

SAM-M10Q

Standard precision GNSS antenna module

Professional grade

Integration manual



Abstract

This document describes the features and application of the u-blox SAM-M10Q GNSS antenna module, an ultra-low-power standard precision GNSS receiver with an integrated antenna.

Document information

Title	SAM-M10Q		
Subtitle	Standard precision GNSS antenna module		
Document type	Integration manual		
Document number	UBX-22020019		
Revision and date	R02	01-Jun-2023	
Disclosure restriction	C1-Public		

This document applies to the following products:

Product name	Type number	FW version	IN/PCN reference	RN reference
SAM-M10Q	SAM-M10Q-00B-01	ROM SPG 5.10	UBX-22024039	UBX-22001426

u-blox or third parties may hold intellectual property rights in the products, names, logos and designs included in this document. Copying, reproduction, or modification of this document or any part thereof is only permitted with the express written permission of u-blox. Disclosure to third parties is permitted for clearly public documents only.

The information contained herein is provided "as is" and u-blox assumes no liability for its use. No warranty, either express or implied, is given, including but not limited to, with respect to the accuracy, correctness, reliability and fitness for a particular purpose of the information. This document may be revised by u-blox at any time without notice. For the most recent documents, visit www.u-blox.com.

Copyright © 2023, u-blox AG.

Contents

1 System description.....	6
1.1 Overview.....	6
1.2 Architecture.....	7
1.2.1 Block diagram.....	7
1.3 Pin assignment.....	7
2 Receiver configuration.....	10
2.1 Basic receiver configuration.....	10
2.1.1 Basic hardware configuration.....	10
2.1.2 GNSS signal configuration.....	10
2.1.3 Communication interface configuration.....	11
2.1.4 Message output configuration.....	11
2.1.5 High performance navigation update rate configuration.....	12
2.2 Navigation configuration.....	13
2.2.1 Dynamic platform.....	13
2.2.2 Navigation input filters.....	14
2.2.3 Navigation output filters.....	14
2.2.4 Odometer filters.....	15
2.2.5 Static hold.....	15
2.2.6 Freezing the course over ground.....	17
2.2.7 Super-Signal (Super-S) technology.....	18
2.3 OTP memory configuration.....	18
3 Receiver functionality.....	19
 3.1 Augmentation systems.....	19
3.1.1 SBAS.....	19
3.1.2 QZSS SLAS.....	20
 3.2 Communication interfaces and PIOs.....	21
3.2.1 UART.....	21
3.2.2 I2C.....	22
3.2.3 PIOs.....	25
 3.3 Forcing receiver reset.....	26
 3.4 Security.....	27
3.4.1 Configuration locking.....	28
 3.5 Power management.....	28
3.5.1 Continuous mode.....	28
3.5.2 Power save mode.....	28
3.5.3 Backup modes.....	34
 3.6 Time.....	35
3.6.1 Receiver local time.....	35
3.6.2 GNSS time bases.....	35
3.6.3 Navigation epochs.....	36
3.6.4 iTow timestamps.....	37
3.6.5 Time validity.....	37
3.6.6 UTC representation.....	38

