when the device is in POR state.

#### 6.8.3.2 IRQ register Status bit

When the device is powered up, the bit $\overline{POR\_STATUS}$ in the IRQ register will be reset to '0'. Once the IRQ register has been fully read, this $\overline{POR\_STATUS}$ is reset again to '1'.

If, between two readings of the IRQ register, a POR event happens once again, the IRQ register status bit will appear as equal to '0' on the second reading. This mechanism can only work when the power supplies are back above the POR thresholds on the analog and digital cores.

#### 6.8.3.3 $\overline{IRQ}$ pin state

A logic-low state is generated on the $\overline{IRQ}$ pin as soon as the $AV_{DD}$ or $DV_{DD}$ monitoring circuits detect a power supply drop below their specified threshold.

This POR Interrupt can only be cleared when both $AV_{DD}$ and $DV_{DD}$ are above their monitoring voltage thresholds. When this condition is met, the POR threshold is cleared by the $\overline{CS}$ falling edge. This means that if a $\overline{CS}$ falling edge does not clear the $\overline{IRQ}$ pin state, the POR event is still in effect.

This feature can be used by the user to know exactly when the chip has powered up by polling with the $\overline{CS}$ pin and checking the $\overline{IRQ}$ pin state at power-up. See Figure 6-16 for more details.

Since this is a high level priority interrupt, this POR interrupt can happen at all times, even when MDAT is enabled. In this case, having a constant logic-low bit stream can indicate in this case a probable POR event (or a fully negative ADC saturation output code induced by a large negative input voltage).

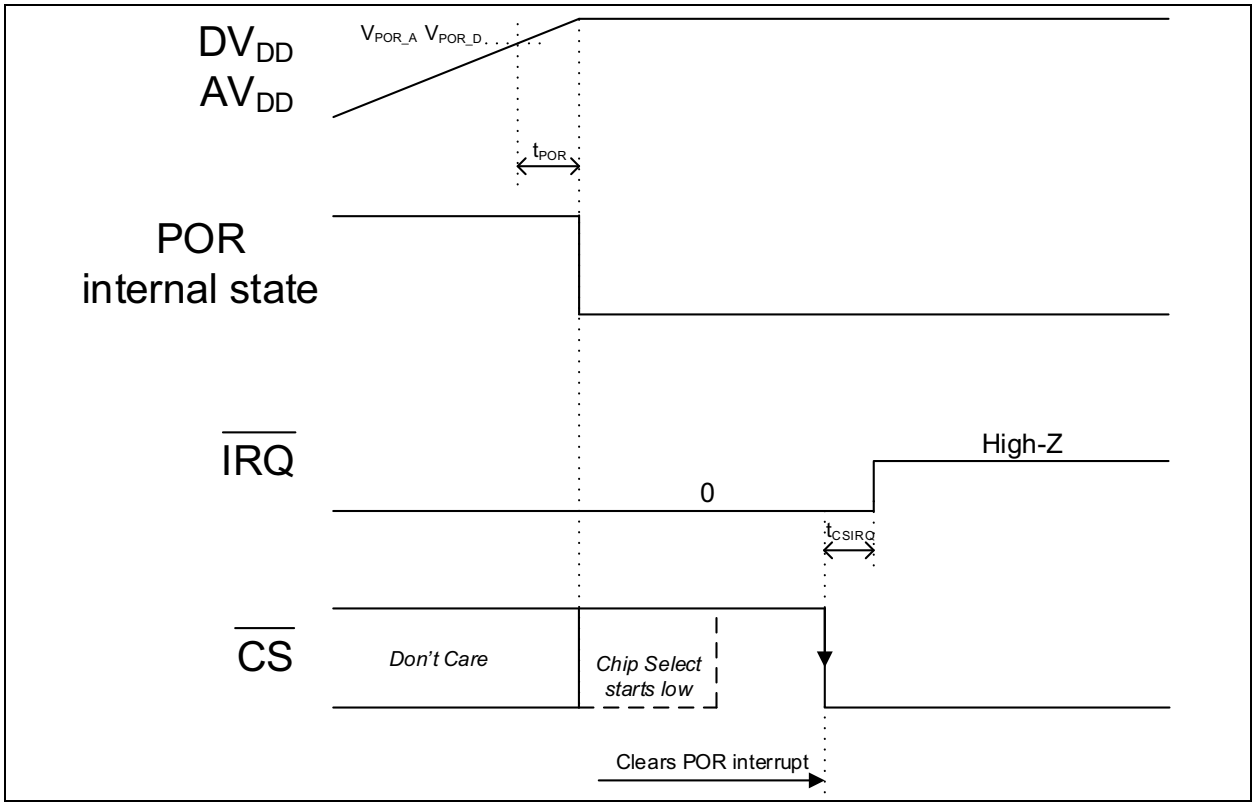



***FIGURE 6-16:*** *POR IRQ Timing Diagram.*

### 6.8.4 CRCCFG ERROR INTERRUPT

The CRCCFG Interrupt happens when an error in the CRC-16 checksum has been detected in the register map CRC calculation.

This interrupt is implemented with three different and independent mechanisms: STATUS byte on SDO, IRQ register status bit, and $\overline{IRQ}$ pin state.

#### 6.8.4.1 STATUS Byte on SDO

When the CRCCFG error happens, on the next STATUS byte transmitted on SDO, the CRCCFG_STATUS bit will be logic-low. Once the STATUS byte has been transmitted, the CRCCFG_STATUS bit will then appear as '1' until a new interrupt occurs. If between two STATUS byte transmissions, the error is detected once again, the CRCCFG_STATUS bit on SDO will appear as equal to '0' on the second reading.

#### 6.8.4.2 IRQ register Status bit

When the CRCCFG error happens, the bit CRCCFG_STATUS in the IRQ register will be set to '0'. Once the IRQ register has been fully read, this CRCCFG_STATUS bit will be reset back to '1'. If between two readings of the IRQ register, the CRCCFG error happens once again, the IRQ register status bit will appear as '0' on the second reading.

#### 6.8.4.3 $\overline{IRQ}$ pin state

The CRCCFG error generates a Logic-Low state on the $\overline{IRQ}$ pin until it is cleared. The clearing of the CRCCFG error can only be made by "unlocking" the device (by writing 0xA5 in the LOCK[7:0] register or by sending a fast command start/restart ADC conversion.). Unlocking the device stops the CRC calculation and clears the associated interrupt. Sending an ADC start/restart conversion Fast command resets the CRC calculation and clears the interrupt as well.

This CRCCFG error can only happen in case of an external perturbation (for example EMI induced) that would cause the continuous calculation of the CRC on the register map to be erroneous, or in the case that the chip integrity has been altered. Since both causes are high priority issues, the CRCCFG error takes priority over all other interrupts (except POR) and over the MDAT output on the $\overline{IRQ}$ pin.

| **Note:** | If MCLK is running before the device has been locked, an interrupt can momentarily occur even if registers have not been corrupted. In such a case, the user needs to send a start/restart conversion Fast command which will clear the unwanted interrupt and restart correctly the CRC calculations. |
|---|---|

## 7.0 BASIC APPLICATION CONFIGURATION

The MCP3461/2/4 devices can be used for various precision Analog-to-Digital Converter applications. The flexibility of its usage is given by the possibility of configuring the ADC to fit the require application.

## 7.1 Typical application for absolute voltage measurement

One application of MCP3461/2/4 is to measure signal provided by sensors with absolute voltage output. For such applications, the MCP3461/2/4 will rely on an external voltage reference. Figure 7-1 provides an example that uses MCP3464 ADC with external voltage reference MCP1501. For best performance an RC filter and operational amplifier have been placed between the OUT pin of the MCP1501 voltage reference and the REFIN+ input of the MCP3464. The op-amp is necessary because the REFIN+ input is not buffered.

