Precision GNSS Module
Evaluation Kit
User Manual
Revision 2

Introduction

This document provides setup and usage reference for the Swift Navigation's Precision GNSS Module (PGM) Evaluation Kit.

The Evaluation Kit - featuring Swift's mPCIe format PGM receiver installed in an Onlogic IoT computer platform - provides a turn-key demonstration of Swift's precision positioning solution. The PGM provides real-time precision GNSS and IMU measurements, and is designed specifically for Swift's Starling® positioning engine running on a host application processor. When used with Swift's Skylark™ cloud-based precise corrections service, the PGM Evaluation Platform (PGM EVP) delivers even stronger performance and higher accuracy levels. The Evaluation Kit is ideal for customers building industrial, last mile and Internet of Things (IoT) platforms that require the ability to quickly install on a target platform and test performance in the field.

The PGM Evaluation Kit includes everything needed to install this GNSS system in your application and quickly get started with precise RTK positioning. It also comes with a 6-months Skylark trial.

This manual describes Evaluation Kit with files version 1.4.0. The software update procedure is described in Appendix A.

Fig 1. PGM Evaluation Platform

Table of Contents

Evaluation Kit Content

Fig 1. PGM Evaluation Kit Content

1. GNSS L1/L5 magnet-mount antenna
2. PGM Evaluation Platform (PGM EVP)
3. Wi-Fi antennas
4. RS232 serial port adapter cable
5. USB drive for data recording
6. RS232 null modem for serial port connection
7. Vehicle signals cable
8. Power supply cable for direct 12 V supply
9. 12 V DC power supply 100-240 V AC input with international plugs
10. RJ45 Ethernet cable
11. DisplayPort to HDMI adapter cable

PGM Evaluation Platform

The PGM Evaluation platform is based on the Onlogic CL200 series industrial computer. The Swift Navigation PGM module occupies one of the internal mPCIe slots. Additionally, the platform contains a Wi-Fi / Bluetooth mPCIe module for wireless connectivity, RS232 serial port for GNSS data output, Ethernet port, display port and three USB ports. Platform runs Ubuntu 20.04 Linux server out of the micro SD card.

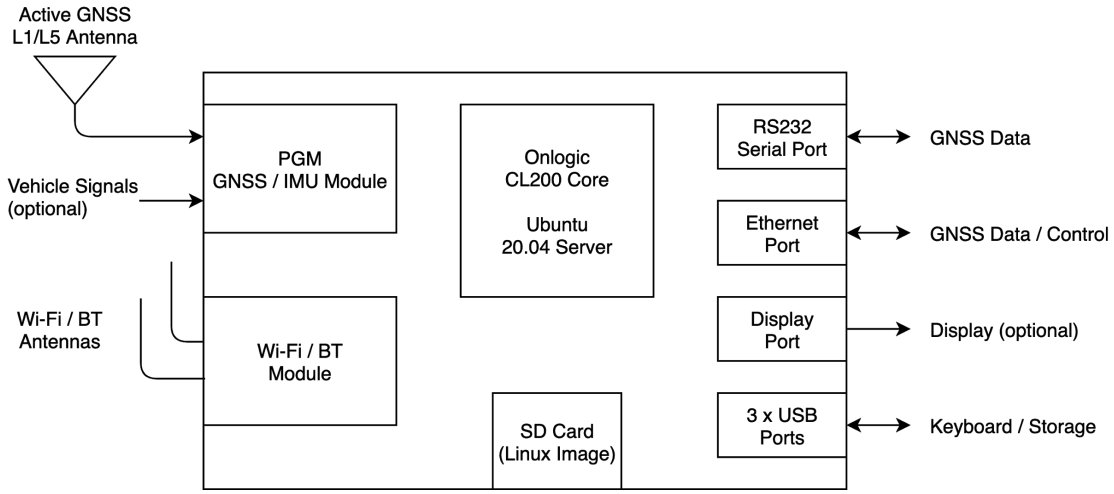


Fig 2. PGM Evaluation Platform Block Diagram

By default, the Wi-Fi module is configured as an access point enabling users to connect to the platform from any modern PC/laptop to configure the system and receive GNSS position data.

The Wi-Fi module can simultaneously work as a client to connect to a cell modem device for Internet access to receive GNSS corrections data. Note: both access point and Wi-Fi client must use the same Wi-Fi channel.

By default, the platform starts after applying power. To stop the platform use the power button or the terminal command for a graceful system shut down or just disconnect the power. Disconnecting power may result in data corruption, especially if recording.

Fig 3. PGM EVP Connectors

1. Wi-Fi antenna RP (Reverse Polarity) SMA connector
2. GNSS antenna SMA connector
3. Vehicle signals receptacle (Hirose HR10A-7R-6P(73))
4. Wi-Fi antenna RP SMA connector
5. Micro SD card slot (Linux image)

Fig 4. PGM EVP Connectors (Continued)

1. 12 V power supply 2.5/5.5 mm barrel jack connector
2. USB 2.0 port connector
3. RS232 serial port connector (use adapter cable for DE 9 connector)

4. Power button and power status LED
5. Two USB 3.0 port connectors
6. Ethernet port RJ45 connector
7. Mini DisplayPort connector
8. IMU orientation labels. The orange dot indicates the IMU center.

Operating Conditions

1. Supply voltage: 12 V DC.
2. Power consumption: about 6 W during normal operation.
3. Operational temperature: 0 to 40°C with a non-condensing relative humidity of 10 - 90%.
4. The device shall not be used outdoors without an additional environmental protection.

Installation in the Vehicle

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

The GNSS antenna needs to be mounted on the vehicle where there are no obstructions and with a correct ground plane. Typical car's roof is a good enough ground plane. Place the antenna at the center of the roof as much as possible. Do not place the antenna close to the roof edge. If the ground plane is not available by vehicle construction, place a round or square metal plate of 10 to 15 cm (4" to 6") radius under the antenna.

To maximize inertial sensor sensitivity, mounting PGM EVP in an orientation orthogonal to the vehicle-body is recommended. This means that PGM EVP 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 EVP with the primary direction of vehicle motion.