3.6.7 Leap seconds.....	39
3.6.8 Date ambiguity.....	39
3.7 Time mark.....	40
3.8 Time pulse.....	41
3.8.1 Recommendations.....	42
3.8.2 Time pulse configuration.....	43
3.9 Time maintenance.....	44
3.9.1 Real-time clock.....	44
3.9.2 Time assistance.....	44
3.9.3 Frequency assistance.....	45
3.9.4 Clock drift assistance.....	45
3.10 Protection level.....	45
3.10.1 Introduction.....	45
3.10.2 Interface.....	46
3.10.3 Expected behavior.....	47
3.11 Multiple GNSS assistance (MGA).....	47
3.11.1 Authorization.....	48
3.11.2 Preserving MGA and operational data during power-off.....	48
3.11.3 AssistNow offline.....	48
3.11.4 AssistNow autonomous.....	51
3.12 Data batching.....	54
3.12.1 Introduction.....	54
3.12.2 Setting up the data batching.....	54
3.12.3 Retrieval.....	55
3.13 CloudLocate.....	55
3.13.1 CloudLocate measurements.....	56
4 Hardware integration.....	57
4.1 Power supply.....	57
4.1.1 VCC.....	57
4.1.2 V_IO.....	57
4.1.3 V_BCKP.....	57
4.2 RF interference.....	58
4.2.1 In-band interference.....	58
4.2.2 Out-of-band interference.....	58
4.2.3 Spectrum analyzer.....	58
4.3 RF design.....	59
4.3.1 Integrated antenna.....	59
4.3.2 RF front-end.....	61
4.4 Layout.....	61
4.4.1 Package footprint, copper and solder mask.....	63
5 Product handling.....	66
5.1 Safety.....	66
5.1.1 ESD precautions.....	66
5.1.2 Safety precautions.....	66
5.2 Packaging.....	67
5.2.1 Reels.....	67
5.2.2 Tapes.....	67
5.2.3 Moisture sensitivity level.....	68

5.3 Soldering.....	68
Appendix.....	71
A Migration.....	71
A.1 Hardware changes.....	71
A.2 Software changes.....	71
B Reference designs.....	73
B.1 Typical design.....	73
Related documents.....	74
Revision history.....	75

1 System description

This section gives an overview of the SAM-M10Q receiver, and outlines the basics of operation with the receiver.

1.1 Overview

SAM-M10Q patch-antenna module features the u-blox M10 standard precision GNSS platform and provides exceptional sensitivity and acquisition time for all L1 GNSS signals.

The M10 platform supports concurrent reception of four GNSS (GPS, GLONASS, Galileo, and BeiDou). The high number of visible satellites enables the receiver to select the best signals. This maximizes the position availability, in particular under challenging conditions such as in deep urban canyons.

u-blox Super-S (Super-Signal) technology offers great RF sensitivity and can improve the dynamic position accuracy in non-line-of-sight scenarios.

The extremely low power consumption of 37 mW in continuous tracking mode with four concurrent GNSS allows great power autonomy for all battery-operated devices, such as asset trackers, without compromising on GNSS performance.

A front-end SAW filter and an LNA is integrated in the SAM-M10Q module. This setup ensures excellent out-band jamming immunity, for example when a cellular modem is nearby. SAM-M10Q also detects jamming and spoofing attempts and reports them to the host, so that the system can react to such events.

The high-gain 15 x 15 mm² patch antenna provides the best balance between performance and small size. The omnidirectional antenna radiation pattern increases flexibility for device installation.

Incorporating the SAM-M10Q module into customer designs is easy and straightforward, thanks to the integrated antenna, robust design, and simple interface. SAM-M10Q is a surface-mount device, enabling simple and automated manufacturing.

SAM-M10Q offers backwards pin-to-pin compatibility with products from the previous u-blox generations, which saves the designer's effort and reduces costs when upgrading designs to the advanced low-power u-blox M10 GNSS technology.

1.2 Architecture

The SAM-M10Q receiver provides all the necessary RF and baseband processing to enable multi-constellation operation. The block diagram below shows the key functionality.

