BCGMCU FW UPGRADE SPECIFICATION

BCGMCU-D01

Table of Contents

1 Programming with the Bootloader

The bootloader (BTL) enables firmware (FW) upgrade after the Debug lock of the MCU has been switched on for field usage. The bootloader is a custom configured version of the Silicon Labs Gecko Bootloader:
https://www.silabs.com/documents/public/user-guides/ug266-gecko-bootloader-user-guide.pdf

The bootloader is factory programmed and can be entered to with a correct startup sequence. The bootloader operates through its designated UART interface using ASCII commands, and it uses the XMODEM-CRC protocol for FW upgrades. Given a properly signed and encrypted firmware image (.gbl file format), the bootloader will decrypt and flash the application image into the MCU.

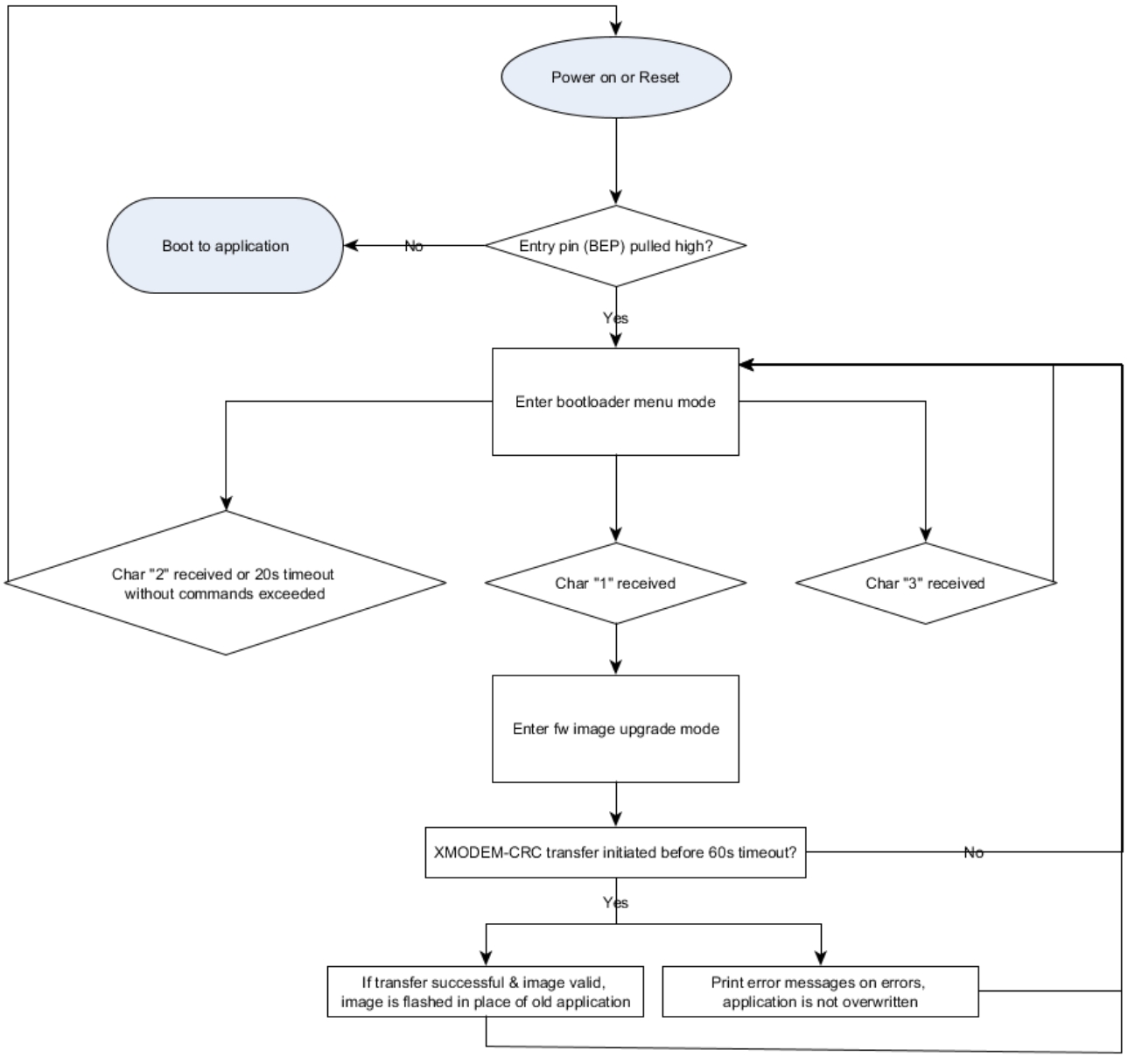

Figure 1 Bootloader workflow

1.1 FW Upgrade Mode Entry Sequence

Reset BCGMCU while pulling pin PF2 (bootloader entry pin) high (3.3V). Bootloader entry pin is checked on power-up and no timeouts or communication is necessary. The TP4 in the sample MCU has a pull-down resistor, so 3.3V voltage for bootloader entry can be supplied through a jump wire for example. The bootloader will enter into menu-mode after which commands introduced below can be used. The menu mode has a 20 second timeout, after which the MCU will attempt to boot into application.

1.2 UART Protocol

The UART protocol:

- Baud rate is 115200.
- No flow control is used.
- Start bit, 8 data bits, no parity, 1 stop bit.

1.3 Bootloader Hardware Interface

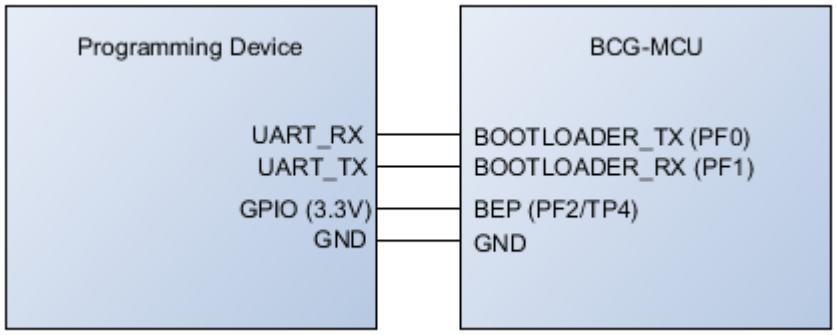

Figure 2 Bootloader hardware interface

2 Bootloader Protocol