PGM EVP 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 EVP 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 orange dots printed on the PGM EVP enclosure to the center of the antenna.

Refer to Appendix C for Starling settings euler angles of typical installations.

Evaluation Platform Connections

The PGM EVP platform requires connections to power source, GNSS antenna, control/recording computer, and correction data source. Connections to the correction data and control/recording computer may be wired or wireless.

Power Connections

The platform requires a 12 V DC power supply with a rate of 3 A. Use included in the kit wall power adapter or other power supply device with 12 V / 3 A DC output.

GNSS Antenna Connections

PGM EVP requires an active GPS/Galileo/BeiDou L1/L5 antenna. The device provides 3.3 V antenna bias voltage through the SMA antenna connector (max. 100 mA). For the best results the antenna LNA gain should be between 15 to 25 dB and with NF < 2-3 dB.

Vehicle Signals

Optionally, the Vehicle Speed Signal (VSS) and Reverse can be connected to the platform for improved Dead Reckoning performance. See Appendix F for connection details.

Control/Recording Computer Connections

Use one of the following methods to control the platform:

1. Connect your PC/laptop over Wi-Fi to PGM EVP's access point named **SwiftNav-PEP-xxxx**, printed on the device. The Wi-Fi password is `swiftnav`. Upon successful connection, open SSH client (like PuTTY on Windows) and connect to PGM EVP using the IP address **10.42.0.1**.
2. Connect your PC via Ethernet. By default, PGM EVP Ethernet is set to the DHCP mode and therefore PGM EVP must be connected to a network router with a DHCP server. Check in the router settings for the assigned IP address or scan the network for the device IP. Once IP is known, open SSH client (like PuTTY on Windows) and connect to PGM EVP using the assigned IP address.
3. Connect a monitor to the DisplayPort and a USB keyboard to control the platform.

Correction Source Connections

To obtain Skylark corrections data PGM EVP needs Internet access. Use one of the following methods to enable PGM EVP's Internet access:

1. Connect PGM EVP to the Internet access point over Wi-Fi (cell modem with router, cell phone with Personal Hotspot, etc.). The connection must use the same Wi-Fi channel as the PGM EVP's Wi-Fi access point.
2. Connect PGM EVP over the Ethernet to the cell modem / router with Internet access. By default, PGM EVP Ethernet is set to the DHCP mode and therefore PGM EVP must be connected to a network with a DHCP server. PGM EVP Ethernet network can also be set to static IP if required for networks without a DHCP server.

Evaluation Platform Usage

The platform runs Ubuntu 20.04 Linux server. Use SSH to control the platform. Default credentials:

Username: `swiftnav`
Password: `swiftnav`

Starling and networking use text configuration files. Connect to the PGM EVP Linux shell to modify the configuration files.

Home Directory Files

Configuration files are saved in the home directory as shown below:

/home/swiftnav/

Name	Size	Changed	Rights	Owner
..		12/4/2020 11:06:05 PM	rwxr-xr-x	root
cfg		5/21/2021 1:37:17 AM	rwxrwxr-x	swiftnav
pgm		6/29/2021 4:37:19 AM	rwxr-xr-x	swiftnav
starling		6/29/2021 4:45:27 AM	rwxr-xr-x	swiftnav
tools		12/5/2020 12:39:31 AM	rwxr-xr-x	swiftnav
check-for-updates.py	4 KB	6/29/2021 4:45:28 AM	rwxr-xr-x	swiftnav
eth-edit.sh	1 KB	6/29/2021 4:45:28 AM	rwxr-xr-x	swiftnav
starling-edit.sh	1 KB	6/29/2021 4:45:28 AM	rwxr-xr-x	swiftnav
starling-restart.sh	1 KB	6/29/2021 4:45:28 AM	rwxr-xr-x	swiftnav
starling-version.sh	1 KB	6/29/2021 4:45:28 AM	rwxr-xr-x	swiftnav
version.yaml	1 KB	6/29/2021 4:45:28 AM	rw-r--r--	swiftnav
wifi-connect.sh	1 KB	6/29/2021 4:45:28 AM	rwxr-xr-x	swiftnav
wifi-delete.sh	1 KB	6/29/2021 4:45:28 AM	rwxr-xr-x	swiftnav
wifi-disconnect.sh	1 KB	6/29/2021 4:45:28 AM	rwxr-xr-x	swiftnav
wifi-edit.sh	1 KB	6/29/2021 4:45:28 AM	rwxr-xr-x	swiftnav
wifi-list.sh	1 KB	6/29/2021 4:45:28 AM	rwxr-xr-x	swiftnav

Fig 5. Home directory files

Note: Instead of using the command line SSH interface you can also use a program like WinSCP (on Windows) to edit configuration files using the GUI interface.

Files Description:

Directory: `cfg/`

`swiftnav-ap.sh`	- PGM EVP Wi-Fi access point configuration
`swiftnav-pgm.sh`	- PGM module serial port to TCP server routing configuration

Directory: `pgm/`

`pgm-update.py`	- PGM firmware updater Python script
`pgm-v1.2.4.zip`	- PGM firmware bundle
`t.sh`	- script to launch mini terminal
`u.sh`	- script to launch PGM firmware updater

Directory: `starling/`

`logs/`	- directory for Starling system logs
`activation-key.txt`	- File with activation key for Starling
`config.yaml`	- Starling configuration file
`igs14.atx`	- Satellite antenna corrections (for Skylark SSR only)
`license.lic`	- Starling license file
`starling`	- Starling program (binary)
`starling-guard.json`	- File for Starling license manager
`start.sh`	- Starling start script (called from starling service)

Directory: `tools/`

`str2str`	- stream to stream routing program (binary)

Home Directory:

`check-for-updates.py`	- Checks and downloads PGM EVK software update
`version.yaml`	- EVK files version information

Control Scripts

Following shell scripts are provided for convenience for common operations over SSH:

`eth-edit.sh`	- Opens Ethernet configuration file for editing
`starling-edit.sh`	- Opens Starling configuration file for editing
`starling-restart.sh`	- Restarts Starling service (required after changing configuration)
`starling-version.sh`	- Displays Starling version

`wifi-connect.sh` - Connects to a Wi-Fi access point. Requires SSID and password Parameters. Usage: `./wifi-connect.sh <SSID> <PASSWORD>`

`wifi-delete.sh` - Deletes (forgets) Wi-Fi network. Requires SSID parameter. Usage: `./wifi-delete.sh <SSID>`

`wifi-disconnect.sh` - Disconnects from Wi-Fi access point

`wifi-edit.sh` - Opens Wi-Fi access point configuration file

`wifi-list.sh` - Scans and lists nearby Wi-Fi networks

Refer to Appendix G for scripts content.

