Description

CambridgeIC's Central Tracking Unit (CTU) chip is a single-chip processor for position measurement. It implements the electronic processing for resonant inductive position sensing technology.

The CTU measures the position of contactless, inductively coupled targets relative to sensors that are built from printed circuit boards to CambridgeIC's design. The CTU supports a number of different sensor types, including rotary and linear.

Features

- Resonant inductive position sensing engine
- Fully ratiometric measurements
- Automatic tuning to target frequency
- SPI communications (slave device)
- User IOs for position triggers and sample indicators
- Measures up to 4 sensors
- Can drive external DAC for up to 4 analog outputs
- Internal software upgradable over SPI

Performance

- Noise Free Resolution 10…14 bits at typical gap
- Tunes to resonators across ±7% frequency range
- < ±0.1% position change across temperature

Product identification	
Part no.	**Description**
Please enquire	CAM204BE in 28-pin SSOP package
	Delivery on Tape and Reel

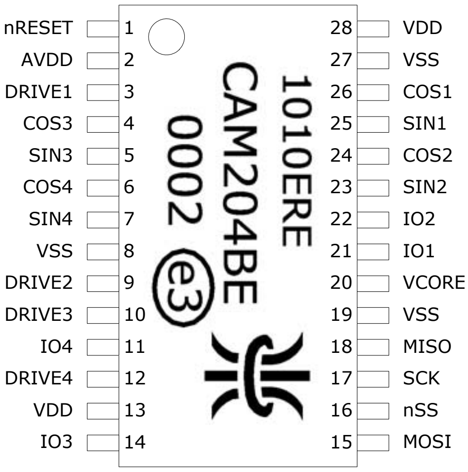


Figure 1 CAM204 28-pin SSOP pinout

Figure 2 CAM204 28-pin SSOP

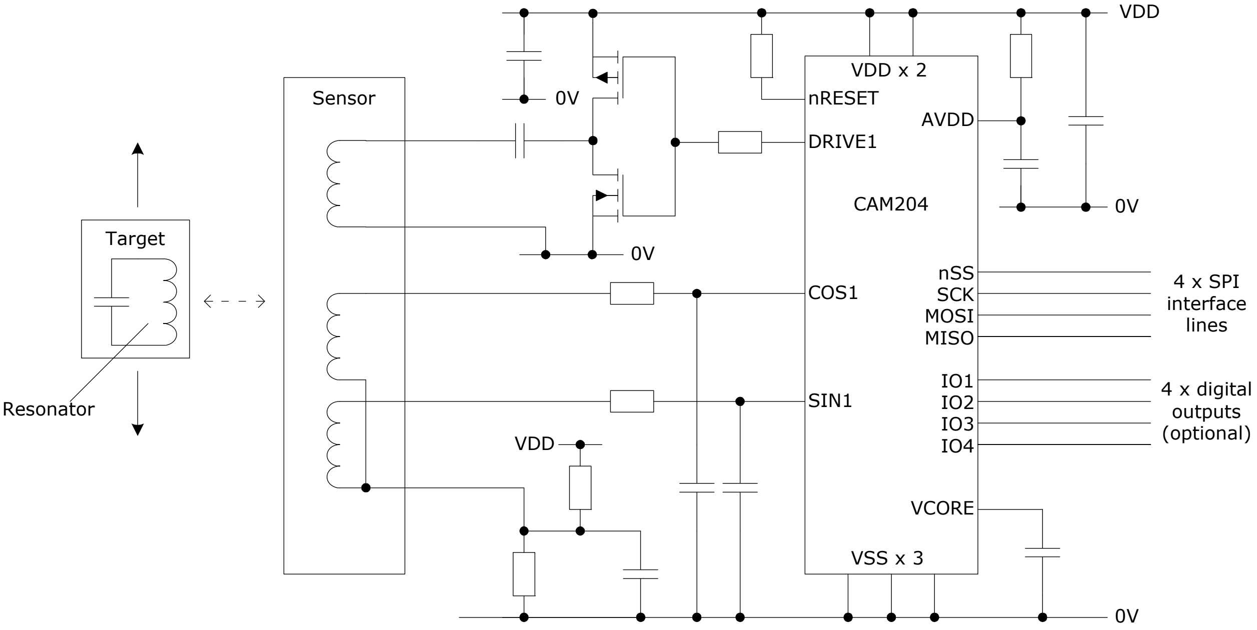


Figure 3 circuit for reading a single Type 1 sensor

1 Resonant Inductive Position Sensing Overview

1.1 System Components

Resonant Inductive Position Sensing is used to track the position of a target without mechanical or electrical contact. The target includes an inductively coupled resonator whose position is measured relative to a sensor.

The sensor comprises a number of coils. It is usually implemented using conventional Printed Circuit Board (PCB) technology, when it is referred to as a Sensor PCB.

A Central Tracking Unit (CTU) chip interacts with the sensor to power the resonator and to detect the signals that it returns. The detected amplitudes of these signals are processed to calculate position.

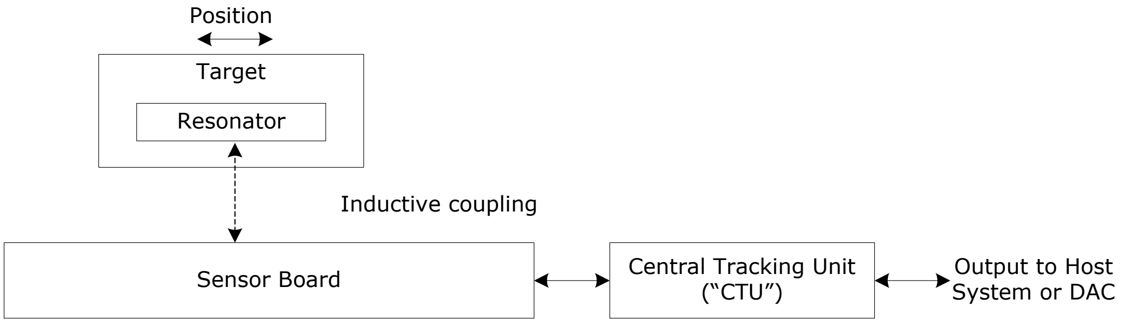


Figure 4 resonant inductive position sensing system

1.2 Sensor Types

The CAM204 chip can be connected to different Types of sensor. Depending on sensor Type, more than 1 sensor may be connected. They must all be the same Type.

Type 1 sensors comprise 3 coils. The excitation coil is used to power the resonator in the target, and a single pair of sensor coils is used to detect resonator signals. The CTU chip measures their amplitudes and uses them to calculate position.

Type 2 sensors are for measuring linear position over longer distances, and use two pairs of sensor coils to determine "fine" and "coarse" position.

Type 3 sensors are for measuring linear position when the target is allowed to rotate. The rotation axis is along the measuring direction. This enables a target to be embedded in a cylindrical float for liquid level applications, for example.

Type 4 sensors are physically the same as Type 1, except their 0V and VREF connections are shared so that only 4 connections are required between the sensor and CTU circuit

Table 1 lists the sensor types currently supported by the CAM204, and includes section references for more details on each Type.

Table 1 comparison of sensor Types

Type	**Number of coils**		**Application**	**Max number of sensors (CAM204)**	**Section number for details**
	Excitation	**Sensing**			
1	1	2	Simple linear and rotary sensing	4	4
2	1	4	Long linear sensors	1	5
3	2	2	Linear sensors, freely rotating target	1	6
4	1	2	Simple linear and rotary sensing, 4-wire sensor connection	2	7

1.3 Electronic Interrogation Method

Sensor Types differ in their details, but the same basic measuring principle applies to them all. The CTU generates signals that drive external miniature MOSFETs. These, in turn, drive AC current into the sensor's excitation coil. The excitation coil current generates an AC field which powers the resonator at its resonant frequency. The energy in the resonator is built up during this *pulse*.

Then the current is removed, and the resonator induces decaying EMFs in the sensor coils. The CTU detects the amplitude of this *echo* in each sensor coil. It then uses the amplitude values to calculate position.

The details of the measuring process and calculation depend on sensor Type. In all cases, sensing and calculation are fully ratiometric for immunity to changes in amplitude due to gap, misalignment, temperature, target frequency and supply voltage.

The *pulse echo* interrogation method separates the excitation and detection processes in time. This yields immunity from stray coupling between excitation and sensor coils, and superior sensing performance.

When the sensor Type allows, the CTU may be connected to more than one sensor. In this case, sensors are measured by the CTU in sequence.

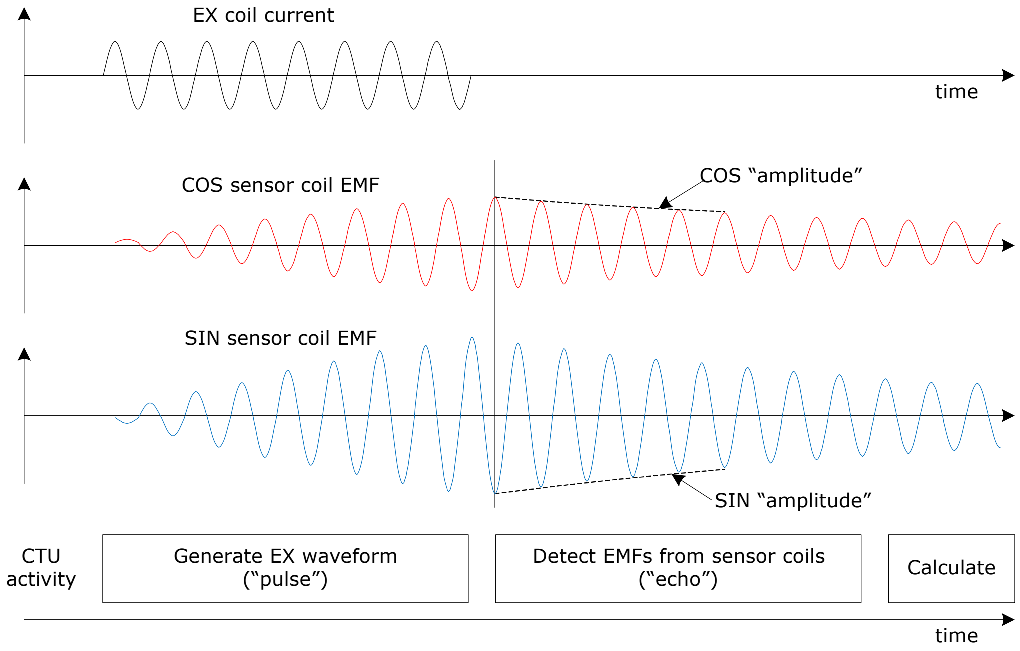


Figure 5 electronic measuring process

1.4 Resonator Frequency Search and Lock

The CAM204 CTU chip is designed to work with resonators having a high Q-factor (see section 9 for specifications). This means that the AC EMFs generated in sensor coils are large: typically on the order of a volt in amplitude. This yields high resolution and immunity from interference.

The CAM204 CTU chip is also designed for operation across a wide tuning range (see section 9). This means that the resonant frequency need not be particularly well controlled, which saves cost at manufacture and accommodates changes of frequency in use.

In order to achieve both of these beneficial features, the CAM204 can change the frequency it uses to excite and detect the resonator within its tuning range. This means it will always operate on resonance, with optimum signal amplitude.

The first few times the CTU performs a measurement on a particular target (after a reset or target out of range) it performs an optimised search of its frequency range for the target frequency. Once sufficient signal has been detected for frequency lock the CTU performs position measurements and reports VALID.

Once frequency locked, the CTU also calculates resonator frequency. It uses this measurement to tune to the optimum excitation frequency for the next measurement. The CTU also reports resonator frequency over the SPI host interface.

2 CAM204 CTU Functional Description

The CAM204 CTU uses sensors built from PCBs to measure the position of contactless targets. These targets comprise an inductively coupled resonant circuit that is energised and detected by the CTU. Sensors can be linear or rotary, and are available in different types for different accuracies and measuring ranges. For more details please see section 1.

The CAM204 CTU has 4 drive lines and 4 pairs of sensor inputs. Depending on sensor Type, these may be connected to one or more sensors. Sections 4, 5, 6 and 7 illustrate the different configurations possible.

The CAM204 has 4 user configurable IOs. They may be used for signaling when new position samples are available (section 12.7), for controlling LEDs (section 12.10), for position triggers (section 12.11) and/or for driving an external DAC (section 13).

Communication with a host device is by a simple register-based SPI interface in which the CTU is a slave device. Alternatively, once appropriately configured over the SPI interface, the CAM204 may run autonomously (section 12.16).

The CAM204 includes internal FLASH memory used to store its application firmware. This may be updated over the SPI interface using an on-chip bootloader (section 14).

Table 2 summarises CAM204 pin functions.

Table 2

Signal Name	**Type**	**Description**
VDD (x2)	Power	Positive supply voltage.
AVDD	Analog Input	Analog supply voltage, decoupled from VDD.
VSS (x3)	Power	0V connection and common return for sensor inputs.
VCORE	Power	Output of on-chip regulator, requires external decoupling capacitor to VSS.
nRESET	Digital Input	Hardware reset, active low. Requires 10K resistor to VDD. Connection to host device is optional.
nSS	Digital Input	SPI Interface line: Slave Select, active low.
SCLK	Digital Input	SPI Interface line: Serial Clock.
MOSI	Digital Input	SPI Interface line: Master Out, Slave In.
MISO	Digital Output	SPI Interface line: Master In, Slave Out.
IO1, IO2, IO3, IO4	Digital Outputs (optional open drain)	User configurable IO. Can be used for position triggers, to signal new or VALID samples, for driving LEDs and/or driving an external DAC
DRIVE1 – DRIVE4	Digital Outputs	Used to drive external MOSFETs for powering the excitation coil(s) of resonant inductive position sensors.
COS1 – COS4 SIN1 – SIN4	Analog Inputs	Used to sense the sensor coil outputs of resonant inductive sensors.

3 Electrical Characteristics

3.1 Operating Characteristics

Table 3 operating characteristics

Item	Min	Max	Comments
Operating Supply Voltage VDD	2.65V	3.60V	
Operating Temperature	-40°C	85°C	Max +125°C for CAM204BE
VDD start voltage relative to VSS		0.1V	For reliable power on reset
VDD rise rate relative to VSS	0.5V/ms		

3.2 Absolute Maximum Ratings

Table 4 absolute maximum ratings

Item	Max	Comments
Ambient temperature under bias	-40°C to +100°C	-40°C to +135°C for CAM204BE
Voltage between VDD and VSS	-0.3V to +4.0V	
Voltage between IO1-4 and SPI lines relative to VSS	-0.3V to +6.0V	
Current into or out of Digital Output	25mA	

3.3 Digital Input Specifications

The specifications of Table 5 apply to digital inputs: the SPI interface lines nSS, SCLK and MOSI and the nRESET line.

Table 5 digital input specifications

Item	Min	Max
Input Low	VSS	0.15 x VDD
Input High	(0.24 x VDD) + 0.8V	5.5V
Input leakage current		±1μA

3.4 Digital Output Specifications

The specifications of Table 6 apply to digital outputs: the 4 user configurable IO pins IO1-IO4 and the SPI interface line MISO. IO1-IO4 can be set to *digital* or *open drain*, see section 11.3. MISO is always open drain.

Table 6 digital output specifications

Item	Min	Max	Comments
Output Low Voltage		0.4V	IOL = 5mA
Output High Voltage (digital setting)	2.1V		VDD=2.5V IOH = -2mA
	3.0V		VDD=3.6V IOH = -3mA
Output High Current (open drain setting)		±1μA	

3.5 Application Memory Characteristics

The CAM204 includes a processor with FLASH memory. This can be updated with new Application Code over its SPI interface, see section 13. It can also be updated with new Configurable Defaults, see section 12.16. FLASH memory related specifications are in Table 7 below.

Table 7 application memory characteristics

Item	Min	Max	Comments
Number of FLASH updates		2000	Across Operating Supply Voltage and Operating Temperature
Retention time	20 years		

4 Type 1 Sensor Application

4.1 Type 1 Sensor Overview

Type 1 sensors are available for linear and rotary position sensing using resonant inductive technology. They have a simple, 3-coil design with the equivalent circuit illustrated in Figure 6.

Figure 7 illustrates the design of a Type 1 linear sensor. An outer excitation coil, EX, is for powering a resonator inside an inductively coupled target. Two sensor coils, SIN and COS, are for measuring signals returned from the resonator.

Up to 4 Type 1 sensors may be connected to a CAM204. Each may have its own or multiple targets, or targets may be shared between them.

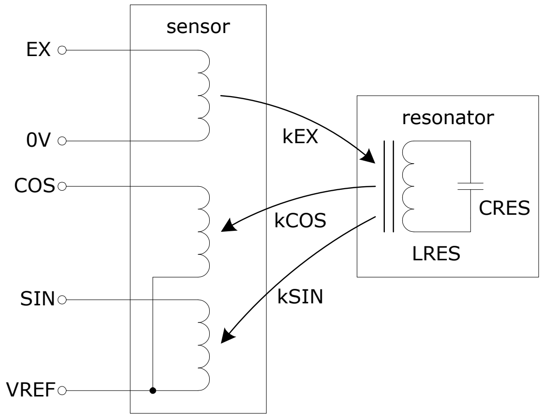


Figure 6 equivalent circuit, Type 1 sensor

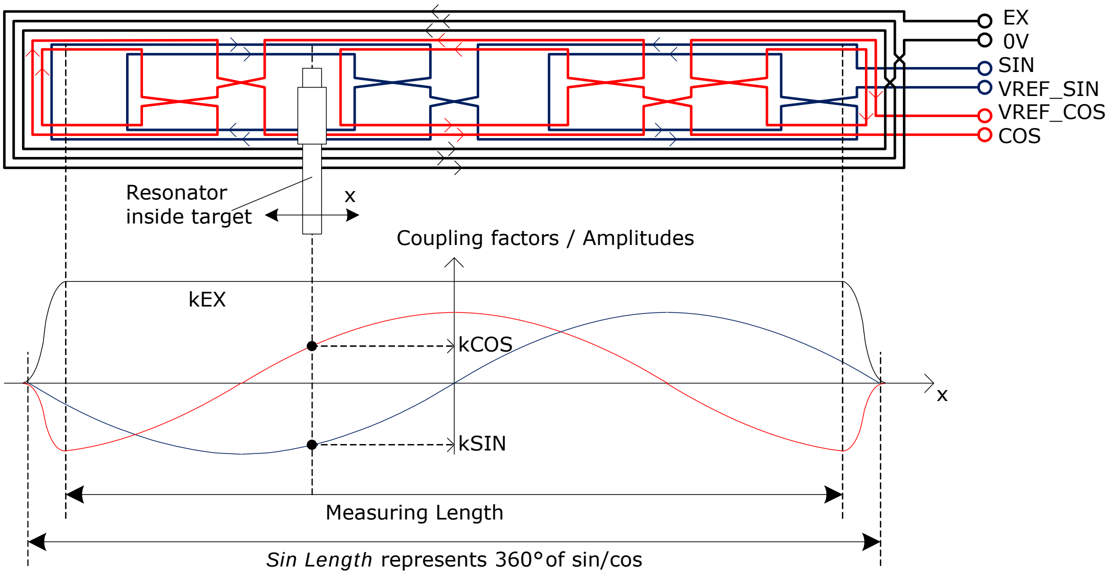


Figure 7 sketch of coils in a Type 1 linear sensor

To determine the position of a target relative to a sensor, the CTU performs a measurement of SIN and COS amplitudes as described in section 1.3. It then uses these to calculate the angle of the (kCOS, kSIN) vector. This angle represents position. The CTU's reported position output is the 16-bit word CtuReportedPosition. This is scaled such that 360° of spatial angle is represented by 65536 reported units.

Position calculation is illustrated in Figure 8. Pr is the spatial angle. The values shown match the target positioned as in Figure 7.

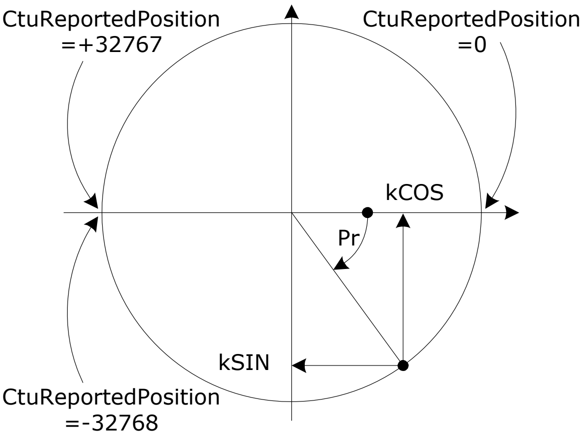


Figure 8 Type 1 position calculation

4.2 Circuit Schematics, Type 1

Figure 9 shows how the CAM204's supplies should be connected, the chip's connections to the host and to the excitation and sensing circuits. Component values are listed in Table 8.

VCORE is an external connection to an on-chip 2.5V regulator. This pin requires an external decoupling capacitor. It must not be connected to other circuitry.

AVDD is the analog supply input to the CAM204, and is used as a reference voltage. The values of R_AVDD and C_AVDD are important for the CAM204 to operate to full specification.

nRESET requires a pull-up resistor to VDD for the CAM204's on-chip power on reset circuit to operate successfully. nRESET may optionally be connected to an open drain output from the host, and used as an active-low reset line for the chip. This simplifies bootloader operation (section 14).

MISO is an open drain output, and requires a pull-up resistor of 4.7kΩ if this is not provided by the host.

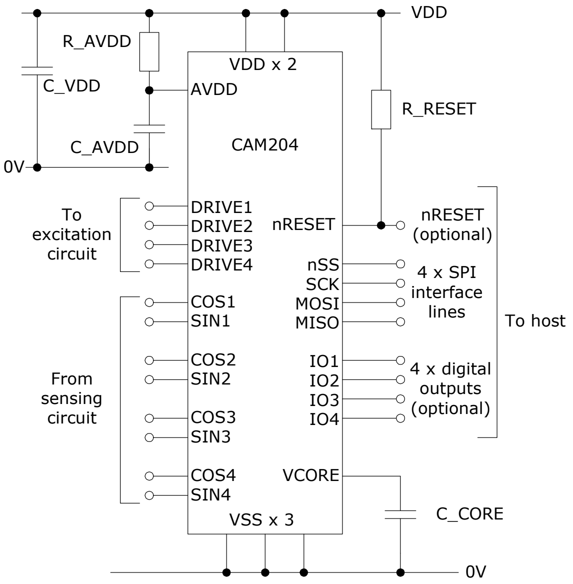


Figure 9 CAM204 connections, Type 1

The CAM204 uses an external complementary MOSFET pair to drive current into each sensor's excitation coil. Figure 10 shows the excitation circuit. When more than one sensor is connected, decoupling capacitor C_EXSUP can be shared across sensors providing connections are kept short. The MOSFET pair, resistor R_DRIVEn and capacitor C_DRIVEn are all repeated for each sensor connected.

MOSFET pair Q1 is available as a single miniature device, see Table 8 for the part required. Resistor(s) R_DRIVEn limit the operating speed of the MOSFET pair, to minimise capacitively coupled emissions. Capacitor(s) C_EXn remove the DC voltage component at the MOSFET drains from the voltage applied to the excitation coil.

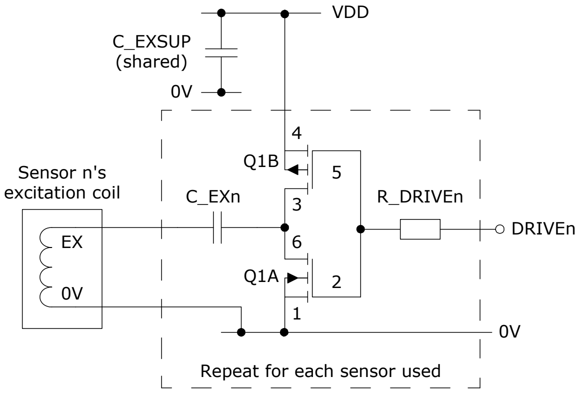


Figure 10 excitation circuit, Type 1

Sensor coil connections to the CAM204 are illustrated in Figure 11. There is a shared bias network, plus an RC filter per sensor coil input to the CAM204. All circuit values quoted in Table 8 are important for function and performance to specification. VREF should not be generated by other means.

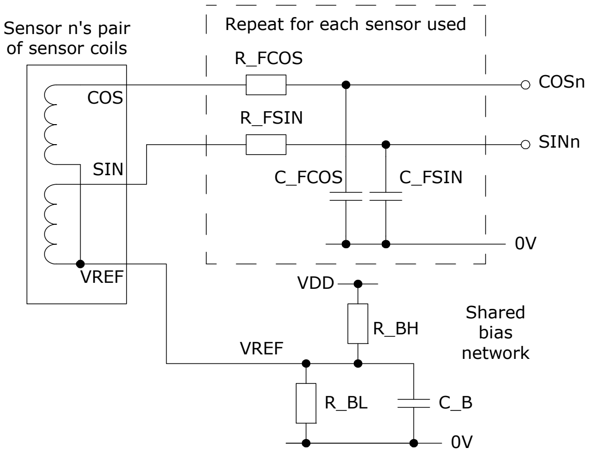


Figure 11 sensor coil connections, Type 1

4.3 Components Required, Type 1

Table 8 lists the components used in the CTU circuits of section 4.2, and their values. It also includes how many are required, depending on the number of Type 1 sensors connected.

Table 8 components required for Type 1 sensors

Circuit Ref	**Value**	**Tolerance**	**Number required for…**			
			1 sensor	**2 sensors**	**3 sensors**	**4 sensors**
R_RESET	10kΩ	±5%	1			
R_AVDD	100Ω	±5%	1			
R_DRIVEn	330Ω	±5%	1	2	3	4
R_FCOSn/SINn	2.7kΩ	±1%	2	4	6	8
R_BH	68kΩ	±1%	1			
R_BL	47kΩ	±1%	1			
C_VDD	470nF	±10%	1			
C_AVDD	4.7μF	±10%	1			
C_CORE	10μF, ESR < 3Ω	-50% / +100%	1			
C_EXSUP	10μF, ESR < 3Ω	-50% / +100%	1			
C_EXn	470nF	±10%	1	2	3	4
C_FCOSn/SINn	47pF	±5%	2	4	6	8
C_B	100nF	±10%	1			
Q1	FDY4000CZ		1	2	3	4
Number of external components			16	23	30	37

4.4 Sample Time and Rate, Type 1

Sample Time and Sample Rate are defined in section 12.8.

There are two Sample Rate values in Table 9. The first is the maximum rate at which sensors can be measured and position triggers generated. This is measured in continuous sampling mode, with INTERVAL set to 0 (see section 11.2). In this case there is no time for SPI transactions.

The second Sample Rate specification is the maximum rate at which a sensor's position can be read over the SPI interface. The SPI Communication Time is assumed to be 120μs.

When the CTU is used to sample more than one sensor at a time, sensors are scanned sequentially. Achievable sample rate is reduced accordingly.

The maximum sample time and minimum sample rate occur when the target's resonant frequency is at the low end of the tuning range for any CTU (see Table 21).