2.1 Bootloader interface

The bootloader has a simple menu structure (menu-mode) where it waits for commands listed below until timeout at 20 seconds. The bootloader communicates through UART with ASCII encoded characters. It will broadcast its menu structure as a simple ASCII string, and accept command characters “1”, “2” and “3” encoded in ASCII (0x31, 0x32 and 0x33 respectively as hex).

2.2 Commands

Table 1 Bootloader commands in menu

Command	Description
“1” / 0x31	“upload gbl“: Start XMODEM-CRC upload of .gbl file. The bootloader will transmit “begin upload” followed by “C”-character until upload is completed. Upload will timeout if XMODEM transmission doesn’t start within 60 seconds.
“2” / 0x32	“run”: Bootloader will reset to application (unless BEP is held high)
“3” / 0x33	“ebl info”: Bootloader will reset into menu-mode (timeout is initialized)

2.2.1 GBL upload

The gbl file will be offered by Murata for use in field FW upgrades. When command “1” is sent to the bootloader (upload gbl), the bootloader will send “\r\nbegin upload\r\n”-string and enter the “gbl upload”-mode. In this mode, the bootloader will transmit a “C” character every second while waiting for XMODEM-CRC transmission to initiate. Upon completed transmission, bootloader will transmit “\r\nSerial upload complete\r\n” string and return to menu-mode. On errors, “Serial upload aborted\r\n”-string followed by an error code will be transmitted. The error codes are listed in table 2. If serial upload is successful, the old application image is overwritten and the newly uploaded application can be entered to through a “2” command (run) or through hard reset.