1.2.1 Block diagram

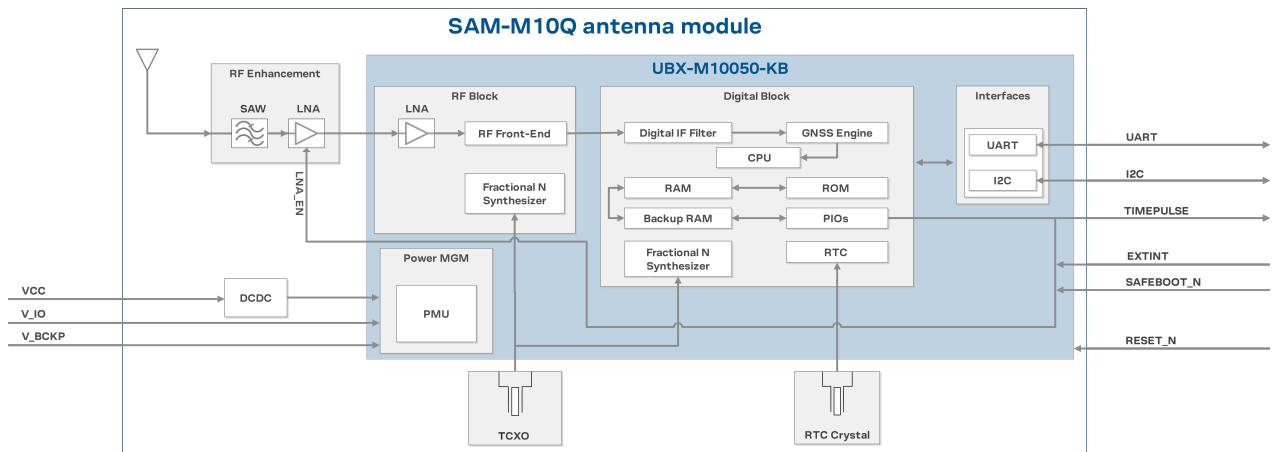


Figure 1: SAM-M10Q block diagram

1.3 Pin assignment

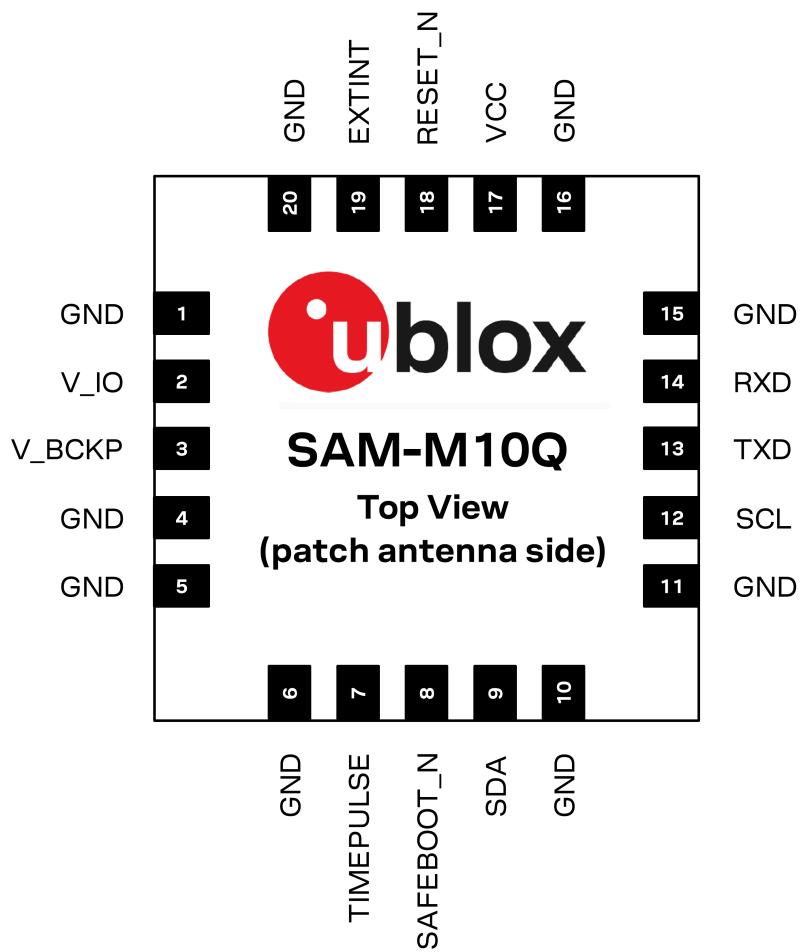


Figure 2: SAM-M10Q pin assignment

Pin no.	Name	PIO no.	I/O	Description
1	GND	-	-	Connect to GND
2	V_IO	-	I	IO voltage supply
3	V_BCKP	-	I	Backup voltage supply
4	GND	-	-	Connect to GND
5	GND	-	-	Connect to GND
6	GND	-	-	Connect to GND
7	TIMEPULSE	4	O	Time pulse signal (shared with SAFEBOOT_N pin) ¹
8	SAFEBOOT_N	-	I	Safeboot mode (leave open) ¹
9	SDA	2	I/O	I2C data
10	GND	-	-	Connect to GND
11	GND	-	-	Connect to GND
12	SCL	3	I	I2C clock
13	TXD	1	O	UART TX

¹ The receiver enters safeboot mode if this pin is low at start up. The SAFEBOOT_N pin is internally connected to TIMEPULSE pin through a 1 kΩ series resistor.

