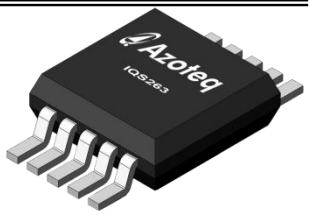
AZD088 IQS263 Communications Interface Guideline



1. Introduction

1.1 Abstract

The IQS263 is a 3-channel I²C compatible device with multiple configuration options via a memory map.

The purpose of this document is:

1. To understand the I²C compatible interface
2. To support the example code provided for evaluation and driver development

Please note that a full description of features can be found in the IQS263 datasheet.

1.2 Connection description

The diagram below describes the hardware requirement for communicating with the IQS263.

**Please note the importance of using the RDY line and not only the SDA and SCL connections.

See section 1.3 for more details.

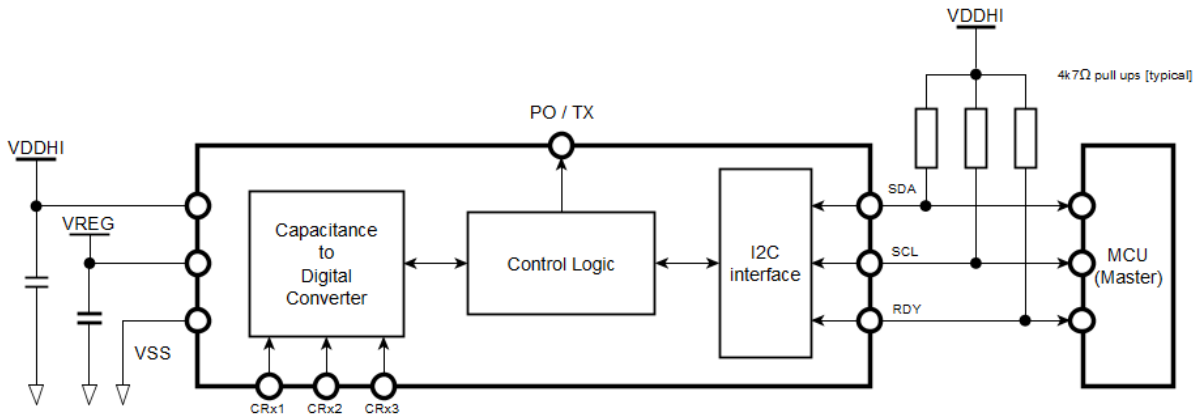
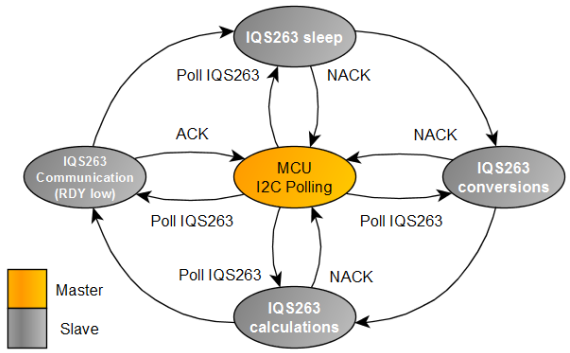


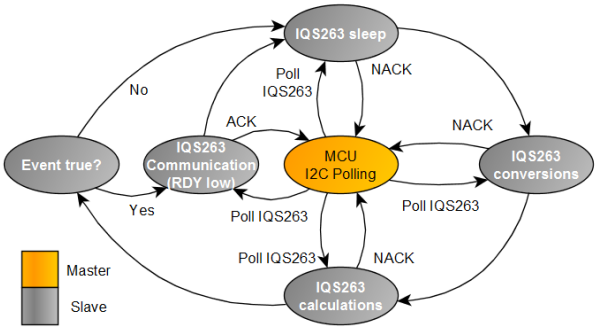
Figure 1.1 IQS263 hardware connection diagram

1.3 Communication window description



Streaming mode (polling at various times – example only)

Recommended: only poll device when RDY=LOW



Event mode (comms window only during event)

Figure 1.2 IQS263 Master-Slave interface options description

The IQS263 will only allow for communication within a certain time-slot. Outside of this “slot” the device will give preference to capacitive conversions, calculations or sleep. With the specific implementation on the IQS263, the master device must:

1. read the bidirectional “RDY” I/O from the IQS263
2. only poll the device during the RDY = low period
3. create a communication window (useful in event mode) by writing the “RDY” low and poll the slave until ACK is received

2. I2C communication protocol

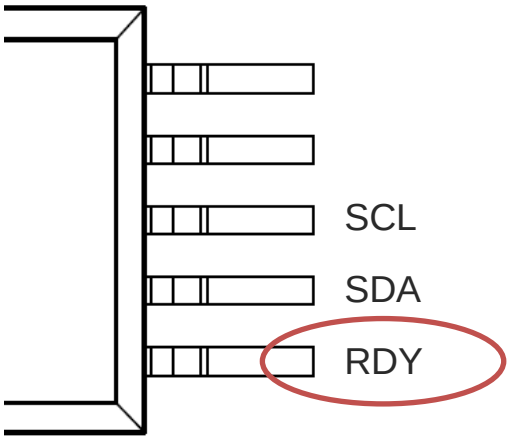
The IQS263 slave device interfaces to a master controller via a 3-wire (SDA, SCL and RDY) serial interface bus and is I²C™ compatible, with a maximum communication speed of 400kbit/s.

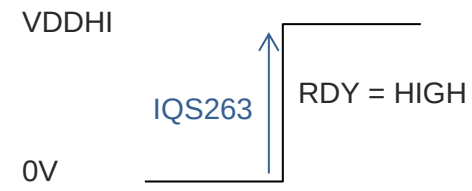
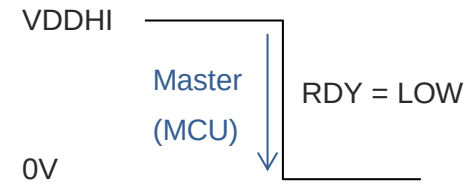
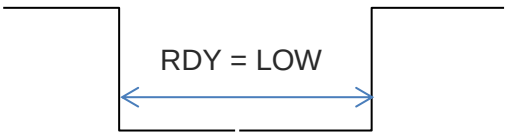
Compatibility with various I²C libraries must be ensured by following the guidelines in this document.