Table 9 sample time and rate, Type 1

Item	**Min**	**Max**
Sample Time		800μs
Maximum Sample Rate (position triggers)	1.2kHz	
Maximum Sample Rate (SPI interface)	1.1kHz	

4.5 Current Consumption, Type 1

The CAM204's supply current depends on the rate at which it is sampling position. Illustrative values are in Table 10. These figures include the current required for the excitation circuit and the biasing network R_BH and R_BL.

There is a power down mode under host control activated over the SPI interface by writing to the PWRDN bit (see section 11.1).

Table 10 current consumption, Type 1

Function	**Typical current at VDD = 3.3V**
Sampling at 1kHz	33mA
Sampling at 100Hz	12mA
Idle	9mA
PWRDN bit set	50µA

4.6 Reproducibility, Type 1

The CAM204 relies only on the geometry of coils printed on a PCB for position sensing, and is fully ratiometric. It is therefore largely immune to temperature changes, and there is minimal chip to chip variability.

Table 11 specifies worst case reproducibility across chips and temperature. Figures are based on the average of a number of successive measurements, so that noise present in individual readings is rendered insignificant.

The reproducibility of successive position readings is measured as the noise present in the CAM204's data stream, and hence Noise Free Resolution. This is addressed in section 4.7.

Table 11 reproducibility, Type 1

Reproducibility (Amplitude ≥ 2000)	**Max**
Across CAM204 chips, constant temperature	±0.1%
Across temperature range -40°C to +85°C, same CAM204 chip	±0.1%

4.7 Noise Free Resolution, Type 1

Position resolution is not limited by the number of bits reported over the SPI interface (16 for a Type 1 sensor). A more useful measure of resolution is Noise Free Resolution. This describes how many different positions the CAM204 can distinguish, considering the noise present in the lower bits of its output.

The Noise Free Resolution is given by Equation 1, and is based on the standard deviation of Reported Position results (calculated from Equation 5) with a stationary target. Equation 1 is based on the standard Noise Free Resolution definition for an ADC. See, for example, "The Data Conversion Handbook" By Walter Allan Kester, Analog Devices Inc, 2005.

$$PkPkNoise = 6.6 \times StandardDeviation$$

$$Noise\ Free\ Resolution = \ log_2 \left[\frac{SinLength}{PkPkNoise}\right]$$

Equation 1

The CAM204's Noise Free Resolution improves with Amplitude, as illustrated in Figure 12. Amplitude depends on a number of factors including the type of sensor and target, and the gap between them. See sensor datasheets for more information. A typical Amplitude at a sensor's nominal operating gap is 3000, so that typical Noise Free Resolution is 10 bits.

When used with an external DAC, the Noise Free Resolution of the analog output is a combination of the CAM204's value and that of the DAC. The analog circuitry used to condition and distribute the analog signal will also contribute noise, and can be minimised through careful analog design.

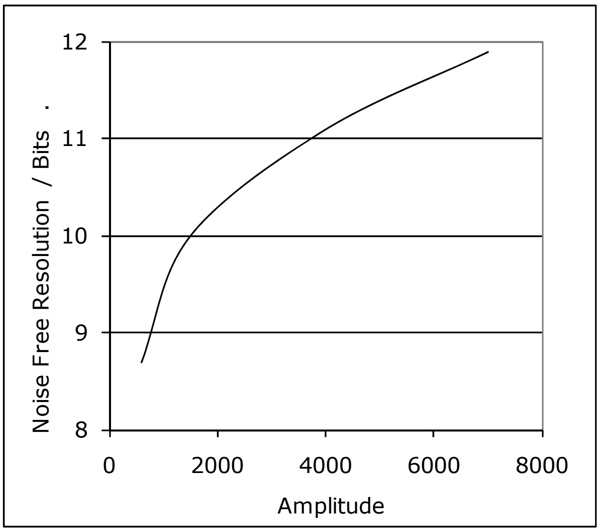


Figure 12 Noise Free Resolution, Type 1

5 Type 2 Sensor Application

5.1 Type 2 Sensor Overview

Type 2 sensors are available for linear position sensing using resonant inductive technology. They differ from Type 1 sensors in having two pairs of SIN/COS sensing coils as illustrated in Figure 13.

The COSA/SINA pair are the "fine" coils, and the COSB/SINB pair are the "coarse" coils. Figure 14 illustrates how the coil amplitudes depend on position. The CTU calculates "coarse" and "fine" position in a similar way to Type 1 (section 4.1).

"Coarse" position is absolute, since Coarse Pitch is less than the Measuring Length. However, this measurement has poor accuracy and resolution.

"Fine" position is *incremental*: it can locate the target over the Fine Pitch distance unambiguously, but the measurement repeats at greater distances. This measurement has the advantage of high precision, however.

To achieve an output which is both precise and absolute, the CTU combines "fine" and "coarse" position measurements. Two methods are available, depending on the setting of the incremental enable bit INCE, as detailed in section 5.2.

The CAM204 CTU chip can be connected to one Type 2 sensor, as illustrated in section 5.4.

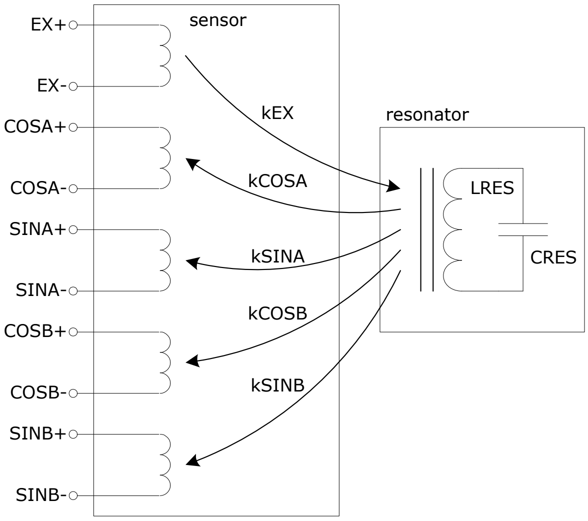


Figure 13 equivalent circuit, Type 2 sensor

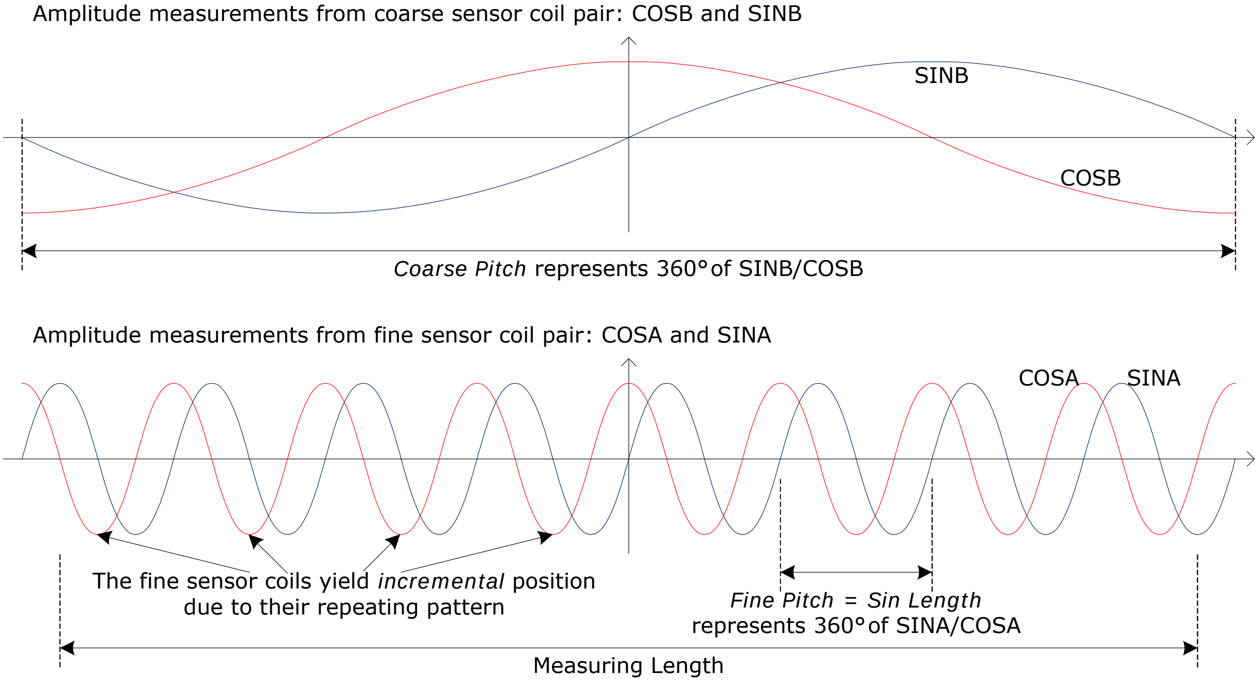


Figure 14 coil amplitudes as a function of position, Type 2 sensor

5.2 Type 2 Sensor Operating Modes

When the host sets the INCE (*incremental enable*) bit to 0 the CTU always takes a fresh measurement from both fine and coarse sensor coils to generate each new reported position. The CTU sets INCF (*incremental flag*) to 0 to indicate incremental mode was not used for the measurement. The INCE and INCF bits are located in the SCW host register, see section 11.9.

When the host sets the INCE bit to 1, incremental mode is enabled. If there is no target present, the CTU's behaviour is the same as INCE=0. It sets INCF to 0, and takes readings from both the coarse and fine sensor coils. When a target is in range and a first valid measurement has been taken, subsequent measurements are based on measurements from the fine sensor coils alone, and INCF is set to 1. In incremental mode, the CTU still reports full absolute position. It uses the previous reported position as its approximate estimate of absolute position, which is then refined with the new incremental data from the fine sensor coils.

The advantage of incremental mode is a shorter Sample Interval, since measurements from the coarse sensor coils are not required.

However there is one well understood and highly repeatable failure mode in incremental mode: *skipping a fine period*. This will occur if the target's position changes by more than the *Fine Period Skip Distance* between CTU position measurements. This is given by:

$$Fine\ Period\ Skip\ Distance = \frac{Fine\ Pitch}{2} - (peak\ to\ peak\ position\ noise) - (peak\ to\ peak\ position\ error)$$

Equation 2

For the purposes of this equation, peak to peak position noise and error must be taken as absolute extreme values, and the following working definition is recommended:

$$Fine\ Period\ Skip\ Distance = 0.4\ \times Fine\ Pitch$$

Equation 3

For a typical Fine Pitch of 50mm, the Fine Period Skipping Distance is therefore 20mm. To prevent *skipping a fine period*, it is recommended that the host only sets INCE to 1 when the interval between successive CTU position measurements is 2ms or less, AND the velocity of the target is known to be less than 10ms^{-1}. These values will yield the required 20mm maximum.

For a sensor with a Fine Pitch of 100mm, the Fine Period Skipping Distance is 40mm. The recommended maxim interval and velocity are 2ms and 20ms^{-1} respectively.

As an additional precaution, the host may opt to alternate between a run of measurements taken with INCE=1 and single measurements taken with INCE=0. This will not prevent the "skipping a fine period" failure mode, but any position offset will be eliminated immediately with an INCE=0 measurement.

5.3 BA Mismatch Output

The CTU reports the results of a Type 2 sensor measurement in the 6 results registers RESA-RESF, described in section 11.11. In addition to position, amplitude and relative frequency measurements, the CTU also reports *BA Position Mismatch*. This is a diagnostic output for the healthiness of the CTU's absolute position calculation.

When INCE=0, BA Position Mismatch is the difference between the B (coarse) and A (fine) position calculations. When INCE=1, BA Position Mismatch is the difference between the most recent position measurement and the one before. BA Position Mismatch is a signed 16-bit value (-32768 to +32767). The value 0 represents perfect health, and extreme positive and negative values are bad. When INCE=0, "bad" means highly inaccurate measurements on fine and/or coarse tracks due to a circuit or sensor PCB fault. When INCE=1, "bad" means excessive distance between successive measurements due to excess time interval and/or velocity.

BA Position Mismatch is scaled in the same way as Reported Position (Equation 5). The extreme values -32768 and +32767 therefore represent ±(Fine Pitch/2).

5.4 Circuit Schematics, Type 2

Figure 15 shows how the CAM204's supplies should be connected, the chip's connections to the host and to the excitation and sensing circuits. Component values are listed in Table 12.

VCORE is an external connection to an on-chip 2.5V regulator. This pin requires an external decoupling capacitor. It must not be connected to other circuitry.

AVDD is the analog supply input to the CAM204, and is used as a reference voltage. The values of R_AVDD and C_AVDD are important for the CAM204 to operate to full specification.

nRESET requires a pull-up resistor to VDD for the CAM204's on-chip power on reset circuit to operate successfully. nRESET may optionally be connected to an open drain output from the host, and used as an active-low reset line for the chip. This simplifies bootloader operation (section 1413).

MISO is an open drain output, and requires a pull-up resistor of 4.7kΩ if this is not provided by the host.

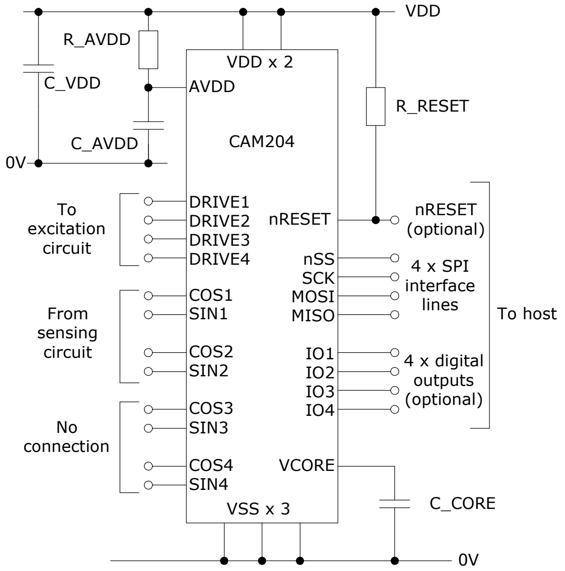


Figure 15 CAM204 connections, Type 2

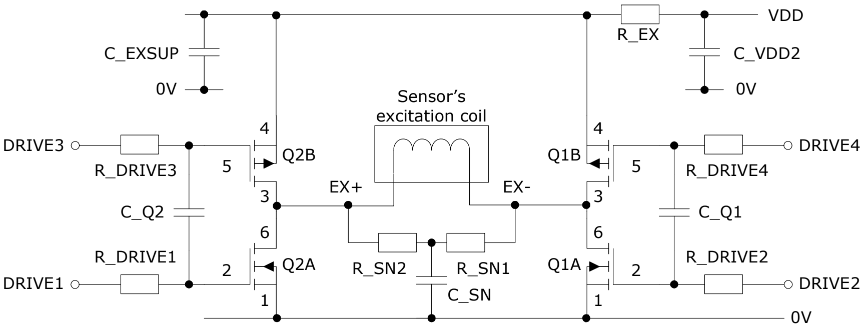


Figure 16 excitation circuit, Type 2

The CAM204 uses an external MOSFET "H" bridge to drive a Type 2 sensor's excitation coil. Figure 16 shows the excitation circuit. MOSFET pairs Q1 and Q2 are each available as a single miniature device, see Table 12. The gate drive circuit uses 2 resistors and 1 capacitor per side, and is designed to enable the CTU to drive the bridge output to a high impedance state. The resistors R_DRIVE limit the operating speed of the MOSFETs, to minimise capacitively coupled emissions. The capacitors C_Q1 and C_Q2 prevent excessive shoot-through current during switching. The network R_SN1, R_SN2 and C_SN absorb the energy in the excitation coil on the transition from low to high impedance. The energy required for each pulse of excitation current is stored in C_EXSUP. R_EX limits the peak charging current.

Figure 17 shows the components required for the sensor coil inputs to the CTU. Each of the 4 coil inputs has two stages of RC filtering formed by the components with prefix R_F2, C_F2, R_F and C_F. The reference voltage VREF is generated by the network R_BH, R_BL and CB. VREF should not be generated by other means.

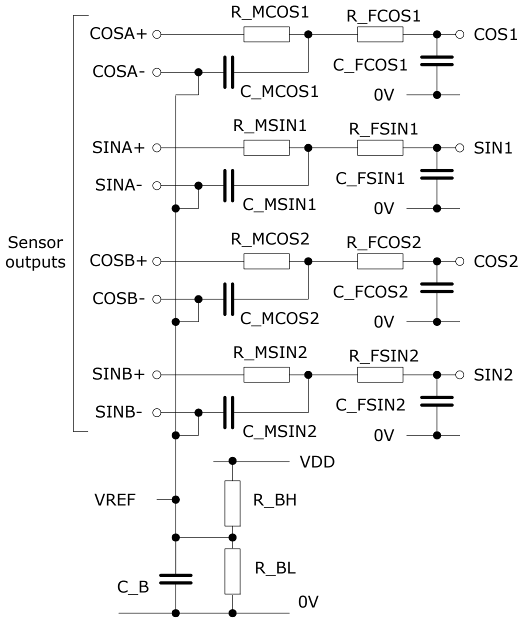


Figure 17 sensor coil connections, Type 2

5.5 Components Required, Type 2

Table 12 lists the component values and numbers required for the schematics of 5.4. The filter components connected to the CTU's sensor 1 inputs are important for reproducibility, and the table includes two grades for them: A and B. Grade A yields the least system to system reproducibility error due to component differences. Grade B components are more cost effective. Grade A and B reproducibility error is compared in Table 13.

Table 12 components required for a Type 2 sensor

Circuit Ref	Value	Tolerance		Number required
		Grade A	Grade B	
R_RESET	10kΩ	±5%		1
R_AVDD	100Ω	±5%		1
R_DRIVE1/2	1.6kΩ	±5%		2
R_DRIVE3/4	330Ω	±5%		2
R_EX	1Ω	±5%		1
R_SN1/2	100Ω	±5%		2
R_MCOS1/SIN1	100Ω	±0.1%	±1%	2
R_MCOS2/SIN2	100Ω	±1%		2
R_FCOS1/SIN1	1kΩ	±0.1%	±1%	2
R_FCOS2/SIN2	1kΩ	±1%		2
R_BH	15kΩ	±1%		1
R_BL	10kΩ	±1%		1
C_VDD	470nF	±10%		1
C_AVDD	4.7μF	±10%		1
C_CORE	10μF, ESR < 3Ω	-50% / +100%		1
C_EXSUP	100μF, ESR < 100mΩ	-50% / +100%		1
C_VDD2	100μF, ESR < 100mΩ	-50% / +100%		1
C_Q1/2	1nF	±10%		2
C_SN	10nF	±10%		1
C_MCOS/SIN1	2.2nF	±1%	±5%	2
C_MCOS/SIN2	2.2nF	±5%		2
C_FCOS/SIN1	220pF	±1%	±5%	2
C_FCOS/SIN2	220pF	±5%		2
C_B	470nF	±10%		1
Q1/2	FDY4000CZ			2
Number of external components				38

Table 13 reproducibility error due to filter components

Grade	Reproducibility as % of Fine Pitch	Reproducibility in μm	
		Fine Pitch = 50mm	Fine Pitch = 100mm
A	±0.03%	±15μm	±30μm
B	±0.17%	±85μm	±170μm

5.6 Sample Time and Rate, Type 2

Sample Time and Sample Rate are defined in section 12.8. The SPI Communication Time is assumed to be 160µs.

The state of the INCF bit reflects the operating mode, see section 5.2. INCF=1 (“incremental”) yields faster sampling.

The maximum sample time and minimum sample rate occur when the target’s resonant frequency is at the low end of the tuning range for any CTU (see Table 21).

Table 14 sample time and rate, Type 2

Item	INCF state	Min	Max
Sample Time	0		1530µs
Maximum Sample Rate (SPI interface)		590Hz	
Sample Time	1		840µs
Maximum Sample Rate (SPI interface)		1000Hz	

5.7 Current Consumption, Type 2

Power supply current consumption is lowest when the CTU is not sampling, and typical values are in Table 15. These figures include the current required for the biasing network R_BH and R_BL. The power down mode draws the lowest current and is activated over the SPI interface by writing to the PWRDN bit (see section 11.1).

Figure 18 illustrates typical power supply current consumption when sampling, for sensors of different Measuring Length. The peak instantaneous supply current occurs when INCF=0. The values shown are for a power supply source resistance of 1Ω. Average supply current depends on sample rate and operating mode.

Table 15 typical supply current, VDD = 3.3V

Function	Typical
Idle	9mA
PWRDN bit set	150µA

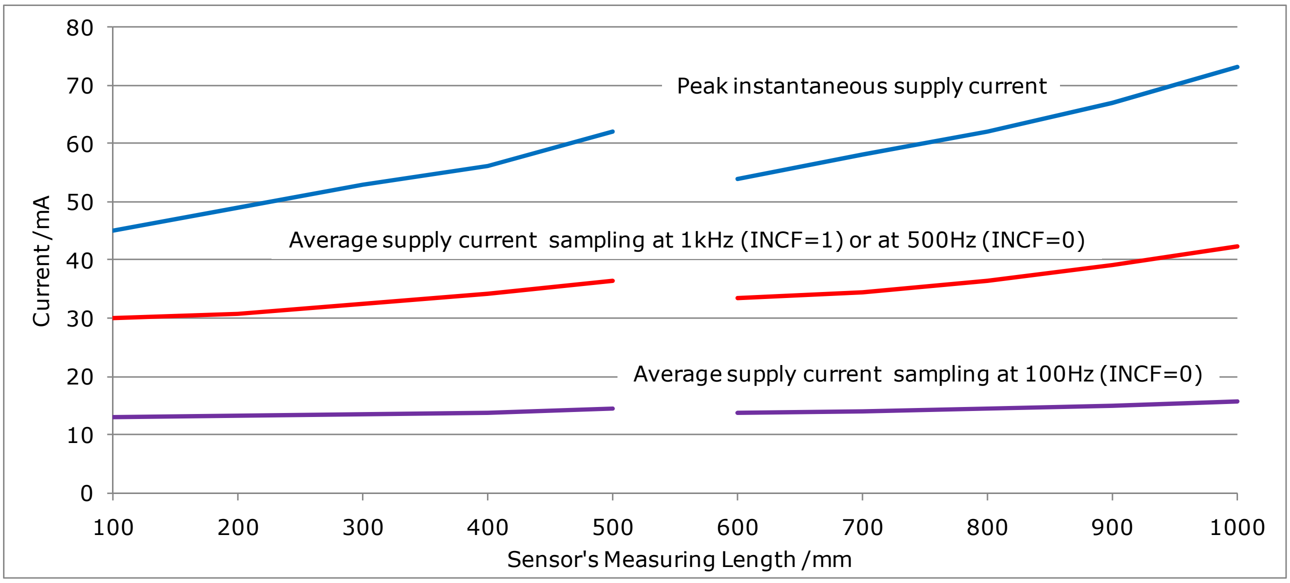


Figure 18 current consumption, VDD = 3.3V, Type 2 sensors

5.8 Noise Free Resolution, Type 2

Noise Free Resolution describes how many different positions the CAM204 can distinguish, considering the noise present in the lower bits of its output. It is generally a more useful measure than interface resolution for a system having excess interface resolution, as described in section 4.7. The typical data in Figure 19 is based on Equation 4. This uses the standard deviation of a number of measurements as a robust measure of peak to peak noise.

$$PkPkNoise = 6.6\ \times StandardDeviation$$

$$Noise\ Free\ Resolution = log_2\left[\frac{Measuring\ Length}{PkPkNoise}\right]$$

Equation 4

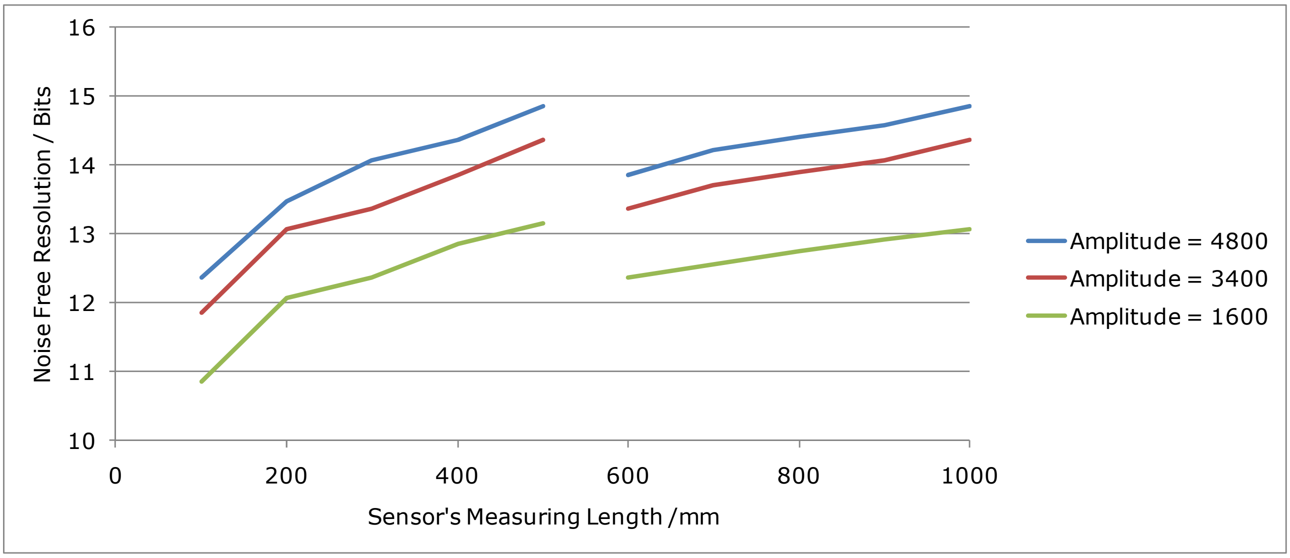


Figure 19 Noise Free Resolution, Type 2

6 Type 3 Sensor Application

6.1 Type 3 Sensor Overview

Type 3 sensors measure the linear position of a freely rotating target, as illustrated in Figure 21.

The magnetic axis of the target points along the measuring direction. This means that a uniform excitation coil, as used in Type 1 and 2 sensors, is not possible. Instead, there are two excitation coils, EXA and EXB. They are sinusoidally patterned, as illustrated in Figure 22. At least one of the excitation coils couples to the target, whatever its location along the measuring length.

Sensor coils are also sinusoidally patterned. They have a different pitch to the excitation coils, chosen to enable absolute position measurement.

The CTU detection process is similar to the one described in section 1.3. The CTU first pulses current into excitation coil EXA and measures the return signals in COS and SIN. It then repeats the measurement with EXB. When combined, the measurements yield full absolute position. The CTU must leave enough time between each measurement for the signal from the target to decay, unlike with Type 1 and 2 sensors. This makes the measuring process slower for Type 3 sensors than for other types.

The equivalent circuit of a Type 3 sensor is shown in Figure 20. The CAM204 CTU chip can be connected to one Type 3 sensor, as illustrated in section 6.2.