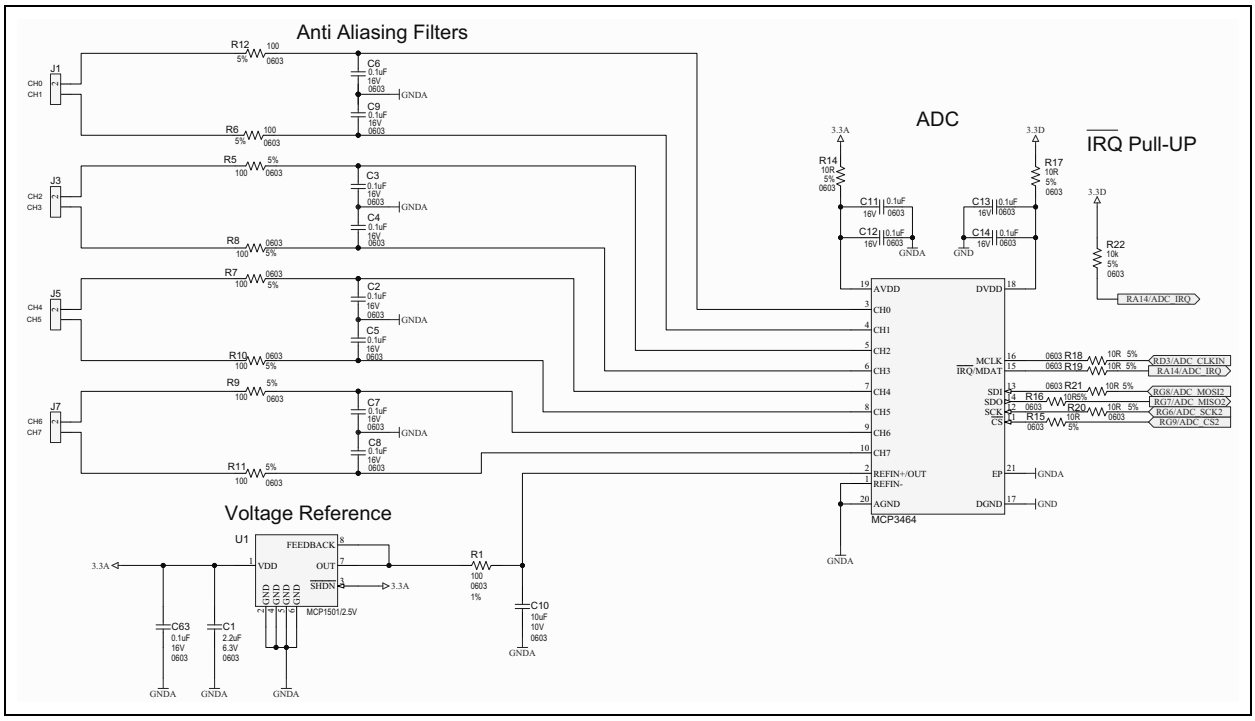



***FIGURE 7-1:*** *MCP3464 Application Example Schematic.*

The ADC can be used either in differential or Single-Ended mode, thanks to the internal dual multiplexer (Figure 5-1). The user can select the input connection settings from the MUX register (**Section 8.7 "MULTIPLEXER (MUX) Register"**) by using the different settings available on the positive and negative inputs of the ADC. The single ended configuration is achieved by selecting $A_{GND}$ for the $V_{IN-}$ input of the ADC (MUX[3:0] = `1000`) or by selecting any CHn input channel for $V_{IN-}$ and connecting the corresponding CHn input channel to $A_{GND}$.

### 7.1.1 HIGH SIDE AND LOW SIDE CURRENT SENSING

The ADC has the ability to perform differential measurements with analog input Common mode equal to or slightly larger than $AV_{DD}$ or equal to or slightly lower than $A_{GND}$ (see **Table "Electrical Characteristics"** ).

The user must use a differential input structure and Kelvin connection to achieve most accurate measurements. An anti-aliasing filter is required to avoid aliasing of the oversampling frequency (DMCLK) back into the baseband of the input signal and possible corruption of the output data. Figure 7-1 provides an example of anti-aliasing filter.

For measurement of voltages that can reach $AV_{DD}$ or few mV higher, a GAIN setting of 0.33x is useful since it will increase the input range to $3xV_{REF}$ value, so a

1.2V $V_{REF}$ will allow a theoretical input range of 3.6V. The maximum voltage that can be measured is always bounded by $AV_{DD}$+0.1V in order to limit excess leakage current at the input pins created by the ESD structures. Therefore, in order to properly measure 3.6V with a 1.2V voltage reference, it is recommended to use an $AV_{DD}$ supply voltage as close as possible to 3.6V.

### 7.1.2 THERMOCOUPLE CONNECTION

One of the most used temperature transducers used in industry is the thermocouple. The thermocouples provide a voltage dependent on the temperature difference between cold junction and hot junction. This voltage is in the order of magnitude of few tens of µV/°C, which require amplification that can be provided by the internal gain stage of the ADC.

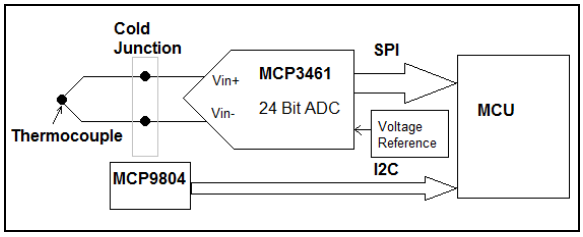



***FIGURE 7-2:*** *Thermocouple Connection to MCP3461.*

The connection of the thermocouple to the ADC require minimal extra components, and it's recommended to use a differential input structure. The cold junction can be measured using a digital temperature sensor like MCP9804 connected to the MCU. If high accuracy is not required, the cold junction temperature can be estimated directly with the internal temperature sensor of the ADC (Figure 7-2).

## 7.2 Typical Application for Ratiometric Voltage Measurement

A wide range of sensors provides an output voltage directly related to the power supply of the sensors. These sensors are known as ratiometric output. These sensors often have Wheatstone bridge structure, like pressure sensors or load cells (Figure 7-3).

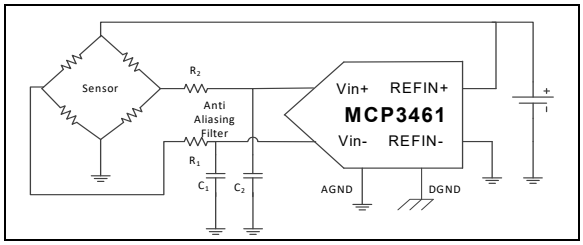



***FIGURE 7-3:*** *Wheatstone Bridge Ratiometric Connection.*

Others act as a single resistor with a value dependent on temperature (pure metal resistance thermometer RTD and negative temperature coefficient resistor NTC). To accurately measure the signal from these sensors most often the REFIN+ is connected to the same power supply of the sensor (Figure 7-4) as long as this will respect **Table "Electrical Characteristics"**.

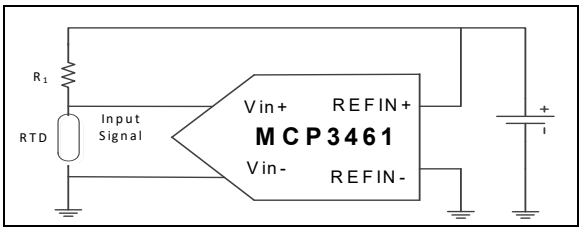



***FIGURE 7-4:*** *RTD Ratiometric Connection.*

## 7.3 Power Supply Design and Bypassing

In any system, the analog ICs (such as references or operational amplifiers) are always connected to the analog ground plane. The MCP3461/2/4 should also be considered as sensitive analog components and connected to the analog ground plane. The ADC features two pairs of power supply voltage pins: $A_{GND}$, $AV_{DD}$, $D_{GND}$ and $DV_{DD}$. For best performance, it is recommended to keep the two pairs of pins connected to two different networks (Figure 7-5). This way, the design will feature two ground traces and two power supplies (Figure 7-6).