The IQS263 can be used in 3 different modes:

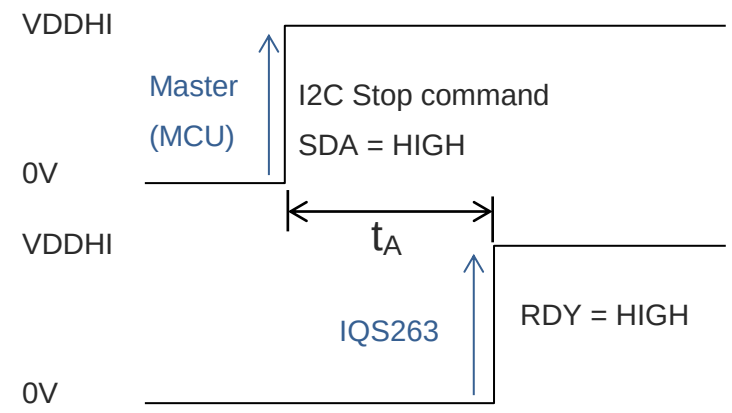
Streaming mode	recommended for initialization periods
	Switch to event mode after initialization
Event mode	recommended for keeping power consumption minimal
	recommended for minimizing bus communications and processor load
Force communication (polling)	recommended for event mode systems
	Implement for case where communications are required, but no events are present force open a communication window

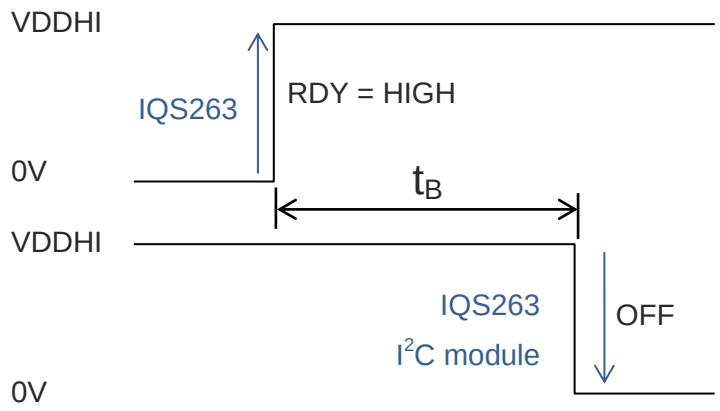
2.1 RDY line description

	RDY line (bi-directional) • Indicator for master device (MCU) • Indicator for when sensing is paused • MCU output to force communications
	• Communication window open • Capacitive sensing paused

	• Communication window closes • Capacitive sensing continues • Only sensing and calculations are done during RDY high when “TUBO MODE” is enabled • Sensing, calculations and sleep are done during RDY high when “TUBO MODE” is disabled
	• In EVENT MODE, the master can request a communication window • Capacitive sensing only pauses after communications are (ACK)nowledged via the SDA line • MCU keeps RDY low until ACK, then IQS263 takes over at ACK time and keeps RDY low
	• When RDY changes low and no I^2C start is sent, the duration of RDY low is as specified in reg 0x0A Byte 7 “I^2C timeout” • When RDY changes low and I^2C start is sent, the RDY duration will be until I^2C stop is sent. • When no I^2C stop is sent the RDY duration will be until the I^2C watchdog timer (WDT) expires. See t_{RDY} in table Table 4.1

2.2 RDY line – known issues and workarounds

	**t_A constraint:** • Do not send I^2C start command during t_A • Result: Bus changes clocked into IQS263 buffer will cause CLK = LOW until next communication window where a NACK will be sent to master • $t_{A\ min}$ = 70us (2 MHz main clock) • Do reseed command +70us • Redo ATI command +112us • Force Sleep command +25us • Therefore: $t_{A\ max}$ = 182us (2MHz) • $t_{A\ min}$ = 35us (4 MHz) • $t_{A\ max}$ = 91us (4 MHz)



t_B constraint:

- Do not send I²C start command during t_B
- Result: Bus changes clocked into IQS263 buffer will cause CLK = LOW until next communication window where a NACK will be sent to master
- t_B = 60us (2 MHz)
- t_B = 8us (4 MHz)

Workaround for issues relating to t_A:

- Ensure communications are effectively done after RDY changes to LOW
- Ensure communications are not initiated (first I²C start) only close to I²C timeout (Reg 0x0A:byte7 – default 5ms)

Workaround for issues relating to t_B:

- Do not re-initiate communications with next device or same device within the specified time.

Further details of fault condition when t_A or t_B are not adhered to:

- As mentioned above, the IQS263 will take control of the I²C bus CLK line by an extended clock stretching event.
- Any address polling at the next communication window will receive a NACK from the IQS263 I²C module and the condition will be reset.
- In the case of event mode with no expected events on the RDY line following, the watchdog timer (t_{WDT}, see Table 4.1) will reset the condition.

3. How to address the IQS263 via I²C

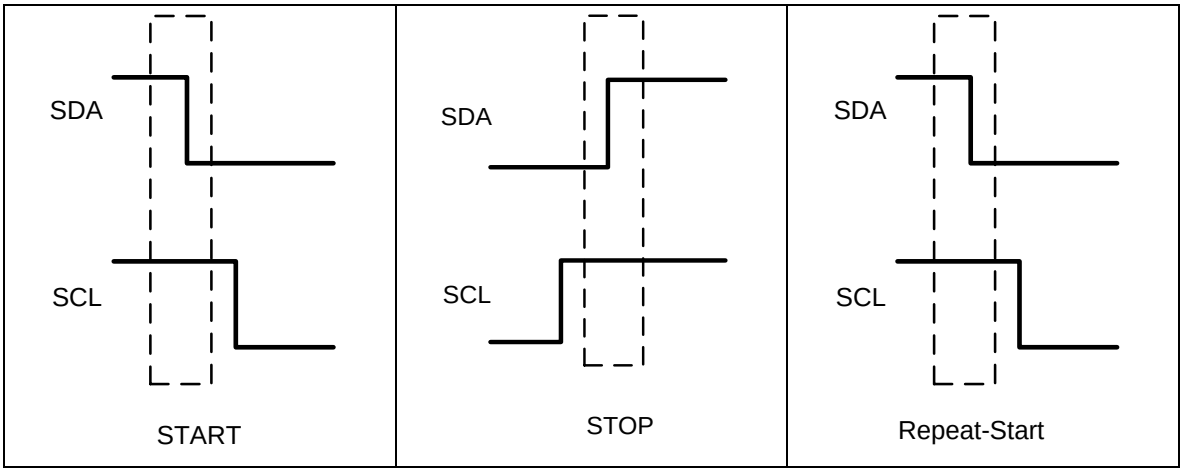


Figure 3.1 I2C Start, stop and repeat-start definitions

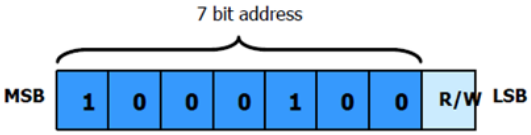


Figure 3.2 Control byte specification

Start	Control Byte		Address-Command		Data n		Data n+1		Stop
S	Adr + WRITE	ACK		ACK		ACK		ACK	S