2.2.2 XMODEM-CRC

The XMODEM-CRC (128byte data packet, 16bit CRC) protocol is used for image file transfer. The protocol is well documented here:
(https://en.wikipedia.org/wiki/XMODEM#XMODEM-CRC)

and for example C- and Python-implementations exist as open source.

C-implementation example:
https://www.menie.org/georges/embedded/xmodem.html
Python implementation example:
https://pypi.org/project/xmodem/

Example Python code for firmware upgrades is available at request.

2.2.3 Upload error codes

Errors are divided to XMODEM errors (codes 0x21-0x27) and file errors (0x41-0x50).
Errors are reported in the following format:
File error: <errorcode> or XModem block error:<errorcode>

Table 2 Upload errors

Command	Description
0x21	BOOTLOADER_ERROR_XMODEM_NO_SOH “No start of header found”
0x22	BOOTLOADER_ERROR_XMODEM_PKTNUM “Packet number doesn't match its inverse”
0x23	BOOTLOADER_ERROR_XMODEM_CRCL “Could not verify lower CRC byte”
0x24	BOOTLOADER_ERROR_XMODEM_CRCH “Could not verify upper CRC byte”
0x25	BOOTLOADER_ERROR_XMODEM_PKTSEQ “Packet number error (unexpected sequence)”
0x27	BOOTLOADER_ERROR_XMODEM_PKTDUP “Packet number error (duplicate)”
0x41	BOOTLOADER_ERROR_PARSER_VERSION “Image file version doesn't match with parser”
0x43	BOOTLOADER_ERROR_PARSER_CRC “Invalid checksum”
0x44	BOOTLOADER_ERROR_PARSER_UNKNOWN_TAG “Unknown data type in image file”
0x45	BOOTLOADER_ERROR_PARSER_SIGNATURE “Invalid signature”
0x4F	BOOTLOADER_ERROR_PARSER_EOF “Image parsing is already done (or has previously errored out)”
0x50	BOOTLOADER_ERROR_PARSER_KEYERROR “Invalid encryption key or no key not present”

3 Example: FW Upgrade via USB to UART Bridge

The BCGMCU application FW can be upgraded using a PC and a simple USB to UART bridge and Tera Term (https://ttssh2.osdn.jp/index.html.en). Teraterm transmits keyboard inputs automatically through UART as ASCII encoded hex. The BEP pin can be driven through use of a jump wire from the 3.3 V wire on the UART bridge, or if SW switching is wanted, through the UART RST wire that can be set high through SW.

Requirements:

- BCGMCU
- PC
- USB to UART cable & bridge (FTDI TTL-232R-3V3)
- A jump wire
- Tera Term

