

Precision GNSS Receiver mPCIe Module User Manual v1.4

Introduction

This document provides setup and usage guidelines for the Swift Navigation Precision GNSS Receiver mPCIe Module (PGM) Board.

Table of Contents

Introduction	1
PGM Board	2
Mounting Board in the Vehicle	3
PGM Firmware Update	3
Operation Check	6
Appendix A - udev Rules Example	7
Appendix B - Orientation Settings	8

PGM Board

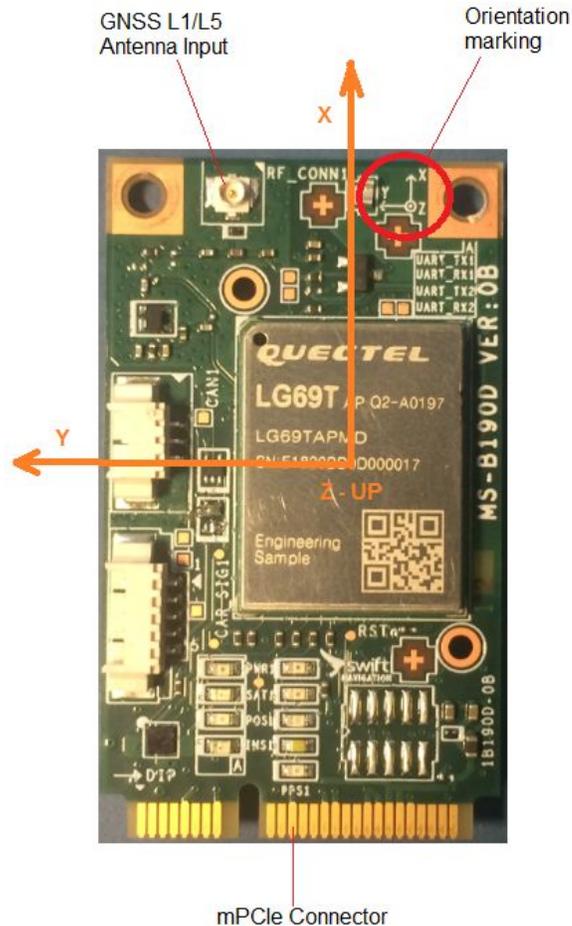


Fig 1. PGM Board (Heat Sink Removed)

Notes:

- For proper operation the PGM requires an active L1/L5, GPS/Galileo/BeiDou antenna. The module provides 3.3 V antenna bias voltage through the U.FL antenna connector.
- Use DC Block to isolate DC voltage if the antenna is connected via RF splitter (if DC block is not included in the splitter).
- The U.FL antenna connector is very fragile. Use caution when plugging/removing antenna cable and attach antenna cable to the device to avoid any tension on the connector.
- See Appendix B for the Starling orientation settings.

Mounting Board in the Vehicle

For proper inertial sensors operation it is essential to mount both the PGM and the GNSS antenna securely and firmly to the vehicle body. During operation, the antenna and PGM must remain in the same position relative to each other (i.e., both must be mounted on the same frame).

To maximize sensor sensitivity, mounting PGM in an orientation orthogonal to the vehicle-body is recommended. This means that PGM should be mounted with all three axes forming angles in multiples of 90° (0°, 90°, 180°, 270°), with respect to the primary direction of vehicle motion.

For easier installation and setup, it is recommended to align the X-axis direction of the PGM with the primary direction of vehicle motion.

PGM must be rigidly mounted to the body of the vehicle. Any vibration which is not directly related to vehicular motion will degrade the quality of the inertial data. Installing PGM on a flexible vehicle rooftop, engine cover, or fender - where the mounting surface can flex and vibrate independently of the vehicle body - should be avoided. Similarly, placing the sensor on plush seating of a vehicle with the antenna on the rigid part of the vehicle will yield poor results.

Record lever arm X, Y and Z vectors after installation. Measure from the center of the Quectel LG69T module to the phase center of the antenna.

Appendix B provides Starling settings euler angles for the typical installations.

PGM Firmware Update

Follow instructions below to update firmware on the PGM device.

To perform the update a PC with Python 3 is required. Example below uses a Linux computer.

PGM firmware update can be done in-circuit when the PGM is installed in the PC or via the USB port with PGM on mPCIe to USB adapter board. In both cases the programming procedure is the same.



Fig 2. PGM on the USB Adapter Board

1. The PGM programming tool is called `pgm-update-vX.X.X.py`. This is a Python 3 script and it's provided in source format.