Figure 3.3 Multiple data write sequence

Start	Control Byte		Address-command		Start	Control Byte		Data n		Stop
S	Adr + WRITE	ACK		ACK	S	Adr + READ	ACK		NACK	S

Figure 3.4 Register read sequence with “repeat-start” after address setup

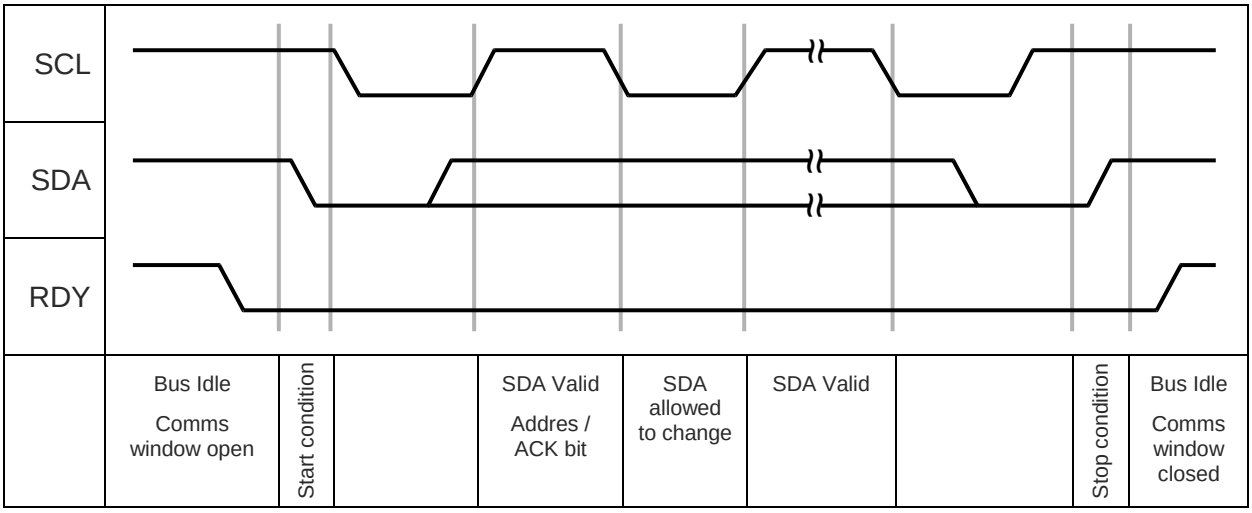


Figure 3.5 Bus state change specifications

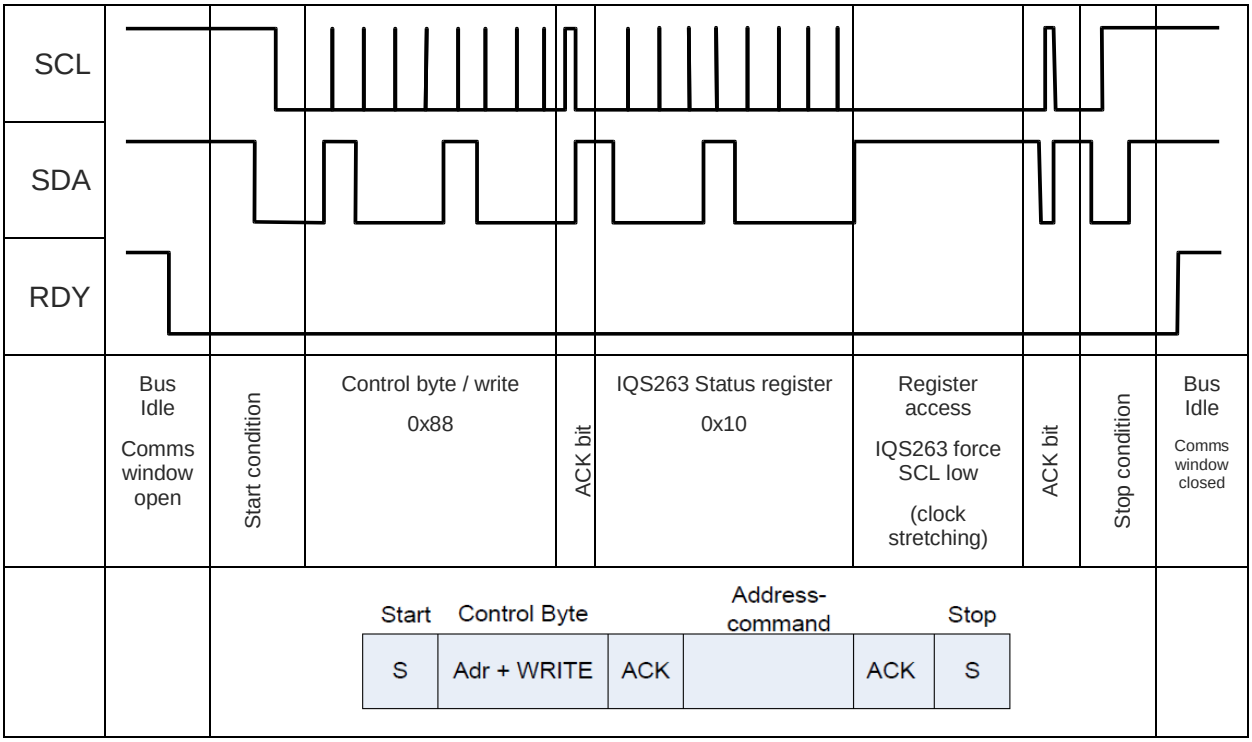
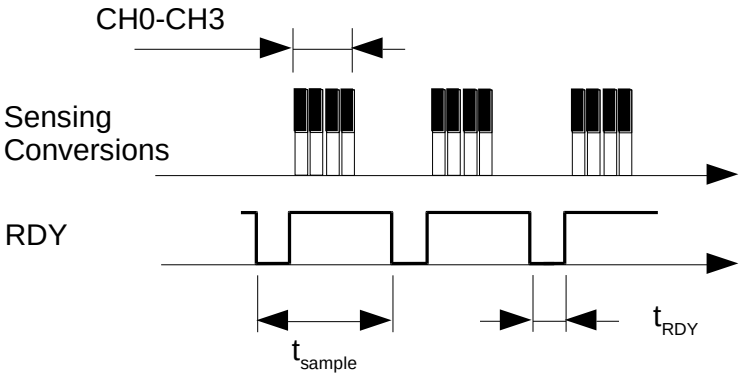