The analog circuitry (including MCP3461/2/4) and the digital circuitry (MCU) should have separate power supplies and return paths to the external ground reference, as described in Figure 7-5. An example of a typical power supply circuit, with different paths for analog and digital return circuit, is shown in Figure 7-6. A possible split example is shown in Figure 7-7, where the ground star connection can be located underneath the device with the exposed pad. The split here between analog and digital can be done under the device, and $AV_{DD}$ and $DV_{DD}$ can be connected together with lines coming under the ground plane. The two separate return paths eventually share a unique connection point (star connection) in order to minimize coupling between the two power supply domains.

Another possibility, sometimes easier to implement in terms of PCB layout, is to consider the MCP3461/2/4 as an analog component and connect both $AV_{DD}$ and $DV_{DD}$ together, and $A_{GND}$ and $D_{GND}$ together, with a star connection. In this scheme, the decoupling capacitors may be larger, due to the ripple on the digital power supply (caused by the digital filters and the SPI interface of the MCP3461/2/4) now causing glitches on the analog power supply.

Figure 7-6 shows an example of a power supply schematic with separate $DV_{DD}$ and $AV_{DD}$. A high current LDO (MCP1825) was used for the $DV_{DD}$ line, in order to be able to power the MCU and other peripherals attached to the MCU. A high PSRR LDO is used (MCP1754) for the $AV_{DD}$ that goes to the ADC and a few other components sensitive to noise. The NET tie is used to separate $D_{GND}$ from $A_{GND}$.

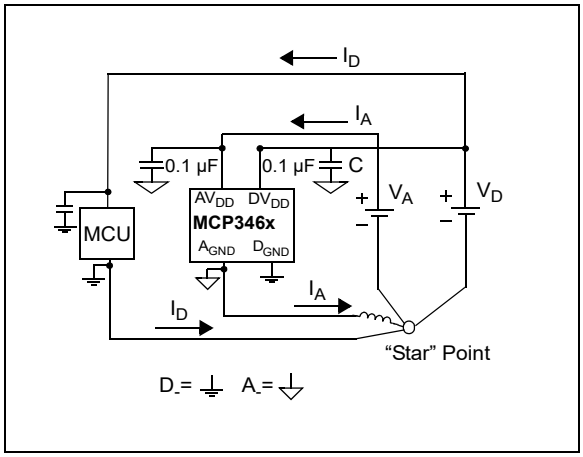



***FIGURE 7-5:*** *Separating Digital and Analog Ground by Using a Star Connection.*

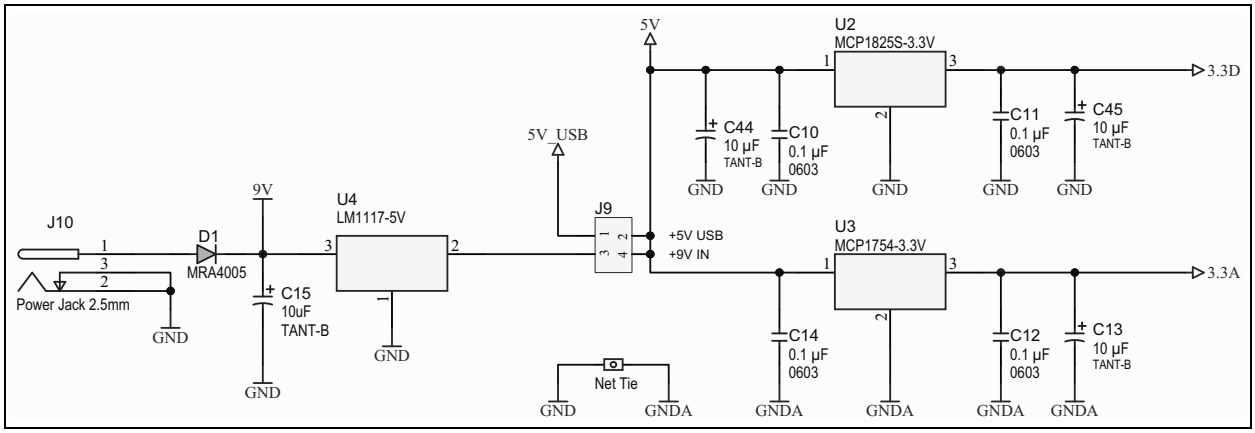



***FIGURE 7-6:*** *Power Supply with Separate Lines for Analog and Digital Sections. Note the "Net Tie" Object that Represents the Star Ground Connection.*

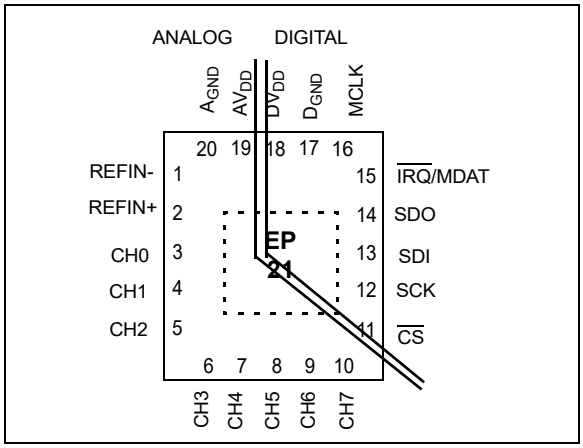



***FIGURE 7-7:*** *Separation of Analog and Digital Circuits on Layout.*

When remote sensors are used, to reduce sensitivity to external influences, such as EMI, the wires that connect the sensor to the ADC should form a twisted pair. Ferrite beads can be used between the digital and analog ground planes to keep high-frequency noise from entering the device. The ferrite bead is recommended to be low-resistance.

## 7.4 SPI Interface Digital Crosstalk

The MCP3461/2/4 incorporates a high-speed 20 MHz SPI digital interface. This interface can induce crosstalk, especially with the outer channels closer to the SPI digital pins (CH7, for example), if it is running at full speed without any precautions. The crosstalk is caused by the switching noise created by the digital SPI signals. This crosstalk would negatively impact the SNR in this case. The noise is attenuated if proper separation between the analog and digital power supplies is put in place (see **Section 7.3 "Power Supply Design and Bypassing"**).

The switching noise is also a linear function of the $DV_{DD}$ supply voltage. In order to reduce further the influence of the switching noise caused by SPI transmissions, the $DV_{DD}$ digital power supply voltage should be kept as low as possible.

In order to further remove the influence of the SPI communication on measurement accuracy, it is recommended to add series resistors on the SPI lines to reduce the current spikes caused by the digital switching noise (see Figure 7-1 where these resistors have been implemented). The resistors also help to keep the level of electromagnetic emissions low.

The measurement graphs provided in this MCP3461/2/4 data sheet have been performed with 100Ω series resistors connected on each SPI I/O pins. Measurement accuracy disturbances have not been observed even at 20 MHz interfacing.

## 8.0 INTERNAL REGISTERS

The device has a total of 16 internal registers, which are made of volatile memory. Table 8-1 shows the summary of the registers. These registers are accessible sequentially.