Platform Setup Example with Wi-Fi Connections

This section provides step-by-step instructions on how to set up the PGM EVP system using Wi-Fi connections to a control laptop and a cellular modem.

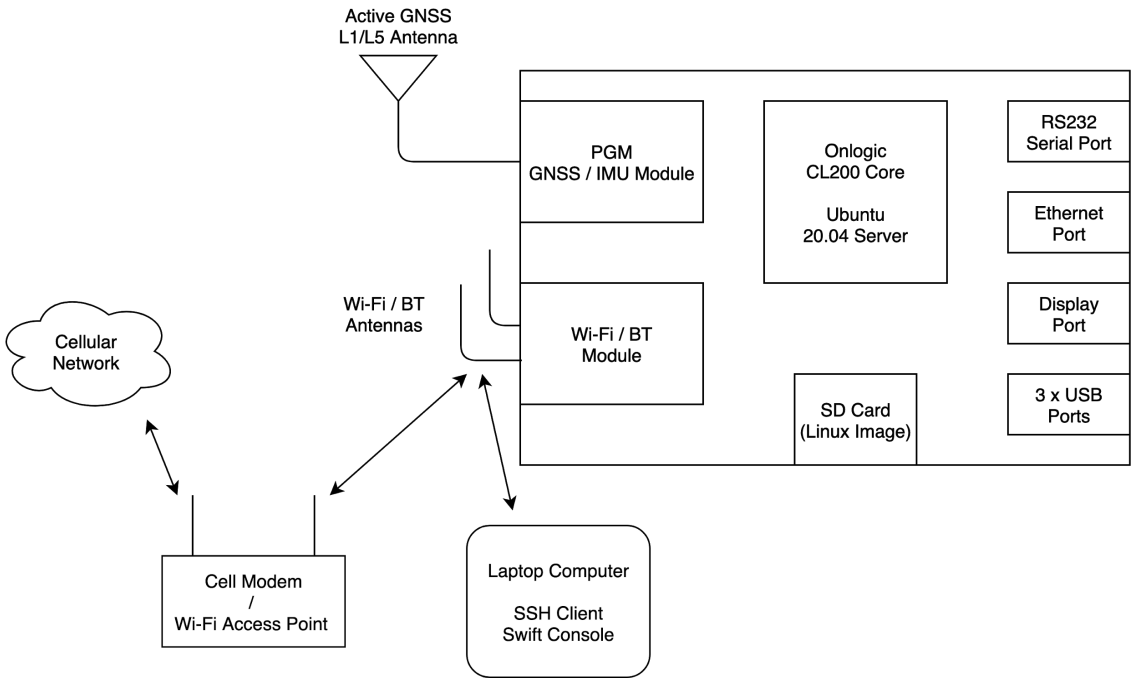


Fig 6. Platform Setup Example with Wi-Fi Connections

Required equipment:

1. PGM EVP device with GNSS and Wi-Fi antennas.
2. Laptop computer (Windows, Linux or macOS) with Wi-Fi networking.
3. Cellular modem with Wi-Fi access point. It can be a dedicated device or a cellphone with a personal hotspot.

Example presented below uses:

- PGM EVP with label reading SwiftNav-PEP-2795
- Cell modem / Wi-Fi access point with SSID “SwiftNav Mobile 4” and password “centimeter”
- Windows laptop for control

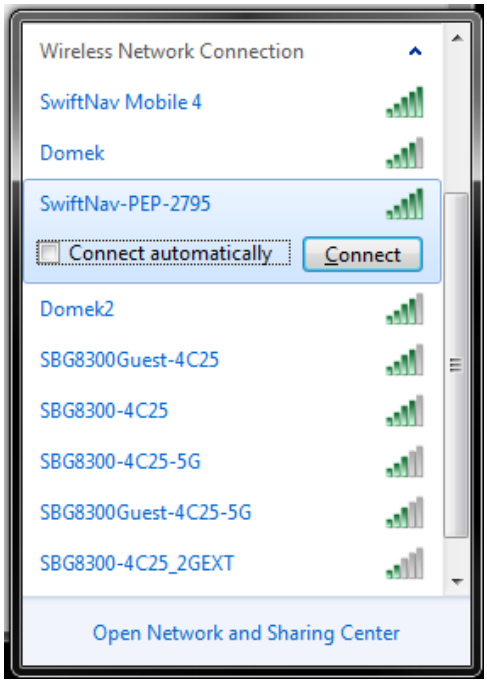
Follow the steps described below to setup and configure the system.

Connecting and bringing up the PGM EVP system

- Connect GNSS and Wi-Fi antennas to PGM EVP.
- The GNSS antenna needs to be outdoors with a good sky view.
- Power up PGM EVP and let it boot up (about 2 minutes). White LED shows power status.

