IQS5xx-B000 I²C Bootloader v2.x Technical User Guide

Table of Contents

1 Introduction

The goal of this document is to describe the working of the Azoteq IQS5xx-B000 I²C Bootloader v2.x from a user point of view. The user in this case will be the engineer that needs to develop the firmware upgrade routines to be used on the master MCU to update the IQS5xx-B000.

2 Overview

The IQS5xx can run in one of two modes:

1. **Bootloader mode:**

   The IQS5xx provides the user with an interface containing numerous USB address commands that can be used to upgrade the application firmware.

2. **Trackpad application mode:**

   The IQS5xx executes the application firmware.

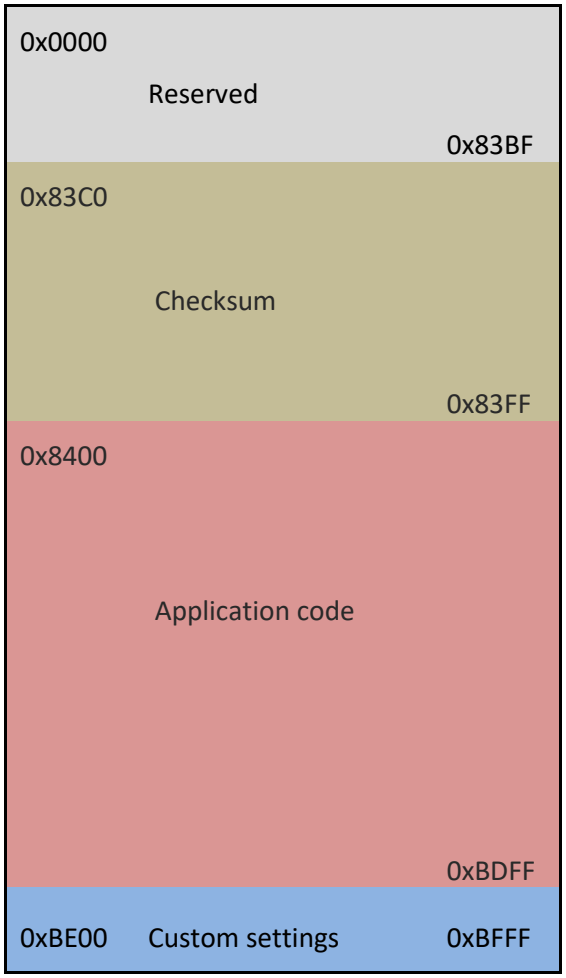
Note that the address commands associated with the bootloader mode are not available when the device runs in the application mode, and vice versa.

The next sections concern the following:

- Simplified IQS5xx memory map.
- Firmware upgrade procedure summary.
- Bootloader entry.
- I²C communication with the bootloader.

3 Simplified IQS5xx memory map

Here follows a graphical representation and a description of the IQS5xx-B000 memory map:



1. 0x0000-0x83BF should be considered reserved space and not written.
2. 0x83C0-0x83FF is the firmware checksum area.
3. 0x8400-0xBDFF contains the application code.
4. 0xBE00-0xBFFF contains the customer specific setup parameters (or defaults if not modified).

When performing a firmware upgrade the master MCU must write data to all addresses 0x83C0 through 0xBFFF (inclusive). All unused memory locations must be written as 0x00.

4 Summary of firmware upgrade procedure for IQS5xx-B000

Suggested order of events:

1. Enter bootloader (see Section 5).
2. Read and verify the bootloader version number (see Section 6.1) to verify bootloader entry and successful communication with correct version of bootloader.
3. Write the new application firmware and settings (0x8400-0xBFFF) to the device (see Section 6.5).
4. Write the checksum descriptor section (0x83C0-0x83FF).
5. Verify all programming was successful:
    - Perform a CRC check to verify the Application code section (see Section 6.4).
    - Read back the non-volatile custom settings section (0xBE00-0xBFFF) which is not included in the CRC calculation (see Section 6.2). Compare this to the data in the HEX file for that section, to verify each byte matches.
6. Exit bootloader mode either by command (see Section 6.3) or by resetting the IQS5xx (toggle NRST low).

5 Bootloader entry

This section describes the polling method that is used to enter bootloader mode. Once in bootloader mode, the master can write the application firmware to the IQS5xx and verify that the upgrade was successful using the available I^2C interface.

Shortly after device reset, an I^2C polling window is available. Since this is only available for a specific time after the IQS5xx powers-up, it is recommended that the master controller has access to the IQS5xx reset (NRST) line, so that the master can control the reset timing accurately, allowing for easy bootloader entry.

If the IQS5xx is polled (continuously addressed until an ACK is received), using the bootloader I^2C address (0x40 xor (IQS5xx I2C slave address)), the device enters bootloader mode.

The polling window stays open for approximately 2ms.