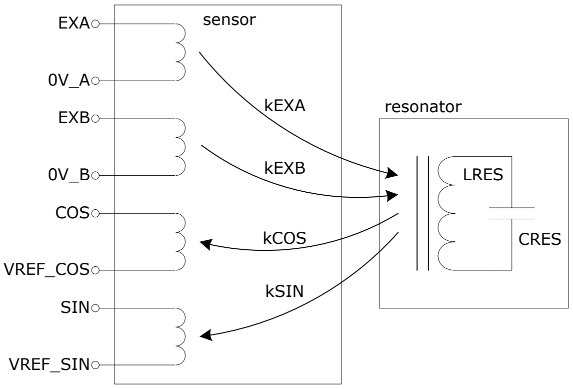


Figure 20 equivalent circuit, Type 3 sensor

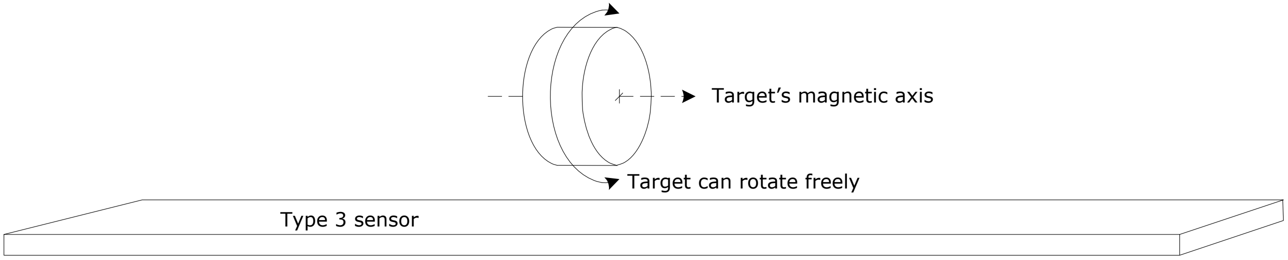


Figure 21 Type 3 sensors work with freely rotating targets

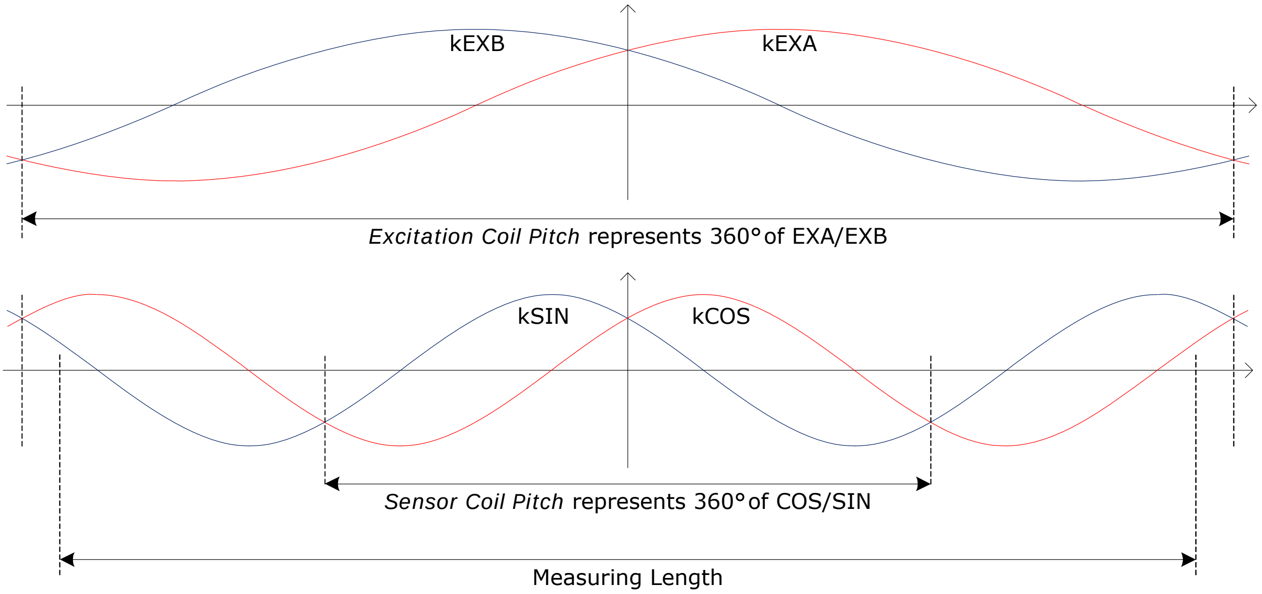


Figure 22 coil coupling factors as a function of position, Type 3 sensor

6.2 Circuit Schematics, Type 3

Figure 23 shows how the CAM204's supplies should be connected, the chip's connections to the host and to the sensor circuit. Component values are listed in Table 16.

VCORE is an external connection to an on-chip 2.5V regulator. This pin requires an external decoupling capacitor. It must not be connected to other circuitry.

AVDD is the analog supply input to the CAM204, and is used as a reference voltage. The values of R_AVDD and C_AVDD are important for the CAM204 to operate to full specification.

nRESET requires a pull-up resistor to VDD for the CAM204's on-chip power on reset circuit to operate successfully. nRESET may optionally be connected to an open drain output from the host, and used as an active-low reset line for the chip. This simplifies bootloader operation (section 14).

MISO is an open drain output, and requires a pull-up resistor of 4.7kΩ if this is not provided by the host.

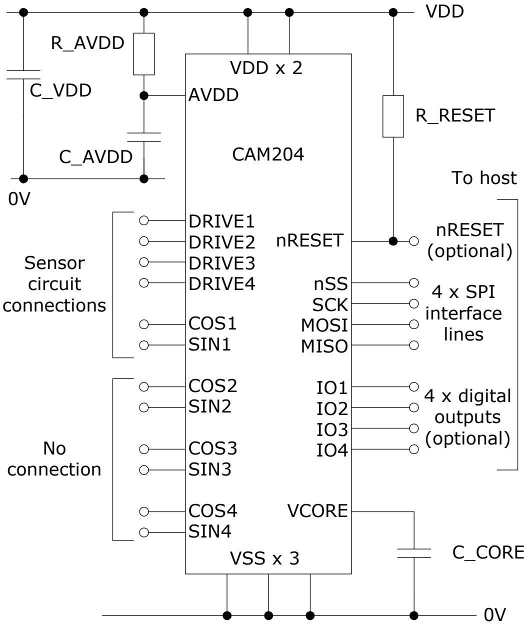


Figure 23 CAM204 connections, Type 3

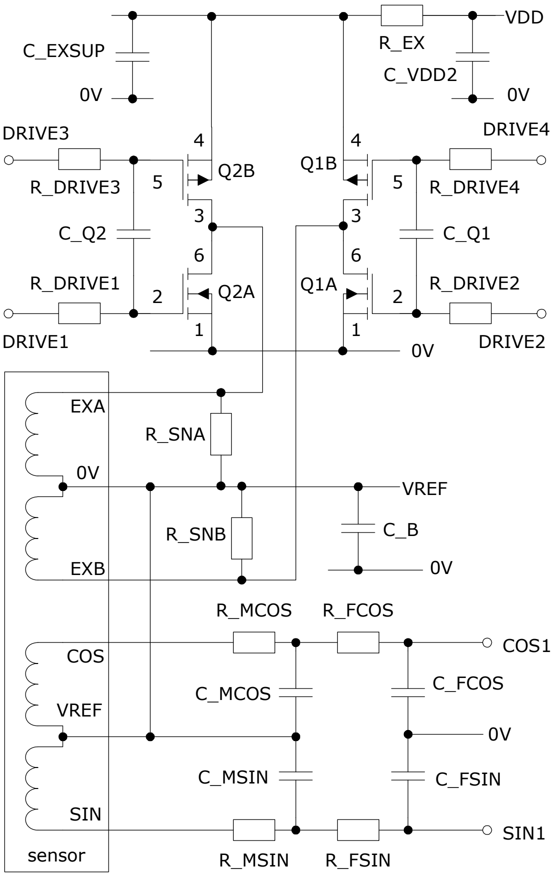


Figure 24 sensor circuit, Type 3

Each of the sensor's two excitation coils is driven from a complementary MOSFET pair, available in a single miniature device. Resistors R_DRIVE limit the slew rate of the MOSFET drains, to minimise capacitively coupled emissions. Capacitors C_Q prevent excessive shoot-through current during switching.

The energy required for each pulse of excitation current is stored in C_EXSUP. R_EX limits the peak charging current. Resistors R_SN absorb the energy in their respective excitation coils on the transition from low to high bridge output impedance.

Capacitor C_B is charged to approximately half supply rail during excitation pulses so that the excitation coils are driven with current in both directions by MOSFET pairs Q1/2. The reference voltage developed, VREF, is also used to bias the CTU's sensor inputs during detection.

Two stages of RC filtering are used between the sensor coils and CTU inputs, using the network formed by resistors R_MCOS/SIN, C_MCOS/SIN, R_FCOS/SIN, C_FCOS/SIN.

6.3 Components Required, Type 3

Table 16 lists the components required for connecting a Type 3 sensor to the CAM204 CTU chip, using the schematics of section 6.2.

Table 16 components required for a Type 3 sensor

Circuit Ref	Value	Tolerance	Number required
R_RESET	10kΩ	±5%	1
R_AVDD	100Ω	±5%	1
R_DRIVE1/2	1.6kΩ	±5%	2
R_DRIVE3/4	330Ω	±5%	2
R_EX	22Ω	±5%	1
R_SN1/2	1kΩ	±5%	2
R_MCOS/SIN	220Ω	±1%	2
R_FCOS/SIN	1kΩ	±1%	2
C_VDD	470nF	±10%	1
C_AVDD	4.7µF	±10%	1
C_CORE	10µF, ESR < 3Ω	-50% / +100%	1
C_EXSUP	10µF	±10%	1
C_VDD2	10µF	±10%	1
C_Q1/2	1nF	±10%	2
C_MCOS/SIN	1nF	±5%	2
C_FCOS/SIN	220pF	±5%	2
C_B	4.7µF	±10%	1
Q1/2	FDY4000CZ		2
Number of external components			27

6.4 Sample Time and Rate, Type 3

Sample Time and Sample Rate are defined in section 12.8. The SPI Communication Time is assumed to be 160µs. The maximum sample time and minimum sample rate occur when the target's resonant frequency is at the low end of the tuning range for any CTU (see Table 21).

The Sample Time for Type 3 sensors is much longer than for Types 1 and 2, since the CTU must allow time for resonator signal decay from each different excitation coil, so that their relative amplitudes are consistent and accuracy is maintained.

Table 17 sample time and rate, Type 3

Item	Min	Max
Sample Time		7.8ms
Maximum Sample Rate (SPI interface)	125Hz	

6.5 Current Consumption, Type 3

Typical supply current for Type 3 sensor operation is in Table 15. Figures include the current required for the excitation circuit.

There is a power down mode under host control activated over the SPI interface by writing to the PWRDN bit (see section 11.1).

Table 18 typical supply current, VDD = 3.3V

Function	Typical current
Sampling at 100Hz	17mA
Idle	9mA
PWRDN bit set	10µA

7 Type 4 Sensor Application

7.1 Type 4 Sensor Overview

Type 4 sensors work in the same way as Type 1 sensors, described in section 4.1, and have the same coil arrangement. The difference is that their 0V and VREF terminals are connected together, as illustrated in Figure 25. This enables a 4-wire connection from the CTU circuit to sensor.

The CAM204 CTU chip can sense up to 2 Type 4 sensors, as shown below.

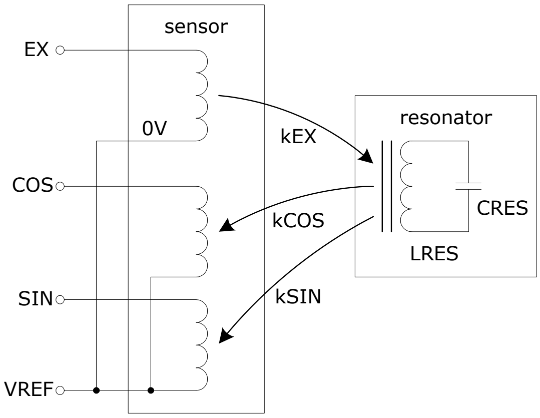


Figure 25 equivalent circuit, Type 4 sensor

7.2 Circuit Schematics, Type 4

Connections to the CAM204 chip itself are as shown in Figure 15. The sensor circuit is shown in Figure 26. This is for a sensor connected to the CAM204's sensor 1 inputs. A sensor may also be connected to the sensor 2 inputs, with the CTU signal connections shown in brackets. Two sensors may be connected. In this case, the components C_B, C_EXSUP, R_EX and C_VDD2 may be shared.

Each sensor's excitation coil is driven from a complementary MOSFET pair, available in a single miniature device. Resistors R_DRIVE limit the slew rate of the MOSFET drains, to minimise capacitively coupled emissions. Capacitors C_Q prevent excessive shoot-through current during switching.

The energy required for each pulse of excitation current is stored in C_EXSUP. R_EX limits the peak charging current. Resistors R_SN absorb the energy in their respective excitation coils on the transition from low to high bridge output impedance.

Capacitor C_B is charged to approximately half supply rail during excitation pulses so that the excitation coils are driven with current in both directions by the MOSFET(s). The reference voltage developed, VREF, is also used to bias the CTU's sensor inputs during detection.

Two stages of RC filtering are used between the sensor coils and CTU inputs, using the network formed by resistors R_MCOS/SIN, C_MCOS/SIN, R_FCOS/SIN, C_FCOS/SIN.

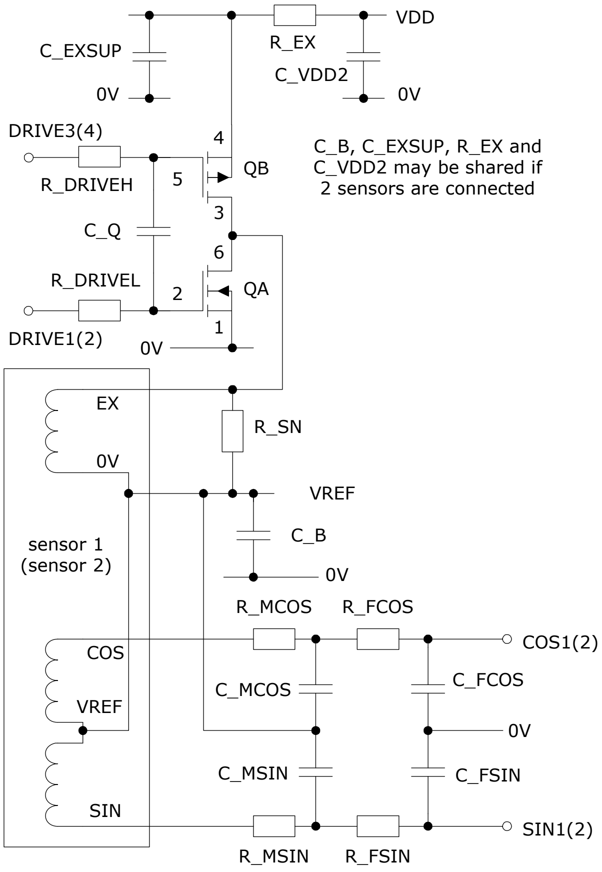


Figure 26 sensor circuit, Type 4

7.3 Components Required, Type 4

Table 19 lists the components required for connecting a Type 3 sensor to the CAM204 CTU chip, using the combined schematics of Figure 15 and Figure 26.

Table 19 components required for type 4 sensors

Circuit Ref	Value	Tolerance	Number required for…	
			1 sensor	2 sensors
R_RESET	10kΩ	±5%	1	
R_AVDD	100Ω	±5%	1	
R_DRIVELn	1.6kΩ	±5%	1	2
R_DRIVEHn	330Ω	±5%	1	2
R_EX	4.7Ω	±5%	1	
R_SNn	1kΩ	±5%	1	2
R_MCOS/SINn	220Ω	±1%	2	4
R_FCOS/SINn	1kΩ	±1%	2	4
C_VDD	470nF	±10%	1	
C_AVDD	4.7μF	±10%	1	
C_CORE	10μF, ESR < 3Ω	-50% / +100%	1	
C_EXSUP	10μF	±10%	1	
C_VDD2	10μF	±10%	1	
C_Qn	1nF	±10%	1	2
C_MCOS/SINn	1nF	±5%	2	4
C_FCOS/SINn	220pF	±5%	2	4
C_B	4.7μF	±10%	1	
Qn	FDY4000CZ		1	2
Number of external components			22	35

8 Circuit Layout

8.1 CAM204 Circuit

Connections between the CAM204's VSS pins should be as wide and short as possible. This is preferably by vias near each pin to a ground plane layer on the PCB.

Connections between the CAM204's VDD pins should also be kept as wide and as short as possible. C_VDD should be placed right next to the chip, with short and wide connections to both VSS and VDD pins. If the connection between VDD pins is difficult, an alternative is to use two decoupling capacitors C_VDD, one adjacent to each VDD pin.

The remaining decoupling capacitors, C_CORE, C_AVDD and filter capacitors connected to the CAM204's SIN/COS1-4 pins must all be placed as close as possible to the CAM204, with short connections to the chip and VSS.

Figure 27 shows a possible layout in the area of the CAM204, and includes the capacitors which need to be kept close to that chip.

The excitation circuit should be laid out to minimise the length of the current path from the VDD end of C_EXSUP through each half of each MOSFET bridge and back to the 0V end of C_EXSUP. Where a single excitation decoupling capacitor C_EXSUP is shared between multiple MOSFETs, the MOSFETs of all circuits should be placed close together so that all such current path lengths are minimised.

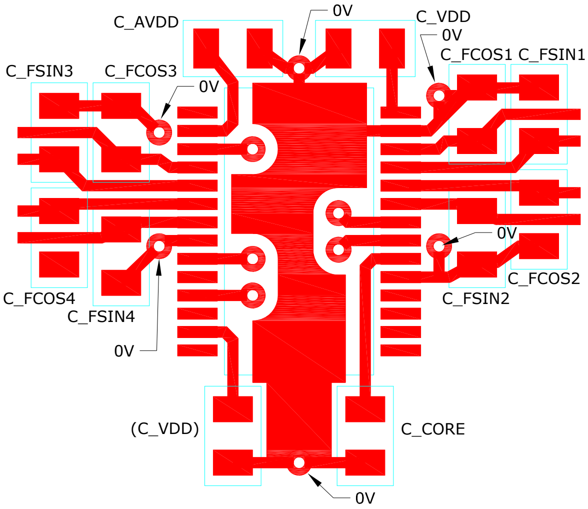


Figure 27 component layout around CAM204

8.2 Sensor Connections Avoiding Loops

Sensors that operate using resonant inductive technology use coils to pick up signals from a resonator in the target. These coils are specially patterned for accurate position sensing. To avoid inaccuracy, and to minimise the effect of any interference, the size of any additional loops formed by connecting conductors must be minimised. Figure 28 (a) and (b) illustrate incorrect connections, where connecting conductors form loops. Figure 28 (c) and (d) illustrate correct connecting options, where the loop size is minimised.

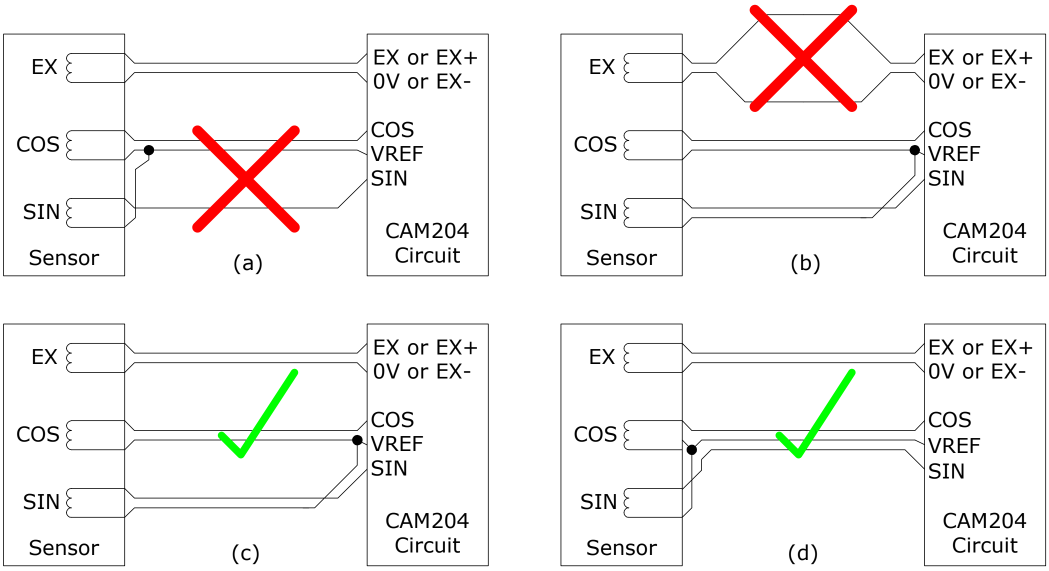


Figure 28 connection between sensor and CAM204 circuit

8.3 Recommended PCB Track Design for Sensor Connection

Figure 29 illustrates how connections may be made using conductors on a 4-layer PCB. Loop sizes are minimised by running the connections for each coil on adjacent layers of the PCB, so that one runs on top of the other.

The excitation coil connecting track widths should preferably be wide to minimise resistance. A loop resistance of less than 0.5Ω is preferable. PCBs with "1oz" copper (34μm thick) have a surface resistivity of 0.5mΩ/square, so connections can be up to 1000 squares long. (The number of squares is the overall length divided by the width). For example, a pair of 200mm long connections has an overall length of 400mm. If the width of each is 0.5mm, then the number of squares is 800.

Sensor coil connecting tracks are less critical. The connecting loop resistance should ideally be less than 5Ω.

If the gap between excitation and sensor connections illustrated in Figure 29 is small and the connections are long, there may be a significant level of coupling between the two, causing small errors in reported position. Examples of limiting values are shown in Table 20.

Table 20

Connection separations = 0.2mm			
Min connecting track width		**Min Gap**	**Max Length**
Excitation	**Sensors**		
0.2mm	0.1mm	0.1mm	100mm
0.8mm	0.1mm	0.4mm	400mm

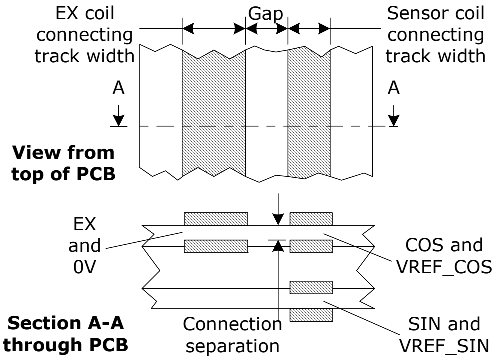


Figure 29 connections using PCB traces

9 Resonator Detection

The CTU is designed to lock onto and track the resonant frequency of inductively coupled resonators inside targets.

The CTU measures the position of each sensor's target in turn. Each time, it also estimates its resonant frequency, providing there is sufficient signal amplitude. If the resonant frequency changes, the CTU changes its excitation frequency to match.

Following power on reset, or if the previous measurement was invalid, the CTU searches its tuning range for the target, trying different excitation frequencies until it achieves lock.

Table 21 lists CTU specifications related to target detection.

Table 21

Item	Min	Typ	Max	Comments
Nominal CTU Operating Frequency		187.5kHz		
CTU Centre Frequency Tolerance			±5%	Across Operating Supply Voltage and Operating Temperature
Tuning Range for Each CTU	±12%			
Tuning range for Any CTU	±7%			
Resonator Q-factor	60		180	
Number of samples before lock			9	Following invalid sample or power on
Minimum Amplitude (Type 1&3 sensors)	250		500	For CTU to report VALID
Minimum Amplitude (Type 2 sensors)	250		700	

Figure 30 illustrates system frequency budgets. The tuning range for any CTU is derived from the CTU frequency tolerance and the tuning range for each CTU, and is the range of resonator frequencies that all CTU chips can tune to. Resonator Frequency Tolerance at Manufacture plus Resonator Frequency Change in Use in use must remain within this range for reliable detection.

The CTU reports relative frequency in its results registers accessible over its SPI interface (see section 11.11). The result is for diagnostic purposes and is approximate. To improve reproducibility of the CTU's relative frequency indication, the host should perform a CTU reset after power on, and the PWRDN state should be avoided (see section 11.1).

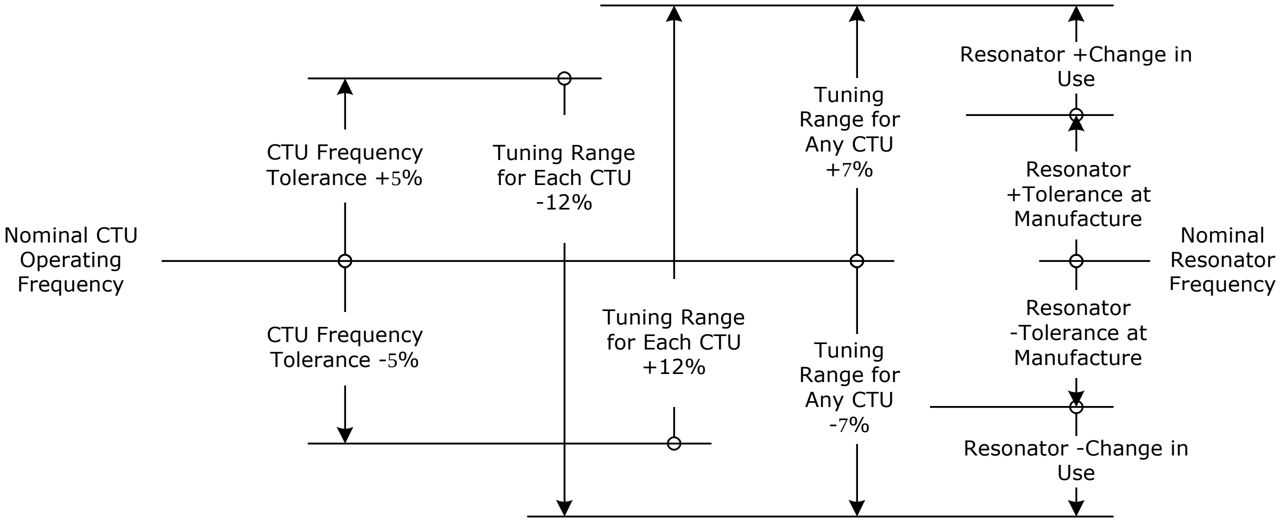

Nominal CTU Operating Frequency
CTU Frequency Tolerance +5%
CTU Frequency Tolerance -5%
Tuning Range for Each CTU -12%
Tuning Range for Each CTU +12%
Tuning Range for Any CTU +7%
Tuning Range for Any CTU -7%
Resonator +Change in Use
Resonator +Tolerance at Manufacture
Resonator -Tolerance at Manufacture
Resonator -Change in Use
Nominal Resonator Frequency

Figure 30 illustration of frequency tolerance budgets

10 SPI Hardware

10.1 Overview

This section describes how data is written to, and read back from, the CAM204 CTU, over its SPI interface. Each read or read-write operation accesses one or more of the CTU's internal registers. The next section, section 11, details the function of each of the available registers. Section 12 describes how to control the CTU to perform sensor position measurements by accessing these registers over the SPI interface, and how to configure the CTU's IO pins to implement position trigger signals.