**TABLE 8-1: INTERNAL REGISTERS SUMMARY**

| Address | Register Name | No. of Bits | R/W | Description |
|---|---|---|---|---|
| 0x0 | ADCDATA | 4/16/32 | R | Latest A/D conversion data output value (16 or 32 bits depending on DATA_FORMAT[1:0]), or modulator output stream (4-bit wide) in MDAT Output mode. |
| 0x1 | CONFIG0 | 8 | R/W | ADC operating mode, Master Clock mode and Input Bias Current Source mode. |
| 0x2 | CONFIG1 | 8 | R/W | Prescale and OSR settings. |
| 0x3 | CONFIG2 | 8 | R/W | ADC boost and gain settings, autozeroing settings for analog multiplexer, voltage reference and ADC. |
| 0x4 | CONFIG3 | 8 | R/W | Conversion mode, data and CRC format settings, enable for CRC on communications, enable for digital offset and gain error calibrations. |
| 0x5 | IRQ | 8 | R/W | IRQ Status bits and IRQ mode settings, enable for Fast commands and for conversion start pulse. |
| 0x6 | MUX | 8 | R/W | Analog multiplexer input selection (MUX mode only). |
| 0x7 | SCAN | 24 | R/W | SCAN mode settings. |
| 0x8 | TIMER | 24 | R/W | Delay value for TIMER between each SCAN cycles. |
| 0x9 | OFFSETCAL | 24 | R/W | ADC Digital Offset calibration value. |
| 0xA | GAINCAL | 24 | R/W | ADC Digital Gain calibration value. |
| 0xB | RESERVED | 24 | R/W | |
| 0xC | RESERVED | 8 | R/W | |
| 0xD | LOCK | 8 | R/W | Password value for SPI Write mode locking. |
| 0xE | RESERVED | 16 | R/W | |
| 0xF | CRCCFG | 16 | R | CRC Checksum for the device configuration. |

## 8.1 ADCDATA REGISTER

| Name | Bits | Address | Cof |
|---|---|---|---|
| ADCDATA | 4/16/32 | 0x0 | R |

**REGISTER 8-1: ADCDATA REGISTER**

| R-0 | |
|---|---|
| ADCDATA[15:0] | |
| bit 15 | bit 0 |

| **Legend:** | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' | |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared | x = Bit is unknown |

bit 23-0 **ADCDATA[15:0]:**

Output code from ADC. The data are post-calibration if the EN_OFFCAL or EN_GAINCAL bits are enabled. The data can be formatted in 16/32-bit modes, depending on the DATA_FORMAT[1:0] settings (see Section 5.6, ADC Output Data Format).

The ADC Channel Data Output registers always contain the most recent A/D conversion data. The register is updated at each data ready internal signal (depends on OSR and CONV_MODE settings). The register is latched at start of each SPI read command. The register is double buffered to avoid loss of data. There is a small time delay $t_{DODR}$ after each data ready where the user has to wait for the data to be available. Otherwise, the data corruption can happen (when the internal data are refreshed).

When the IRQ_MODE[1:0] = 1X, this register becomes a 4-bit wide register containing the MDAT output codes, which are the outputs of the modulator that are represented by 4 comparator outputs (COMP[3:0] see **Section 5.4.2 "Modulator Output Block"**).

## 8.2 CONFIG0 REGISTER

| Name | Bits | Address | Cof |
|---|---|---|---|
| CONFIG0 | 8 | 0x1 | R/W |

**REGISTER 8-2: CONFIG0 REGISTER**

| R/W-1 | R/W-1 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| CONFIG0[7:6] | | CLK_SEL[1:0] | | CS_SEL[1:0] | | ADC_MODE[1:0] | |
| bit 7 | | | | | | | bit 0 |

| **Legend:** | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' | |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared | x = Bit is unknown |

bit 7-6 **CONFIG0[7:6]:** These bits are writable but have no effect except that they force Full-Shutdown mode when they are set to 00 and when all other CONFIG0 bits are set to 0.

bit 5-4 **CLK_SEL[1:0]:** Clock Selection bits

11 = Internal clock is selected and AMCLK is present on the analog master clock output pin.
10 = Internal clock is selected and no clock output is present on the CLK pin.
01 = External digital clock.
00 = External digital clock. **(default)**

bit 3-2 **CS_SEL[1:0]:** Current Source/Sink Selection bits for sensor bias. (Source on $V_{IN+}$/Sink on $V_{IN-}$)

11 = 15 μA is applied to the ADC inputs.
10 = 3.7 μA is applied to the ADC inputs.
01 = 0.9 μA is applied to the ADC inputs.
00 = No current source is applied to the ADC inputs. **(default)**

bit 1-0 **ADC_MODE[1:0]:** ADC operating mode Selection bits

11 = ADC Conversion mode.
10 = ADC Standby mode.
01 = ADC Shutdown mode.
00 = ADC Shutdown mode. **(default)**

## 8.3 CONFIG1 REGISTER

| Name | Bits | Address | Cof |
|---|---|---|---|
| CONFIG1 | 8 | 0x2 | R/W |

**REGISTER 8-3: CONFIG1 REGISTER**

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-1 | R/W-1 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| PRE[1:0] | | OSR[3:0] | | | | RESERVED[1:0] | |
| bit 7 | | | | | | | bit 0 |

| **Legend:** | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' | |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared | x = Bit is unknown |

bit 7-6 **PRE[1:0]:** Prescaler value selection for analog master clock (AMCLK)
11 = AMCLK = MCLK/8
10 = AMCLK = MCLK/4
01 = AMCLK = MCLK/2
00 = AMCLK = MCLK **(default)**

bit 5-2 **OSR[3:0]:** Oversampling ratio for delta sigma A/D conversion
1111 = OSR: 98304
1110 = OSR: 81920
1101 = OSR: 49152
1100 = OSR: 40960
1011 = OSR: 24576
1010 = OSR: 20480
1001 = OSR: 16384
1000 = OSR: 8192
0111 = OSR: 4096
0110 = OSR: 2048
0101 = OSR: 1024
0100 = OSR: 512
0011 = OSR: 256 **(default)**
0010 = OSR: 128
0001 = OSR: 64
0000 = OSR: 32

bit 1-0 **RESERVED[1:0]:** Should be set to '00' at all times

## 8.4 CONFIG2 REGISTER

| Name | Bits | Address | Cof |
|---|---|---|---|
| CONFIG2 | 8 | 0x3 | R/W |

**REGISTER 8-4: CONFIG2 REGISTER**

| R/W-1 | R/W-0 | R/W-0 | R/W-0 | R/W-1 | R/W-0 | R/W-1 | R/W-1 |
|---|---|---|---|---|---|---|---|
| BOOST[1:0] | | GAIN[2:0] | | | AZ_MUX | RESERVED[1:0] | |
| bit 7 | | | | | | | bit 0 |

| **Legend:** | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' | |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared | x = Bit is unknown |

bit 7-6 **BOOST[1:0]:** ADC Bias current selection
- 11 = ADC channel has current x2
- 10 = ADC channel has current x1 **(default)**
- 01 = ADC channel has current x0.66
- 00 = ADC channel has current x0.5

bit 5-3 **GAIN[2:0]:** ADC Gain selection
- 111 = Gain is x64 (x16 analog, x4 digital)
- 110 = Gain is x32 (x16 analog, x2 digital)
- 101 = Gain is x16
- 100 = Gain is x8
- 011 = Gain is x4
- 010 = Gain is x2
- 001 = Gain is x1 **(default)**
- 000 = Gain is x1/3

bit 2 **AZ_MUX:** Auto-zeroing MUX setting
- 1 = ADC auto-zeroing algorithm is enabled. This setting multiplies by two the conversion time and does not allow Continuous Conversion mode operation (which is then replaced by a series of consecutive one-shot mode conversions).
- 0 = Analog input multiplexer auto-zeroing algorithm is disabled. **(default)**

bit 1-0 **RESERVED[1:0]:** Should always be set to '11'