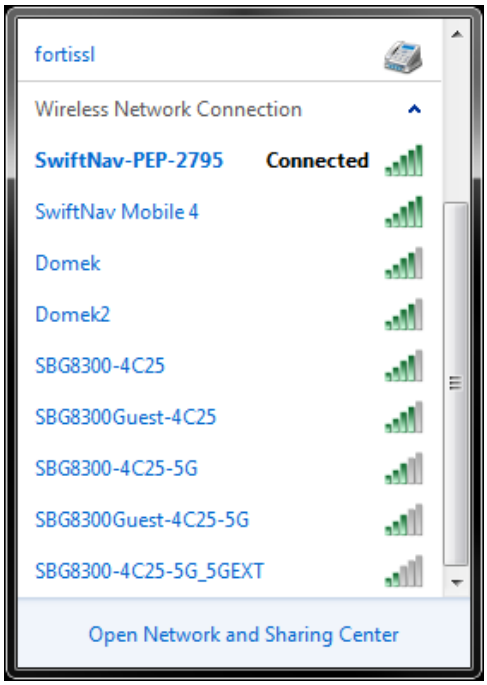
Connecting computer to PGM EVP over Wi-Fi for the system control

- Open Wi-Fi network selection window and select **SwiftNav-PEP-xxxx** network:

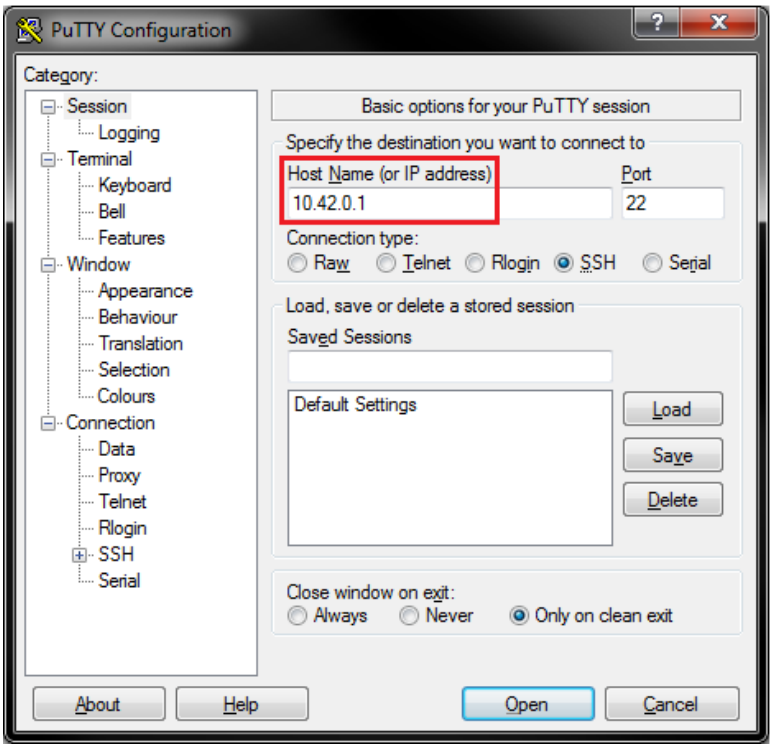


- Enter password (`swiftnav` by default):

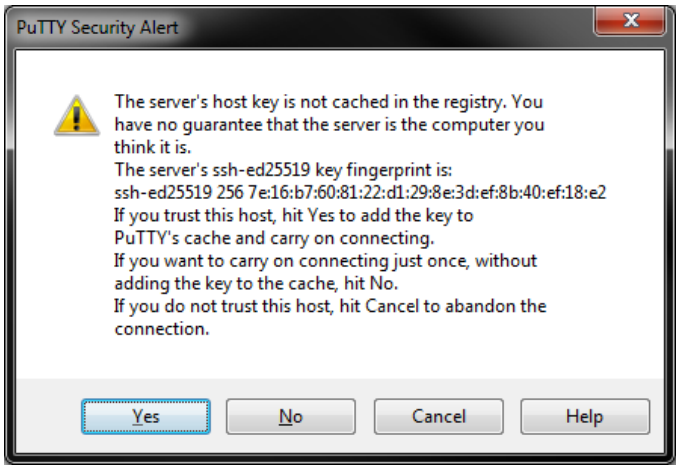
- Network is connected:



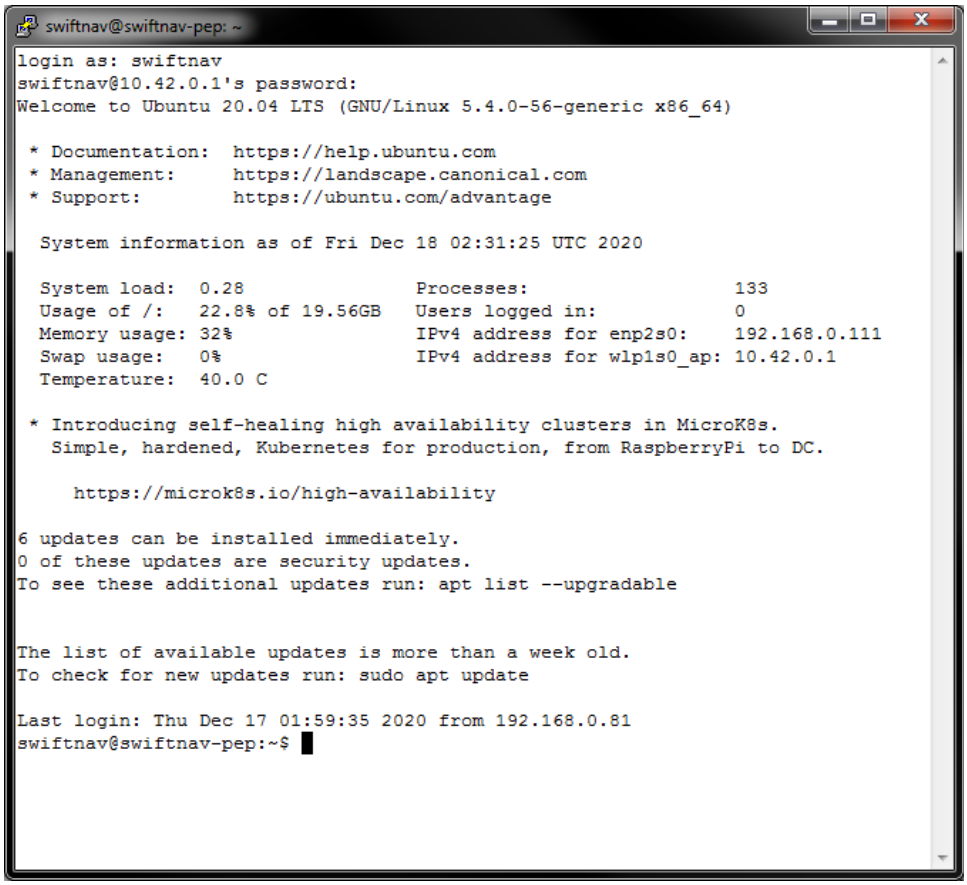
- Open SSH client (eg. PuTTY on Windows) and open connection at IP address **10.42.0.1**:



- Accept server's key (this will show only once):

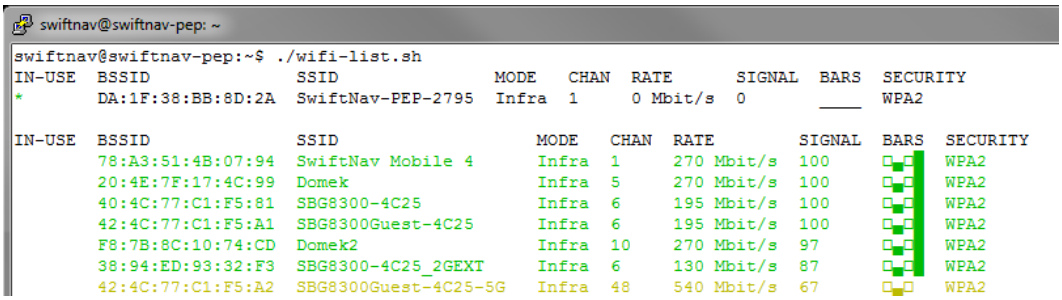


- Login as `swiftnav` and enter password (`swiftnav` by default):



Connecting PGM EVP to the Wi-Fi access point for Internet access

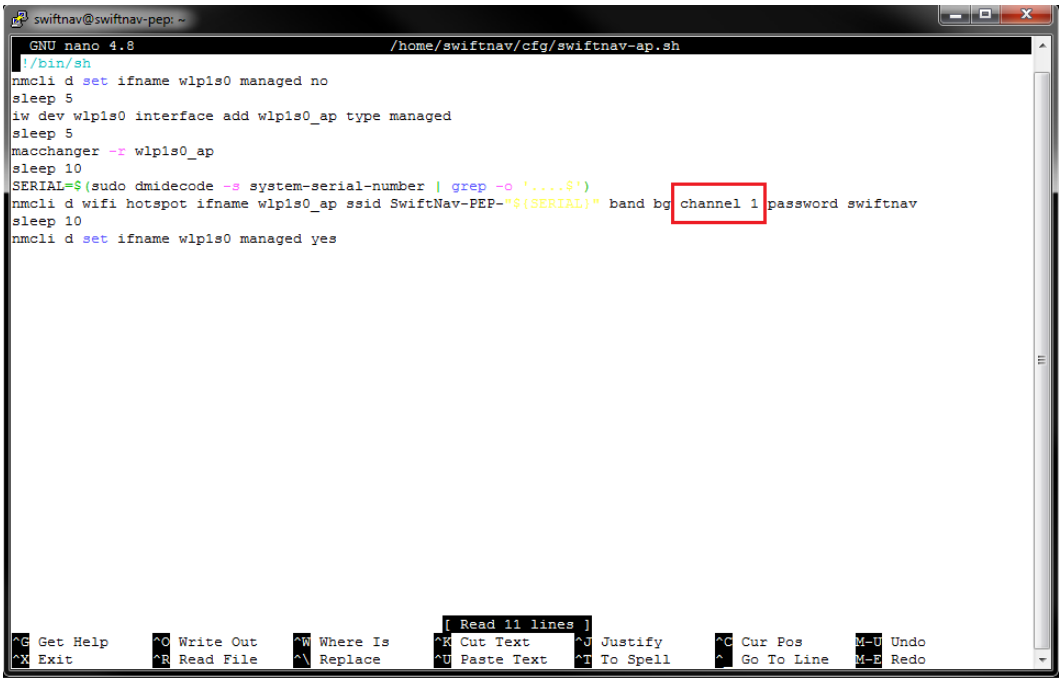
- Use `wifi-list.sh` script to list nearby networks:



- Check for the network to connect (SSID) and note the channel it uses. The Wi-Fi channel must be the same to the one PGM EVP uses for its access point.

  If the PGM EVP's channel is the same as the cell modem's channel, skip this step.

  If the cell modem channel is different from the PGM EVP channel (by default 1) then use `wifi-edit.sh` script to open PGM EVP Wi-Fi configuration file and change the PGM EVP's channel number to match the cell modem:



After changing the channel, restart PGM EVP (power cycle) for changes to take effect.

- Use `wifi-connect.sh` script to connect to the cell modem access point. The script uses two parameters: SSID and password. If any of them contain spaces use double quotes.

```
swiftnav@swiftnav-pep: ~
swiftnav@swiftnav-pep:~$ ./wifi-connect.sh "SwiftNav Mobile 4" centimeter
Device 'wlp1s0' successfully activated with 'f557b1ee-f775-43ab-a487-106fc6b71c1e'.
swiftnav@swiftnav-pep:~$
```

  Command can also be used directly:

  `sudo nmcli d wifi connect <SSID> password <PASSWORD> ifname wlp1s0`

- Run `wifi-list.sh` again to check connection:

```
swiftnav@swiftnav-pep: ~
swiftnav@swiftnav-pep:~$ ./wifi-list.sh
IN-USE  BSSID              SSID               MODE   CHAN  RATE       SIGNAL  BARS  SECURITY
*       DA:1F:38:BB:8D:2A  SwiftNav-PEP-2795  Infra  1     0 Mbit/s   0             WPA2

IN-USE  BSSID              SSID               MODE   CHAN  RATE       SIGNAL  BARS  SECURITY
*       78:A3:51:4B:07:94  SwiftNav Mobile 4  Infra  1     270 Mbit/s 95      ▂▄▆▇  WPA2
swiftnav@swiftnav-pep:~$
```

- A simple test to verify Internet connection is to ping server at IP address 8.8.8.8:

```
swiftnav@swiftnav-pep: ~
swiftnav@swiftnav-pep:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=117 time=40.8 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=117 time=17.5 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=117 time=14.3 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=117 time=17.1 ms
^C
--- 8.8.8.8 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 14.289/22.406/40.777/10.676 ms
swiftnav@swiftnav-pep:~$
```

Configuring and Running Starling

For the best results PGM Evaluation Platform should run the latest software. PGM EVK software updating procedure is described in Appendix A.