For communication with a host system, the CAM204 CTU is always an SPI slave. All communication is initiated by the host.

10.2 Data Transfer Method

The CAM204 CTU always operates as an SPI slave device. The host device signals a data transfer by driving nSS low. It clocks data presented on the MOSI line into the CTU on the positive going transition of SCK. The CTU outputs successive data bits on its MISO line on negative going SCK transitions. This is commonly referred to as SPI Mode 0.

The beginning and end of an SPI transaction is illustrated in Figure 31.

All SPI transactions MUST be bounded by the Slave Select (nSS) line being driven low at their start and being driven high at their end. The SPI interface will not function if nSS is tied permanently low.

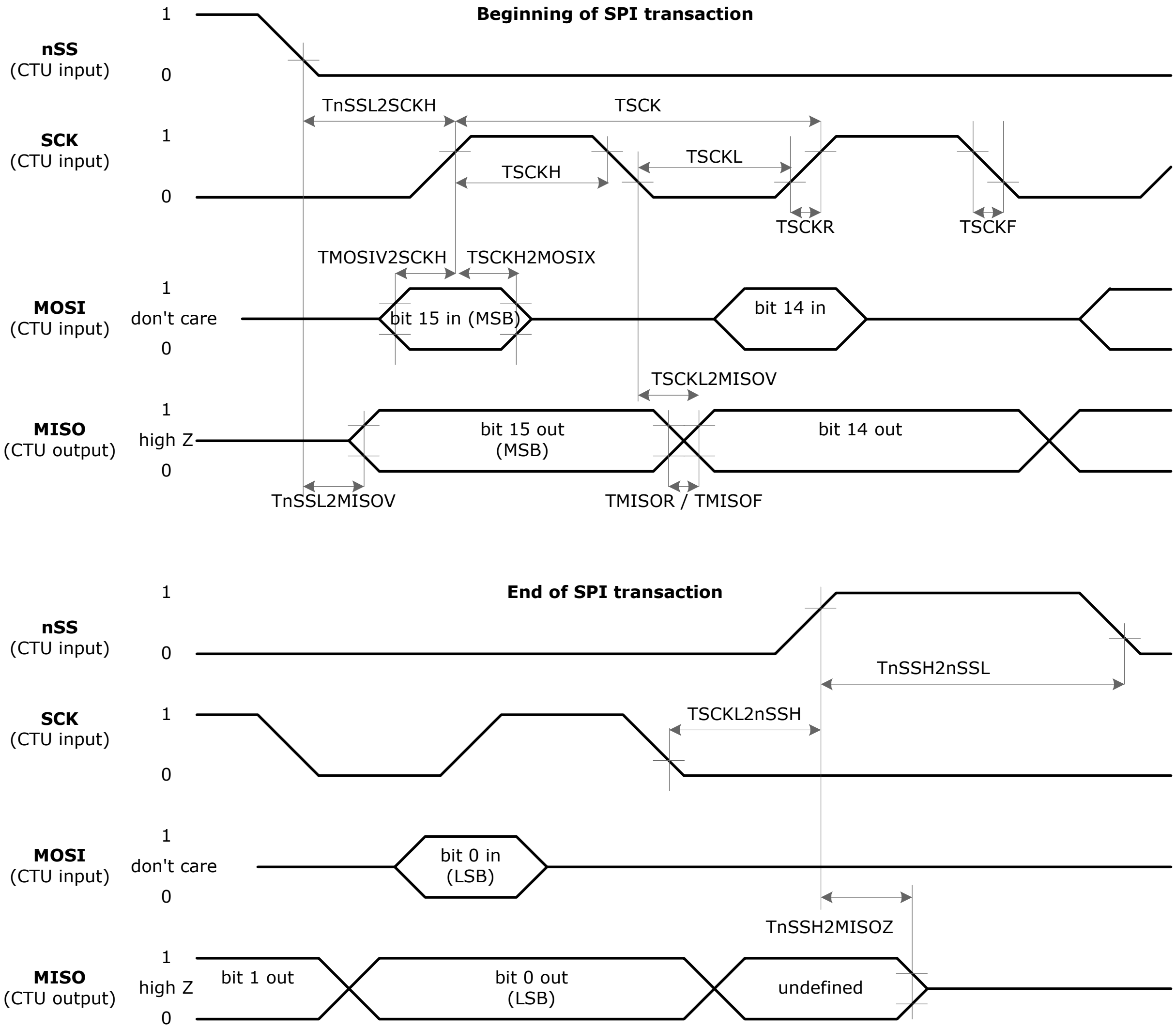


Figure 31 SPI data transfer

10.3 Register Access

The host operates the CAM204 CTU by writing to and reading from registers. There are a number of registers with different functions, as detailed in section 11. Each register is 16 bits wide, and has its own 12 bit address.

There are two modes of data transfer, *read* and *write read*. The host signals which transfer mode it wishes to use with the first 4 bits of an SPI transaction. A read starts with 0x0, and a write read with 0xF. The host then specifies the register address with the following 12 bits.

In a read operation, the next 16 bits clocked out of MISO are the data contained in the register at the specified address. The CAM204 ignores the state of MOSI during data transfer.

In a write read operation, the data to be written to the specified address is sent after the address, and the data previously at that address is clocked out of MISO at the same time.

This sequence, for a single register access, is illustrated in Figure 32.

A time TAD is required between the last address bit and first data bit clocked into the CAM204.

The CTU outputs the contents of its SYSID register (section 11.8) as the first word of each SPI transaction. This defaults to 0xABCD, and may be changed by the host. It is recommended that the host checks the value read back against the expected value as a test for SPI communication integrity.

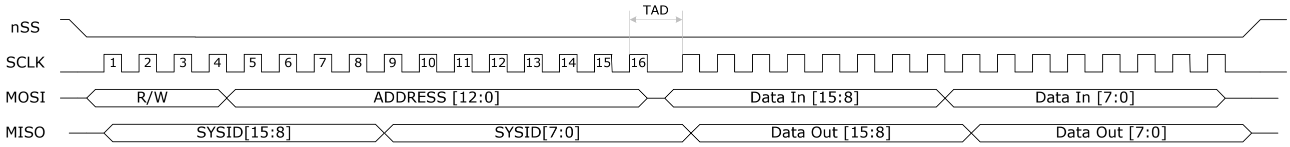


Figure 32 single register access

Instead of accessing a single register, the host may access a set of consecutive registers by extending the SPI transaction as illustrated in Figure 33. Data is transferred one register at a time, starting at the first, specified address.

Registers are arranged in blocks dedicated to system functions, and to each sensor connected, see section 11. Multiple register access may not span across different blocks.

It is not essential to transfer complete 16 bit words into the CAM204. However it is recommended, since data from any incomplete word will be discarded.
A delay TWW is required between the rising edge of SCK that clocks out the LSB of adjacent data words. These timings are illustrated in Figure 33 and specified in Table 22.

If the host attempts to access a reserved or unimplemented address, incoming data will be discarded and the state of MISO is undefined.

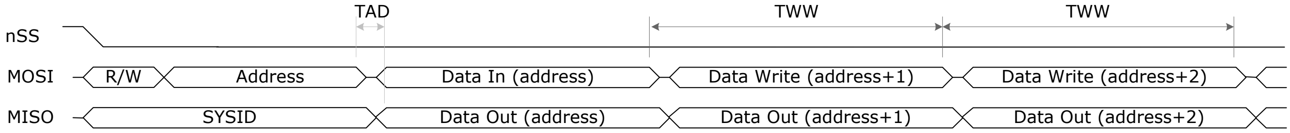


Figure 33 multiple register access

10.4 SPI Flag Clearing

User configurable CTU output lines can be configured as sample indicators which indicate when the CTU has new (or new valid) position data available for a sensor. They may also be configured as position triggers.

There are different mechanisms to clear these signals. One is for the host to clear flags with an SPI transaction, illustrated in Figure 34. There is a delay between the end of the SPI transaction and the IO state change, TnSSH2IOch, specified in Table 22.

TnSSH2IOch takes different values depending on the registers written by the host. The CTU's response is faster when the SIF or PTF bits are cleared (see sections 11.9 and 11.10). The response takes longer when IO functions are modified (e.g. changes to SIC, PTAH, PTC and PTP registers).

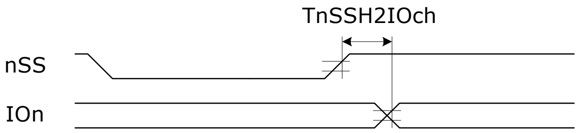


Figure 34 IO reset with SPI transaction

10.5 Interface Timing Specifications

Table 22 specifies the timing parameter values for SPI interface transactions. Parameters are defined in Figure 31, Figure 32, Figure 33 and Figure 34. TVALIDCHECK is defined in Figure 40.

Table 22 interface timing specifications

Parameter	Description	Min	Typ	Max	Units
TVALIDCHECK	Time for CTU internal validity checks following reset	12	-	20	ms
TSCKL	SCK Input Low Time	30	-	-	ns
TSCKH	SCK Input High Time	30	-	-	ns
TSCK	SCK clock period	100	-	-	ns
TSCKR	SCK Input Rise Time	-	10	25	ns
TSCKF	SCK Input Fall Time	-	10	25	ns
TMISOR	MISO Rise Time (50pF load)	-	-	25	ns
TMISOF	MISO Fall Time (50pF load)	-	-	25	ns
TMOSIV2SCKH	MOSI Setup Time	20	-	-	ns
TSCKH2MOSIX	MOSI Hold Time	20	-	-	ns
TSCKL2MISOV	MISO valid after SCK edge	-	-	30	ns
TnSSL2MISOV	MISO valid after nSS edge	-	-	5	µs
TnSSH2MISOZ	nSS high to MISO high Z	-	-	2	µs
TnSSL2SCKH	nSS low to SCK edge	5	-	-	µs
TSCKL2nSSH	Last SCK edge to nSS high	11	-	-	µs
TAD	Idle time between address end and first data start	7	-	-	µs
TWW	Data word to data word interval	20	-	-	µs
TnSSH2nSSL	nSS high time following read of register(s)	6	-	-	µs
	nSS high time following write to register(s) controlling IO, LED or DAC configuration	200	-	-	µs
	nSS high time, any other SPI transaction	40	-	-	µs
TnSSH2IOch	Time to clear an IO by clearing SIF or PTF bits	-	-	40	µs
	Time to change an IO following write to register(s) controlling IO, LED or DAC configuration	-	-	200	µs

10.6 Timing SPI Transactions and Position Measurements

An SPI transaction may occur while a position measurement is in progress. However, if it does, the position measurement may be abandoned and re-started. It is preferable to time SPI transactions and position measurements so that they do not coincide. This can be done by connecting using sample indicators, see section 12.7.

Alternatively, where maximum speed is not critical, SPI transactions may be separated by sufficient time for measurements to have finished. That is, at least the Sample Time (defined in section 12.8, sensor Type specific) plus adequate margin, per sensor.

11 Register Description

The host configures and controls the CAM204 CTU by writing to its internal registers. It determines status and measurement results by reading back from internal registers. This section describes the function of each of the CTU's registers.

Reading and writing to registers is done over the SPI interface, using the procedures described in section 10. Section 12 describes how to control the CTU to perform sensor position measurements, and how to configure the CTU's IO pins to implement position trigger signals.

The CTU's register map is arranged in to distinct sections: a system control section and one section for each sensor. The arrangement for a 4 sensor system is illustrated in Figure 35 and Figure 36.

The map has been arranged to reduce SPI overheads once the whole system has been initialised (following a reset). Once initialised, the host system will normally only need to access a small contiguous block of registers to control each sensor, which can be performed within a single SPI transaction.

	Address	Register
SENSOR 4	0x 4 1F	PTP8
	0x 4 1E	PTP7
	0x 4 1D	PTP6
	0x 4 1C	PTP5
	0x 4 1B	PTP4
	0x 4 1A	PTP3
	0x 4 19	PTP2
	0x 4 18	PTP1
	0x 4 17	PTAH
	0x 4 16	PTC87
	0x 4 15	PTC65
	0x 4 14	PTC43
	0x 4 13	PTC21
	0x 4 12	SIC
	0x 4 11	TYPE
	0x 4 10	DACCW
	0x 4 0F	DACLIMY
	0x 4 0E	DACLIMX
	0x 4 0D	DACPOSD
	0x 4 0C	DACPOSC
	0x 4 0B	DACPOSB
	0x 4 0A	DACPOSA
	0x 4 09	LEDCW
	0x 4 08	
	0x 4 07	RESF
	0x 4 06	RESE
	0x 4 05	RESD
	0x 4 04	RESC
	0x 4 03	RESB
	0x 4 02	RESA
	0x 4 01	PTEF
	0x 4 00	SCW

	Address	Register
SENSOR 3	0x 3 1F	PTP8
	0x 3 1E	PTP7
	0x 3 1D	PTP6
	0x 3 1C	PTP5
	0x 3 1B	PTP4
	0x 3 1A	PTP3
	0x 3 19	PTP2
	0x 3 18	PTP1
	0x 3 17	PTAH
	0x 3 16	PTC87
	0x 3 15	PTC65
	0x 3 14	PTC43
	0x 3 13	PTC21
	0x 3 12	SIC
	0x 3 11	TYPE
	0x 3 10	DACCW
	0x 3 0F	DACLIMY
	0x 3 0E	DACLIMX
	0x 3 0D	DACPOSD
	0x 3 0C	DACPOSC
	0x 3 0B	DACPOSB
	0x 3 0A	DACPOSA
	0x 3 09	LEDCW
	0x 3 08	
	0x 3 07	RESF
	0x 3 06	RESE
	0x 3 05	RESD
	0x 3 04	RESC
	0x 3 03	RESB
	0x 3 02	RESA
	0x 3 01	PTEF
	0x 3 00	SCW

	Address	Register
SENSOR 2	0x 2 1F	PTP8
	0x 2 1E	PTP7
	0x 2 1D	PTP6
	0x 2 1C	PTP5
	0x 2 1B	PTP4
	0x 2 1A	PTP3
	0x 2 19	PTP2
	0x 2 18	PTP1
	0x 2 17	PTAH
	0x 2 16	PTC87
	0x 2 15	PTC65
	0x 2 14	PTC43
	0x 2 13	PTC21
	0x 2 12	SIC
	0x 2 11	TYPE
	0x 2 10	DACCW
	0x 2 0F	DACLIMY
	0x 2 0E	DACLIMX
	0x 2 0D	DACPOSD
	0x 2 0C	DACPOSC
	0x 2 0B	DACPOSB
	0x 2 0A	DACPOSA
	0x 2 09	LEDCW
	0x 2 08	
	0x 2 07	RESF
	0x 2 06	RESE
	0x 2 05	RESD
	0x 2 04	RESC
	0x 2 03	RESB
	0x 2 02	RESA
	0x 2 01	PTEF
	0x 2 00	SCW

	Address	Register	Description
SENSOR 1	0x 1 1F	PTP8	Pos Trig Position 8
	0x 1 1E	PTP7	Pos Trig Position 7
	0x 1 1D	PTP6	Pos Trig Position 6
	0x 1 1C	PTP5	Pos Trig Position 5
	0x 1 1B	PTP4	Pos Trig Position 4
	0x 1 1A	PTP3	Pos Trig Position 3
	0x 1 19	PTP2	Pos Trig Position 2
	0x 1 18	PTP1	Pos Trig Position 1
	0x 1 17	PTAH	PT Auto-Clear & Hysteresis
	0x 1 16	PTC87	Position Trigger 8&7 Control
	0x 1 15	PTC65	Position Trigger 6&5 Control
	0x 1 14	PTC43	Position Trigger 4&3 Control
	0x 1 13	PTC21	Position Trigger 2&1 Control
	0x 1 12	SIC	Sample Indicator Control
	0x 1 11	TYPE	Sensor Type
	0x 1 10	DACCW	DAC Control Word
	0x 1 0F	DACLIMY	DAC Limit Y
	0x 1 0E	DACLIMX	DAC Limit X
	0x 1 0D	DACPOSD	DAC Position D
	0x 1 0C	DACPOSC	DAC Position C
	0x 1 0B	DACPOSB	DAC Position B
	0x 1 0A	DACPOSA	DAC Position A
	0x 1 09	LEDCW	LED Control Word
	0x 1 08		(Reserved)
	0x 1 07	RESF	Result Register F
	0x 1 06	RESE	Result Register E
	0x 1 05	RESD	Result Register D
	0x 1 04	RESC	Result Register C
	0x 1 03	RESB	Result Register B
	0x 1 02	RESA	Result Register A
	0x 1 01	PTEF	Pos Trig Enables/Flags
	0x 1 00	SCW	Sensor Control Word

System	0x 0 0F	SYSID	Device ID
	0x 0 0E	SYSVER	System Version Number
	0x 0 0D	BOOTVER	Bootloader Version Number
	0x 0 0C	SAVEKEY	Save Key
	0x 0 0B		(Reserved)
	0x 0 0A		(Reserved)
	0x 0 09		(Reserved)
	0x 0 08		(Reserved)
	0x 0 07		(Reserved)
	0x 0 06		(Reserved)
	0x 0 05		(Reserved)
	0x 0 04		(Reserved)
	0x 0 03	SYSDAC	System DAC Register
	0x 0 02	SYSIO	IO Pin Types
	0x 0 01	SYSI	Continuous Sample Interval
	0x 0 00	SYSCW	System Control Word

Figure 35 Register map overview

	Address	Description	Register	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SENSOR 1	0x 1 1F	Pos Trig Position 8	PTP8	PTP8 [15:0]															
	0x 1 1E	Pos Trig Position 7	PTP7	PTP7 [15:0]															
	0x 1 1D	Pos Trig Position 6	PTP6	PTP6 [15:0]															
	0x 1 1C	Pos Trig Position 5	PTP5	PTP5 [15:0]															
	0x 1 1B	Pos Trig Position 4	PTP4	PTP5 [15:0]															
	0x 1 1A	Pos Trig Position 3	PTP3	PTP3 [15:0]															
	0x 1 19	Pos Trig Position 2	PTP2	PTP2 [15:0]															
	0x 1 18	Pos Trig Position 1	PTP1	PTP1 [15:0]															
	0x 1 17	PT Auto-Clear & Hysteresis	PTAH	PTAUTOCLR	HYSTERESIS [14:0]														
	0x 1 16	Position Trigger 8&7 Control	PTC87	PT8ACT [1:0]		PT8DIR [1:0]		PT8MAP [3:0]				PT7ACT [1:0]		PT7DIR [1:0]		PT7MAP [3:0]			
	0x 1 15	Position Trigger 6&5 Control	PTC65	PT6ACT [1:0]		PT6DIR [1:0]		PT6MAP [3:0]				PT5ACT [1:0]		PT5DIR [1:0]		PT5MAP [3:0]			
	0x 1 14	Position Trigger 4&3 Control	PTC43	PT4ACT [1:0]		PT4DIR [1:0]		PT4MAP [3:0]				PT3ACT [1:0]		PT3DIR [1:0]		PT3MAP [3:0]			
	0x 1 13	Position Trigger 2&1 Control	PTC21	PT2ACT [1:0]		PT2DIR [1:0]		PT2MAP [3:0]				PT1ACT [1:0]		PT1DIR [1:0]		PT1MAP [3:0]			
	0x 1 12	Sample Indicator Control	SIC									SAUTOCLR [1:0]		SCTRL [1:0]		SMAP [3:0]			
	0x 1 11	Sensor Type	TYPE															TYPE [1:0]	
	0x 1 10	DAC Control Word	DACCW	DACNV [15:8]								DACEN	-	-	DACADDR [2:0]			DACOP [1:0]	
	0x 1 0F	DAC Limit Y	DACLIMY	DACLIMY[15:0]															
	0x 1 0E	DAC Limit X	DACLIMX	DACLIMX[15:0]															
	0x 1 0D	DAC Position D	DACPOSD	DACPOSD[15:0]															
	0x 1 0C	DAC Position C	DACPOSC	DACPOSC[15:0]															
	0x 1 0B	DAC Position B	DACPOSB	DACPOSB[15:0]															
	0x 1 0A	DAC Position A	DACPOSA	DACPOSA[15:0]															
	0x 1 09	LED Control Word	LEDCW	LEDTHRESHOLD [11:0]												LEDMAP [3:0]			
	0x 1 08	(Reserved)																	
	0x 1 07	Result Register F	RESF	RESF [15:0]															
	0x 1 06	Result Register E	RESE	RESE [15:0]															
	0x 1 05	Result Register D	RESD	RESD [15:0]															
	0x 1 04	Result Register C	RESC	RESC [15:0]															
	0x 1 03	Result Register B	RESB	RESB [15:0]															
	0x 1 02	Result Register A	RESA	RESA [15:0]															
	0x 1 01	Pos Trig Enables/Flags	PTEF	PT8E	PT7E	PT6E	PT5E	PT4E	PT3E	PT2E	TPT1E	PT8F	PT7F	PT6F	PT5F	PT4F	PT3F	PT2F	PT1F
	0x 1 00	Sensor Control Word	SCW	SENSOR [3:0]				0	0	INCE	INCF	SIE	SIF	NEW	VALID		TRIG	CONT	GO

	Address	Description	Register	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
System	0x 0 0F	Device ID	SYSID	SYSID [15:0]															
	0x 0 0E	System Version Number	SYSVER	SYSVER [15:0]															
	0x 0 0D	Bootloader Version Number	BOOTVER	BLVER [15:0]															
	0x 0 0C	Save Key	SAVEKEY	SAVEKEY [15:0]															
	0x 0 0B	(Reserved)																	
	0x 0 0A	(Reserved)																	
	0x 0 09	(Reserved)																	
	0x 0 08	(Reserved)																	
	0x 0 07	(Reserved)																	
	0x 0 06	(Reserved)																	
	0x 0 05	(Reserved)																	
	0x 0 04	(Reserved)																	
	0x 0 03	System DAC Register	SYSDAC	DACCAL	-	-	-	-	-	CPOL	CPHA					DACFORMAT [2:0]			DACSON
	0x 0 02	IO Pin Types	SYSIO	-	-	-	-	-	-	-	-	INT4AH	INT4DO	INT3AH	INT3DO	INT2AH	INT2DO	INT1AH	INT1DO
	0x 0 01	Continuous Sample Interval	SYSI								SYSI [9:0]								
	0x 0 00	System Control Word	SYSCW	RESET	BOOTLOAD	SAVE	FACTORY												PWRDN

🟩 Read Only ☐ Read / Write 🟨 Read / Write to 0

Figure 36 Register map detail

11.1 SYSCW: System Control Word

SYSCW	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	RESET	BOOTLOAD	SAVE	FACTORY	-	-	-	-	-	-	-	-	-	-	-	PWRDN
Access	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Factory default value = 0x0000

Writing a 1 to the RESET bit will result in a device reset at the end of the SPI transaction. Registers will return to their default values and measurements will be aborted. SPI transactions will be ignored until the device comes out of reset.

Writing a 1 to the PWRDN bit pauses measurements in progress and puts the CTU into a low power state. To take the CTU out of this state and resume any measurements in progress, toggle nSS low. This operation may take the form of a dummy SPI read, whose results should be discarded.

Writing a 1 to the BOOTLOAD bit resets the CTU at the end of the SPI transaction. When the CTU comes out of reset, it will remain in Bootloader Mode (section 14.6) until the next reset.

Writing a 1 to the SAVE bit, together with 0x0C1C to the SAVEKEY register, will result in the current register contents being made the Configurable Defaults (section 12.16). Writing a 1 to the FACTORY bit, together with 0x0C1C to the SAVEKEY register, will result in restoration of the factory default register values. After the SAVE or FACTORY operations the CTU will reset.

11.2 SYSI: System Interval Register

SYSI	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x001	-	-	-	-	-	-	-	INTERVAL [8:0]								
Access	R	R	R	R	R	R	R	R/W								

Factory default value = 0x000A

SYSI controls the interval between measurements on each active sensor in continuous mode. The interval is measured in ms, and is approximate.

If the sample interval is less than the Sample Time specified for the sensor Type then the CTU will extend the sample interval to the minimum possible for the system. This may not yield a constant sample rate, since measurement and calculation times may vary.

If the host performs an SPI transaction while a measurement is in progress, the CTU may need to re-start the measurement, causing a delay.

Table 23 effect of INTERVAL, Type 1 sensors

INTERVAL	Time between measurements
0	0.8ms
1	1 ms
2	2 ms
-	-
510	510 ms
511	511 ms

11.3 SYSIO: System IO Configuration

SYSIO	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x002	-	-	-	-	-	-	-	-	INT 4AH	INT 4DO	INT 3AH	INT 3DO	INT 2AH	INT 2DO	INT 1AH	INT 1DO
Access	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Factory default value = 0x0000

The SYSIO register controls each IO pin's behaviour. Each pin may be configured for active low or active high with the coding of Table 24. Independently, each pin may be configured for open drain or digital with the coding of Table 25.

Table 24

INTnAH	Function
0	Active Low
1	Active High

Table 25

INTnDO	Function
0	Open Drain
1	Digital Output

11.4 SYSDAC: System DAC Register

SYSDAC	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x003	DACCAL	-	-	-	-	-	CPOL	CPHA	NVMODE [1:0]		-	-	DACFORMAT [2:0]			DACSON
Access	R/W	R	R	R	R	R	R/W	R/W	R/W		R	R	R/W			R/W

Factory default value = 0x0000

The SYSDAC register controls how the CTU interfaces to an external Digital to Analog Converter (DAC), if present. The host should set the DACSON bit if an external DAC is connected (Table 26). DACFORMAT[2:0] must be set to match the DAC's data format (Table 38). NVMODE[1:0] defines how the DAC output(s) behave when its matching sensor's last measurement was not valid (Table 42). CPOL and CPHA must be set to match the DAC SPI interface's mode (Table 27). DACCAL allows the host to set the DAC data directly for calibration purposes, if set (Table 28).

Table 26

DACSON	**Function**
0	No DAC output(s)
1	Control external DAC output(s)

Table 27

CPOL	**CPHA**	**DAC SPI Mode**
0	0	0
0	1	1
1	0	2
1	1	3

Table 28

DACCAL	**Function**
0	Normal DAC operation
1	DAC output forced, for calibration

11.5 SAVEKEY: Save Key

BOOTVER	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x00D	SAVEKEY[15:0]															
Access	R/W															

Default value = 0x0000

This system register is used to prevent accidental changes being made to register defaults. For the SAVE and FACTORY operations to take effect, its value should be set to 0x0C1C.

11.6 BOOTVER: Bootloader Version Number

BOOTVER	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x00D	Bootloader Version Number [15:0]															
Access	R															

Factory default value = 0x????

The BOOT register contains the fixed revision number for the CTU's Bootloader Code, and is read-only.

11.7 SYSVER: System Version

SYSVER	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x00E	System Version Number [15:0]															
Access	R															

Factory default value = 0x????

The SYSVER register contains the version number of the CTU's Application Code, and is read-only. It is updated when new code is successfully updated using the bootloader (section 13).