Figure 3.6 Dummy status register read setup

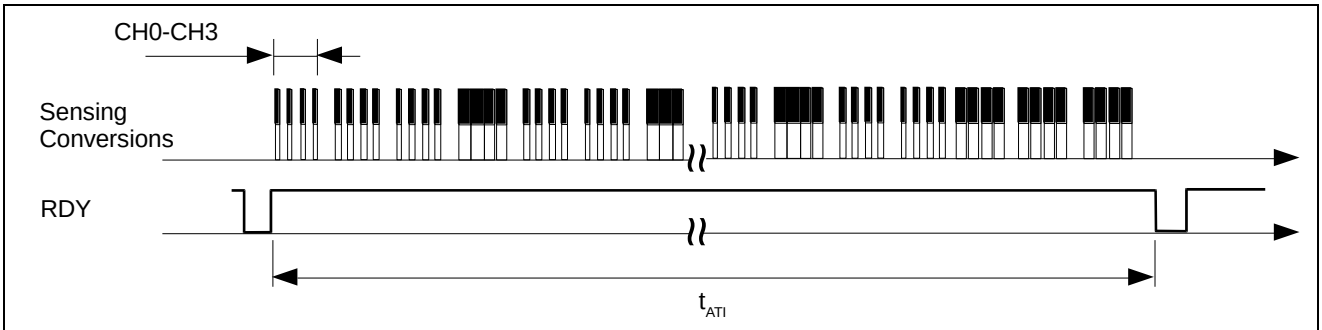
4. Streaming mode specification

4.1 Streaming and sampling times

Table 4.1 Sample and ready timing description

Streaming (RDY low after each conversion)				
Settings	2 MHz osc (40Hz sampling) (Reg 0x09:byte0:bit1 = 0)	4 MHz osc (80Hz sampling) (Reg 0x09:byte0:bit1 = 1)	Turbo Mode Enabled (Reg 0x09:byte3:bit2 = 0)	
t_{sample}	≈25ms	≈12ms	<25ms (2MHz)	<12ms (4MHz)
Example Case	Default (no comms) (Reg 0x0A:byte7 = 0x04)	Typical (with comms) (Read 3 registers @ 400kHz)	WDT (Start condition, no stop condition)	
t_{RDY}	≈5ms	TBD	260ms (2MHz)	130ms (4MHz)

4.2 ATI time (with streaming during ATI disabled)



	Typical	Max
t_{ATI}	400ms	850ms

See datasheet for full description of ATI time variations.

5. Event mode specification

- Event mode is recommended for most applications
- Default streaming mode is intended for initialization periods and applications that utilize reading raw sample data

5.1 Idle state

- In the idle state (RDY high) the device will not act on any communications on the SDA and SCL lines
- The device will do sampling and calculations while communication procedures are idle

5.2 Event flagged description

- Events are flagged in the IQS263 until read by the master device.
- Newer events will overwrite the older.
- The following is an example of event accumulation and overwriting:

Events occurred:	Event flags read:
• ATI complete • RDY toggle, • movement, • RDY toggle, • proximity, • RDY toggle, • touch, • RDY toggle, • no touch, RDY low, poll IQS263, read flags	• ATI complete, • movement, • proximity

- Figure 5.1 shows the behaviour of RDY when missing RDY signal indicators in event mode:

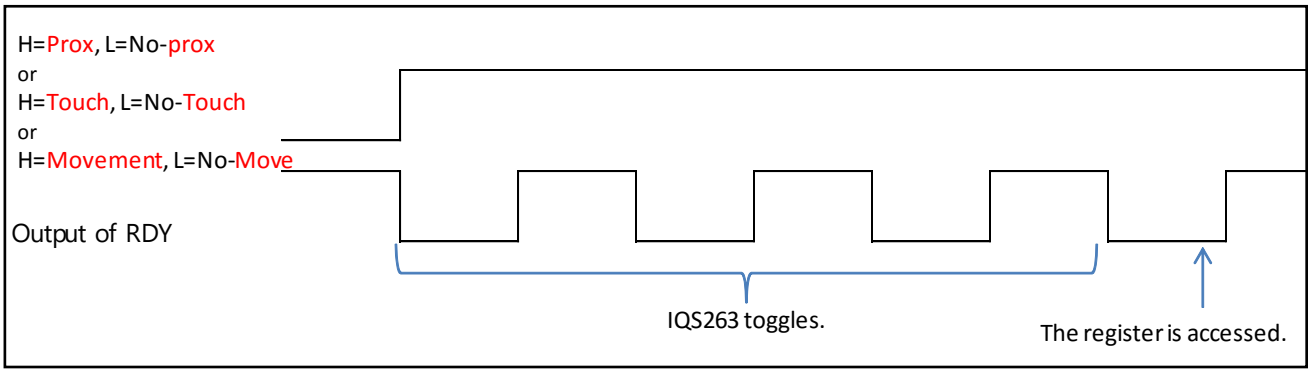


Figure 5.1 Proximity / Touch / Movement Event (When the host reads out registers at the fourth RDY)