Starting from version 1.4 Starling program requires license activation to operate. PGM Evaluation Kits shipped after June 2021 already have the license activated. Older units require license activation after upgrading to Starling v1.4. Refer to Appendix B for detailed license activation instructions.

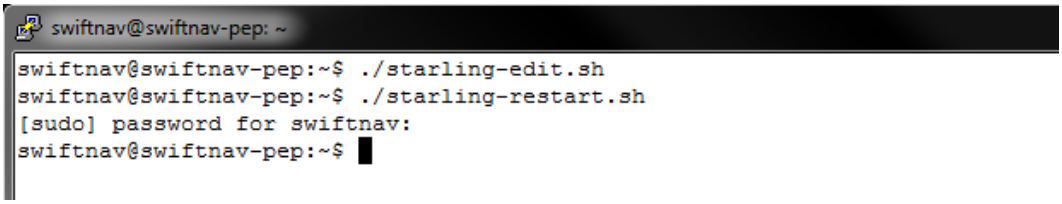
Starling GNSS position engine requires a simple configuration before using it. Configuration is stored in `config.yaml` file in the `starling/` directory. Following items need to be configured:

- Corrections service
- Inertial Fusion settings

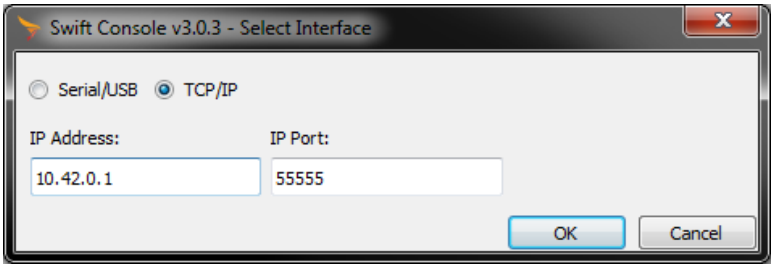
- Use `skylark-edit.sh` to open the Starling configuration file for editing.
- Select Skylark host and update NTRIP credentials as provided by Swift:

- Update device orientation and lever arm. See Appendix C for typical orientations.

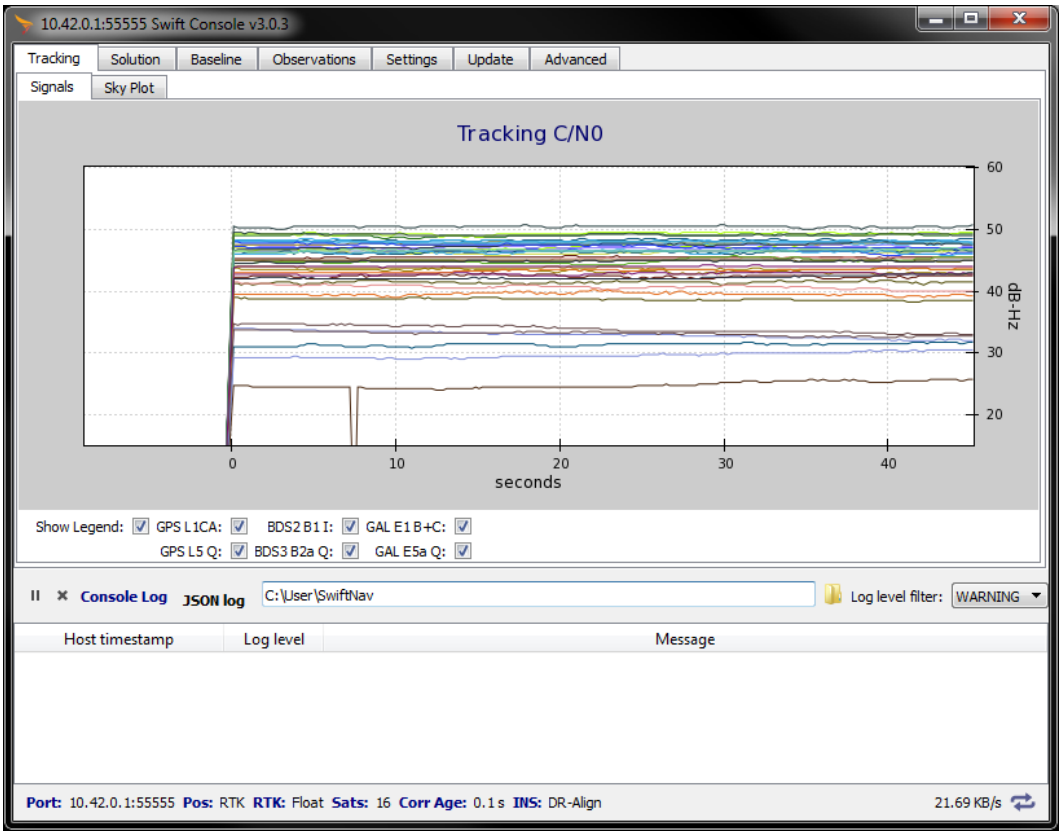
- Save file, close editor and restart Starling service for changes to take effect:



- Open Swift Console to check GNSS operation and to save logs:



Swift Console main window:



- To start logging click the **JSON Log** button on the Swift Console. To stop logging click the same button again.

Appendix A - PGM EVK Software Update

It's recommended to update your evaluation platform to the latest software version to get the latest features and improvements, and for the best navigation performance.

The software update comprises two main components: EVK platform files (which includes the Starling executable) and the PGM firmware.

Software updating takes about 5 minutes to complete (including the PGM firmware update).

Either of the following procedures may be used to update the PGM EVK system to the latest software version.

Option 1: Automated Procedure

1. Run `./check-for-updates.py` script from the home directory. During the execution, the script will compare the current and the latest available versions and ask to download the latest version if it's newer than the current version. After the download the script will also ask to perform the installation.