11.8 SYSID: System Device ID

SYSID	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x00F	DEVICEID[15:0]															
Access	R/W															

Factory default value = 0xABCD

The System Device ID is a host-writeable word. Its only function is that SYSID is output to the SPI master as the first word of every transaction (see Figure 32). It is recommended that the host verifies this value against the expected one, to assist detection of communication errors.

The Device ID may be used to identify different SPI slave devices connected to the same host. Any value may be chosen; it is of no significance to the CTU. It is preferable to avoid 0x0000 and 0xFFFF, since these may also typically result from communication errors. 0x10AD should also be avoided since the bootloader uses this value to signal missing Application Code (section 14).

The Device ID may be used to detect or verify that a CTU reset has occurred. The host should write a new Device ID, and any subsequent reset will cause the Device ID to return to its default value.

11.9 SCW: Sensor Control Word

SCW	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0xn00	SENSOR[3:0]				0	0	INCE	INCF	SIE	SIF	NEW	VALID	-	TRIG	CONT	GO
Access	R				R	R	R/W	R	R/W	R/W0	R/W0	R	R	R/W	R/W	R/W

Factory default value = 0xn000

The registers described above relate to system functions. The Sensor Control Word, and the other registers described below, relate to sensors connected to the CTU. More than one sensor may be connected, depending on sensor type, and these registers are distinguished by bits 8 to 11 of the address. For example, up to 4 Type 1 sensors may be connected, and the addresses of their SCW registers are 0x100, 0x200, 0x300 and 0x400.

The host can start a measurement on a sensor by setting its GO bit to 1. If the host also sets the CONT bit to 1 the CTU will measure that sensor continuously. In this case, the sample interval between measurements on each active sensor is configured with SYSI, see section 11.2. If a sensor's CONT bit is set to 0, setting its GO bit to 1 will result in a *single shot* measurement. The CTU will clear GO to 0 upon completion.

The TRIG bit is a global enable bit which, when set, instructs the CTU to look for position trigger events after each sample from that sensor. TRIG must be set for the CTU to generate position triggers. Please refer to section 12.11 for more on position triggers.

The VALID bit indicates that the last measurement result for the sensor was valid. The CTU sets the VALID flag when it determines the channel's target to be *in range*. The meaning of in range depends on the sensor and target, but typically means that it is physically aligned within specification and its resonant frequency is within specified limits.

The NEW flag indicates that new measurement data is available for the sensor. The CTU sets NEW to 1 when a measurement is completed. It is recommended that the host clears NEW back to 0 when reading result registers, using a multiple register write/read SPI transaction spanning SCW and the relevant results registers. That way, the host can unambiguously verify that the results it has collected from the CTU are new.

The SIF bit is the sample indicator flag, and SIE is the sample indicator enable bit. A sensor's sample indicator can be configured to activate an IO when a new, or a new valid, sensor measurement is available. When suitably configured (see section 11.17), the SIE bit controls whether sample indicators are enabled. Once activated, the host may clear a sample interrupt by writing a 0 to SIF (see also section 10.4 concerning timing).

INCE and INCF are for sensor Type 2, which supports incremental mode. The INCE bit controls whether incremental mode is allowed. The INCF bit is a flag indicates when an incremental measurement was actually performed. See section 5.2 for details.

SENSOR[3:0] field is read-only and equals the sensor number.

11.10 PTEF: Position Trigger Enables and Flags (Type 1 Sensors)

PTEF	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xn01	PT 8E	PT 7E	PT 6E	PT 5E	PT 4E	PT 3E	PT 2E	PT 1E	PT 8F	PT 7F	PT 6F	PT 5F	PT 4F	PT 3F	PT 2F	PT 1F
Access	R/W								R/W0							

Factory default value = 0x0000

A sensor's PTEF register is used to control position triggers mapped to user IO pins. Each sensor can have up to 8 position triggers. The host can individually enable each using the PTmE bits. m is the position trigger index for that sensor, not to be confused with n, used to denote sensor number.

When a position trigger occurs causing an IO pin to activate (PTmACT set to 0, see section 11.18), it may be cleared by writing a 0 to the respective PTmF bit. In some applications, position triggers are required in the absence of SPI transactions. In this case, position triggers may be auto-cleared by the CTU instead, see section 0.

11.11 RESA…RESF: Results Registers

RESA … RESF	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xn02 … 0xn07	RESA [15:0] … RESF [15:0]															
Access	R															

Factory default values = 0x0000

A sensor's results registers contain the results of the last measurement to complete on that sensor. The CTU supports different sensor types. The interpretation of result registers depends on the sensor type as described in Table 29.

Table 29

Register	Sensor Type		
	1 and 4	2	3
RESA	CtuReportedPositionI16	CtuReportedPositionI32 high word	CtuReportedPositionI32 high word
RESB	AmplitudeU16	CtuReportedPositionI32 low word	CtuReportedPositionI32 low word
RESC	RelativeFrequencyI16	AmplitudeAU16	AmplitudeAU16
RESD	Not used	AmplitudeBU16	AmplitudeBU16
RESE	Not used	RelativeFrequencyI16	RelativeFrequencyI16
RESF	Not used	BAPositionMismatchI16	Not used

CtuReportedPosition may be converted into measurement units with Equation 5 below. *Sin Length* is a characteristic of the sensor, and is quoted in sensor datasheets. Type 2 sensors have fine and coarse pitches. Sin Length should be set to Fine Pitch for use in Equation 5.

Section 4 specifies the performance of the CAM204 CTU with Type 1 sensors. In this case it is convenient to define Reported Position in % so that it is independent of individual sensors, with SinLength = 100%.

$$ReportedPosition = \frac{CtuReportedPosition}{65536} \times SinLength$$

Equation 5

Note that Reported Position is a signed quantity which is nominally 0 when the Target Origin is aligned with the Sensor Origin. This is usually at the centre point of the sensor's travel.

CtuReportedPosition will be forced to 0 following an invalid measurement when the target is out of range. The host can use the VALID bit of the SCW register to distinguish this condition from zero Reported Position.

Amplitude is a measure of inductive signal strength and influences system performance, particularly resolution (for example see section 4.7). Type 2 sensors have two Amplitude results, one for fine (AmplitudeA) and one for coarse (AmplitudeB). The coarse value is significantly less than the fine.

Relative Frequency is the CTU's approximate measurement of the frequency difference in Hz between the target and the CTU's Centre Frequency.

BA Position Mismatch is specific to Type 2 sensors, and is described in section 5.3.

11.12 LEDCW: LED Control Word

LEDCW	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xn09	LEDTHRESHOLD[11:0]												LEDMAP[3:0]			
Access	R/W												R/W			

Factory default value = 0x0000

User IOs can be configured to activate when the result of a sensor's last measurement was VALID. Each sensor's LEDMAP[3:0] controls which user IO is activated by that sensor. The mapping is as in Table 31. The IO will toggle at approximately 2Hz for Amplitudes less than LEDTHRESHOLD, and it will activate continuously for Amplitudes greater or equal to LEDTHRESHOLD. Please refer to section 12.10 for more details on using this feature.

11.13 DACPOSA, DACPOSB, DACPOSC, DACPOSD

RESA … RESF	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xn0A … 0xn0D	DACPOSA[15:0] … DACPOSD[15:0]															
Access	R/W															

Factory default values = 0x0000

DACPOSA, DACPOSB, DACPOSC and DACPOSD are used when an external DAC is connected. They define the relationship between measured sensor position and the DAC output mapped to the sensor (the *DAC Transfer Function*). They are in signed, 16-bit position units. DACPOSA and DACPOSB define the beginning and end of the *up slope*. DACPOSC and DACPOSD define the beginning and end of the *down slope*. Please see section 13.4 for more details.

11.14 DACLIMX, DACLIMY

DACLIMX,Y	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xn0E, 0xn0F	DACLIMX[15:0] … DACLIMY[15:0]															
Access	R/W															

Factory default values = 0x0000

DACLIMX and DACLIMY are used when an external DAC is connected. They define the relationship between measured sensor position and the DAC output mapped to the sensor (the *DAC Transfer Function*). They are in unsigned, 16-bit DAC Code units. DACLIMX defines the lower DAC value, and DACLIMY the upper DAC value. Please see section 13.4 for more details.

11.15 DACCW: DAC Control Word

DACCW	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xn10	DACNV[15:8]								DACEN	-	-	DACADDR[2:0]			DACOP[1:0]	
Access	R/W								R/W	R	R	R/W			R/W	

Factory default value = 0xn000

DACCW, the DAC Control Word, is used when an external DAC is connected. DACEN is a sensor-specific enable bit for CTU control of an external DAC. Where the CTU is connected to a DAC with multiple outputs, a sensor's DACADDR[2:0] bits control which DAC output the sensor controls. DACOP[1:0] contains DAC output control bits. Their function is DAC-specific, and usually includes a power-down mode. DACNV[15:8] controls the DAC output when the sensor's last measurement was not valid. The exact behaviour depends on NVMODE[1:0]. DACNV[15:0] is in DAC Code units. Please see section 13 for more details.

11.16 TYPE: Sensor Type Register

TYPE	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xn11	-	-	-	-	-	-	-	-	-	-	-	-	-	TYPE[2:0]		
Access	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W (sensor1 only)		

Factory default value = 0x0001

The CTU is designed to operate with different types of sensor, see Table 1. The default is Type 1. To operate different sensor Types, the host must write the appropriate number to sensor 1's TYPE register before any position measurements. Other sensors' TYPE registers are read only. They will read either 0 if measurement on this sensor is not allowed or sensor 1's TYPE if measurement is allowed.

For example, two Type 4 sensors can be measured. Write 4 to sensor 1's TYPE register. A read from either sensor 1 or 2's TYPE register will then yield 4. A read from sensor 3 or 4's TYPE register will yield 0.

11.17 SIC: Sample Indicator Control Register

SIC	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xn12	-	-	-	-	-	-	-	-	SAUTOCLR[1:0]		SCTRL[1:0]		SMAP[3:0]			
Access	R	R	R	R	R	R	R	R	R/W		R/W		R/W			

Factory default value = 0x0000

Sample indicators signal whether the CTU has new, or new valid, measurement results available for a sensor. The function is configured with SCTRL[1:0], see Table 30.

Table 30

SCTRL[1:0]	Function
0	No sample indicators
1	Set on any new position
2	Set if VALID flag set
3	Reserved

Sample indicators can be configured to control the CTU's digital IOs. SMAP[3:0] configures which of the available IOs is used for the sensor's sample indicator. See Table 31 for how this bit field is encoded.

Table 31

SMAP[3:0]	IO pin mapping
0	Not mapped to an IO
1	IO1
2	IO2
3	IO3
4	IO4
5-15	reserved

Sample indicators can be reset by the host over the SPI interface by writing a 0 to the sample indicator flag SIF, see section 11.9. Alternatively, they may be auto-cleared. SAUTOCLR[1:0] controls sample indicator auto-clear behaviour. Table 32 defines how this bit field is encoded.

Table 32

SAUTOCLR[1:0]	Auto clear behaviour
0	Autoclear OFF
1	Clear before measurement
2	Clear after measurement
3	Reserved

Sample indicators can be used to generate an IO whose state reflects whether the sensor's last measurement was VALID or not. Set SCTRL to 2 and SAUTOCLR to 2. An LED control function behaves similarly (section 12.10), except for the addition of a flashing state for low Amplidue.

11.18 PTC21–PTC87: Position Trigger Control Registers

PTC21 – 87	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xn13 -0xn16	PT2ACT[1:0] - PT8ACT[1:0]		PT2DIR[1:0] - PT8DIR[1:0]		PT2MAP[3:0] - PT8MAP[3:0]				PT1ACT[1:0] - PT7ACT[1:0]		PT1DIR[1:0] - PT7DIR[1:0]		PT1MAP[3:0] - PT7MAP[3:0]			
Access	R	R	R/W		R/W				R	R	R/W		R/W			

Factory default value = 0x0000

Each sensor has 4 position trigger control registers controlling the behaviour of its 8 position triggers. The registers each control 2 position triggers.

The PTmMAP[3:0] bits control which user configurable IO pin, if any, the mth position trigger is assigned to. The mapping is defined in Table 33.

The PTmDIR[1:0] bits control which direction(s) the position trigger is triggered.

The PTmACT[1:0] bits control the Position Trigger Action. The available actions are summarised in Table 35.

When set to "Activate IO", the position trigger activates the selected IO. The INTnAH bits in the SYSIO register (section 11.3) determine whether the active state is high or low. The IO is reset by writing a 0 to the appropriate PTmF bit of the PTEF register (section 11.10). Alternatively, for generating position trigger pulses independent of SPI operation, set PTAUTOCLR to 1 (section 11.19).

The position trigger action can alternatively be set to "posLnegH" or "posHnegL". When set to posLnegH the selected IO will be forced low when it triggers in the positive direction, and high when it triggers in the negative direction, see Figure 37. When set to posHnegL the selected IO will be forced low when it triggers in the negative direction, and high when it triggers in the positive direction, see Figure 38.

When position trigger action is set to Toggle IO, the selected IO changes state when the associated position trigger is triggered.

Section 12.11 describes how to use position triggers.

Table 33 Position Trigger Mapping

PTmMAP[3:0]	Function
0	Not mapped to a user configurable IO pin
1	IO1
2	IO2
3	IO3
4	IO4
5-15	reserved

Table 34 Position Trigger Direction

PTmDIR [1:0]	When position trigger m is set
0	Never
1	Positive trigger event
2	Negative trigger event
3	Both directions

Table 35 Position Trigger Action

PTmACT [1:0]	Position Trigger Action
0	Activate IO
1	posLnegH
2	posHnegL
3	Toggle IO

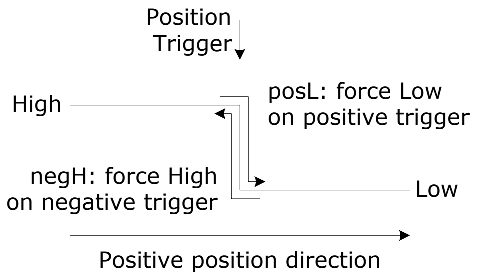


Figure 37 posLnegH

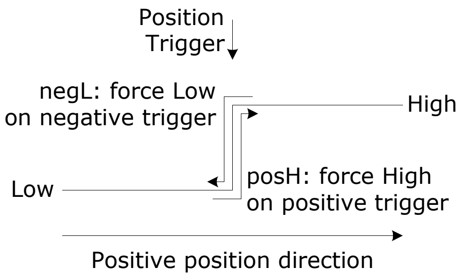


Figure 38 posHnegL

11.19 PTAH: Position Trigger Auto-Clear and Hysteresis

PTAH	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0xn17	PTAUTOCLR	HYSTERESIS[14:0]														
Access	R/W	R/W														

Factory default value = 0x0000

Each sensor's PTAUTOCLR bit determines how position triggers are cleared when PTmACT is set to "Activate IO". If the host sets PTAUTOCLR to 1, Auto-Clear is On and the CTU automatically clears each position trigger during the measurement following the one that set it. This way trigger events can be configured to generate transitions on IO pins independent of host SPI activity. Alternatively, if PTAUTOCLR is 0, Auto-Clear is Off and position triggers can only be cleared by the host, by writing a 0 to the appropriate bit(s) of the PTEF register (section 11.10).

HYSTERESIS[14:0] sets the level of hysteresis applied to position triggers and DAC output. It is used to prevent multiple position triggers and DAC noise, when that is required. Its function is described in section 12.12. It is scaled in the same way as CtuReportedPositionI16.

11.20 PTP1–PTP8: Position Trigger Position Registers

PTP1 – PTP8	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0xn18 – 0xn1F	PTP1[15:0] – PTP8[15:0]															
Access	R/W															

Factory default value = 0x0000

Each sensor has a set of 8 position trigger position registers. They set the positions at which position triggers activate, and can be in any order. They are scaled in the same way as CtuReportedPositionI16. Their operation is described in section 12.11.

12 System Operation

12.1 Overview

The CAM204 CTU operates up to 4 resonant inductive sensors, depending on sensor type. It measures the position of contactless targets relative to each sensor. The results of measurements are available from the CTU's SPI interface. The CTU can also generate position triggers and sample indicators on digital IO.

The CTU can be operated in a number of modes to suit the host system. This main section describes how the host can configure and operate each mode. All configuration is through the SPI interface hardware described in section 10, and is performed by writing to and reading from the registers described in section 11.

Register settings may be saved to non-volatile memory, for applications requiring autonomous operation.

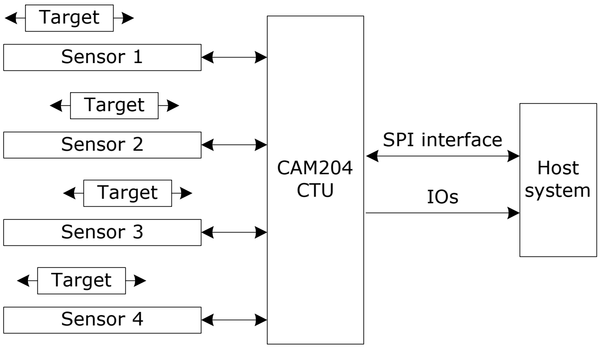


Figure 39 functional block diagram

12.2 Internal Validity Checking

The CTU is internally reset following power on, when nRESET is toggled or when the host writes a 1 to the RESET bit in the SYSCW register (section 11.1).

Following a reset, the CTU performs internal validity checks. These include a checksum of its FLASH memory contents, including Application Code. The time taken for these checks is TVALIDCHECK, as specified in Table 22. For normal operation, the host should allow at least TVALIDCHECK(MAX) between a reset and the first communication with the CTU. Then the CTU will return the SYSID register contents (factory default 0xABCD) as the first word of each SPI transaction. If the internal validity checks fail, for example due to missing or incomplete Application Code, the CTU will return 0x10AD instead.

The CTU may be configured to measure one or more sensors continuously following a reset (section 12.16). Measurements will begin at the end of validity checks.

The CTU includes Bootloader Code which may be used to upload new Application Code into the CTU (section 14). Bootloader mode can be entered by interrupting the internal validity checks with an SPI transaction, as illustrated in Figure 40. If the host accidentally interrupts the CTU's internal validity checks, the CTU will report 0x10AD to indicate it is in bootloader mode. In this case, The CTU tests whether the first 2 words sent by the host are a valid start to new Application Code. If not, the CTU resumes validity checks.

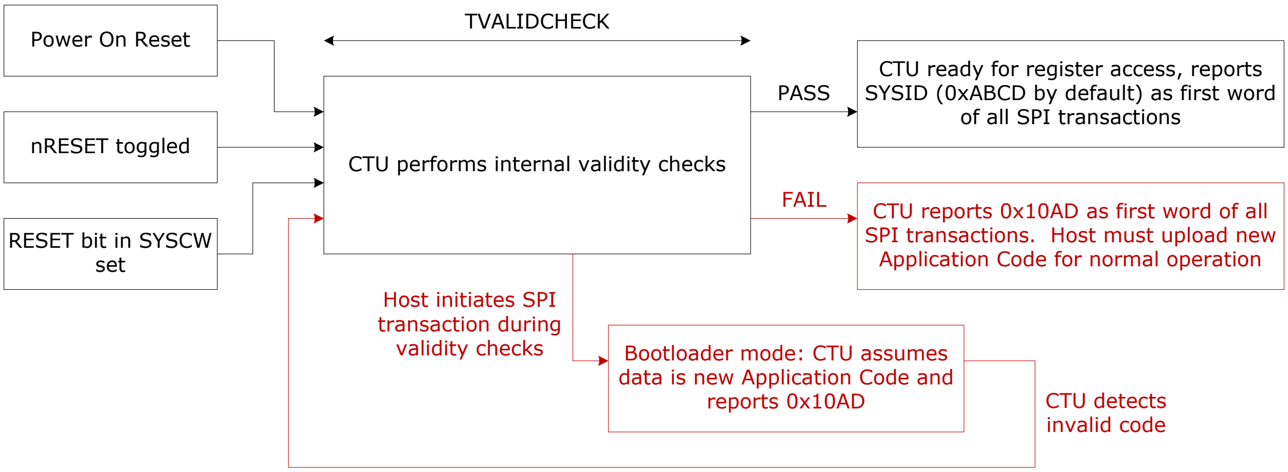

Power On Reset
nRESET toggled
RESET bit in SYSCW set
TVALIDCHECK
CTU performs internal validity checks
PASS
CTU ready for register access, reports SYSID (0xABCD by default) as first word of all SPI transactions
FAIL
CTU reports 0x10AD as first word of all SPI transactions. Host must upload new Application Code for normal operation
Host initiates SPI transaction during validity checks
Bootloader mode: CTU assumes data is new Application Code and reports 0x10AD
CTU detects invalid code

Figure 40 CTU reset process

12.3 Setting Sensor Type

The default sensor type is Type 1. The CTU may be connected to sensors of other Types, summarised in section 1.2. The maximum number of sensors that can be connected depends on the Type. Where this is more than one, the sensors must all be of the same Type. The TYPE register (section 11.16) is used to indicate to the CTU which Type is connected. Before any measurement, sensor 1's TYPE register must be written with the appropriate Type of all sensors that are connected.

12.4 Single Shot Measurement

The CTU begins a single shot measurement when the host writes a 1 to a sensor's GO bit in its SCW register, with CONT = 0 (section 11.9). The results of the measurement are available from the sensor's results registers upon completion (section 11.11). The time it takes to complete a measurement is the Sample Time and is specific to each type of sensor.

When reading measurements, it is recommended to use a multiple register SPI transaction spanning the SCW, PTEF and results registers (see section 10.3). That way, there is no risk of data skew. The VALID and NEW bits in the SCW, amplitude and position readings will all come from the same measurement.

The NEW bit in the SCW register can be used to verify that the host has collected the results of a new measurement. To do this, the host should clear the NEW bit in the SCW register to 0 when writing to an SCW register, and check that the NEW bit read back from the CTU is set to 1 in the data returned. If NEW is not set, the results have not been updated since the last measurement on that sensor, most likely because the interval between setting GO to 1 and reading results was too short, or because the measurement was interrupted by other CTU SPI activity.

When measuring a sensor's target for the first time after power on, or following measurements with the target absent, the CTU may take a number of measurements before it is able to lock on to the target's frequency and report VALID. Section 9 specifies the maximum number.

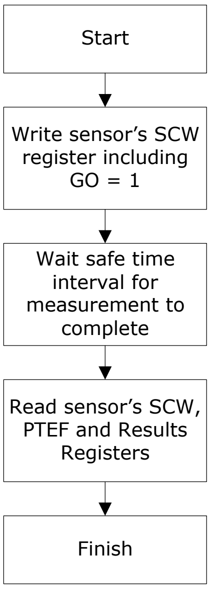

Start
Write sensor's SCW register including GO = 1
Wait safe time interval for measurement to complete
Read sensor's SCW, PTEF and Results Registers
Finish

Figure 41 single shot measurement

12.5 Repeated Single Shot

The host may repeat single shot measurements in a loop. The procedure is similar to above, except that it is possible to use a single write read SPI transaction to read results and kick off the next measurement. This process is illustrated in Figure 42.

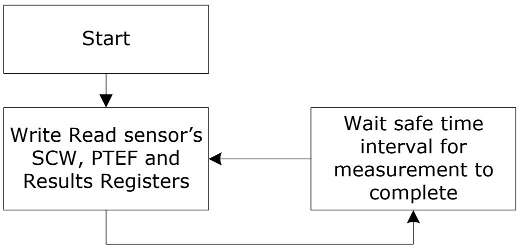

Start
Write Read sensor's SCW, PTEF and Results Registers
Wait safe time interval for measurement to complete

Figure 42 repeated single shot, 1 sensor

12.6 Repeated Single Shot, Multiple Sensors

The procedure described in section 12.7 can be extended to the repeated measurement of multiple sensors. Figure 43 illustrates an example with 4 sensors.

The SPI transactions should ideally be grouped so that there is a minimum of time in between them, so that the CTU does not attempt to begin measurements on any sensor until SPI activity has completed. This minimum time is determined by the parameter TnSSH2nSSL which specifies the time that nSS is held high between SPI transactions, and its value is in Table 22. It is recommended that the maximum time be no more than twice this value.

Sensors should ideally be addressed in numeric order as illustrated in Figure 43, since the CTU measures them in this order.

When all the SPI transactions are complete, the CTU will sample each activated sensor in turn. The time taken will be the number of activated sensors times the Sample Interval.

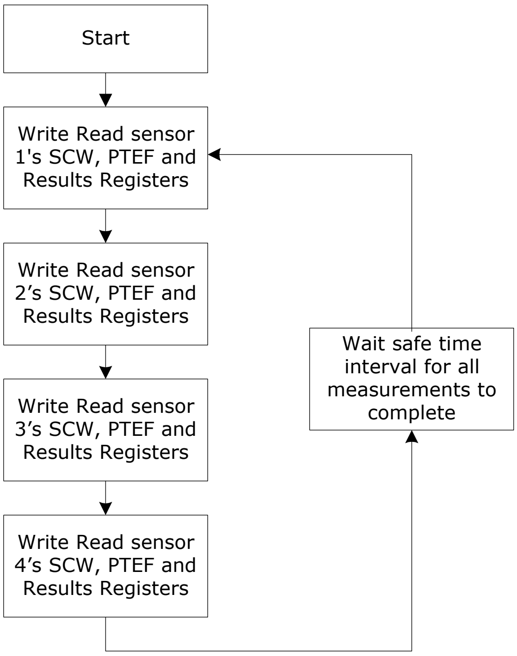

Start
Write Read sensor 1's SCW, PTEF and Results Registers
Write Read sensor 2’s SCW, PTEF and Results Registers
Write Read sensor 3’s SCW, PTEF and Results Registers
Write Read sensor 4’s SCW, PTEF and Results Registers
Wait safe time interval for all measurements to complete

Figure 43 repeated single shot, 4 sensors

12.7 Single Shot Measurement using Sample Indicators

The sections above described a single shot measurement procedure where the host was responsible for leaving a safe time for measurements to complete. Leaving a safe time may be acceptable for simple systems where timing is not critical.

For higher speed and deterministic behaviour it is preferred to use one or more of the CTU’s sample indicators to signal to the host when measurements are complete. The signal indicator is mapped to an IO. This may be used to trigger an interrupt in the host system. Alternatively, it may be polled by the host. Unlike reading a sensor’s NEW bit in its SCW register, no SPI activity is required, so CTU measurements need not be interrupted.

A sensor’s sample indicators are configured by writing to its SIC register (see section 11.17). The designer should choose which IO to use, and configure SMAP[3:0] accordingly. The SCTRL[1:0] bits should be configured so that sample indicators are triggered on a new sample. SAUTOCLR[1:0] should normally be set to 0, so that the selected IO is not automatically cleared by the CTU.

The CTU’s IOs are configured using the SYSIO register, see section 11.3. This determines whether IOs are active high or low, and whether they are digital or open drain.