Note: it is suggested to update PGM's default ports in the script to match the PC so they don't need to be entered at every run.

```
DEFAULT_AUX_DEV    = '/dev/pgm1aux'
DEFAULT_RESET_DEV = '/dev/pgm1reset'
```

2. Connect PGM to your PC and turn the power on. Two virtual serial ports (`/dev/ttyUSBn`) and two GPIO devices (`/dev/gpiochipn`) will be created on your PC. For example:

```
swift@fitlet1:~/update$ ls /dev/ttyUSB*
/dev/ttyUSB0 /dev/ttyUSB1
swift@fitlet1:~/update$ ls /dev/gpiochip*
/dev/gpiochip0 /dev/gpiochip1 /dev/gpiochip2
swift@fitlet1:~/update$
```

3. It is recommended to use udev rules to assign PGM constant device names. At Swift we use:

```
swift@fitlet1:~/update$ ls -l /dev/pgm*
lrwxrwxrwx 1 root root 7 Jun 12 18:33 /dev/pgm4aux -> ttyUSB1
lrwxrwxrwx 1 root root 7 Jun 12 18:33 /dev/pgm4main -> ttyUSB0
lrwxrwxrwx 1 root root 9 Jun 12 18:33 /dev/pgm4reset -> gpiochip5
swift@fitlet1:~/update$
```

See Appendix A for udev rules example.

4. If this is the first time PGM is used on the PC then following system settings will be required:
 - a. Add `gpio` group and add current user to that group. Example:

- ```
sudo groupadd gpio
sudo adduser <user> gpio
```
- b. Add current user to dialout group
 

```
sudo adduser <user> dialout
```
  - c. Log out and log in again
5. The PGM firmware is provided in a ZIP file. It contains TeseoV and MCU firmware images. Do not unzip that file.
  6. Use -h option to display pgm-update.py parameters and options

```
swift@onlogic1:~/update$./pgm-update-v2.1.1.py -h
usage: pgm-update-v2.1.1.py [-h] [-b BAUD] [-t {stm32,sta8100,combined}] [-s] [--version] [-g RESET_DEV] [-p DATA_DEV]
 binary

Swift Navigation PGM Programming Tool

positional arguments:
 binary binary file or combined archive

optional arguments:
 -h, --help show this help message and exit
 -b BAUD, --baud-rate BAUD
 baud rate, default: 460800
 -t {stm32,sta8100,combined}, --target {stm32,sta8100,combined}
 target to program, default: combined
 -s, --no-sync skip sync
 --version show program's version number and exit
 -g RESET_DEV, --gpio-reset RESET_DEV
 GPIO device for reset, default: /dev/pgm8reset
 -p DATA_DEV, --serial-port DATA_DEV
 serial port, default: /dev/pgm8aux
swift@onlogic1:~/update$
```

7. Launch pgm-update.py with required parameters. Use PGM's AUX serial port (for data transfer) and PGM's AUX gpio (for reset) for module programming. Example:

```
swift@onlogic1:~/update$./pgm-update-v2.1.1.py pgm-v1.1.5.zip
Swift Navigation PGM Programming Tool v2.1.1
Waiting for sync...
Synchronisation success
Send address success
Send binary information success
.....
Send start command success
Programming 1, block 0 (0.0%)
Programming 1, block 1 (0.7%)
Programming 1, block 2 (1.3%)
Programming 1, block 3 (2.0%)
```

8. Programming takes about 2 minutes. "Programming 1" programs TeseoV firmware and "Programming 2" programs MCU.
9. Programming completed:

```
Programming 2, block 16 (72.7%)
Programming 2, block 17 (77.3%)
Programming 2, block 18 (81.8%)
Programming 2, block 19 (86.4%)
Programming 2, block 20 (90.9%)
Programming 2, block 21 (95.5%)
Programming completed successfully
swift@onlogic1:~/update$
```

10. After programming PGM is ready to use. Additional reset is not required.

## Operation Check

To verify the basic firmware operation, read data from the PGM's AUX port. Text data (NMEA Swift proprietary message) is output at 460800 bps and prints the firmware version information every 30 seconds.

Example:

```
python3 -m serial.tools.miniterm /dev/pgm8aux 460800 --raw <Enter>
```

```
swift@onlogic1:~/update$ python3 -m serial.tools.miniterm /dev/pgm8aux 460800 --raw
--- Miniterm on /dev/pgm8aux 460800,8,N,1 ---
--- Quit: Ctrl+] | Menu: Ctrl+T | Help: Ctrl+T followed by Ctrl+H ---
$PSWTL,NOTICE,MCU FW version: v1.1.5*3F
$PSWTL,NOTICE,MCU Unique ID: 0046001E-31395119-35333634*5F
$PSWTL,NOTICE,STA FW config: PR6A 07dd3045c*22
$PSWTL,NOTICE,STA FW version: GNSSLIB_9.4.0.0_RC_ARM*1C
$PSWTL,NOTICE,PGM HW version: 01097-02 Rev 02*20
$PSWTL,NOTICE,PGM process version: 01109-01 Rev 03*50
```