Pin no.	Name	PIO no.	I/O	Description
14	RXD	0	I	UART RX
15	GND	-	-	Connect to GND
16	GND	-	-	Connect to GND
17	VCC	-	I	Main voltage supply
18	RESET_N	-	I	System reset (active low). Has to be low for at least 1 ms to trigger a reset.
19	EXTINT	5	I	External interrupt
20	GND	-	-	Connect to GND

Table 1: SAM-M10Q pin assignment

2 Receiver configuration

The configuration determines all aspects of the GNSS receiver operation and therefore this section is essential reading for the successful integration of the SAM-M10Q.

SAM-M10Q is configured using UBX configuration interface keys. The configuration database in the receiver's RAM holds the current configuration, which is used by the receiver at runtime. It is constructed on startup of the receiver from several sources of configuration. For more information on receiver configuration, see the interface description [3].

A configuration setting stored in the RAM remains effective until power-down or reset. The RAM content is cleared by any of the following:

- Activating the RESET_N pin
- A UBX-CFG-RST message excluding GNSS stop (resetMode 0x08) and GNSS start (resetMode 0x09)
- Entering software standby mode
- Using external control in power save mode (PSM) to enter the inactive state
- Entering the off state of the PSM on/off operation (PSMOO)

It is therefore recommended to apply runtime configuration on both RAM and battery-backed RAM (BBR) layers.

The configuration stored in BBR is also cleared by a RESET_N signal or a UBX-CFG-RST message with reset mode set to a hardware reset (resetMode 0x00 and 0x04), but otherwise will be used as long as the backup battery supply remains.

 **CAUTION** The configuration interface has changed from earlier u-blox positioning receivers. Users must adopt the configuration interface described in this document.

The configuration interface settings are stored in a database consisting of separate configuration items. An item is made up of a pair consisting of a key ID and a value. Related items are grouped together and identified under a common group name: CFG-GROUP-*; a convention used in u-center 2 and within this document. Within u-center 2, a configuration group is identified as "Group name" and the configuration item is identified as the "item name" in the "Device configuration" window.

The UBX messages available to change or poll the configurations are the UBX-CFG-VALSET, UBX-CFG-VALGET, and UBX-CFG-VALDEL messages. For more information about these messages and the configuration keys, see the configuration interface section in the interface description [3].

2.1 Basic receiver configuration

This section summarizes the basic receiver configuration most commonly used.

2.1.1 Basic hardware configuration

The SAM-M10Q receiver is configured with the default setting during the module production. The receiver starts up and is fully operational as soon as power supplies and a communication interface are connected.

2.1.2 GNSS signal configuration

SAM-M10Q supports concurrent reception of four major GNSS constellations using the GPS L1C/A, Galileo E1, BeiDou B1C, and GLONASS L1OF signals. The default configuration is concurrent reception of GPS, Galileo, GLONASS, and BeiDou B1C with QZSS and SBAS enabled.

GNSS constellations and signals can be configured using the CFG-SIGNAL-* configuration group. Each GNSS constellation can be enabled or disabled independently except for QZSS and SBAS, which are functional only with GPS. In addition to the configuration key for each constellation, there is a configuration key for each signal supported by the firmware. Unsupported combinations are rejected with a UBX-ACK-NAK message, and the warning "inv sig cfg" is sent via UBX-INF and NMEA-TXT messages (if enabled).

For example, if CFG-SIGNAL-GPS_ENA is set to zero, all signals from the GPS constellation are disabled. Alternatively, if CFG-SIGNAL-GPS_L1CA_ENA is set to zero, only the GPS L1 C/A signal is disabled.