## 8.5 CONFIG3 REGISTER

| Name | Bits | Address | Cof |
|---|---|---|---|
| CONFIG3 | 8 | 0x4 | R/W |

**REGISTER 8-5: CONFIG3 REGISTER**

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R-0 |
|---|---|---|---|---|---|---|---|
| CONV_MODE[1:0] | | DATA_FORMAT[1:0] | | CRC_FORMAT | EN_CRCCOM | EN_OFFCAL | EN_GAINCAL |
| bit 7 | | | | | | | bit 0 |

| **Legend:** | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' | |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared | x = Bit is unknown |

bit 7-6 **CONV_MODE[1:0]:** Conversion mode Selection
- 11 = Continuous Conversion mode or continuous conversion cycle in SCAN mode.
- 10 = One-shot conversion or one-shot cycle in SCAN mode and sets ADC_MODE[1:0] to 10 (Standby) at the end of the conversion or at the end of the conversion cycle in the SCAN mode.
- 0X = One-shot conversion or one-shot cycle in SCAN mode and sets ADC_MODE[1:0] to 0X (Shutdown) at the end of the conversion or at the end of the conversion cycle in the SCAN mode. **(default).**

bit 5-4 **DATA_FORMAT[1:0]:** ADC output data format selection
- 11 = 32-bit (17-bit right justified data plus Channel ID: CHID[3:0] plus SGN extension (12 bits) plus 16-bit ADC data. Allows over-range with the SGN extension.
- 10 = 32-bit (17-bit right justified data): SGN extension (16-bit) plus 16-bit ADC data. Allows over-range with the SGN extension.
- 01 = 32-bit (16-bit left justified data): 16-bit ADC data plus 0x0000 (16-bit). Does not allow over-range (ADC code locked to 0xFFFF or 0x8000).
- 00 = 16-bit (default ADC coding): 16-bit ADC data. Does not allow over-range (ADC code locked to 0xFFFF or 0x8000). **(default)**

bit 3 **CRC_FORMAT:** CRC checksum format selection on read communications (does not affect CRCCFG coding)
- 1 = CRC-16 followed by 16 zeros (32-bit format)
- 0 = CRC-16 only (16-bit format) **(default)**

bit 2 **EN_CRCCOM:** CRC checksum selection on read communications (does not affect CRCCFG calculations)
- 1 = CRC on communication enabled.
- 0 = CRC on communication disabled **(default)**

bit 1 **EN_OFFCAL:** Enable digital offset calibration
- 1 = Enabled
- 0 = Disabled **(default)**

bit 0 **EN_GAINCAL:** ADC operating mode selection bits
- 1 = Enabled
- 0 = Disabled **(default)**

## 8.6 IRQ REGISTER

| Name | Bits | Address | Cof |
|---|---|---|---|
| IRQ | 8 | 0x5 | R/W |

**REGISTER 8-6: IRQ REGISTER**

| U-0 | R-1 | R-1 | R-1 | R/W-0 | R/W-0 | R/W-1 | R/W-1 |
|---|---|---|---|---|---|---|---|
| — | $\overline{\text{DR\_STATUS}}$ | $\overline{\text{CRCCFG\_STATUS}}$ | $\overline{\text{POR\_STATUS}}$ | IRQ_MODE[1:0] | | EN_FASTCMD | EN_STP |
| bit 7 | | | | | | | bit 0 |

| **Legend:** | | |
|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as ‘0’ |
| -n = Value at POR | ‘1’ = Bit is set | ‘0’ = Bit is cleared x = Bit is unknown |

bit 7 **Unimplemented:** Read as ‘0’

bit 6 **$\overline{\text{DR\_STATUS}}$:** Data Ready status flag bit
- 1 = ADCDATA has not been updated since last reading or last Reset **(default)**
- 0 = New ADCDATA ready for reading.

bit 5 **$\overline{\text{CRCCFG\_STATUS}}$:** CRC error status flag bit for internal registers
- 1 = CRC error has not occurred for the configuration registers. **(default)**
- 0 = CRC error has occurred for the configuration registers.

bit 4 **$\overline{\text{POR\_STATUS}}$:** POR status flag bit
- 1 = POR has not occurred since the last reading. **(default)**
- 0 = POR has occurred since the last reading.

bit 3-2 **IRQ_MODE[1:0]:** Configuration for the $\overline{\text{IRQ}}$/MDAT pin.(1)

**IRQ_MODE[1]:** $\overline{\text{IRQ}}$/MDAT selection
- 1 = MDAT output is selected. Only POR and CRC interrupts can be present on this pin and take priority over MDAT output.
- 0 = $\overline{\text{IRQ}}$ output is selected. All interrupts can appear on the $\overline{\text{IRQ}}$/MDAT pin. **(default)**

**IRQ_MODE[0]:** $\overline{\text{IRQ}}$ pin Inactive state selection
- 1 = The inactive state is logic High. (does not require a pull-up resistor to $DV_{DD}$)
- 0 = The inactive state is High Z (requires a pull-up resistor to $DV_{DD}$). **(default)**

bit 1 **EN_FASTCMD:** Enable fast commands in the command byte.
- 1 = Fast commands are enabled. **(default)**
- 0 = Fast commands are disabled.

bit 0 **EN_STP:** Enable Conversion start Interrupt output
- 1 = Enabled **(default)**
- 0 = Disabled

**Note 1:** When IRQ_MODE[1:0] = 10 or 11, the modulator output codes (MDAT stream) are available at both MDAT pin and ADCDATA register (0x0).

## 8.7 MULTIPLEXER (MUX) REGISTER

| Name | Bits | Address | Cof |
|---|---|---|---|
| MUX | 8 | 0x6 | R/W |

**REGISTER 8-7: MUX REGISTER**

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-1 |
|---|---|---|---|---|---|---|---|
| MUX_VIN+[3:0] | | | | MUX_VIN-[3:0] | | | |
| bit 7 | | | | | | | bit 0 |

| **Legend:** | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' | |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared | x = Bit is unknown |

bit 7-4 **MUX $V_{IN+}$ Input Selection:**(2, 3)
1111 = Internal VCM
1110 = Internal Temperature Sensor diode M (Temp Diode M)(1)
1101 = Internal Temperature Sensor diode P (Temp Diode P)(1)
1100 = REFIN-
1011 = REFIN+
1010 = Reserved (do not use)
1001 = $AV_{DD}$
1000 = $A_{GND}$
0111 = CH7
0110 = CH6
0101 = CH5
0100 = CH4
0011 = CH3
0010 = CH2
0001 = CH1
0000 = CH0 **(default)**

bit 3-0 **MUX $V_{IN-}$ Input Selection:** (2, 3)
1111 = Internal VCM
1110 = Internal Temperature Sensor diode M (Temp Diode M)(1)
1101 = Internal Temperature Sensor diode P (Temp Diode P)(1)
1100 = REFIN-
1011 = REFIN+
1010 = Reserved (do not use)
1001 = $AV_{DD}$
1000 = $A_{GND}$
0111 = CH7
0110 = CH6
0101 = CH5
0100 = CH4
0011 = CH3
0010 = CH2
0001 = CH1 **(default)**
0000 = CH0

**Note 1:** Selects the internal temperature sensor diode and forces a fixed current through it. For a correct temperature reading, the MUX[7:0] selection should be equal to 0xDE.
**2:** For MCP3462, the codes 0111/0110/0101/0100 correspond to a floating input and should be avoided.
**3:** For MCP3461, the codes 0111/0110/0101/0100/0011/0010 correspond to a floating input and should be avoided.

## 8.8 SCAN REGISTER

| Name | Bits | Address | Cof |
|---|---|---|---|
| SCAN | 24 | 0x7 | R/W |

**REGISTER 8-8: SCAN REGISTER**

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | U-0 |
|---|---|---|---|---|
| DLY[2:0] | | | RESERVED | — |
| bit 23 | | | | bit 16 |

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| OFFSET | VCM | $AV_{DD}$ | TEMP | SCAN_DIFF_CHD | SCAN_DIFF_CHC | SCAN_DIFF_CHB | SCAN_DIFF_CHA |
| bit 15 | | | | | | | bit 8 |

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R-0 |
|---|---|---|---|---|---|---|---|
| SCAN_SE_CH7 | SCAN_SE_CH6 | SCAN_SE_CH5 | SCAN_SE_CH4 | SCAN_SE_CH3 | SCAN_SE_CH2 | SCAN_SE_CH1 | SCAN_SE_CH0 |
| bit 7 | | | | | | | bit 0 |

| **Legend:** | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' | |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared | x = Bit is unknown |

bit 23-21 **DLY[1:0]:** Delay time ($T_{DLY\_SCAN}$) between each conversion during a SCAN cycle.
111 = 512 * DMCLK
110 = 256 * DMCLK
101 = 128 * DMCLK
100 = 64 * DMCLK
011 = 32 * DMCLK
010 = 16 * DMCLK
001 = 8 * DMCLK
000 = 0: No Delay **(default)**

bit 20 **Reserved:** Should be set to '0'

bit 19-16 **Unimplemented:** Read as '0'

bit 15-0 **SCAN channel selection bits:** See Table 5-13 for the complete description of the list.