Option 2: Manual Procedure

1. Download the latest PGM EVK system update tarball from Swift Navigation using one of the following methods:
    - Download directly to the PGM EVK platform with command

      ```
      wget -N -q --show-progress --content-disposition -P
      /home/swiftnav/update https://swiftnav.com/latest/pgm-evp-onlogic-files
      ```

    - Download the update (`pgm-evp-onlogic-files-v1.4.0.tar.gz`) from Swift Navigation Support portal and copy it (SCP) to the new directory, e.g. `/home/swiftnav/update`
2. Untar the file with command `tar -xvf pgm-evp-onlogic-files-v1.4.0.tar.gz`
3. Run `./install-update.sh` script.
    - Note: During execution the script will ask to also update PGM firmware. It's recommended to update the PGM firmware as well.

Warning: the installation script will overwrite all PGM and Starling files. Make a backup copy of modified files to retain your changes.

Note: during the update process the previous Starling configuration file `config.yaml` is renamed to `config-backup-<current time>.yaml`.

Appendix B - Starling License Activation

Starting from version 1.4 Starling program requires license activation to operate.

PGM Evaluation Kits shipped after July 2021 have the license activated already. Older units require license activation after upgrading to Starling v1.4.

The license activation process must be performed only once during the first execution of Starling. In order to activate a licence, the user must obtain a so-called *guard* file and an activation code from Swift Navigation. The device hosting Starling must have access to the Internet to perform the activation procedure.

The license activation procedure is as follows:

1. Obtain a guard file and activation code from Swift Navigation by submitting a support request ticket on the Swift Navigation Support Portal.
    - Note: A user account is required to submit a support request. If you do not have an account yet, click the Support Request button, then click the SIGN UP WITH US button on the login screen, fill in the form, and click REGISTER. After your email is verified, you will be able to submit your support request.
2. Ensure that the hosting platform has access to the Internet.
3. Copy guard file `starling-guard.json` to the `/home/swiftnav/starling` directory.
4. Write the activation code (a 16 digit number in the form XXXX-XXXX-XXXX-XXXX) to the `activation-key.txt` file, e.g. `nano starling/activation-key.txt`
5. Restart Starling service by running `./starling-restart.sh` script.

Upon successful activation, a license file (`/home/swiftnav/starling/license.lic`) will be created and Starling will start. In case of error, an error message will be written to the log file in the `/home/swiftnav/starling/logs` directory.

Appendix C - Orientation Settings

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

X: Forward **Y: Left** Z: 0.0 Y: 0.0 X: 180.0	**X: Right** **Y: Forward** Z: 90.0 Y: 0.0 X: 180.0	**X: Backward** **Y: Right** Z: 180.0 Y: 0.0 X: 180.0	**X: Left** **Y: Backward** Z: -90.0 Y: 0.0 X: 180.0
X: Forward **Y: Right** Z: 0.0 Y: 0.0 X: 0.0	**X: Right** **Y: Backward** Z: -90.0 Y: 0.0 X: 0.0	**X: Backward** **Y: Left** Z: 180.0 Y: 0.0 X: 0.0	**X: Left** **Y: Forward** Z: 90.0 Y: 0.0 X: 0.0
X: Forward **Y: Up** Z: 0.0 Y: 0.0 X: 90.0	**X: Right** **Y: Up** Z: 0.0 Y: 90.0 X: 90.0	**X: Backward** **Y: Up** Z: 180.0 Y: 0.0 X: -90.0	**X: Left** **Y: Up** Z: 0.0 Y: -90.0 X: 90.0
X: Forward **Y: Down** Z: 0.0 Y: 0.0 X: -90.0	**X: Right** **Y: Down** Z: 0.0 Y: -90.0 X: -90.0	**X: Backward** **Y: Down** Z: 180.0 Y: 0.0 X: 90.0	**X: Left** **Y: Down** Z: 0.0 Y: 90.0 X: -90.0
X: Up **Y: Left** Z: 0.0 Y: 90.0 x:-180.0	**X: Up** **Y: Forward** Z: 90.0 Y: 0.0 X: -90.0	**X: Up** **Y: Right** Z: 0.0 Y: -90.0 X: 0.0	**X: Up** **Y: Backward** Z: -90.0 Y: 0.0 X: 90.0
X: Down **Y: Left** Z: 0.0 Y: -90.0 X: 180.0	**X: Down** **Y: Forward** Z: 90.0 Y: 0.0 X: 90.0	**X: Down** **Y: Right** Z: 0.0 Y: 90.0 X: 0.0	**X: Down** **Y: Backward** Z: -90.0 Y: 0.0 X: -90.0

Appendix D - Default Settings

Linux:

Server name: `swiftnav-pep`
Username: `swiftnav`
Password: `swiftnav`

Wi-Fi Access Point:

Band: 2.4 GHz
RF channel: 1
SSID: SwiftNav-PEP-xxxx as printed on the device
Password: `swiftnav`
IP: 10.42.0.1

Wi-Fi Client:

Not configured.

Bluetooth:

Not configured.

Ethernet Port:

Network configuration: DHCP

RS232 Serial Port:

Baud rate: 115200 bps, 8N1.
Flow control: Disabled

Starling:

10 Hz essential SBP messages output on TCP server port 55555
10 Hz all SBP messages output on TCP server port 55556
10 Hz NMEA output on TCP server port 55557
10 Hz NMEA output on RS232 serial port

Skylark:

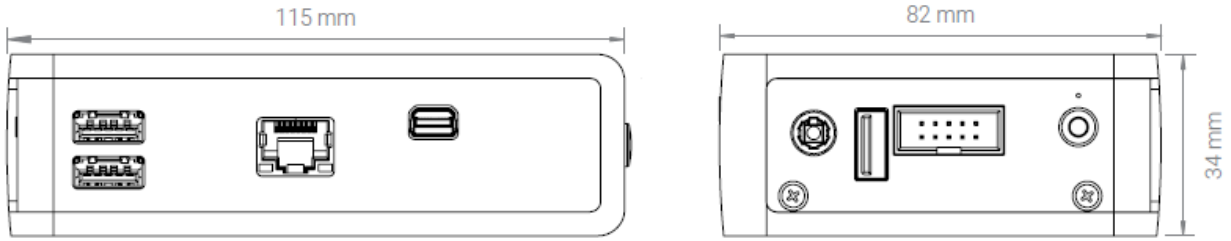
Region: North America
Credentials: Factory with short expiration