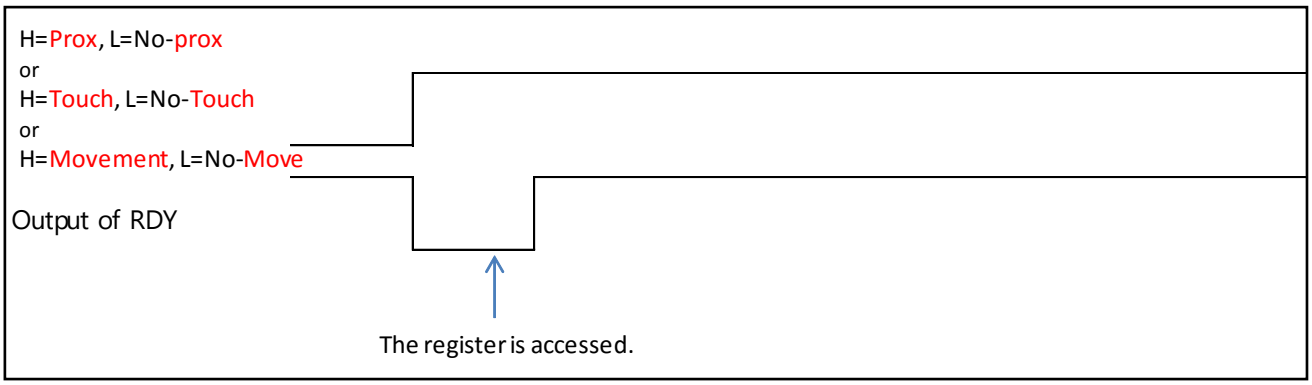


Figure 5.2 Proximity / Touch / Movement Event (When the host reads out registers at the first RDY)

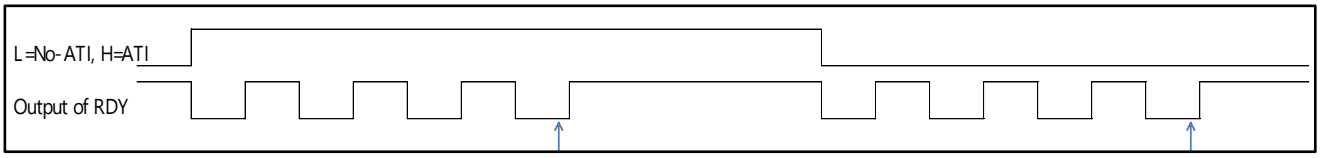


Figure 5.3 ATI Event (When the host reads out registers at the fourth RDY.)

5.3 Event mode force communications

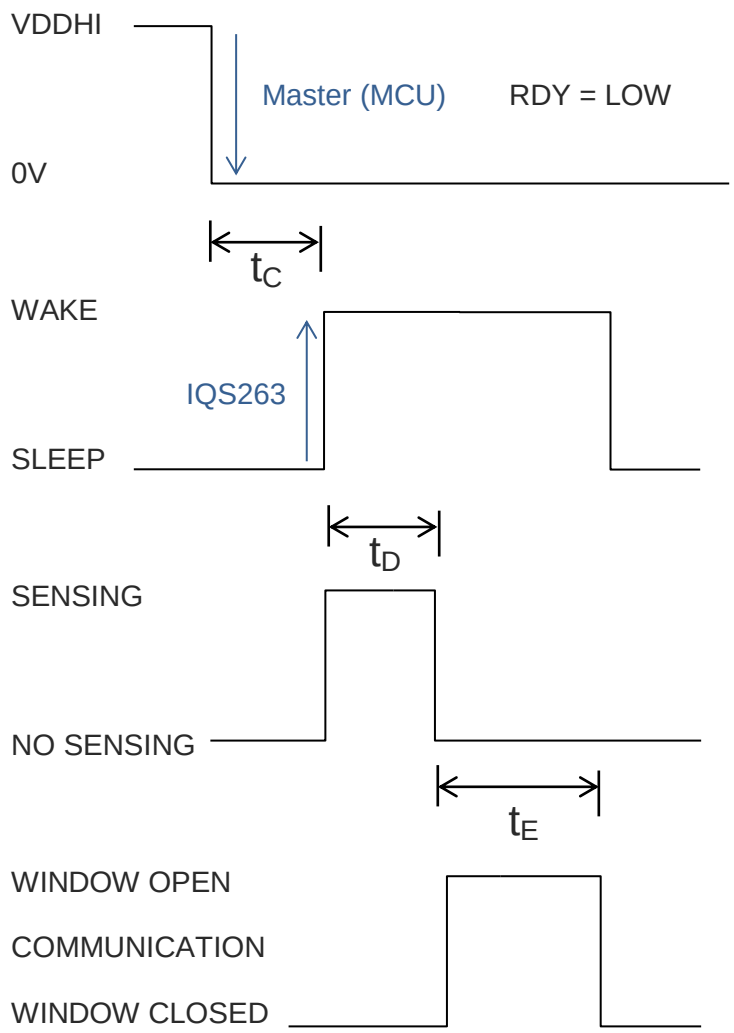
When using the IQS263 in event mode it will be required to implement a force communications procedure in the driver. The force communication is to be used as follows:

- **Failsafe:** using the device in event mode, a failsafe “check” will be required to ensure the IC is still in good working order while not generating any events.
- **Reseed:** When no movement is detected for a time period determined by the master MCU, a “reseed” command will clear such non-moving activation. This event will require forced communications.
- **ATI-error:** When ATI is somehow disrupted while being executed, an ATI-error may result. In this case, the MCU is responsible to call a redo-ATI in order to repeat a calibration attempt. Instead of engaging in multiple rapid retries, the master can schedule the retry according to application requirements by forcing communications when suited.

The handshake method previously recommended has a limited response rate and is not recommended for use.