## 8.9 TIMER REGISTER

| Name | Bits | Address | Cof |
|---|---|---|---|
| TIMER | 24 | 0x8 | R/W |

**REGISTER 8-9: TIMER REGISTER**

| R/W-0 | | | |
|---|---|---|---|
| TIMER[23:0] | | | |
| bit 23 | | | bit 0 |

| **Legend:** | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' | |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared | x = Bit is unknown |

bit 23-0 **TIMER[23:0]:** Selection bits for the time-interval ($T_{TIMER\_SCAN}$) between two consecutive SCAN cycles (when CONV_MODE[1:0] = 11).

0xFFFFFF: $T_{TIMER\_SCAN}$ = 16777215 * DMCLK.
0xFFFFFE: $T_{TIMER\_SCAN}$ = 16777214 * DMCLK.

.
.
.

0x000002: $T_{TIMER\_SCAN}$ = 2 * DMCLK.
0x000001: $T_{TIMER\_SCAN}$ = 1 * DMCLK.
0x000000: $T_{TIMER\_SCAN}$ = 0: No Delay **(default).**

## 8.10 OFFSET CALIBRATION REGISTER

| Name | Bits | Address | Cof |
|---|---|---|---|
| OFFSETCAL | 24 | 0x9 | R/W |

**REGISTER 8-10: OFFSETCAL REGISTER**

| R/W-0 |
|---|
| OFFSETCAL[23:0] |
| bit 23 ... bit 0 |

| **Legend:** | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' | |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared | x = Bit is unknown |

bit 23-0 **OFFSETCAL[23:0]:** Offset error digital calibration code (2's complement MSB first coding). See **Section 5.12 "Digital System Offset And Gain Calibrations"**.

## 8.11 GAIN CALIBRATION REGISTER

| Name | Bits | Address | Cof |
|---|---|---|---|
| GAINCAL | 24 | 0xA | R/W |

**REGISTER 8-11: GAINCAL REGISTER**

| R/W-1 | R/W-0 |
|---|---|
| GAINCAL[23:0] | |
| bit 23 ... bit 0 | |

| **Legend:** | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' | |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared | x = Bit is unknown |

bit 23-0 **GAINCAL[23:0]:** Gain error digital calibration code (Unsigned MSB first coding). GAINCAL[23:0] default value is 800000, which provides a gain of 1x. See **Section 5.12 "Digital System Offset And Gain Calibrations"**.

## 8.12 RESERVED REGISTER

| Name | Bits | Address | Cof |
|---|---|---|---|
| RESERVED | 24 | 0xB | R/W |

**REGISTER 8-12: RESERVED REGISTER**

| R/W-0x900000 | |
|---|---|
| RESERVED[23:0] | |
| bit 23 | bit 0 |

| **Legend:** | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as ‘0’ | |
| -n = Value at POR | ‘1’ = Bit is set | ‘0’ = Bit is cleared | x = Bit is unknown |

Bit 23-0 **RESERVED[23:0]**: Should be set to 0x900000

## 8.13 RESERVED REGISTER

| Name | Bits | Address | Cof |
|---|---|---|---|
| RESERVED | 8 | 0xC | R/W |

**REGISTER 8-13: RESERVED REGISTER**

| R/W-0x50 | |
|---|---|
| RESERVED[7:0] | |
| bit 7 | bit 0 |

| **Legend:** | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as ‘0’ | |
| -n = Value at POR | ‘1’ = Bit is set | ‘0’ = Bit is cleared | x = Bit is unknown |

Bit 7-0 **RESERVED[7:0]**: Should be set to 0x50

## 8.14 Lock Register

| Name | Bits | Address | Cof |
|---|---|---|---|
| LOCK | 8 | 0xD | R/W |

**REGISTER 8-14: LOCK REGISTER**

| R/W-1 | R/W-0 | R/W-1 | R/W-0 | R/W-0 | R/W-1 | R/W-0 | R/W-1 |
|---|---|---|---|---|---|---|---|
| LOCK[7:0] | | | | | | | |
| bit 7 | | | | | | | bit 0 |

| **Legend:** | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' | |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared | x = Bit is unknown |

bit 7-0 **LOCK[7:0]:** Write Access Password Entry Code

0xA5 = Write access is allowed on the full register map. CRC on register map values is not calculated (CRCCFG[15:0] = 0x0000). **(default)**

Any code except 0xA5: Write access is not allowed on the full register map. Only LOCK register is writable. CRC on register map is calculated continuously only when DMCLK is running.

## 8.15 RESERVED REGISTER

| Name | Bits | Address | Cof |
|---|---|---|---|
| RESERVED | 16 | 0xE | R/W |

**REGISTER 8-15: RESERVED REGISTER**

| R/W (default depends on product denomination) | |
|---|---|
| RESERVED[15:0] | |
| bit 15 | bit 0 |

| **Legend:** | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' | |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared | x = Bit is unknown |

Bit 15-0 **RESERVED[15:0]**: Should be set to:
MCP3461: 0x0008
MCP3462: 0x0009
MCP3464: 0x000B

## 8.16 CRC CONFIGURATION REGISTER

| Name | Bits | Address | Cof |
|---|---|---|---|
| CRCCFG | 16 | 0xF | R |

**REGISTER 8-16: CRCCFG REGISTER**

| R/W-0 | | | |
|---|---|---|---|
| CRCCFG[15:0] | | | |
| bit 15 | | | bit 0 |

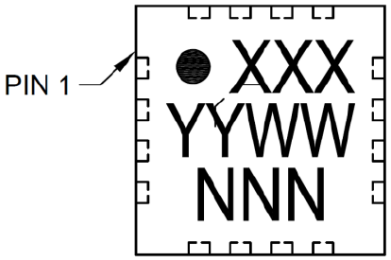
| **Legend:** | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' | |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared | x = Bit is unknown |

bit 15-0 **CRCCFG[15:0]:** CRC-16 Checksum continuously calculated internally based on the register map configuration settings when the device is locked (LOCK[7:0] different than 0xA5).

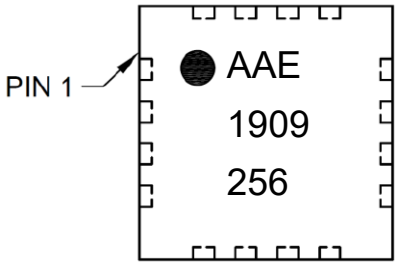
# 9.0 PACKAGING INFORMATION

## 9.1 Package Marking Information

20-Lead UQFN (3 mm x 3 mm)




Example:




| Part Number | Code | SPI Device Address |
|---|---|---|
| MCP3461T-E/NC | AAE | 01(2) |
| MCP3462T-E/NC | AAF | 01(2) |
| MCP3464T-E/NC | AAG | 01(2) |