- ☞ Any change to the signal configuration items triggers a restart of the GNSS subsystem. This takes a short time period, so the host application should wait for message acknowledgement and a margin of 0.5 seconds prior to sending any further commands.
- ☞ It is recommended to enable QZSS L1C/A when GPS L1C/A is enabled in order to mitigate possible cross-correlation issues between the signals.

Refer to the Interface description [3] for more information on the CFG-SIGNAL-* configuration group.

2.1.3 Communication interface configuration

Several configuration groups allow configuring the operation mode of the communication interfaces. These include parameters for the data framing, transfer rate and enabled input/output protocols. See [Communication interfaces and PIOs](#) section for details. The configuration groups available for each interface are:

Interface	Configuration groups
UART	CFG-UART1-*, CFG-UART1INPROT-*, CFG-UART1OUTPROT-*
I2C	CFG-I2C-*, CFG-I2CINPROT-*, CFG-I2COUTPROT-*, CFG-TXREADY-*

Table 2: Interface configuration

2.1.4 Message output configuration

The receiver supports two protocols for output messages: industry-standard NMEA and u-blox UBX. Any message type can be enabled or disabled individually and the output rate is configurable.

The message output rate is related to the frequency of an event. For example, the output message UBX-NAV-PVT (position, velocity, and time solution) is related to the navigation event, which generates a navigation epoch. In this case, the rate for each navigation epoch is defined by the configuration keys CFG-RATE-MEAS and CFG-RATE-NAV. For example, a value of 1000 ms in CFG-RATE-MEAS indicates that a measurement is done every second. If CFG-RATE-NAV is set to one (1), the solution is calculated for every measurement. This means that a navigation epoch is calculated every 1000 ms. If the rate is set to two (2), only the second measurement is used and the navigation epoch is calculated every two seconds. The same result is obtained if CFG-RATE-MEAS is set to 2000 ms, and CFG-RATE-NAV is set to one (1). Every 2000 ms a measurement is done, and in every measurement, a navigation epoch is calculated. However, this second option demands fewer resources and is the correct procedure when the navigation rate is changed. Setting a navigation rate value higher than one (1) is only needed when it is required that the raw measurement data is output at a higher rate than the navigation data.

The output rate for each message is defined in the CFG-MSGOUT-* configuration group. If the output rate of the message is set to one (1) on the UART interface, CFG-MSGOUT-UBX_NAV_PVT_UART1 = 1, the message is output for every navigation epoch. If the rate is set to two (2), the message is output for every other navigation epoch. If the rate is zero (0), then corresponding

message is not output. As seen in this example, the rates of the output messages are individually configurable per communication interface.

Some messages, such as UBX-MON-VER, are non-periodic and are only output as an answer to a poll request.

The UBX-INF-* and NMEA-Standard-TXT information messages are non-periodic output messages that do not have a message rate configuration. Instead they can be enabled for each communication interface via the CFG-INFMSG-* configuration group.

-  All message output is additionally subject to the protocol configuration of the communication interfaces. Messages of a given protocol are not output unless the protocol is enabled for output on the interface. See [Communication interface configuration](#) for details.

2.1.5 High performance navigation update rate configuration

The navigation update rate is a GNSS configuration item which determines how many position fixes the receiver calculates and outputs per second. The maximum achievable navigation update rate depends on the number of GNSS signals the receiver is tracking. Depending on the geographical location and GNSS constellations enabled, the number of tracking channels in use may vary. In particular, in Asia there is a large number of BeiDou satellites available potentially limiting the maximum achievable navigation update rate. In regions with a lower number of available BeiDou satellites, a higher navigation rate is achieved.

The maximum navigation update rate given in the datasheet is provided for a minimum of 98% position fix rate, i.e., up to 2% position fixes can get lost in case of high availability of signals. The navigation update rate can be increased beyond the maximum value stated in the datasheet. However, this may result in a reduced fix rate if a very large number of satellites is tracked.

u-blox M10 devices are optimized for low power consumption and come with the default CPU clock rate that supports the default navigation update rate stated in the product datasheet. However, it is possible to achieve a higher navigation update rate by configuring the device for a higher clock rate. This supports the high performance navigation update rate with minor increase in power consumption.