The force communications technique is the fastest method of “catching” the IQS263 communications window. The following method is recommended:

- Master (MCU) write RDY = LOW
- Master initiates I^2C repeated polling until ACK is received from IQS263 (if NACK is detected then master must continue polling)
- When ACK is received, master should release the RDY line by making the I/O an input
- When the ACK is received the communications window is open until a STOP-command is sent
- Background: when IQS263 is in sleep mode in between sensing cycles, the device will initiate wake-up procedure with the RDY pin change.
- Background: The wake-up procedure always starts with sensing procedure on one of the channels. After the sensing and calculations the communications window will allow for a response on address polling.



t_C specification:

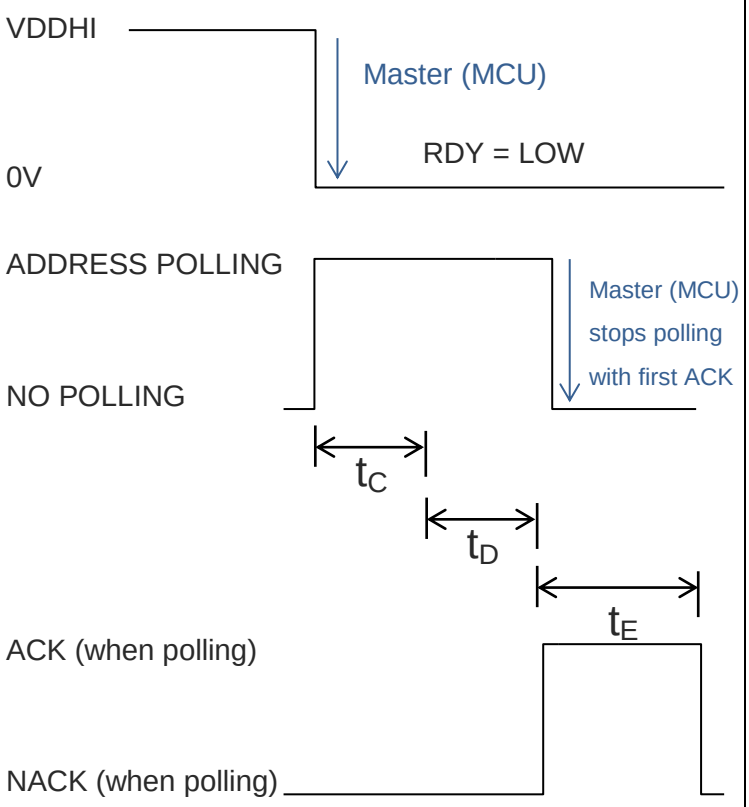
- After forcing the RDY low, the IQS263 will initiate a wake-up procedure.
- t_C = 4ms (typical)
- t_C < 5ms

t_D specification:

- One channel sensing data is gathered before allowing time for communications
- Channel sensing time is dependent on target setting and sensing frequency
- t_D = 3ms (typical)

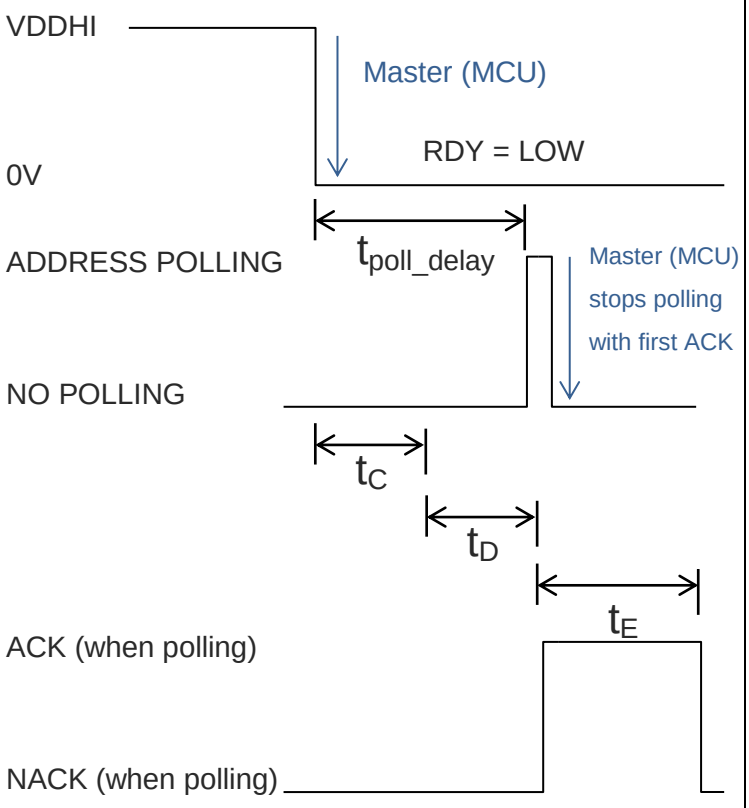
t_E specification:

- Time from IQS263 ready to respond to polling requests.
- Addressing the device in this window should result in an expected response (acknowledge)
- t_E = 5ms (default)
- t_E = Reg 0x0A:byte7
- Reg 0x0A:byte7 time-out value is cancelled after successful polling



Polling option: Brute force

- Recommended polling interval: 1ms



Polling option: Time delay

- Recommended:

  t_{poll_delay} > 8ms
- Change the I2C time-out to match the chosen polling delay

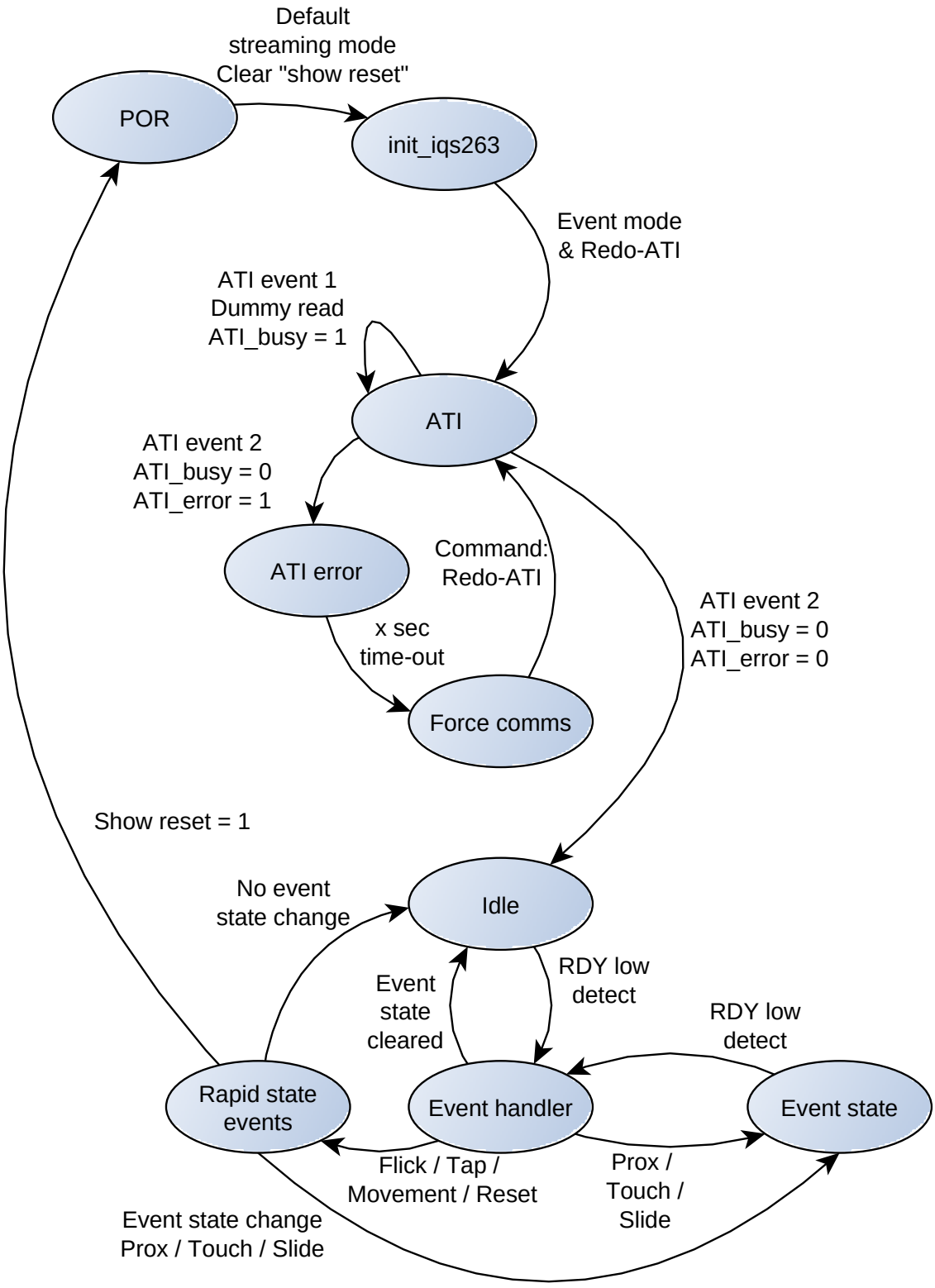
6. Recommended master implementations

Please see the IQS263 datasheet for a full specification of options that are typically fixed during run-time of the IQS263.

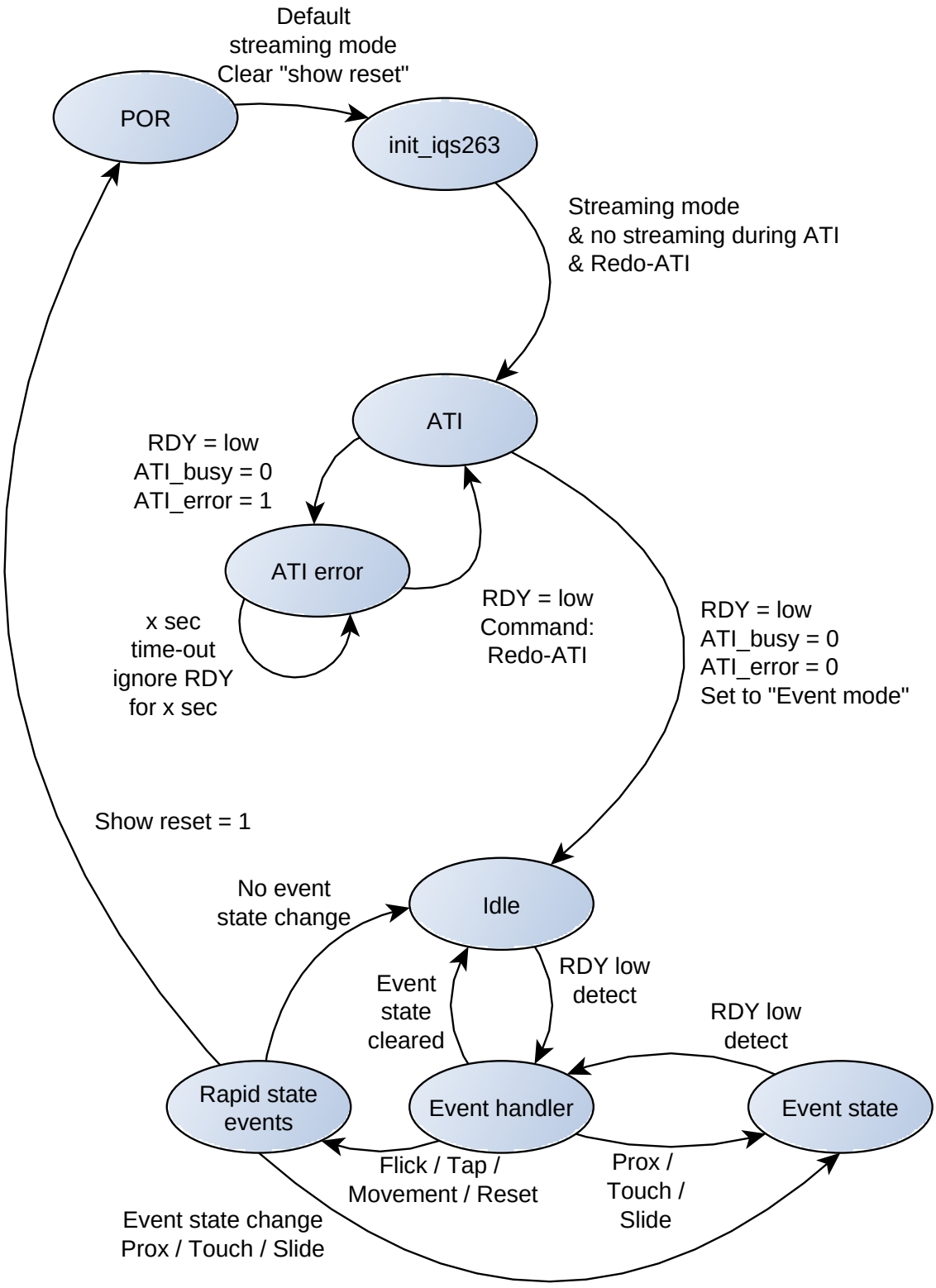
Most popular initialization options are as follows:

Channel setup	CH0 (distributed), CH1, CH2, CH3 CH0 cannot be disabled
Sensing specification	Sensing frequency and sample rate selection
Threshold adjustment	Filter halt Proximity threshold (CH0) Touch thresholds (CH1, CH2, CH3) Movement threshold (CH0 or CH3
Sensitivity adjustment	Target value (higher target – more sensitivity) Base value (lower base – more sensitivity)
Communication specification	Streaming, Event mode Streaming disabled during ATI option I2C timeout
Filter options	LTA beta Filtered touches Touch debounce
UI options	Slider, swipe, wheel etc.