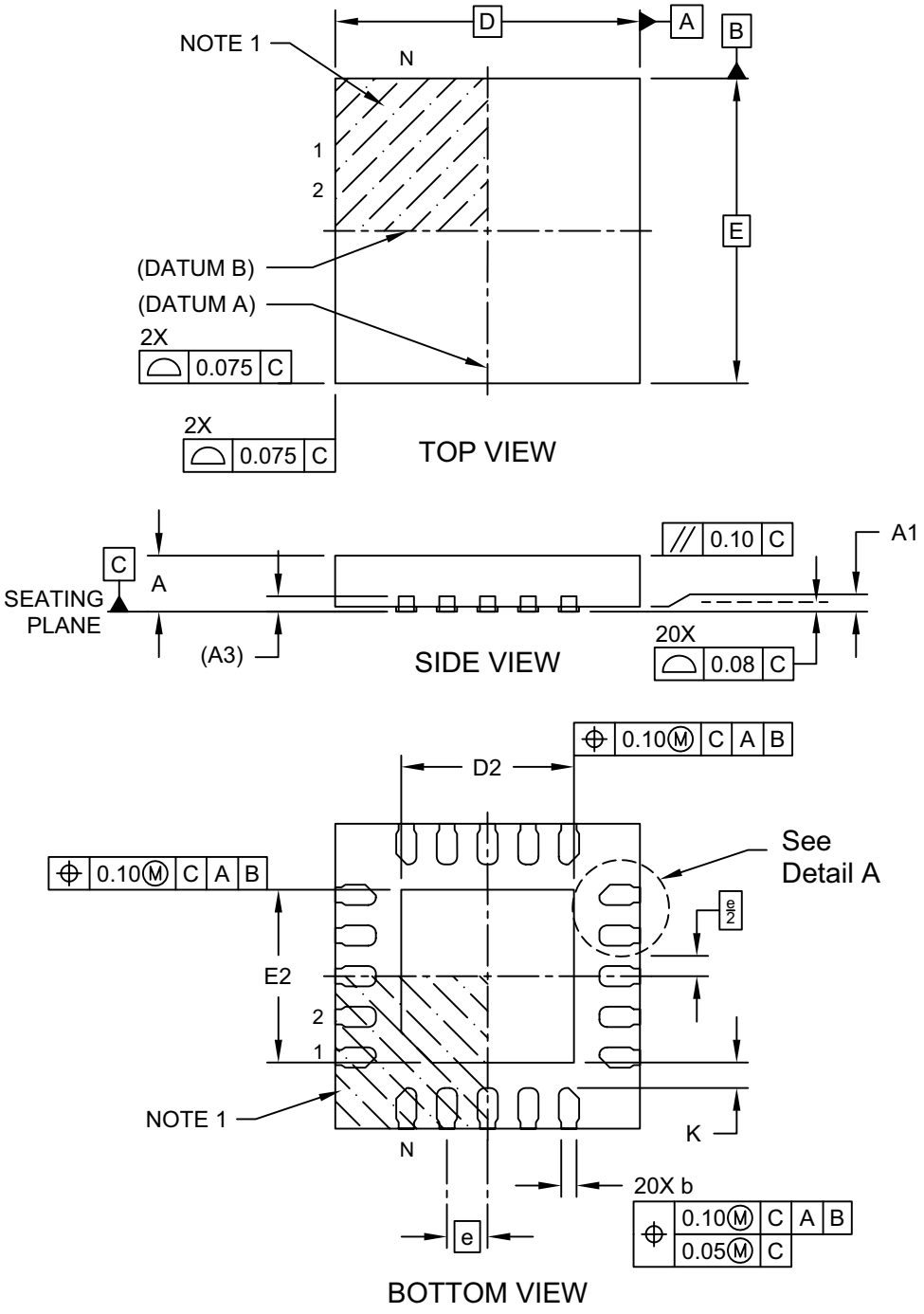
| | | |
|---|---|---|
| **Legend:** | XX...X | Customer-specific information |
| | Y | Year code (last digit of calendar year) |
| | YY | Year code (last 2 digits of calendar year) |
| | WW | Week code (week of January 1 is week ‘01’) |
| | NNN | Alphanumeric traceability code |
| | (e3) | Pb-free JEDEC designator for Matte Tin (Sn) |
| | * | This package is Pb-free. The Pb-free JEDEC designator ((e3)) can be found on the outer packaging for this package. |

**Note 1:** In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.

**2:** Denotes the device default SPI Address option. Device only responds to SPI commands if CMD[7:6] matches the SPI device address for each command (see **Section 6.2.2 “Device Address BITS (CMD[7:6])”**).

### 20-Lead Ultra Thin Plastic Quad Flat, No Lead Package (NC) - 3x3 mm Body [UQFN] (Formerly Q3DE; SST Legacy Package)

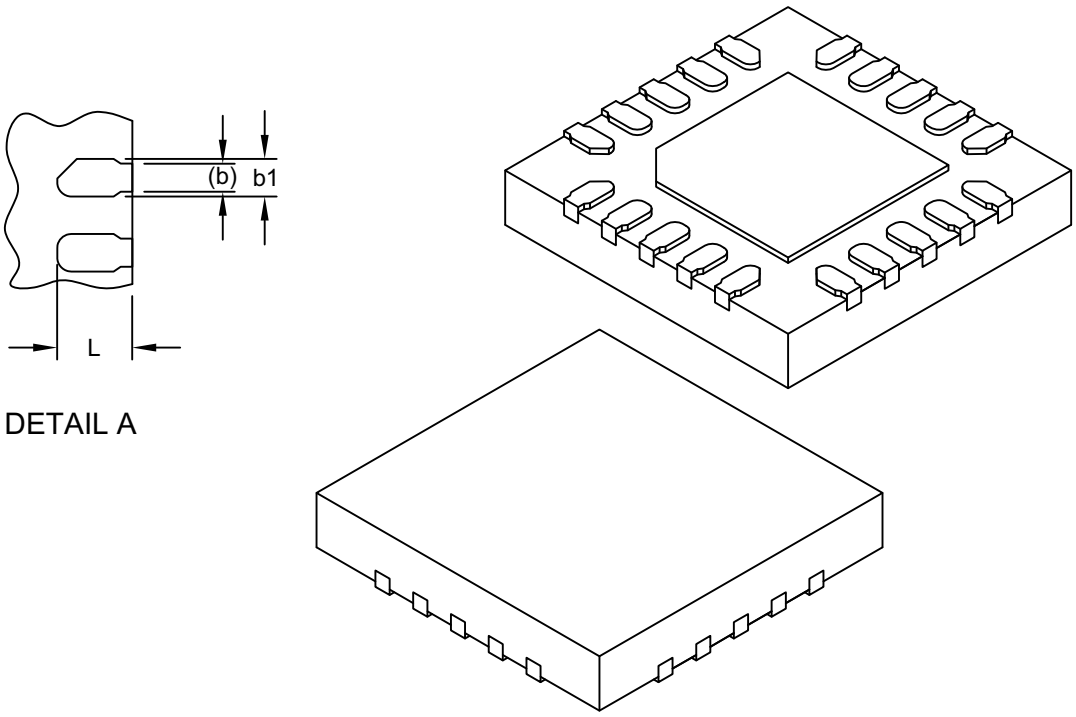
**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




Microchip Technology Drawing C04-264A Sheet 1 of 2

## 20-Lead Ultra Thin Plastic Quad Flat, No Lead Package (NC) - 3x3 mm Body [UQFN] (Formerly Q3DE; SST Legacy Package)

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Number of Terminals | N | | 20 | |
| Pitch | e | | 0.40 BSC | |
| Overall Height | A | 0.50 | 0.55 | 0.60 |
| Standoff | A1 | 0.00 | 0.02 | 0.05 |
| Terminal Thickness | A3 | | 0.15 REF | |
| Overall Length | D | | 3.00 BSC | |
| Exposed Pad Length | D2 | 1.60 | 1.70 | 1.80 |
| Overall Width | E | | 3.00 BSC | |
| Exposed Pad Width | E2 | 1.60 | 1.70 | 1.80 |
| Terminal Width (Inner) | b | | 0.15 REF | |
| Terminal Width (Outer) | b1 | 0.15 | 0.20 | 0.25 |
| Terminal Length | L | 0.35 | 0.40 | 0.45 |
| Terminal-to-Exposed-Pad | K | 0.20 | - | - |