Once this configuration is complete, repeated single shot measurements can be taken as before. This time two more bits in the SCW register are important. The host should write a 1 to the SIE bit in the SCW register of any sensor required to trigger a sample interrupt, to enable the indicator. It should also write a 0 to the SIF bit to clear the previous sample indicator.

Figure 44 illustrates this process for the repeated measurement of a single sensor. Figure 45 is for two sensors, 1 and 2. In this case it is only necessary to configure a sample indicator for the completion of measurement on the highest numbered sensor 2, whose measurement will complete last.

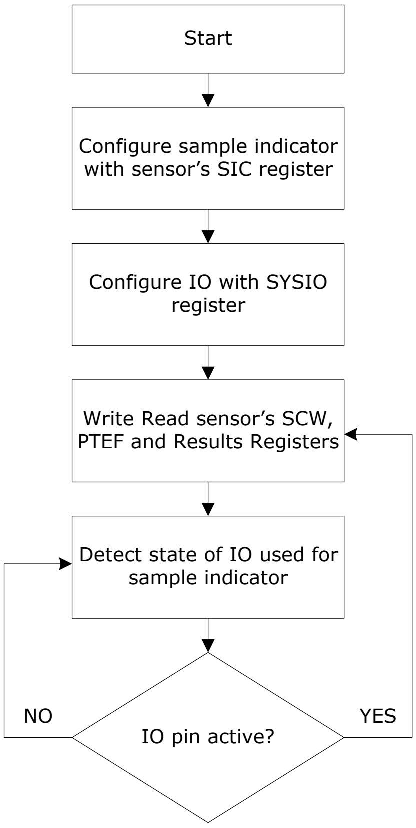

Start
Configure sample indicator with sensor's SIC register
Configure IO with SYSIO register
Write Read sensor's SCW, PTEF and Results Registers
Detect state of IO used for sample indicator
IO pin active?
NO
YES

Figure 44 with sample indicator

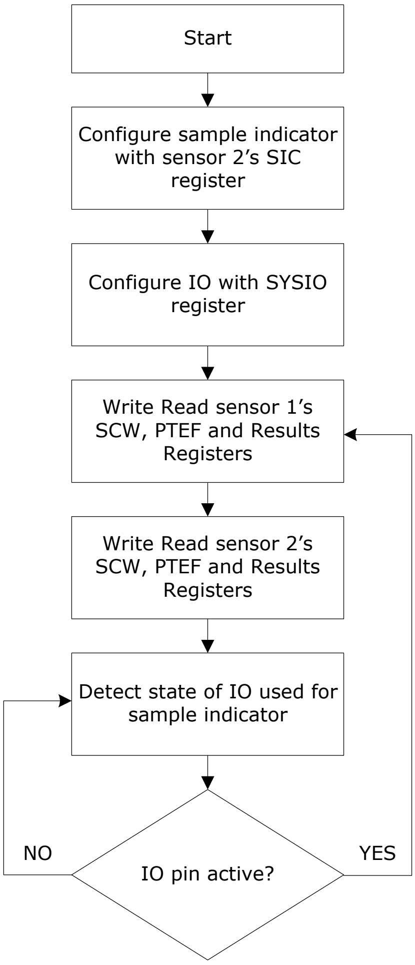

Start
Configure sample indicator with sensor 2's SIC register
Configure IO with SYSIO register
Write Read sensor 1's SCW, PTEF and Results Registers
Write Read sensor 2's SCW, PTEF and Results Registers
Detect state of IO used for sample indicator
IO pin active?
NO
YES

Figure 45 measuring sensor 1 and 2

12.8 Sample Time and Sample Rate

Sample Time is defined for single shot measurement using sample indicators, as in section 12.7. It is defined as the interval between the host requesting a measurement and the results being available. It is measured between the rising edge of the nSS signal used to write GO to the sensor's SCW register to the activation of a sample indicator mapped to that sensor.

Sample Rate is the frequency at which independent position measurements can be made with the CAM204 CTU. The maximum Sample Rate when communicating results over the SPI interface is given by Equation 6. The minimum *SPI Communication Time* is determined by the SPI timings of section 10.5 and the number of results registers that are read in the same transaction. This number is specific to the sensor Type and the information that the host requires.

$$Sample\ Rate\ (SPI\ Communication) = \frac{1}{Sample\ Time + SPI\ Communication\ Time}$$

Equation 6

12.9 Continuous Measurement

The CTU can perform measurements continuously. The host selects a time interval between sensor measurements, which it should write to the INTERVAL bits in the SYSI register (section 11.2).

Once INTERVAL is set, the host should write a 1 to the sensor's CONT bit in its SCW register, and a 1 to the GO bit to start measurements.

When a host system is to read the results of continuous measurement over the CTU's SPI interface, it is recommended that sample indicators are used to signal to the host when the results of a measurement are available. That way, the host can collect data before the next measurement begins, so that the SPI operation does not cause a measurement restart. Configuration is similar to above in section 12.7.

It is also possible to read a sensor's measurement results at any time over the SPI interface. However, if a measurement is in progress, the CTU may need to re-start it and subsequent measurements will be delayed.

When more than one sensor is configured to measure continuously, sensors will be measured in numeric order. When a set of sensors is performing continuous measurement, it is possible for the CTU to request single shot measurement on other sensors. Measurements will be performed in numeric sensor order.

Continuous measurement is stopped sensor by sensor by writing a 0 to each one's CONT bit. Writing a 0 to its GO bit will also prevent a single shot measurement from following.

Applications of continuous operation include driving an LED (section 12.10), position triggers (section 11.17) and driving an external DAC (section 13). Used in conjunction with configurable defaults (section 12.16), the CTU can take measurements autonomously without a host connected to the SPI interface.

12.10 LED Control

Some applications may benefit from a visual indication of sensor status. Figure 46 shows how an LED may be controlled by one of the CTU's user IOs using a MOSFET and current limiting resistor.

The IO can be configured for signaling status with LED(s) by configuring the LEDCW appropriately (section 11.12). Each sensor can be mapped to any user IO with the LEDMAP[3:0] bits.

The IO will flash the LED at approximately 2Hz when the sensor's last measurement was VALID and Amplitude was less than LEDTHRESHOLD[11:0]. This is an indication that the target is in range, but Amplitude has not yet reached a desired minimum value to yield the performance required.

The IO will light the LED continuously when the sensor's last measured Amplitude was greater or equal to LEDTHRESHOLD[11:0] to indicate a healthy signal level.

LED control can operate independently of host SPI transactions when the CTU is operating in continuous mode (section 12.9).

IOs used to drive LEDs are configured for active high or active low using the SYSIO register (section 11.3).

In the example shown in Figure 46, the IO should be configured for active high. For changes to the SYSIO register to apply to an IO controlled LED, they must be made with LEDMAP[3:0] set to 0.

LED control is typically used in conjunction with Position Triggers (section 12.11) or with an external DAC (section 13). Both of these functions also require user IOs, limiting the number of remaining user IOs available for driving LEDs. For example, the CAM204 CTU chip has 4 user IOs. IO2, IO3 and IO4 are required for an external DAC connected using SPI, leaving IO1 available for LED control.

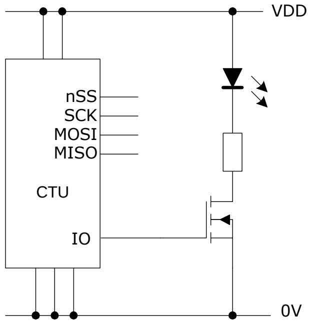


Figure 46 driving an LED

12.11 Position Triggers

The CTU can generate position trigger signals on its digital IOs when a sensor's measured position has passed preset thresholds. If the host configures the CTU for continuous operation, there is no need for further SPI activity. Position triggers may be configured in different ways:

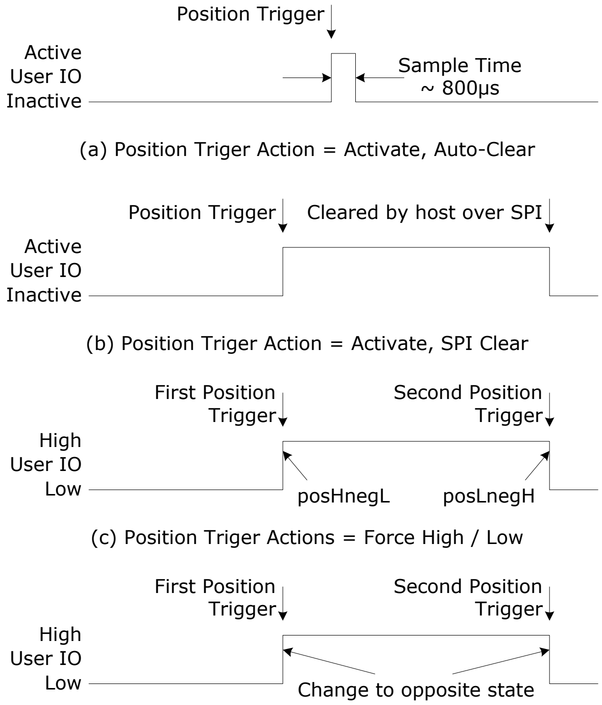
In Figure 47(a), the IO is configured to pulse active when a position trigger occurs. This could be used to trigger digital circuitry independent of the SPI host.

In Figure 47(b), the IO is configured to activate when a position trigger occurs, and to remain active until cleared by the CTU. This configuration can be used to alert an SPI host processor to a sensor's position passing a preset threshold.

In Figure 47(c), the IO is controlled by two position triggers at different positions. When the sensor's position measurement moves to between the two positions the IO is forced high, and when it moves outside the two positions it is forced low. This behaviour approximates the function of an opto-interrupter or proximity switch. Note, however, that the state of the IO is only confirmed once the sensor's position *crosses* a position trigger.

In Figure 47(d), the IO is again controlled by two position triggers at different positions. This time the IO toggles each time a position trigger is triggered.

Table 36 summarises the register settings required for each type of behaviour. It also shows how many of the 8 position triggers available are used for each configuration. Remaining position triggers can be configured to generate additional edges on the same or a different IO.



(a) Position Triger Action = Activate, Auto-Clear

(b) Position Triger Action = Activate, SPI Clear

(c) Position Triger Actions = Force High / Low

(d) Position Triger Action = Toggle

Figure 47 Position Trigger configurations

Table 36 Configuring Position Triggers

Configuration	**Pulse**	**Activate, SPI clear**	**Force high/low**	**Toggle**
Example	Figure 47(a)	Figure 47 (b)	Figure 47 ©	Figure 47 (d)
System IO Config INTnAH	Set to active high or active low		Set inactive IO state to desired state before a crossing has occurred	
Pos Trigs used	1	1	2	2
Pos Trig Map PTmMAP	Set to the desired IO			
Pos Trig Position(s) PTP	Set to where position trigger required		Set to where edges required	
Pos Trig Direction PTmDIR	Set to positive, negative or both		Set to both	Set to positive, negative or both
Pos Trig Auto-Clear PTAUTOCLR	Auto-Clear On	Auto-Clear Off	No effect	No effect
Pos Trig Action PTmACT	Set to Activate		Set one to posLnegH and other to posHnegL	
Pos Trig Hysteresis PTH	Usually set to minimum value that prevents multiple triggers, see section 12.12			
Pos Trig Flags PTmF	Don't care	Clear: 0	Don't clear: 1	
Pos Trig Enables PTmE	Set to 1 for each position trigger used, otherwise 0			
SYSI:INTERVAL	0 (fastest) or larger value (lower power)			
SCW	Set CONT and GO in the sensor's SCW register to start continuous measurements, set TRIG to enable position triggers			

12.12 Hysteresis

Each sensor has a HYSTERESIS[14:0] setting in its PTAH register (section 11.19). This is used to prevent multiple triggers due to CTU and/or mechanical noise. Its action is illustrated in Figure 48. A typical value for 10 bit resolution is 0x0040 (2^16 / 2^10 = 64 decimal).

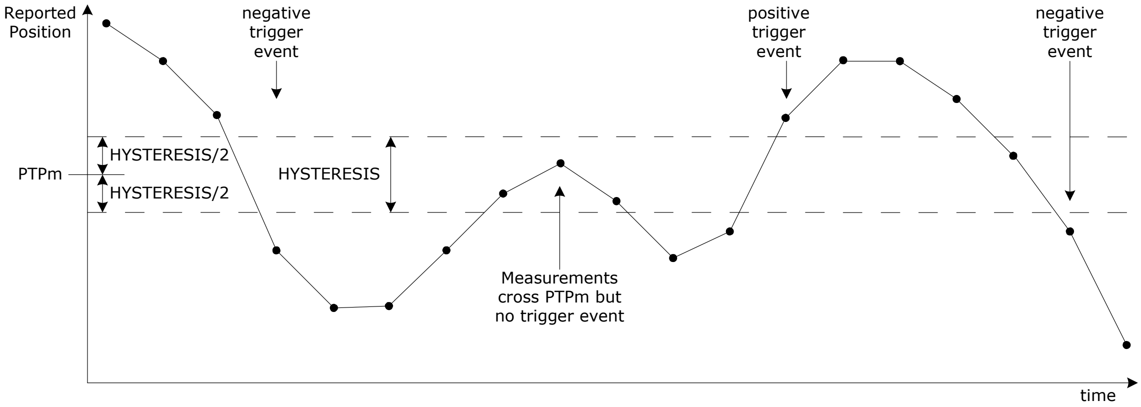


Figure 48 position trigger events and hysteresis

12.13 Position Scaling for Position Triggers

The Position Trigger Position (PTP) registers represent position as signed, 16-bit numbers. For Type 1 and 4 sensors this matches the format of CtuReportedPositionI16 as reported in the results registers (section 11.11). For other sensor types the scaling is defined in Table 37. HYSTERESIS is an unsigned quantity, and is scaled the same as the Position Trigger Positions.

Table 37 position measurement scaling for Position Trigger operation

Sensor Type	CtuReportedPositionI16	1 LSB represents...
1 and 4	As reported in results registers	SinLength/65536
2	= CtuReportedPositionI32 / 16	Sin Length / 4096
3	= CtuReportedPositionI32 / 4	Sin Length / 16384

12.14 Position Trigger Behaviour on Invalid

Position triggers are activated when a sensor's measured position *passes* the preset threshold. This condition requires a minimum of two valid measurements, one on each side of the threshold. This means that the first time a position trigger can be activated following one or more invalid samples is the *second* valid one.

12.15 Position Trigger Behaviour across Angle Discontinuity

Position triggers are designed to operate with both rotary and linear sensors. Rotary sensors have special requirements, since there is an apparent *angle discontinuity* between CtuReportedPosition = +32767 and -32768. In reality these values represent rotary positions a single LSB apart as illustrated in Figure 49. The CTU's triggers are designed to treat these values as adjacent. The transition from point A to E of Figure 49 is treated as the positive direction. If point E were set as a position trigger, then movement from A to D via E would create a positive trigger event, even though the numeric change in Reported Position is negative.

The CTU determines direction by assuming the actual position between two measurements takes the *shortest* route between them. This can cause erroneous triggering if the sample interval is too long. For example, if the actual position is sampled at A in Figure 49, and moves through B and C before being measured a second time at D, the CTU will interpret this as a positive change in position, even though actual position change was always negative. For correct operation, the sample interval should be set so that each sensor is sampled at least twice for a movement equivalent to a change in CtuReportedPosition of half full scale: 32768. This consideration also holds for a linear sensor, since the CTU performs the same underlying detection algorithms.

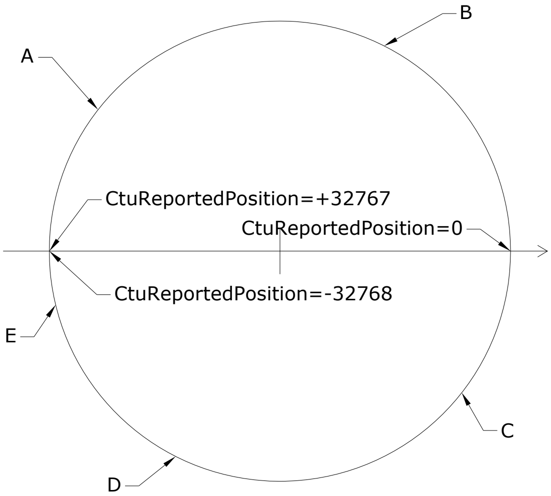


Figure 49 trigger behaviour

12.16 Configurable Defaults

The default values of the CAM204 CTU chip's register contents can be configured over the SPI interface (*Configurable Defaults*). This allows the CTU to operate autonomously, for example driving a DAC, LED and/or position triggers.

To change to new Configurable Defaults, the host should first write the required settings to the CTU's registers. Settings will normally include continuous measurement on one or more sensor (section 12.9) and either position triggers (section 12.11) or driving an external DAC (section 13). One or more status LEDs may also be controlled (section 12.10).

The host should then write 0x0C1C to the SAVEKEY register (section 11.5) followed by a 1 to the SAVE bit (section 11.1). The CTU will then reset itself. After this reset, and each subsequent one, the CTU's operation is determined by the saved Configurable Defaults.

The CTU's Factory Defaults can be restored by writing 0x0C1C to the SAVEKEY register (section 11.5) followed by a 1 to the FACTORY bit (section 11.1). The CTU will reset itself following this operation.

The Configurable Defaults are stored in FLASH memory. The number of FLASH updates are limited (section 3.5), so updates to Configurable Defaults should be rare. The SAVE and FACTORY operations each count as one FLASH update. For most applications the update will be done only once when a product containing the CAM204 is manufactured.

Configurable Defaults revert to factory settings after an update of CTU Application Firmware (section 14). If it is important that custom settings survive an update, they should be read out of the CTU before the update process and written back and saved afterwards.

13 Analog Output from an External DAC

13.1 DAC Overview

The CAM204 CTU chip has an SPI interface which can be used to read the results of sensor measurements. This interface yields the best performance, due to its digital nature, and is inherently low-cost. However in some applications, such as those lacking a digital processor or requiring support for a legacy interface, analog output may be required. To support these requirements, the CTU is able to drive an external Digital to Analog Converter (DAC).

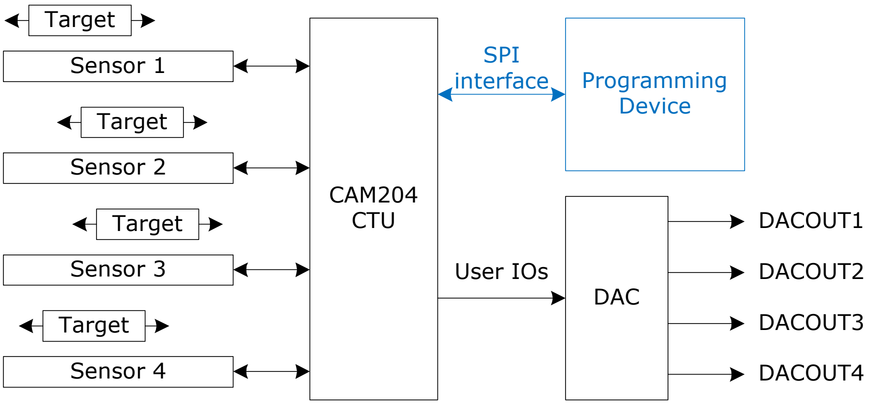


Figure 50 CAM204 CTU chip driving and external DAC

The CAM204 CTU chip drives external DACs from its user IO pins, as illustrated in the block diagram of Figure 50. The CAM204 supports up to 4 Type 1 sensors, and can drive DACs with up to 4 outputs.

When driving an external DAC, the CAM204 is usually operated stand-alone, without an external host device connected. This requires the chip's configuration to be programmed into non-volatile memory over the SPI interface (section 12.16). Programming is usually performed when the product containing the CAM204 chip is manufactured. Programming may be performed by a device that implements the SPI interface and register protocols described in this datasheet. Alternatively, a CTU Adapter from CambridgeIC may be used to configure the chip from a PC running CambridgeIC CTU Software.

13.2 Circuit Connections between CTU and DAC

The CAM204 CTU drives an external DAC with an SPI interface. The CTU is the SPI master device. When configured for DAC operation, IO2 is the serial clock, IO3 is the serial slave select/ synchronise and IO4 is the serial data. Figure 51 shows how these signals are connected to a DAC having typical SPI interface names.

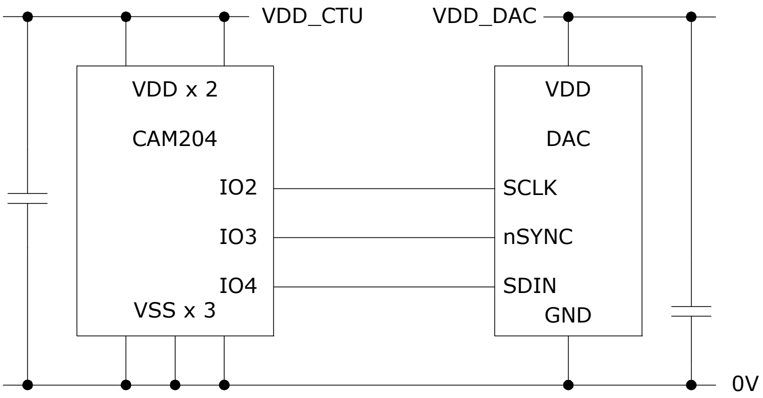


Figure 51 connecting the CTU to an external DAC

Figure 51 illustrates the CTU and DAC with different power supplies VDD_CTU and VDD_DAC respectively. This is allowable where the two supplies are within the operating ranges of each device, and where the CTU's IO output voltage level (section 3.4) is compatible with the DAC's logic inputs.

13.3 Supported DACs and Formats

The CAM204 CTU supports DACs having SPI interfaces with the message formats listed in Table 38. The DACFORMAT[2:0] bits within the SYSDAC register (section 11.4) select which format the CTU uses. Bit number 15 is the first one clocked into the DAC, and bit number 0 the last. The SPI clock rate specification is in Table 43.

When DACFORMAT[2:0]=0, each SPI message includes the two least significant bits of the DACADDR[2:0] bits from a sensor's DACCW register (section 11.15). These enable each sensor to address a different DAC output when the CTU is connected to an appropriate DAC. The function of the DACOP[1:0] bits is specific to each DAC family. DACOP[1:0] settings usually include power down, and provision for synchronously updating DACs in multi-output systems.

DACFORMAT[2:0]=1 is similar, except it excludes the DACADDR bits and is therefore only used for DACs with a single output.

In both cases, DACDATA is the data controlling the DAC output voltage. Its value depends on sensor position measurements when they are VALID (section 13.4) and on NVMODE[1:0] and DACNV[15:8] when Not VALID (section 13.5). The CTU calculates DACDATA to 16-bit precision internally. The number of bits sent over the SPI interface depends on DACFORMAT[2:0].

DAC families often include parts with different resolutions, typically from 8 to 12 bits as illustrated in Table 39. The same CTU configuration is valid for each part, since DACs with lower resolutions ignore the state of the appropriate number of least significant bits.

Table 38 SPI message format

DACFORMAT [2:0]	**SPI Data Bits in SPI message from CTU to DAC**															
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	DACADDR[1:0]		DACOP[1:0]		DACDATA[15:4]											
1	DACOP[1:0]		DACDATA[15:2]													

Table 39 Examples of supported DACs

Application	**Outputs per DAC**	**DAC Resolution**			**DAC FORMAT [2:0]**	**DACOP[2:0] normal operation**	**SPI Mode**
		8 bit	**10 bit**	**12 bit**			
Low-cost	1	DAC081S101	DAC101S101	DAC121S101	0	1	0
	2	DAC082S085	DAC102S085	DAC122S085	0	1	0
	4	DAC084S085	DAC104S085	DAC124S085	0	1	0
Precision	1			AD5621B	1	0	0

13.4 DAC Transfer Function

When a sensor measurement completes and an external DAC is on and enabled for that sensor, the CTU calculates the required SPI message and sends it to the DAC. The message includes formatting information (section 13.3) plus the upper bits of DACDATA[15:0]. DACDATA[15:0] is the code representing DAC output, with 0x0000 (decimal 0) typically representing the minimum output voltage and 0xFFFF (decimal 65535) representing the maximum .

When the sensor measurement is valid, the CTU calculates DACDATA[15:0] from CtuReportedPositionI16. This is a signed, 16-bit integer that represents measured position. Where sensor measurements are reported over the SPI interface to 16-bit precision, it equals the reported value (section 11.11). Where measurements are reported as a 32-bit integer, a scaling is applied according to Table 40.

Table 40 position measurement scaling for DAC operation

Sensor Type	**CtuReportedPositionI16**	**1 LSB represents…**
1 and 4	As reported in results registers	SinLength/65536
2	= CtuReportedPositionI32 / 16	Sin Length / 4096
3	= CtuReportedPositionI32 / 4	Sin Length / 16384

The CTU performs two operations on CtuReportedPositionI16 to yield DACDATA[15:0]. First, it applies a Hysteresis Function. This may be used to stabilise the DAC output in systems requiring a stable output code when stationary. The amount of hysteresis is set using the HYSTERESIS[14:0] bits in the PTAH register (section 11.19). Hysteresis compromises dynamic performance, and it is recommended that it remain at the default value of 0 for most applications.

Next, the CTU applies the *DAC Transfer Function* to the position data. This is the relationship between measured position (after hysteresis) and DACCODE[15:0]. It can be configured using the parameters listed in Table 41.

Table 41 parameters defining the DAC Transfer Function

Parameter name	**Name in Figure 52**	**Summary of parameter's function**	**Register description**
DACPOSA	A	Position at start of up slope	Section 11.13
DACPOSB	B	Position at end of up slope	
DACPOSC	C	Position at start of down slope	
DACPOSD	D	Position at end of down slope	
DACLIMX	X	Minimum DAC value when valid	Section 11.14
DACLIMY	Y	Maximum DAC value when valid, must be greater than DACX	

Figure 52 illustrates the range of DAC Transfer Functions available. Figure 52(a) is a key to the other graphs. The x-axis is CtuReportedPositionI16 after hysteresis is applied, running from -32768 to +32767. The y-axis is the resulting DAC value.

Figure 52 (b) to Figure 52(e) illustrate the range of possible Transfer Functions applicable to linear sensors. The shape of the Transfer Function is configured using the programmable parameters listed in Table 41. DACPOSA-F define the start and end positions of the sloping portions of the Transfer Function. DACX and DACY define minimum and maximum DAC values when the sensor's measurement is valid.

Figure 52(b) is the *general case* from which all other Transfer Functions are derived. The DAC output increases linearly between DACX and DACY for positions between DACPOSA and DACPOSB. It remains at DACY between DACPOSB and DACPOSC. It decreases linearly from DACY to DACX for positions between DACPOSC and DACPOSD. It remains at DACX from DACPOSD upwards (back to DACPOSA in the case of a rotary sensor).

Figure 52(c) illustrates a common application where the DAC output increases linearly between DACPOSA and DACPOSB, typically at the ends of the sensor's measuring length. When position is valid but less than DACPOSA the DAC output remains at DACX, and when it is valid but greater than DACPOSB the DAC output remains at DACY. DACPOSD should be set to -32768 and DACPOSC to +32767.

Figure 52(d) is similar to (c), except that the DAC output slopes in the opposite direction. This can only be achieved as shown, with DACPOSA-D repositioned, and not by selecting a value of DACY smaller than DACX.