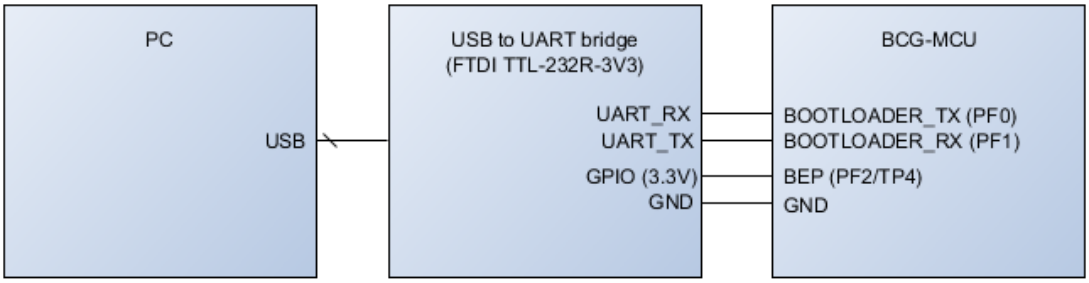

Figure 3 HW interface example

1. Connect the USB-UART bridge's RX to the PF0 (BOOTLOADER_TX) and TX to PF1 (BOOTLOADER_RX)
2. Connect USB-UART bridge to your PC
3. Open Tera Term on your PC
4. Connect to the COM port that your USB-UART bridge occupies, and set Serial connection settings ("Setup->Serial port…") to the ones displayed on figure 5.
5. Reset the BCG-MCU while holding BEP (PF2) high at 3.3V (using a jump wire for example).
6. If no menu text appeared on Tera Term, send command "3" by pressing 3 on keyboard while on Tera Term window (see text on figure 5).
7. The bootloader is now on menu-mode, and will timeout to application within 20 seconds (or if BEP is held high, will reset itself back to bootloader menu every 20 sec)
8. Enter "1" on Tera-term window to enter FW upload mode.
9. Within 60 seconds, open tab "File->Transfer->XMODEM->Send" in Tera Term and select a valid .gbl firmware file to send. Transfer will start when open is clicked and figure 6 view pops up.
10. On successful transfer, the bootloader returns to menu mode (see figure 7), and application can be started by sending the "2" (run) command while making sure BEP is no longer pulled high.