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M

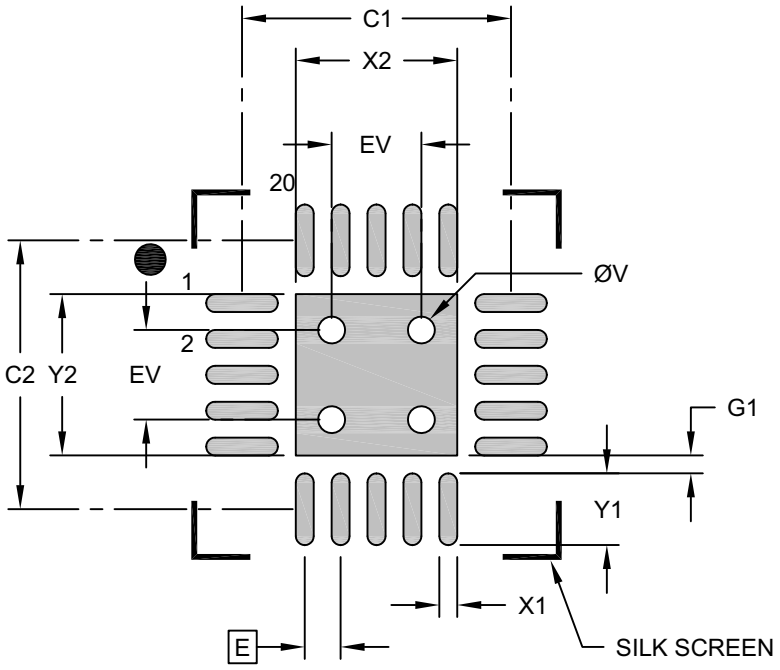
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-264A Sheet 2 of 2

## 20-Lead Ultra Thin Plastic Quad Flat, No Lead Package (NC) - 3x3 mm Body [UQFN] (Formerly Q3DE; SST Legacy Package)

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




RECOMMENDED LAND PATTERN

| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Contact Pitch | E | | 0.40 BSC | |
| Optional Center Pad Width | X2 | | | 1.80 |
| Optional Center Pad Length | Y2 | | | 1.80 |
| Contact Pad Spacing | C1 | | 3.00 | |
| Contact Pad Spacing | C2 | | 3.00 | |
| Contact Pad Width (X20) | X1 | | | 0.20 |
| Contact Pad Length (X20) | Y1 | | | 0.80 |
| Contact Pad to Center Pad (X20) | G1 | 0.20 | | |
| Thermal Via Diameter | V | | 0.30 | |
| Thermal Via Pitch | EV | | 1.00 | |

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
2. For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

Microchip Technology Drawing C04-2264A

# APPENDIX A: REVISION HISTORY

## Revision A (March 2019)

- Initial release of this document.

**NOTES:**

## PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.

| PART NO. | X(1) | — | X | /XX |
|---|---|---|---|---|
| Device | Tape and Reel | | Temperature Range | Package |

| | | |
|---|---|---|
| **Device:** | MCP3461/2/4: | One/Two/Four-Differential Channel High Precision 16-bit Delta Sigma ADCs, with SPI device address ‘01’ |
| **Tape and Reel:** | T = | Tape and Reel(1) |
| **Temperature Range:** | E = | -40°C to +125°C (Extended) |
| **Package:** | NC = | Ultra Small Leadless Package, 3 mm x 3 mm UQFN-20 |

**Examples:**

a) MCP3461T-E/NC: Single-Channel ADC, Tape and Reel, Extended Temperature, 20LD UQFN

b) MCP3462T-E/NC: Dual-Channel ADC, Tape and Reel, Extended Temperature, 20LD UQFN

c) MCP3464T-E/NC: Quad-Channel ADC. Tape and Reel, Extended Temperature, 20LD UQFN

**Note 1:** Tape and Reel identifier only appears in the catalog part number description. This identifier is used for ordering purposes and is not printed on the device package. Check with your Microchip Sales Office for package availability with the Tape and Reel option.

**2:** The device SPI Address ‘01’ is the default address option. Contact Microchip Sales for other device address option ordering procedure.

**NOTES:**

**Note the following details of the code protection feature on Microchip devices:**

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.



*Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.*

**QUALITY MANAGEMENT SYSTEM**
**CERTIFIED BY DNV**
**═ ISO/TS 16949 ═**

**Trademarks**



ISBN: 978-1-5224-4276-9

# Worldwide Sales and Service

## AMERICAS

**Corporate Office**
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/support
Web Address:
www.microchip.com

**Atlanta**
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

**Austin, TX**
Tel: 512-257-3370

**Boston**
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

**Chicago**
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

**Dallas**
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

**Detroit**
Novi, MI
Tel: 248-848-4000

**Houston, TX**
Tel: 281-894-5983

**Indianapolis**
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453
Tel: 317-536-2380

**Los Angeles**
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608
Tel: 951-273-7800

**Raleigh, NC**
Tel: 919-844-7510

**New York, NY**
Tel: 631-435-6000

**San Jose, CA**
Tel: 408-735-9110
Tel: 408-436-4270

**Canada - Toronto**
Tel: 905-695-1980
Fax: 905-695-2078

## ASIA/PACIFIC

**Australia - Sydney**
Tel: 61-2-9868-6733

**China - Beijing**
Tel: 86-10-8569-7000

**China - Chengdu**
Tel: 86-28-8665-5511

**China - Chongqing**
Tel: 86-23-8980-9588

**China - Dongguan**
Tel: 86-769-8702-9880

**China - Guangzhou**
Tel: 86-20-8755-8029

**China - Hangzhou**
Tel: 86-571-8792-8115

**China - Hong Kong SAR**
Tel: 852-2943-5100

**China - Nanjing**
Tel: 86-25-8473-2460

**China - Qingdao**
Tel: 86-532-8502-7355

**China - Shanghai**
Tel: 86-21-3326-8000

**China - Shenyang**
Tel: 86-24-2334-2829

**China - Shenzhen**
Tel: 86-755-8864-2200

**China - Suzhou**
Tel: 86-186-6233-1526

**China - Wuhan**
Tel: 86-27-5980-5300

**China - Xian**
Tel: 86-29-8833-7252

**China - Xiamen**
Tel: 86-592-2388138

**China - Zhuhai**
Tel: 86-756-3210040

## ASIA/PACIFIC

**India - Bangalore**
Tel: 91-80-3090-4444

**India - New Delhi**
Tel: 91-11-4160-8631

**India - Pune**
Tel: 91-20-4121-0141

**Japan - Osaka**
Tel: 81-6-6152-7160

**Japan - Tokyo**
Tel: 81-3-6880- 3770

**Korea - Daegu**
Tel: 82-53-744-4301

**Korea - Seoul**
Tel: 82-2-554-7200

**Malaysia - Kuala Lumpur**
Tel: 60-3-7651-7906

**Malaysia - Penang**
Tel: 60-4-227-8870

**Philippines - Manila**
Tel: 63-2-634-9065

**Singapore**
Tel: 65-6334-8870

**Taiwan - Hsin Chu**
Tel: 886-3-577-8366

**Taiwan - Kaohsiung**
Tel: 886-7-213-7830

**Taiwan - Taipei**
Tel: 886-2-2508-8600

**Thailand - Bangkok**
Tel: 66-2-694-1351

**Vietnam - Ho Chi Minh**
Tel: 84-28-5448-2100

## EUROPE

**Austria - Wels**
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

**Denmark - Copenhagen**
Tel: 45-4450-2828
Fax: 45-4485-2829

**Finland - Espoo**
Tel: 358-9-4520-820

**France - Paris**
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

**Germany - Garching**
Tel: 49-8931-9700

**Germany - Haan**
Tel: 49-2129-3766400

**Germany - Heilbronn**
Tel: 49-7131-67-3636

**Germany - Karlsruhe**
Tel: 49-721-625370

**Germany - Munich**
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

**Germany - Rosenheim**
Tel: 49-8031-354-560

**Israel - Ra'anana**
Tel: 972-9-744-7705

**Italy - Milan**
Tel: 39-0331-742611
Fax: 39-0331-466781

**Italy - Padova**
Tel: 39-049-7625286

**Netherlands - Drunen**
Tel: 31-416-690399
Fax: 31-416-690340

**Norway - Trondheim**
Tel: 47-7288-4388

**Poland - Warsaw**
Tel: 48-22-3325737

**Romania - Bucharest**
Tel: 40-21-407-87-50

**Spain - Madrid**
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

**Sweden - Gothenberg**
Tel: 46-31-704-60-40

**Sweden - Stockholm**
Tel: 46-8-5090-4654

**UK - Wokingham**
Tel: 44-118-921-5800
Fax: 44-118-921-5820