If the polling window expires, and the device has not been polled on the bootloader I^2C address, the device will enter the application mode and start executing the application firmware. The following flow chart describes bootloader entry using the polling method:

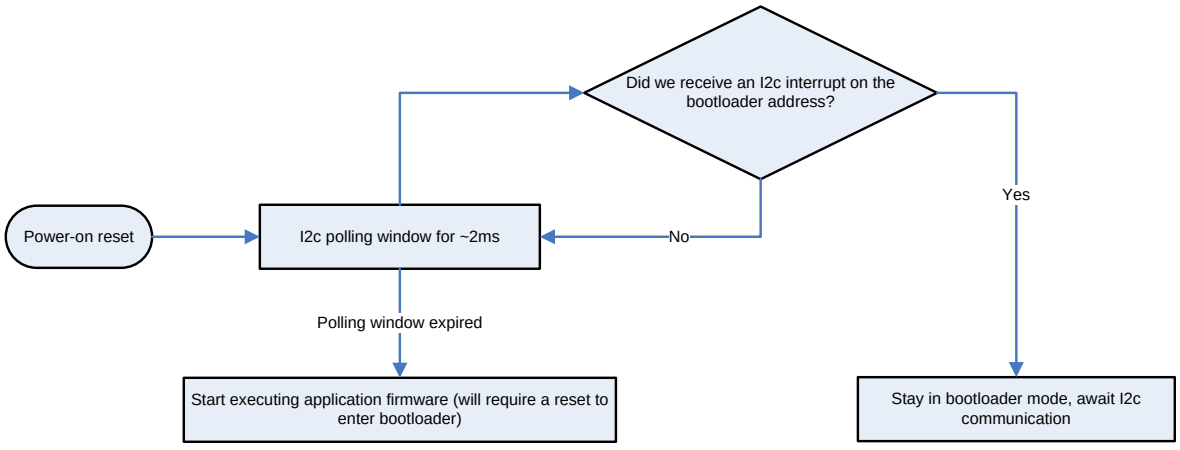


Figure 5.1 Bootloader entry

6 I²C communication with the bootloader

By default the IQS5xx-B000 has an I²C address of 0x74, the bootloader address is the application address XOR 0x40, hence by default 0x34 (0x74 ^ 0x40).

The following commands are supported by the bootloader:

I2C Command	Description
0x00	Read Bootloader version (Read only, 2 data bytes)
0x01	Read 64 bytes from memory map. (see description below)
0x02	Execute application firmware (Write only, 0 data bytes)
0x03	Do CRC check on current firmware. (Read one byte, 1=fail, 0=success)
Other	Write data bytes to firmware application address if valid address presented

Note that these commands are only available in bootloader mode.

6.1 Read bootloader version (0x00):

A single command byte (0x00) is written to the slave, the first two bytes read from the device will return the bootloader version. Detailed I²C will look like this:

Start, (slave address << 1), master writes 0x00, stop.

Start, ((slave address << 1) | 0x01), master reads two version bytes, stop.

It is recommended to read the bootloader version to ensure that bootloader entry was successful.

For the current bootloader version, the two bytes read should be 0x02 and 0x00 respectively.

6.2 Read block from memory (0x01):

The read command is intended for the master to verify that the firmware download has been successful. The CRC check may be trusted to verify the application program space, but note that certain blocks such as the non-volatile custom settings section is not included in the CRC check.

By writing 0x01 to the slave, followed by a two byte starting address, the slave will return 64 consecutive bytes starting at the specified address.

It is possible to verify the entire memory written to the slave with this command, but to optimise the bootloader programming routine; only the sections not covered by the CRC can be read back for verification.

Detailed I²C:

Start, (slave address << 1), master writes 0x01, start address MSB, start address LSB, stop.

Start, ((slave address << 1) | 0x01), master reads up to 64 bytes from slave, stop.

6.3 Execute application firmware (0x02):

This command can be seen as an ‘Exit bootloader’ command. After this command is sent, the bootloader will exit, and the application firmware will begin to execute.

The master writes a single command byte (0x02) to the slave, at which point the slave will jump to the application start address. I²C details:

Start, (slave address << 1), master writes 0x02, stop.

6.4 Calculate CRC of current application firmware (0x03):

The checksum is automatically calculated on the IQS5xx, and therefore the user does not need to do any CRC computing.

The checksum command (0x03) instructs the bootloader to perform a checksum on the actual data in the application code space, and compare it to the value found in the checksum data area (0x83C0-0x83FF).

If these two CRCs match, the application code space is correct and a 'pass' result is returned via I^2C. Alternatively a 'fail' can be returned, which would indicate that the data is incorrectly programmed, and the complete programming process must be repeated.

The master writes a single command byte (0x03) to the slave, one byte is read from the slave, which will be the CRC check pass or fail. A 1 indicates failure, 0 indicates success.

Start, (slave address << 1), master writes 0x03, stop

Start, ((slave address << 1) | 0x01), master reads single byte result, stop.

6.5 Program application firmware to device:

The device will program a block of application firmware once the entire block has been received via I^2C and the starting address of the block is valid.

Note: This action is different to the other 4 since it does not contain an address-command (0x00-0x03) as the others do.

Block size: 64 bytes.

Valid starting addresses are in multiples of 64 byte. (Hence the start address has to end with 6 zero bits, e.g. 0x83C0, 0x8400, 0x8440, 0x8480 etc.)

I^2C details:

Start, (Slave address << 1), (starting address MSB), (starting address LSB), data byte 1, data byte 2, … , data byte 64, stop.

NOTE: All 64 bytes MUST be sent in order to trigger the write on the bootloader.

Writes are allowed on blocks from 0x83C0 to 0xBFFF. Write the complete programmable area, filling all unused spaces with 0x00 (memory not defined in the program HEX file).

The firmware will be provided in standard intel hex format.

After a block write command has been given, and the 64 bytes have been written to the IQS5xx, then roughly 7ms should be allowed for the write to complete on the IQS5xx.

7 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	info@azoteq.com	info@azoteq.com

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