6.1 Default recommended master implementation

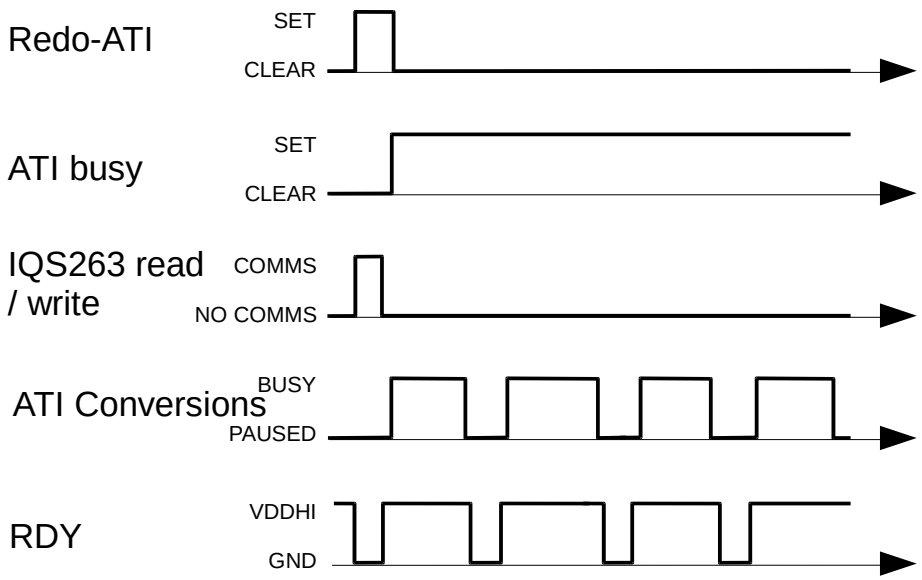


6.2 Alternate recommended master implementation

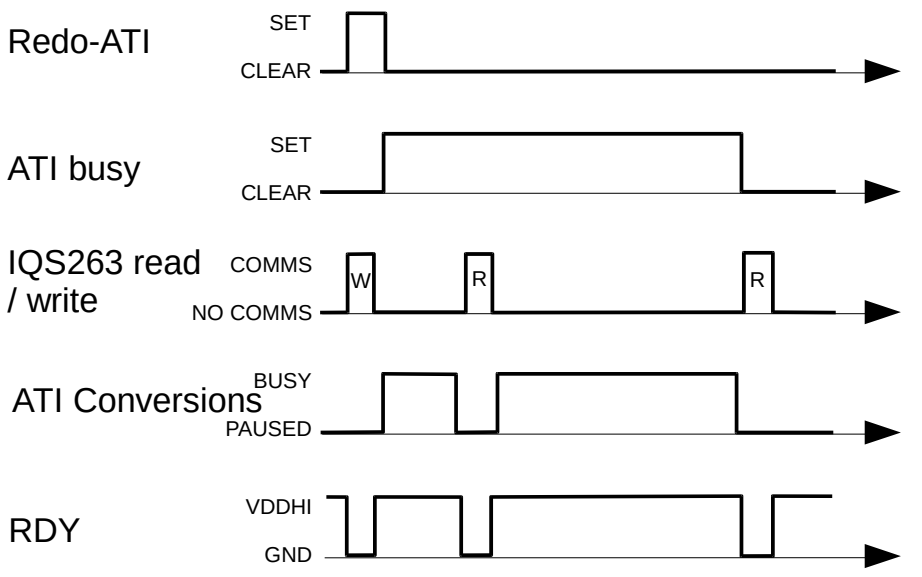


7. ATI (calibration) state

7.1 Event mode ATI events active



- After calling a redo-ATI, a dummy event is created to indicate the ATI start
- Read this dummy event in order to prevent repeated dummy events.
- Repeated ATI events (or streaming during ATI) causes a significant decrease in calibration speed



- By reading the dummy ATI event, the conversions required for calibration can rapidly execute without being slowed down by communication windows.
- Be sure to read events as soon as possible to prevent reoccurring events.

- IQS263 generates dummy ATI event after redo-ATI
- This must be read to clear the event, otherwise the RDY will toggle continuously while doing ATI (slows ATI considerably)
- Recommended: disable ATI events (mask), read ATI_busy 1second after redo-ATI command – check ATI-error.

8. Reseed case

8.1 Event mode reseed

- Usually RESEED is done automatically and the master device does not need to be aware of the event.
- The master may force a reseed in the case of an unwanted latched condition.
- Force communications and calling a reseed may have two outputs:
    1. Reseed only (LTA = counts)
    2. Redo-ATI (LTA=counts & maintain correct target)

- IQS263 does Reseed and ATI
    - Does force communications procedure
    - Sends reseed command to IQS263
    - Sends stop command
    - Waits for RDY=L
    - Reads the ATI Busy
    - Waits for RDY=L
    - Set RDY=L time-out (100ms)
    - If RDY = L then Read the ATI Busy & ATI Error bits to confirm ATI

- IQS263 does Reseed but not ATI
    - Does force communications procedure
    - Sends reseed command to IQS263
    - Sends stop command
    - Waits for RDY=L
    - Set RDY=L time-out (100ms)
    - If RDY ≠L then no ATI, only reseed was done

Appendix A Contact Information

	USA	Asia	South Africa
Physical Address	6507 Jester Blvd Bldg 5, suite 510G Austin TX 78750 USA	Rm2125, Glittery City Shennan Rd Futian District Shenzhen, 518033 China	109 Main Street Paarl 7646 South Africa
Postal Address	6507 Jester Blvd Bldg 5, suite 510G Austin TX 78750 USA	Rm2125, Glittery City Shennan Rd Futian District Shenzhen, 518033 China	PO Box 3534 Paarl 7620 South Africa
Tel	+1 512 538 1995	+86 755 8303 5294 ext 808	+27 21 863 0033
Fax	+1 512 672 8442		+27 21 863 1512
Email	info@azoteq.com	linayu@azoteq.com.cn	info@azoteq.com

Please visit www.azoteq.com for a list of distributors and worldwide representation.