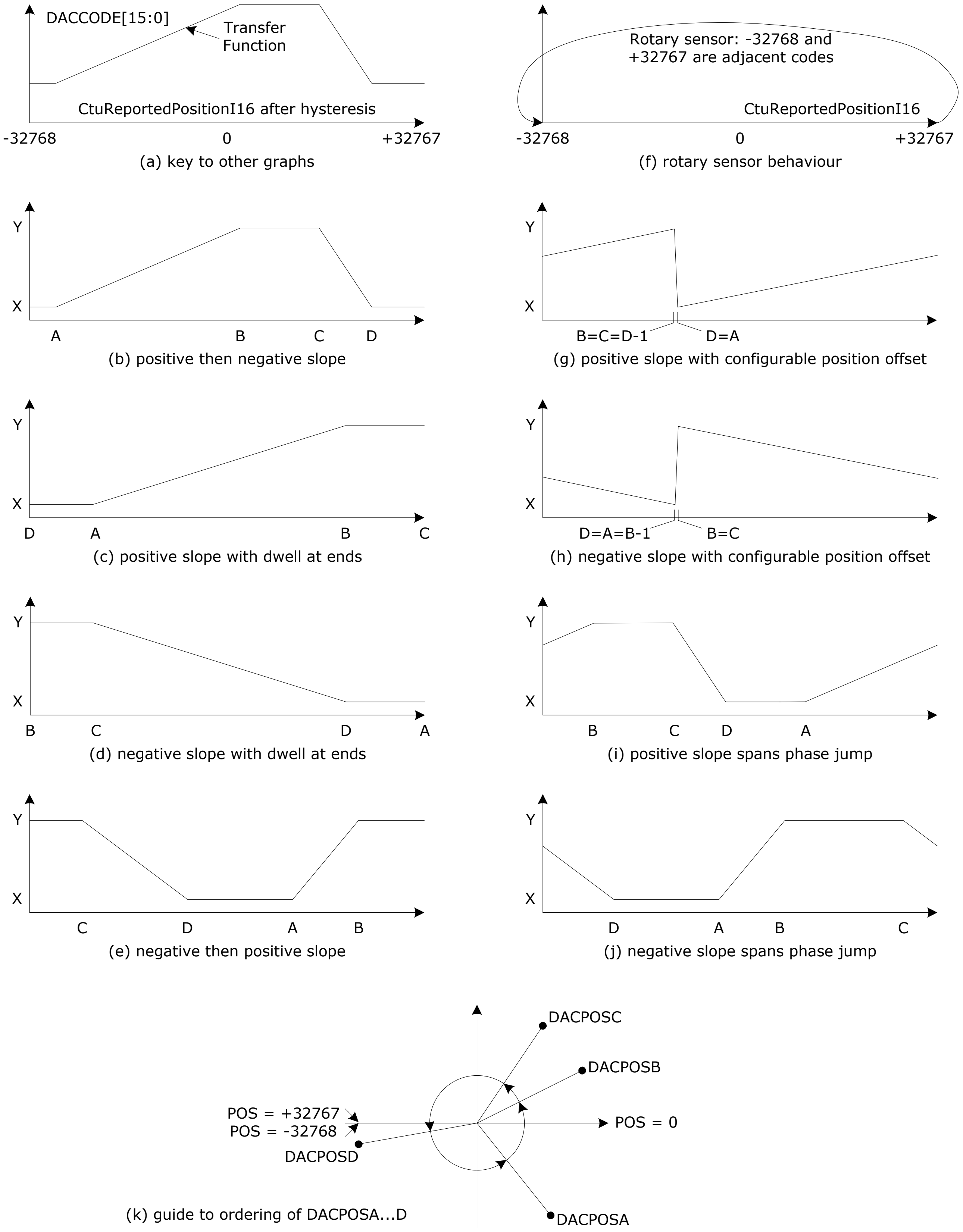
The Transfer Function of Figure 52(e) arises when DACPOSA-D are re-ordered from (b).

Rotary sensors behave in the same way, except that *modulo arithmetic* applies. Positions -32768 and +32767 are adjacent. The transfer functions can be *rotated* to start at any angle offset, as illustrated in Figure 52(f).

Figure 52(g) illustrates how to achieve a DAC output which increases linearly with rotation angle. The start angle of the slope is defined by DACPOSA = DACPOSD. The end angle is defined by DACPOSB = DACPOSC, which should be set to 1 less than DACPOSA and DACPOSD. (h) is similar, except with sloping in the opposite direction.
Figure 52 (i) and Figure 52(j) are more general cases applicable to rotary sensors, and are rotated versions of (b).

DACPOSA…D must be correctly ordered for the CTU to generate meaningful data for the DAC. The general rule is illustrated in Figure 52(k). Position is treated as an angle, so that the values +32767 and -32768 represent adjacent positions. DACPOSA must be anticlockwise from or equal to DACPOSD. DACPOSB must be anticlockwise from DACPOSA. DACPOSC must be anticlockwise from or equal to DACPOSB. Finally DACPOSD must be anticlockwise from DACPOSC.

DACLIMY must be greater than or equal to DACLIMX. If a "flipped" transfer function is required, the values of DACPOSA and DACPOSC must be swapped, and those of DACPOSB and DACPOSD.



(a) key to other graphs

(f) rotary sensor behaviour

(b) positive then negative slope

(g) positive slope with configurable position offset

(c) positive slope with dwell at ends

(h) negative slope with configurable position offset

(d) negative slope with dwell at ends

(i) positive slope spans phase jump

(e) negative then positive slope

(j) negative slope spans phase jump

(k) guide to ordering of DACPOSA...D

Figure 52 configuring the DAC Transfer Function

13.5 DAC Output When Measurements are Not Valid

When the CTU's DAC output is on and a sensor's DAC output is enabled, the CTU sends DACDATA[15:0] to the DAC to control its output after each measurement. When sensor measurements are VALID, the value of DACDATA is defined by the Transfer Function, described in the previous section. When Not VALID, the behaviour depends on the NVMODE bits in the SYSDAC register (section 11.4). There are 3 possible modes, defined in Table 42.

Table 42 DAC output on invalid measurements

NVMODE	Mode Name	DACDATA when last sensor measurement was invalid
0	Default Value Mode	Forced to DACNV
1	Last Value Mode	Holds value from last measurement. Defaults to DACNV if there was no valid measurement since the last reset.
2	End Latch Mode	If last valid measurement was in region DA, outputs DACX. If last measurement was in region BC, outputs DACY. Otherwise outputs DACNV.
3	Reserved	

In Default Value Mode, the DAC is driven to the DACNV value stored in the sensor's DACCW register (section 11.15). DACNV[15:8] is an 8-bit wide field representing the upper 8 bits of DACCODE[15:0]. The CTU sets the lower 8 bits to 0. The value of DACNV may be selected such that the DAC output falls significantly outside the range DACLIMX…Y. That way, a device connected to the analog output alone can distinguish between a valid and invalid measurement.

In Last Value Mode, the DAC is driven with the value of DACDATA from the last VALID measurement when available. This may suit some systems that can experience a loss of target signal, e.g. due to momentarily increase gap, and where the DAC output should still reflect an estimate of position.

End Latch Mode is similar to Last Value Mode, except that the last position only "latches" when the last value fell in regions DA or BC. This behaviour is useful for linear sensors where the target is allowed to move beyond the ends of the sensor's measuring length. With the Transfer Function configured as Figure 52(c) or (d), the DAC outputs either DACLIMX or DACLIMY depending on the direction the target left the sensor. If the last measured position was in region AB or CD, the DAC is driven with DACCODE = DACNV to indicate that the target did not leave the sensor in one of the expected locations.

Note that the first few measurements from a sensor after reset may not be VALID due to the frequency search process that the CTU undertakes (section 9). Appropriate DACNV and NVMODE settings should therefore always be considered.

13.6 DAC Calibration

In some applications it may be possible to establish DAC configuration in advance, so that all products are programmed with the same CTU configuration.

Alternatively, DAC configuration may be established as part of the manufacturing and test process of the product. For example, there may be mechanical tolerances requiring the optimum values of DACPOSA…D to be established, and/or the need to calibrate out errors in the DAC output and its signal conditioning.

Figure 53 illustrates a typical CTU-based system undergoing calibration. A target positioning system positions the target at pre-selected positions, for example at the required ends of the measuring region. The CTU's reported position at these locations may be measured over the SPI interface and averaged by the customer's Test and Configuration System, ready for programming back into the CTU's DACPOSA…D registers as appropriate.

To calibrate out errors in the DAC and any signal conditioning, the Test and Configuration System may need to set the value of DACCODE controlling the DAC output to a specific value. The DACCAL bit in the SYSDAC register (section 11.4) is available for this purpose. When set, the CTU sends an SPI message to the DAC with the DACCODE value equal to the required sensor's DACLIMX, DACLIMY or DACNV each time one of those registers is written. DAC configuration should otherwise be as detailed in section 13.8.

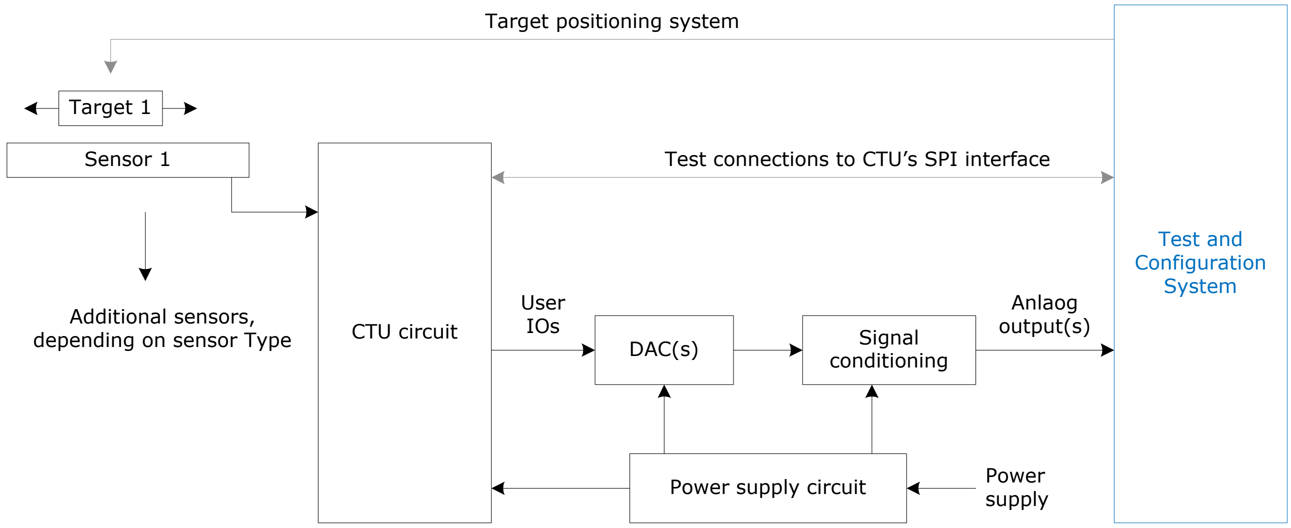

Target positioning system
Target 1
Sensor 1
Additional sensors, depending on sensor Type
CTU circuit
Test connections to CTU's SPI interface
User IOs
DAC(s)
Signal conditioning
Anlaog output(s)
Power supply circuit
Power supply
Test and Configuration System

Figure 53 block diagram of a CTU-based sensor system undergoing calibration

13.7 Synchronisation of DAC Outputs

When the CTU is connected to more than one sensor it measures each in sequence. If a multi-output DAC is connected, the CTU updates the DAC register associated with each sensor when each individual measurement completes. With the multi-output DACs of Table 39 and the value for DACOP[1:0] shown, each DAC output will be updated when its register is updated, with the updates staggered in time accordingly.

Some systems may require that changes in the output voltages of a multi-output DAC occur at the same time. The multi-output DACs shown in Table 39 allow synchronisation. DACOP[1:0] should be set to 1 for one "trigger" sensor and 0 for the remaining "follower" sensors. When the CTU updates DAC register values for the "follower" sensors the analog outputs will not change until the "trigger" sensor is measured and its DAC register is updated. That way, all DAC outputs will be updated synchronously.

13.8 Summary of Register Settings for DAC Operation

This sub-section summarises the register settings typically required for autonomous CTU operation with an external DAC.

Register(:Bit)	**Function Controlled**	**Section Ref**
SYSI	Interval between measurements. Set to 0 for fastest update rate, or greater values for lower power operation.	11.2, 12.9
SYSDAC:DACSON	Set to 1	11.4
SYSDAC:CPOL,CPHA	Set to match the selected DAC's SPI Mode	
SYSDAC:DACFORMAT	Set to match the selected DAC's SPI format	13.3
SYSDAC:NVMODE	Select the most appropriate behaviour on invalid measurements	13.5
SCW:GO	Set to 1 for each active sensor	12.9
SCW: CONT	Set to 1 for each active sensor to operate continuously	
SCW:TRIG	Set to 0, unless a spare IO is used as a position trigger output	12.11
SCW:INCE	Set to 0 to operate Type 2 sensors in Absolute Mode	5.2
SCW:SIE	Set to 0, unless a spare IO (IO1 in CAM204) is used as a sample indicator, e.g. to indicate VALID or to activate after a DAC update	11.17
SIC		
PTEF	Set to 0 unless a spare IO (IO1 in CAM204) is used as a Position Trigger	12.11
LEDCW: LEDMAP	Set to IO1=1 if a spare IO (IO1 in CAM204) is used to drive an LED	12.10
LEDCW:LEDTHRESHOLD	Set to the desired Amplitude threshold if LED control is required	

Register(:Bit)	Function Controlled	Section Ref
DACPOSA…D	Defines the positions of the start and end of the up and down slopes	13.4
DACLIMX,Y	Set to the minimum and maximum DAC values required when VALID	
DACCW:DACEN	Set to 1 for each sensor that generates a DAC output	13.3, 13.7
DACCW:DACOP	Setting depends on the DAC SPI data format and timing of updates	
DACCW:DACADDR	Defines which output of a multi-output DAC each sensor updates	
DACCW:DACNV	Sets value of DAC output when a sensor's measurement is invalid	13.5
TYPE	Set to the type of sensor connected to the CTU	12.3
PTC	Ignore unless a Position Trigger output is required	12.11
PTAH:PTAUTOCLR	Set to 1 if a Position Trigger output is required	
PTAH:HYSTERESIS	Set to 0 unless hysteresis is required in the DAC output	13.4
SAVEKEY SYSCW:SAVE	Once all other registers are programmed, set SAVEKEY to 0x0C1C and write a 1 to the SAVE bit to set configuration as the non-volatile default	12.16

13.9 Layout and Filtering Considerations

The noise present at the output of the DAC is a combination of the CTU's own measurement noise, noise that is introduced into the analog output of the DAC by the DAC itself, and system interference. This interference can be mimimised by following the DAC manufacturer's circuit and layout recommendations. Coupling mechanisms include the power supply. Care should be taken to isolate the CTU and DAC power supplies sufficiently, since the CTU includes significant digital processing and its excitation circuit handles AC current.

It is possible to further reduce the noise with appropriate filtering, for example using an RC filter illustrated in Figure 54. Rfilt and Cfilt are the filter resistor and capacitor. Rload represents the circuit's load, for example an amplifier. The filter capacitor's 0V connection is shown close to the load to minimise noise coupled along the 0V connection.

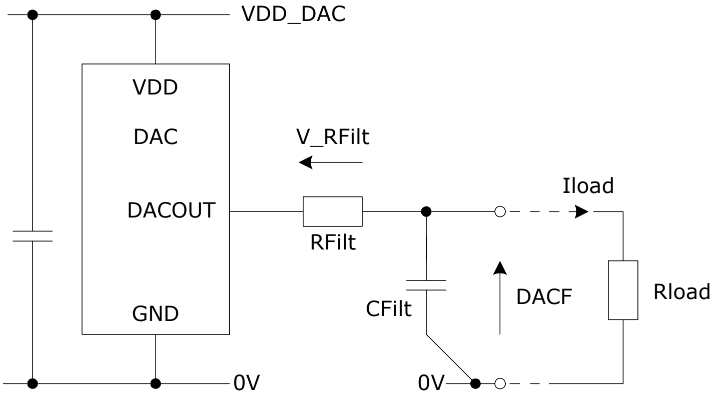


Figure 54 simple filter for DAC output

The filter bandwidth can be chosen to pass the signal of interest representing position. With a maximum update rate on the order of 1000 samples per second, a filter bandwidth of 500Hz will typically make little difference to the dynamic performance. The other consideration is loss of accuracy due to load current Iload flowing through filter resistor Rfilt. If the load resistance is 1Mohm or greater, Rfilt can be 1kohm to yield 1LSB error in 10 bits, and Cfilt 330nF to yield 500Hz filter bandwidth.

Active filters and signal conditioning circuits may be required for more demanding applications, for example when the product's output is distributed over an interface cable and is subject to EMI.

13.10 DAC Related Specifications

Table 43 lists specifications that relate to the CTU’s DAC driving function.

Table 43 DAC related specifications

Item	Min	Typ	Max	Units
SPI clock frequency		2.2	3	Mbit/s

The rate at which each DAC output is updated depends on the type of sensor, the number of sensors connected and the SYSI setting (section 11.2). Table 44 lists sample rates with SYSI set to 0, and Position Triggers disabled, to yield the highest possible rate.

Table 44 DAC update rate (SYSI=0)

Sensor Type	Number of active sensors	Min update rate	Units
1 or 4	1	1200	Samples/s
1 or 4	2	600	
1	3	400	
1	4	300	
2	1	680	
3	1	125	

14 Bootloader Operation

14.1 Overview

The CTU chip's internal software is partitioned into two fields: Application Code and Bootloader Code. The Application Code is responsible for normal CTU operation including measurements, with communication through the register interface described in section 11. The Bootloader Code can be used to update the Application Code. In normal operation, the version number of the Application Code (the System Version Number) can be read over the SPI interface from the SYSVER register (section 11.7). The version number of the Bootloader Code can be read from the BOOTVER register (section 11.6).

The procedure for uploading new Application Code over the CTU chip's SPI interface is specified in this section. These details will be required if the host processor is required to perform the update process. The upload procedure may alternatively be performed by a PC communicating with the CTU chip over the SPI interface using CambridgeIC's CTU Adapter.

14.2 Applications

Updating Application Code can be useful for adapting CTU chips to work with newly supported sensors, sensing circuits and peripherals. Instead of buying a new stock of chips, a customer can program existing chips with the new code.

14.3 CTU Firmware File Format

New CTU Application Code is provided in the CTU Firmware File format (.cff). This comprises a Header Block, a Data Block and a Checksum Word. The Header Block is plain text, while the Data Block and Checksum Word are binary.

The Header Block can be viewed on a PC as a text file, for example using *WordPad* or dedicated hex and binary code editing software such as *Hex Editor Neo*. The text includes version number, build date and a description of the CRC algorithm that can be used to checksum the Data Block.

The Header Block comprises rows that start with the "#" character and end in <CR><LF> (0x0D, 0x0A). The data block can be identified by searching for the first character after <CR><LF> that is not "#". This is the first character of the data block.

The Data Block comprises the remaining characters in the .cff file, except for the last two, which are the Checksum Word for the Data Block. The Checksum Word may be used to verify that the Data Block is valid and uncorrupted.

14.4 Choice of Bootloader Method

There are two Bootloader methods. The simplest, recommended, approach requires the host to control the CTU's nRESET line in real time, and is described in section 14.5. An alternative is provided when the host does not have control over the nRESET line, described in 14.6.

14.5 Bootloader Operation With nRESET

This method is illustrated in Figure 56. To start, the host toggles nRESET to reset the CTU. It then begins the Data Block Transfer Process described in section 14.7. For correct operation, a minimum time period is required between the reset and first SPI transaction, TnRST2nSS, illustrated in Figure 55 and specified in Table 45. The second word of the Data Block must have been sent within a time TVALIDCHECK(Min) of the reset, otherwise the CTU may resume normal operation instead of updating Application Code. TVALIDCHECK is specified in Table 22.

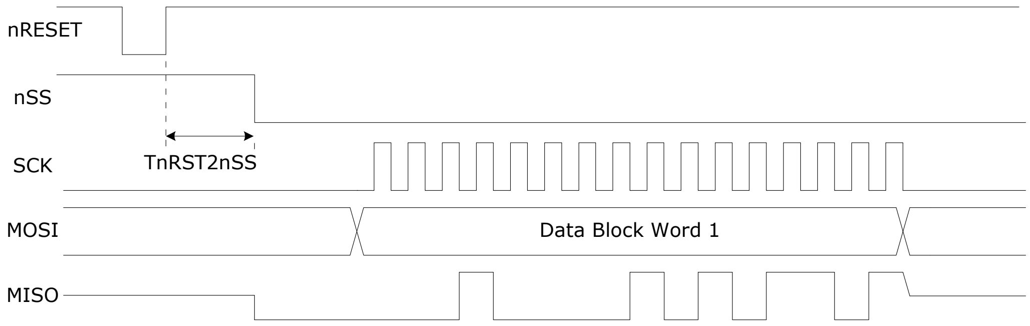


Figure 55 timing of first Data Block Word SPI transaction after reset

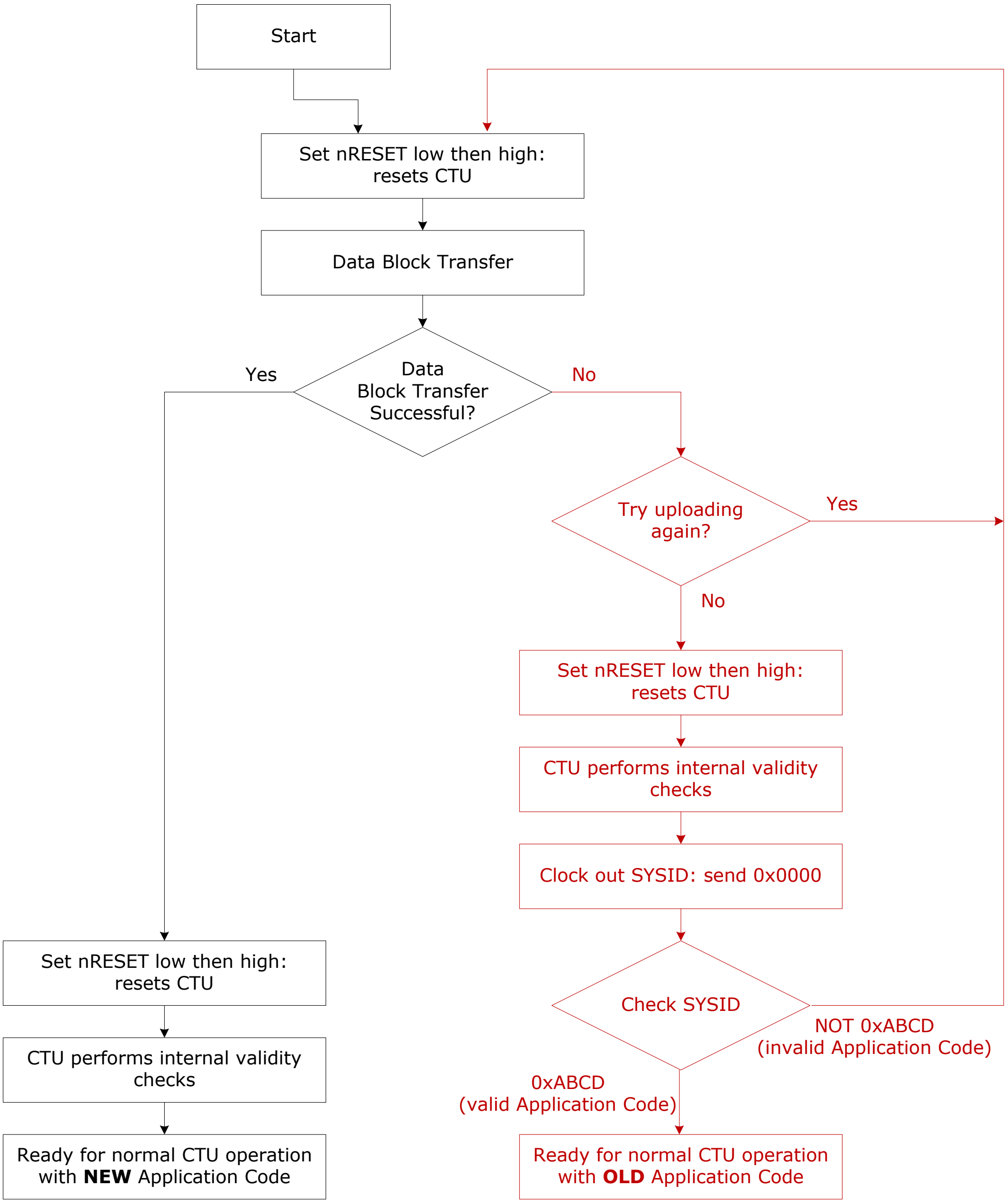

Start
Set nRESET low then high: resets CTU
Data Block Transfer
Data Block Transfer Successful?
Yes
No
Try uploading again?
Yes
No
Set nRESET low then high: resets CTU
CTU performs internal validity checks
Clock out SYSID: send 0x0000
Check SYSID
NOT 0xABCD (invalid Application Code)
0xABCD (valid Application Code)
Ready for normal CTU operation with OLD Application Code
Set nRESET low then high: resets CTU
CTU performs internal validity checks
Ready for normal CTU operation with NEW Application Code

Figure 56 Bootloader Method, with nRESET

If the Data Block Transfer is successful, the host should reset the CTU, wait for validity checks to complete (section 12.2) and resume normal operation. It is recommended that the host checks that the System Version Number reported in the SYSVER register (section 11.7) matches the expected value for the *new* Application Code.

If the Data Block is not successful, the host can repeat the process. If the upload failed early in the process, it is possible that old Application Code will remain valid. The host can attempt to run *old* Application Code, by following the process illustrated in red shown in Figure 56. Valid Application Code is indicated by SYSID equal to 0xABCD (the factory default) or by an alternative valid value programmed as a Configurable Default.

14.6 Bootloader Operation Without nRESET

This method is illustrated in Figure 58. Before Data Block Transfer can begin, the CTU must be put into its Bootloader Mode. There are two methods, depending on whether or not the CTU contains valid Application Code already.

If the CTU contains valid Application Code, the host can set the BOOTLOAD bit in the SYSCW register while the CTU is operating normally (section 11.1). The CTU will reset itself, and when it comes out of reset it will remain in Bootloader Mode. The host can then perform the Data Block Transfer process described in section 14.7 to load new Application Code. The first Data Block Word should be timed as in Figure 57, so that it does not violate the minimum time TnRST2nSS specified in Table 45.

If the Data Block Transfer is successful, the host should then send the Boot Command: 0xB002, to the CTU. The CTU will reset itself and begin validity checks on the new Application Code. The host should wait for these checks to complete before resuming normal operation (see section 12.2). It is recommended that the host checks that the System Version Number reported in the SYSVER register (section 11.7) matches the expected value for the *new* Application Code.