Inertial Fusion:

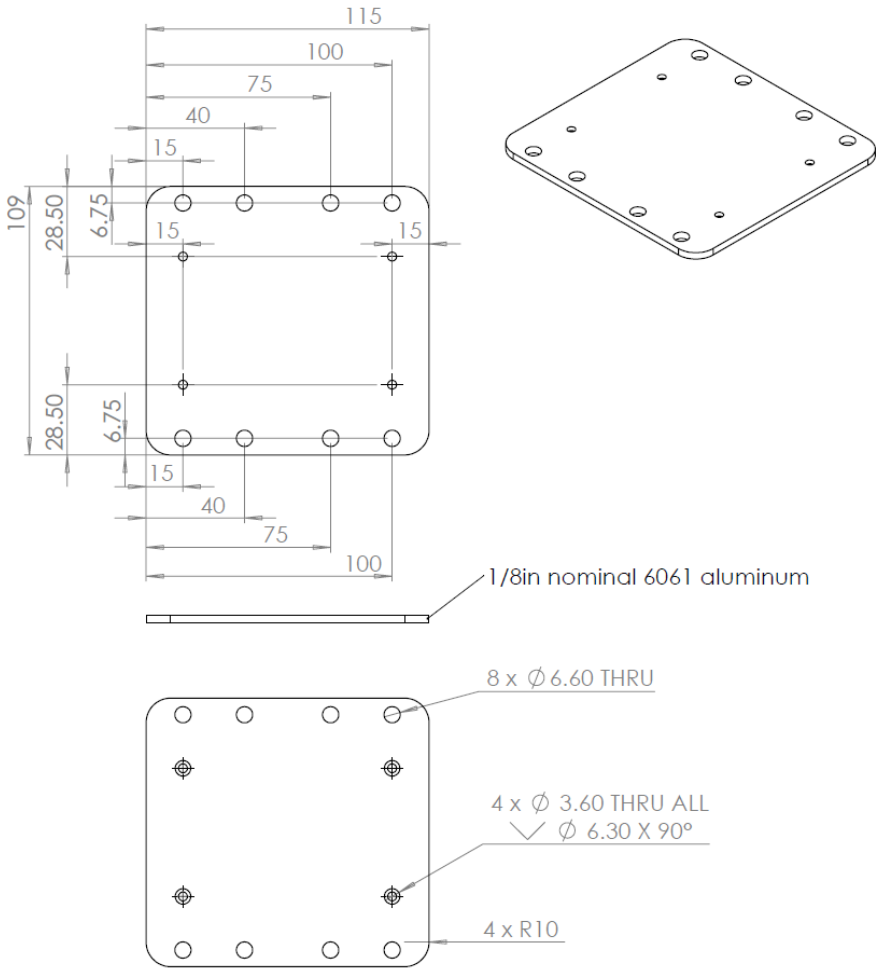
Not configured.

Appendix E - Enclosure Dimensions

PGM EVP Device:



Mounting Bracket:



Dimensions are in mm.

Appendix F - Vehicle Signals

Evaluation kit includes a M8 connector to pigtails cable for optional vehicle signals.

M8 6 Pin Connector	Cable Color	Signal
Pin 1	Black	Ground
Pin 2	Red	Reverse
Pin 3	Green	Vehicle Speed Signal (VSS)
Pin 4	Yellow	CAN Lo
Pin 5	Brown	CAN Hi
Pin 6	Orange	Pulse Per Second (PPS)

Refer to PGM Data Sheet for detailed signal description.

Plug connector type: Hirose HR10A-7P-6S(73).

Appendix G - Control Scripts Content

Following shell scripts are provided for convenience for common operations over SSH:

`eth-edit.sh` - Opens Ethernet configuration file for editing

```
#!/bin/sh
sudo nano /etc/netplan/00-installer-config.yaml
```

`starling-edit.sh` - Opens Starling configuration file for editing

```
#!/bin/sh
nano /home/swiftnav/starling/config.yaml
```

`starling-restart.sh` - Restarts Starling service (required after changing configuration)

```
#!/bin/sh
sudo systemctl restart starling
```

`starling-version.sh` - Displays Starling version

```
#!/bin/sh
/home/swiftnav/starling/starling --version
```

`wifi-connect.sh` - Connects to a Wi-Fi access point. Requires SSID and password params

```
#!/bin/sh
#Usage: ./wifi-connect.sh <SSID> <PASSWORD>
sudo nmcli d wifi connect "$1" password "$2" ifname wlp1s0
```

`wifi-delete.sh` - Deletes (forgets) Wi-Fi network. Requires SSID parameter.

```
#!/bin/sh
#Usage: ./wifi-delete.sh <SSID>
sudo nmcli connection delete "$1"
```

`wifi-disconnect.sh` - Disconnects from Wi-Fi access point

```
#!/bin/sh
sudo nmcli d disconnect wlp1s0
```

`wifi-edit.sh` - Opens Wi-Fi access point configuration file

```
#!/bin/sh
nano /home/swiftnav/cfg/swiftnav-ap.sh
```

`wifi-list.sh` - Scans and lists nearby Wi-Fi networks

```
#!/bin/sh
nmcli d wifi list
```


Appendix H - Additional Resources

Swift Navigation Support Portal

support.swiftnav.com

Starling Specification and Reference Manual

support.swiftnav.com > Products > Starling

PGM Evaluation Kit Software

support.swiftnav.com > General > Downloads > PGM Evaluation Kit Software

Swift Console

support.swiftnav.com > General > Downloads > Swift Console

Onlogic Device

www.onlogic.com/cl200g-11/
www.onlogic.com/computers/industrial/fanless/cl200-series/

Windows Tools

PuTTY - SSH Client

www.putty.org

WinSCP - File Manager

winscp.net

macOS Tools

Transmit - File Manager

panic.com/transmit/

Cyberduck - File Manager

cyberduck.io/

Linux Tools

FileZilla - File Manager

filezilla-project.org/