## Appendix A - udev Rules Example

File name: 99-swift-pgm.rules

File directory: /etc/udev/rules.d/

### File content:

```
SUBSYSTEMS=="usb", ATTRS{product}=="Swift Navigation Precision GNSS Module",
ENV{usb_num}="$attr{devpath}"
SUBSYSTEM=="tty", ATTRS{interface}=="SwiftNav PGM Main Port", SYMLINK+="pgm$env{usb_num}main"
SUBSYSTEM=="tty", ATTRS{interface}=="SwiftNav PGM Aux Port", SYMLINK+="pgm$env{usb_num}aux"
SUBSYSTEM=="gpio", MODE="0660", GROUP="gpio"
SUBSYSTEM=="gpio", ATTRS{interface}=="SwiftNav PGM Aux Port", SYMLINK+="pgm$env{usb_num}reset"
```

In this example each PGM will create a unique set of devices in format:

```
/dev/pgm1aux
/dev/pgm1main
/dev/pgm1reset
```

where the PGM number reflects the USB port where PGM is connected to.

## Appendix B - Orientation Settings

Use the worksheet below to determine corresponding euler angles for the Starling orientation settings (*rotation-sensor-vehicle-degrees*). Angles are in degrees.

|                                                                     |                                                                       |                                                                         |                                                                        |
|---------------------------------------------------------------------|-----------------------------------------------------------------------|-------------------------------------------------------------------------|------------------------------------------------------------------------|
| <b>X: Forward</b><br><b>Y: Left</b><br>Z: 0.0<br>Y: 0.0<br>X: 180.0 | <b>X: Right</b><br><b>Y: Forward</b><br>Z: 90.0<br>Y: 0.0<br>X: 180.0 | <b>X: Backward</b><br><b>Y: Right</b><br>Z: 180.0<br>Y: 0.0<br>X: 180.0 | <b>X: Left</b><br><b>Y: Backward</b><br>Z: -90.0<br>Y: 0.0<br>X: 180.0 |
| <b>X: Forward</b><br><b>Y: Right</b><br>Z: 0.0<br>Y: 0.0<br>X: 0.0  | <b>X: Right</b><br><b>Y: Backward</b><br>Z: -90.0<br>Y: 0.0<br>X: 0.0 | <b>X: Backward</b><br><b>Y: Left</b><br>Z: 180.0<br>Y: 0.0<br>X: 0.0    | <b>X: Left</b><br><b>Y: Forward</b><br>Z: 90.0<br>Y: 0.0<br>X: 0.0     |
| <b>X: Forward</b><br><b>Y: Up</b><br>Z: 0.0<br>Y: 0.0<br>X: 90.0    | <b>X: Right</b><br><b>Y: Up</b><br>Z: 0.0<br>Y: 90.0<br>X: 90.0       | <b>X: Backward</b><br><b>Y: Up</b><br>Z: 180.0<br>Y: 0.0<br>X: -90.0    | <b>X: Left</b><br><b>Y: Up</b><br>Z: 0.0<br>Y: -90.0<br>X: 90.0        |
| <b>X: Forward</b><br><b>Y: Down</b><br>Z: 0.0<br>Y: 0.0<br>X: -90.0 | <b>X: Right</b><br><b>Y: Down</b><br>Z: 0.0<br>Y: -90.0<br>X: -90.0   | <b>X: Backward</b><br><b>Y: Down</b><br>Z: 180.0<br>Y: 0.0<br>X: 90.0   | <b>X: Left</b><br><b>Y: Down</b><br>Z: 0.0<br>Y: 90.0<br>X: -90.0      |
| <b>X: Up</b><br><b>Y: Left</b><br>Z: 0.0<br>Y: 90.0<br>X: -180.0    | <b>X: Up</b><br><b>Y: Forward</b><br>Z: 90.0<br>Y: 0.0<br>X: -90.0    | <b>X: Up</b><br><b>Y: Right</b><br>Z: 0.0<br>Y: -90.0<br>X: 0.0         | <b>X: Up</b><br><b>Y: Backward</b><br>Z: -90.0<br>Y: 0.0<br>X: 90.0    |
| <b>X: Down</b><br><b>Y: Left</b><br>Z: 0.0<br>Y: -90.0<br>X: 180.0  | <b>X: Down</b><br><b>Y: Forward</b><br>Z: 90.0<br>Y: 0.0<br>X: 90.0   | <b>X: Down</b><br><b>Y: Right</b><br>Z: 0.0<br>Y: 90.0<br>X: 0.0        | <b>X: Down</b><br><b>Y: Backward</b><br>Z: -90.0<br>Y: 0.0<br>X: -90.0 |