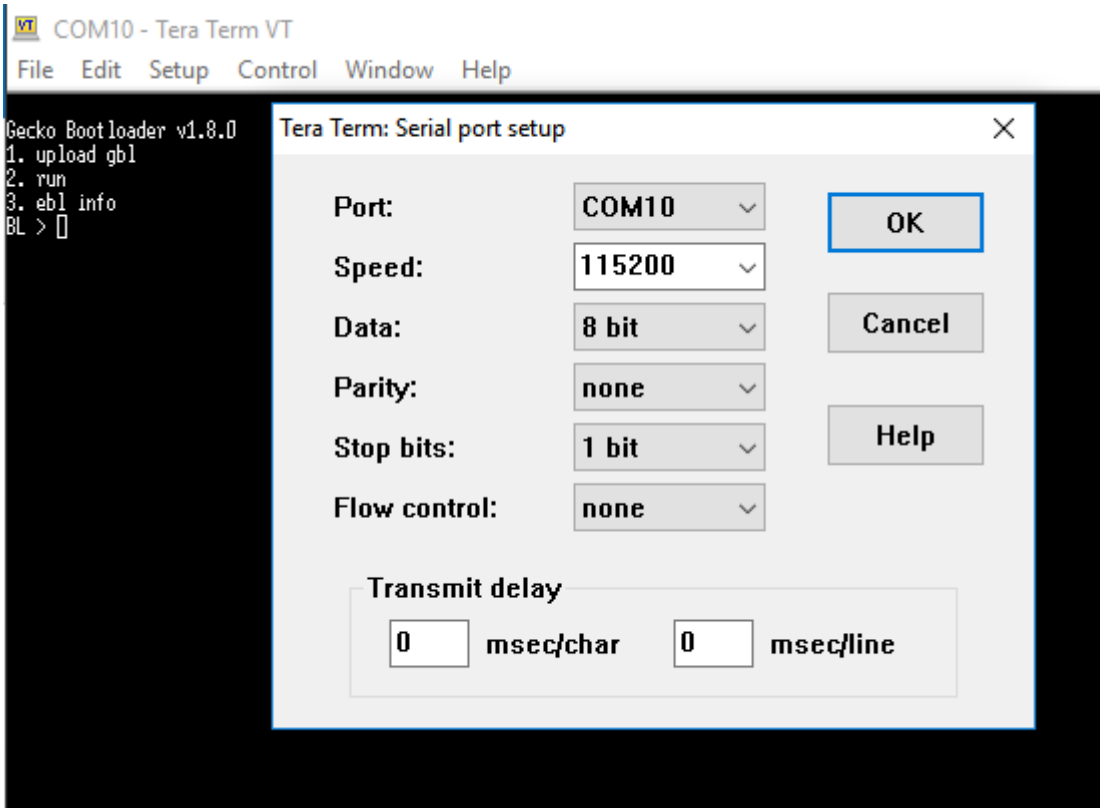

Figure 4 Tera Term settings

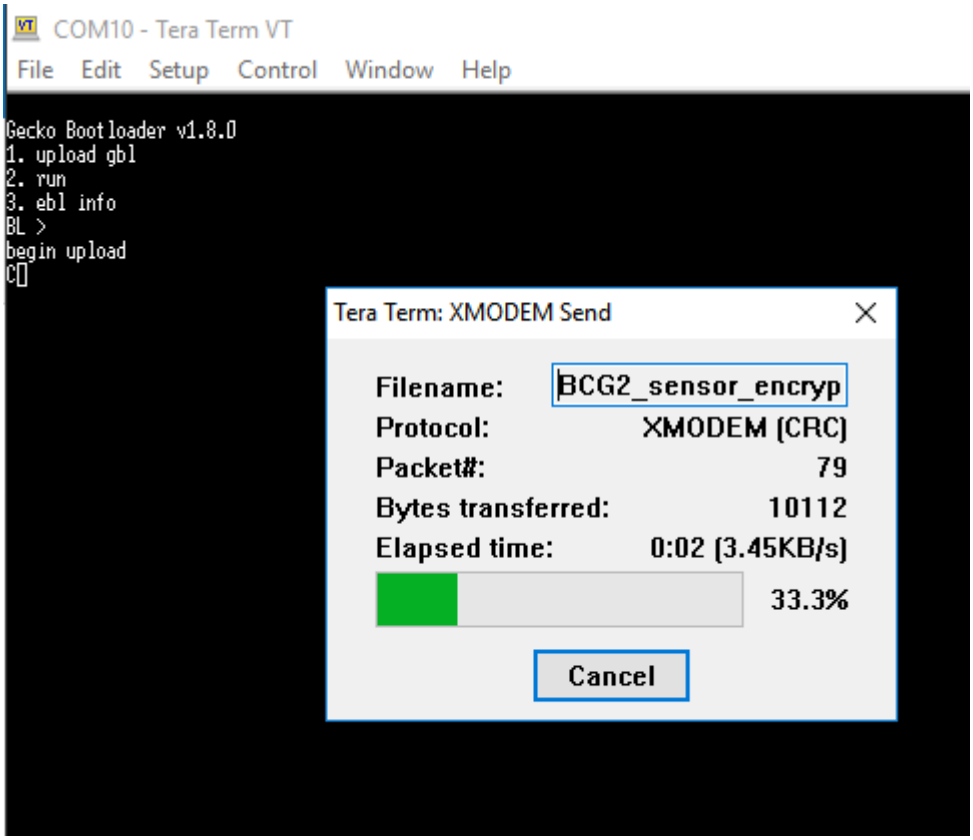

Figure 5 Tera Term XMODEM Send

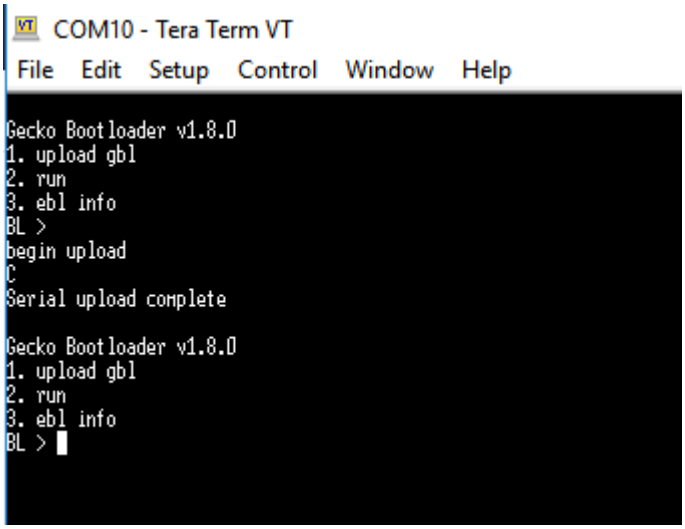

Figure 7 Tera Term XMODEM Serial upload complete and successful

4 Document Change Control

Rev.	Date	Change Description
1	18-Mar-19	First revision for BCG-MCU bootloader definition.
2	15-Oct-19	Fixed product name usage