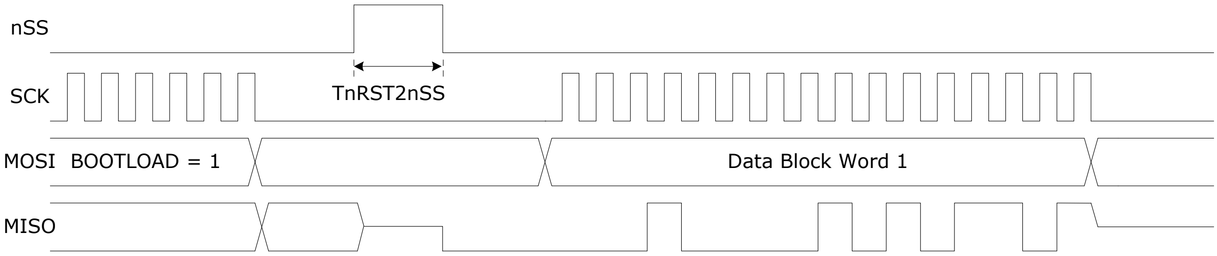


Figure 57 timing of first Data Block Word SPI transaction after BOOTLOAD bit set

If the Data Block Transfer process fails, the host should send the Boot Command and wait for the CTU's internal validity checks to complete, as illustrated in Figure 58. If the CTU reports the default SYSID (factory default 0xABCD or as set up as a Configurable Default) the *old* Application Code remains valid. In this case, the host can attempt the upload process again using the BOOTLOAD bit to enter Bootloader Mode as above.

If the Data Block Transfer process fails and the SYSID test does not result in the expected value, then the CTU's *old* Application Code has become corrupted. The BOOTLOAD bit can not be used to enter Bootloader Mode since the register map is defined in Application Code. Without host control over nRESET, the only way remaining to enter Bootloader Mode is to power cycle the CTU. The host should then begin Data Block Transfer, subject to the minimum timing TnRST2nSS specified in Table 45. There is no upper limit on the time between power cycling and the start of Data Block Transfer. However, the host must not send any data to the CTU over its SPI interface before Data Block Transfer, otherwise it will be interpreted as the Data Block and a subsequent upload will fail.

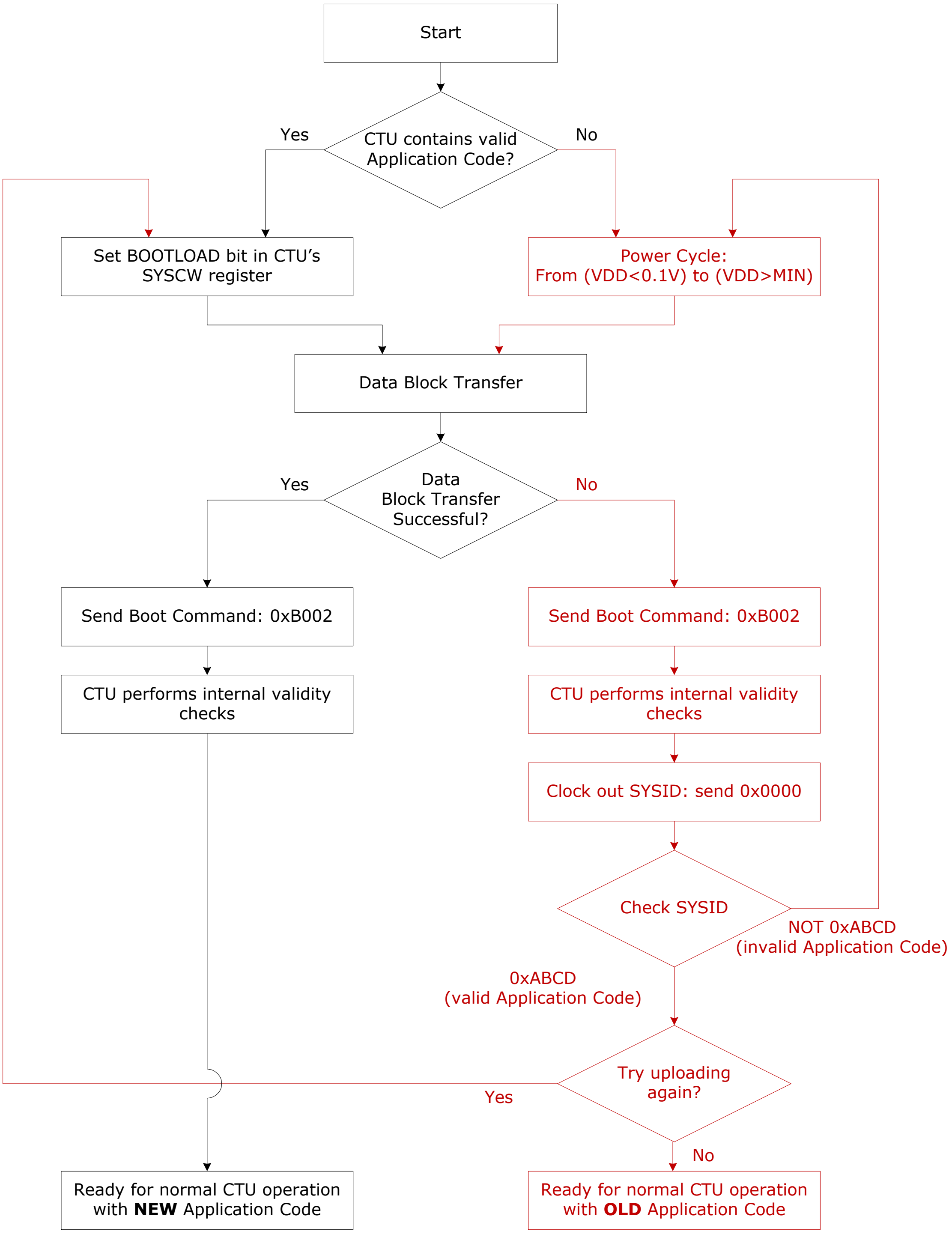

Start
CTU contains valid Application Code?
Yes
No
Set BOOTLOAD bit in CTU's SYSCW register
Power Cycle: From (VDD<0.1V) to (VDD>MIN)
Data Block Transfer
Data Block Transfer Successful?
Yes
No
Send Boot Command: 0xB002
CTU performs internal validity checks
Send Boot Command: 0xB002
CTU performs internal validity checks
Clock out SYSID: send 0x0000
Check SYSID
NOT 0xABCD (invalid Application Code)
0xABCD (valid Application Code)
Try uploading again?
Yes
No
Ready for normal CTU operation with NEW Application Code
Ready for normal CTU operation with OLD Application Code

Figure 58 Bootloader Method, without nRESET

14.7 Data Block Transfer Process

The Data Block Transfer process is for sending new Application Code to the CTU, and is illustrated in Figure 59. It forms part of both Bootloader Methods described in section 14.5 and section 14.6.

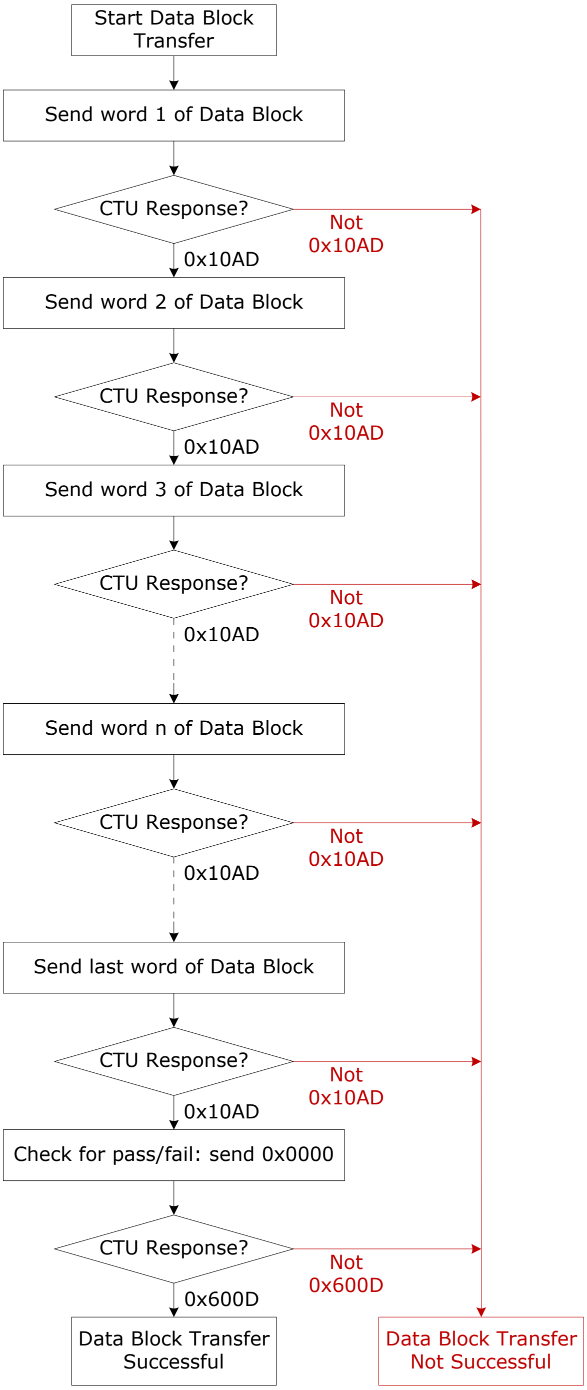

Start Data Block Transfer
Send word 1 of Data Block
CTU Response?
0x10AD
Not 0x10AD
Send word 2 of Data Block
CTU Response?
0x10AD
Not 0x10AD
Send word 3 of Data Block
CTU Response?
0x10AD
Not 0x10AD
Send word n of Data Block
CTU Response?
0x10AD
Not 0x10AD
Send last word of Data Block
CTU Response?
0x10AD
Not 0x10AD
Check for pass/fail: send 0x0000
CTU Response?
0x600D
Not 0x600D
Data Block Transfer Successful
Data Block Transfer Not Successful

Figure 59 Data Block Transfer Process

Data is transferred in 16-bit words over the SPI interface, as described in section 14.8. Each time the host sends a word, the CTU should respond with 0x10AD. If not, Data Block Transfer has failed. When the last word of the Data Block has been sent and the CTU has signaled it is ready as described in section 14.8, the host can check whether the Data Block Transfer was successful. It does this by sending 0x0000, and checking the response from the CTU is 0x600D. If not (typically responding with 0x0BAD) the Data Block Transfer has failed.

14.8 SPI Communication with the CTU in Bootloader Mode

Communication with the CTU in Bootloader Mode uses the same SPI timings and bit ordering as described in section 10.

There is one difference: the MISO output from the CTU is used for handshaking. After each Data Block word, the CTU sets MISO to high impedance (it will normally be pulled high by a pull-up resistor). When the CTU is ready for the next Data Block word, it pulls MISO low. The host should detect the state of MISO and only send the next Data Block word when it has been pulled low by the CTU.

This behaviour is illustrated in Figure 60. The time between the last clock edge of a Data Block word and the CTU signaling it is ready for the next Data Block word is denoted TBOOTWAIT. The maximum value of TBOOTWAIT is specified in Table 45. This maximum is only required on a small number of Data Block words when the CTU is performing internal tests. In most cases it is much smaller.

The Data Block Transfer process may be undertaken as a single SPI transaction with nSS low throughout. Alternatively, Data Block words or groups of words may be sent as individual SPI transactions with nSS high in between. In all cases the host should check MISO as described above before sending the next Data Block word.

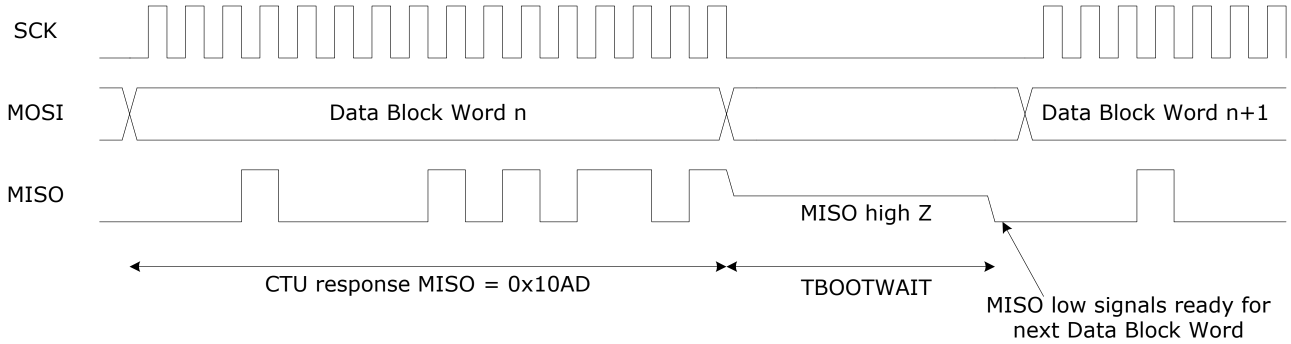

SCK
MOSI
MISO
Data Block Word n
Data Block Word n+1
MISO high Z
CTU response MISO = 0x10AD
TBOOTWAIT
MISO low signals ready for next Data Block Word

Figure 60 SPI communication, sending Data Block

14.9 Bootloader Timing Specifications

The specifications for the Bootloader timings referred to above are in Table 45.

Table 45 Bootloader timings

Parameter	Description	Min	Typ	Max	Units
TBOOTLOAD	Overall time to update Application Code	-	2.6	-	s
TBOOTWAIT	Variable pause required between Data Block Words	-	0.009	300	ms
TnRST2nSS	Pause required before start of Data Block Transfer	20	-	-	µs

15 Package Details

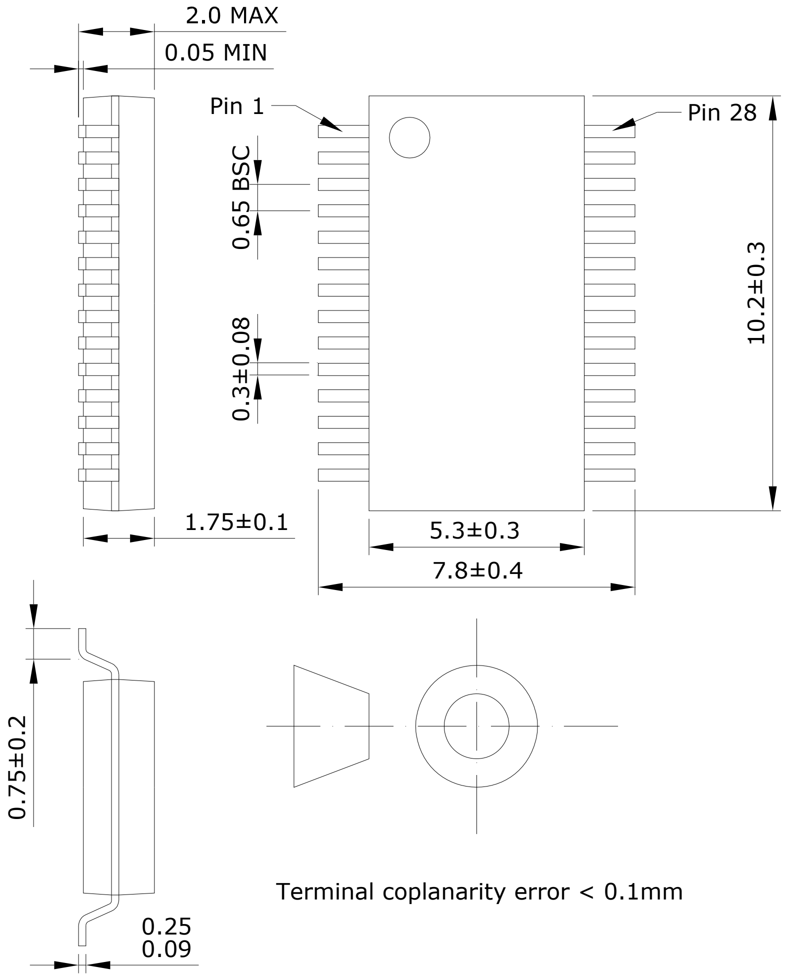


Figure 61 28-pin Plastic Shrink Small Outline 5.30mm Body (SSOP)

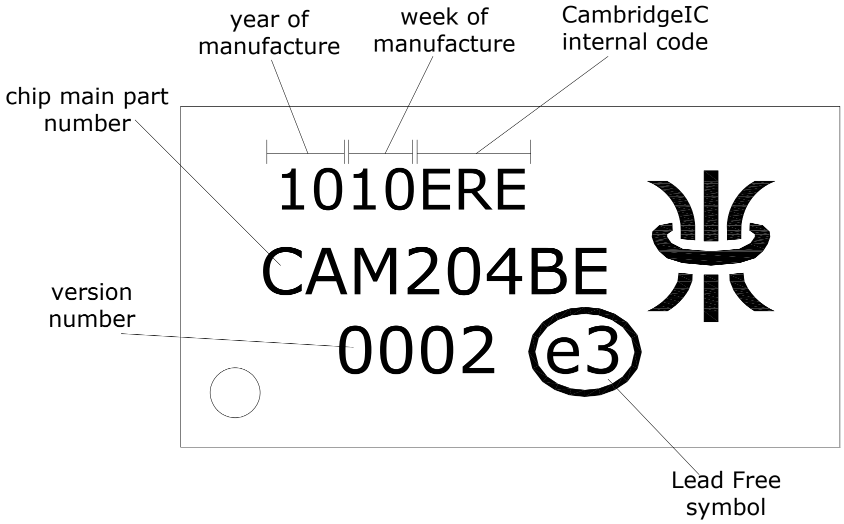


Figure 62 product markings

16 Tape and Reel Specifications

The CAM204 is available on complete reels of 2100 parts. The carrier tape is illustrated in Figure 63, and dimensions are specified in Table 46.

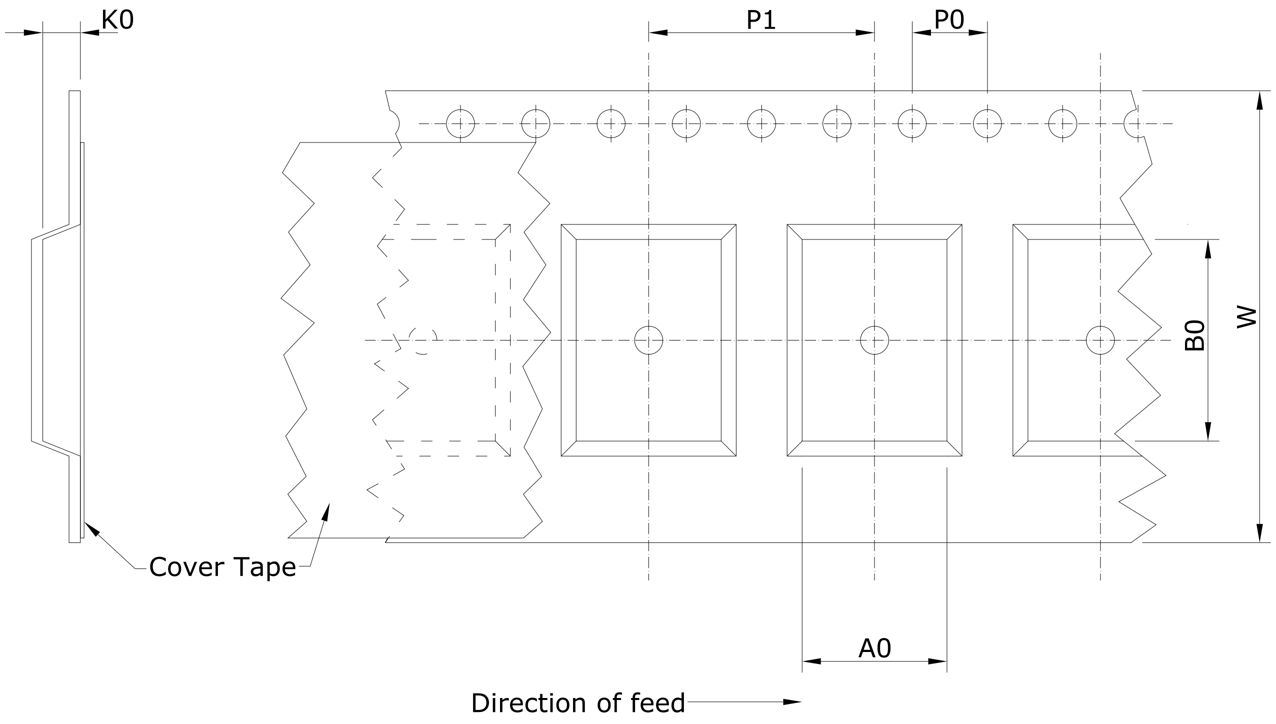


Figure 63 carrier tape dimensions

Table 46

Tape and Reel Specifications		**Dimensions in mm**						
Package	**Units per reel**	**Reel diameter**	**W**	**P0**	**P1**	**A0**	**B0**	**K0**
28-pin SSOP	2100	300	24	4	12	8.3	10.7	2.2

17 Reflow Soldering Recommendations

The CAM204 is available in lead free packaging only. The recommended reflow soldering temperature profile is illustrated in Figure 64. Values are shown in Table 47.

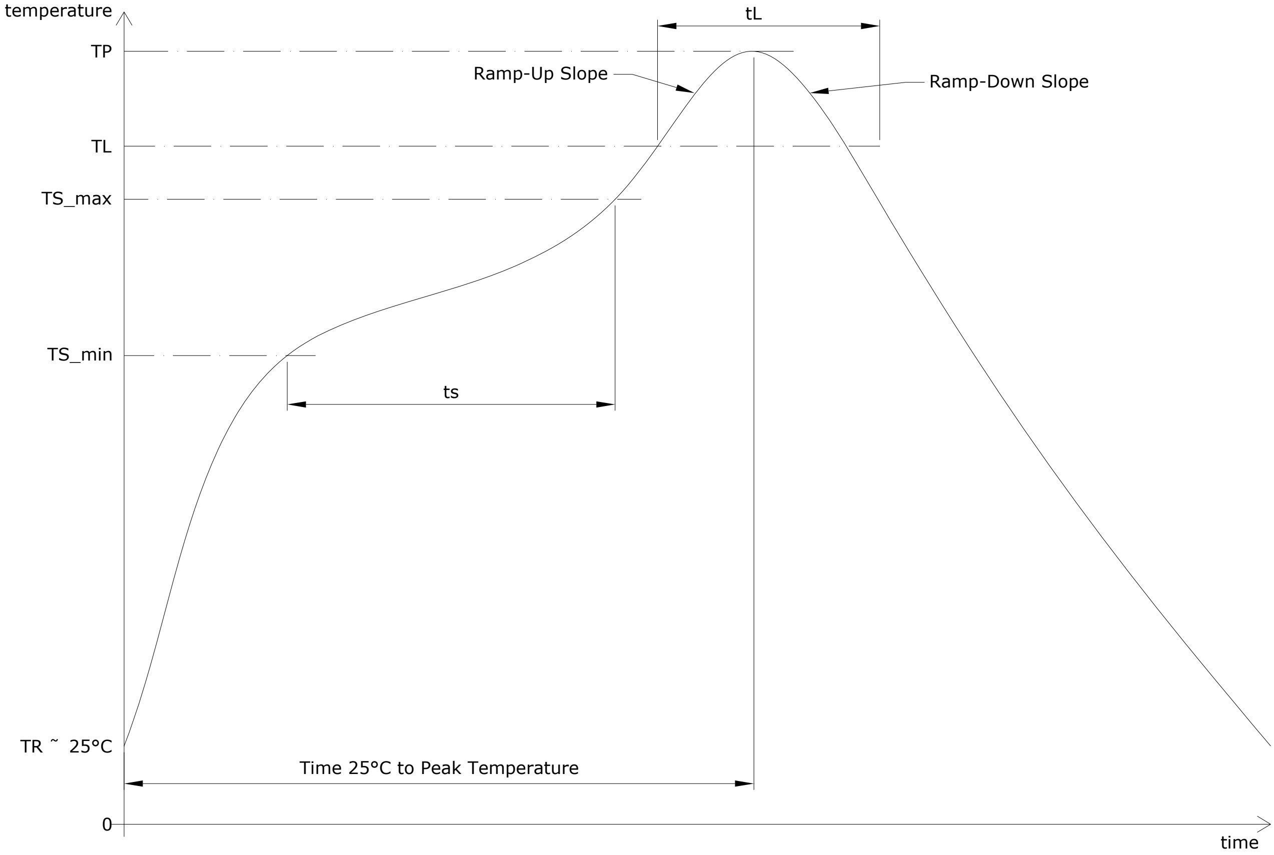


Figure 64 reflow soldering temperature profile definitions

Table 47

Profile feature	**Value**		**Comments**
TS_min	150°C		Preheat temperature range
TS_max	200°C		
ts	60s min	120s max	Preheat time
TL	217°C		Liquidous temperature
TP	225°C min	260°C max	Peak temperature
tL	60s min	150s max	Time maintained above Liquidous temperature
Ramp-Up Slope	3°C/s max		
Ramp-Down Slope	6°C/s max		

18 Environmental

Table 48

Item	Min	Max
Storage temperature	-65°C	150°C

19 RoHS Compliance

The CAM204 uses Matte Tim (Sn) pin finish. CambridgeIC certifies, to the best of its knowledge and understanding, that the CAM204 chip is in compliance with EU RoHS, China RoHS and Korea RoHS.

20 Document History

Table 49 main changes

Rev	Date	Comments
0006	1 February 2010	Added Bootloader details Specified TVALIDCHECK and behaviour following reset Corrected values of TSCKL2MISOV, TnSSL2MISOV
0007	14 April 2010	Updates for CAM204BE version of CAM204 Increased TSCKL2nSSH to 11µs Updated position triggers with new Position Trigger Action (SYSVER ≥ 0x0404)
0008	19 August 2010	Changed Type 1 C_B to 100nF Changes relating to SYSVER ≥ 0x0406: Added Type 4 sensor details Sensor 1 must be used to configure sensor Type; sensors 2…4 are read only Added details on driving an external DAC and/or LED Added details on saving configurable defaults to non-volatile memory Updated Position Triggers, allowing operation with all sensor Types SPI timings TnSSH2MISOZ, TnSSH2nSSL, TnSSH2IOch increased to accommodate new functions

21 Contact Information

Cambridge Integrated Circuits Ltd
21 Sedley Taylor Road
Cambridge
CB2 8PW
UK

Tel: +44 (0) 1223 413500

info@cambridgeic.com

22 Legal

This document is © 2009-2010 Cambridge Integrated Circuits Ltd (CambridgeIC). It may not be reproduced, in whole or part, either in written or electronic form, without the consent of CambridgeIC. This document is subject to change without notice. It, and the products described in it ("Products"), are supplied on an as-is basis, and no warranty as to their suitability for any particular purpose is either made or implied. CambridgeIC will not accept any claim for damages as a result of the failure of the Products. The Products are not intended for use in medical applications, or other applications where their failure might reasonably be expected to result in personal injury. The publication of this document does not imply any license to use patents or other intellectual property rights.