-  For the high navigation update rates, increase the communication speed and reduce the number of enabled messages.

The high performance navigation update rate can be configured in the device's one-time programmable (OTP) memory. The OTP configuration is only done once, and is subsequently applied automatically at every startup. The configuration string for setting the high CPU clock rate in the OTP memory is given in [Table 3](#). This occupies 18 bytes of OTP memory space.

-  Changes made in the OTP configuration are permanent and cannot be reverted.

CPU clock	Configuration string
Default CPU clock	Default
High CPU clock	B5 62 06 41 10 00 03 00 04 1F 54 5E 79 BF 28 EF 12 05 FD FF FF 8F 0D B5 62 06 41 1C 00 04 01 A4 10 BD 34 F9 12 28 EF 12 05 05 00 A4 40 00 B0 71 0B 0A 00 A4 40 00 D8 B8 05 DE AE

Table 3: High performance navigation update rate configuration in OTP memory

To configure the high performance navigation update rate in OTP memory:

1. Power up the device.
2. Test the communication interface by polling the UBX-MON-VER message.

3. Send the configuration string provided in [Table 3](#). The device returns two UBX-ACK-ACK messages with sequence of bytes B5 62 05 01 02 00 06 41 4F 78.
4. Set the device reset to a hardware reset. Power the device off and on or send the UBX-CFG-RST message to the device. The higher clock setting is applied at startup.
5. To verify that the configuration item is correctly set,
 - send the following sequence to the receiver: B5 62 06 8B 14 00 00 04 00 00 01 00 A4 40 03 00 A4 40 05 00 A4 40 0A 00 A4 40 4C 15
 - receive a UBX-CFG-VALGET message with the following sequence: B5 62 06 8B 24 00 01 04 00 00 01 00 A4 40 00 B0 71 0B 03 00 A4 40 00 B0 71 0B 05 00 A4 40 00 B0 71 0B 0A 00 A4 40 00 D8 B8 05 76 81
 - receive a UBX-ACK-ACK message with the following sequence: B5 62 05 01 02 00 06 8B 99 C2
6. The OTP memory configuration is completed and verified.

2.2 Navigation configuration

This section presents various configuration options related to the navigation engine. These options can be configured through CFG-NAVSPG-* configuration keys.

2.2.1 Dynamic platform

The dynamic platform model can be configured through the CFG-NAVSPG-DYNMODEL configuration item. For the supported dynamic platform models and their details, see [Table 4](#) and [Table 5](#).

Platform	Description
Portable	Applications with low acceleration, e.g. portable devices. Suitable for most situations.
Stationary	Used in timing applications (antenna must be stationary) or other stationary applications. Velocity restricted to 0 m/s. Zero dynamics assumed.
Pedestrian	Applications with low acceleration and speed, e.g. how a pedestrian would move. Low acceleration assumed.
Automotive	Used for applications with equivalent dynamics to those of a passenger car. Low vertical acceleration assumed.
At sea	Recommended for applications at sea, with zero vertical velocity. Zero vertical velocity assumed. Sea level assumed.
Airborne <1g	Used for applications with a higher dynamic range and greater vertical acceleration than a passenger car. No 2D position fixes supported.
Airborne <2g	Recommended for typical airborne environments. No 2D position fixes supported.
Airborne <4g	Only recommended for extremely dynamic environments. No 2D position fixes supported.
Wrist	Only recommended for wrist-worn applications. Receiver will filter out arm motion.

Table 4: Dynamic platform models

Platform	Max altitude [m]	Max horizontal velocity [m/s]	Max vertical velocity [m/s]	Sanity check type	Max position deviation
Portable	12000	310	50	Altitude and velocity	Medium
Stationary	9000	10	6	Altitude and velocity	Small
Pedestrian	9000	30	20	Altitude and velocity	Small
Automotive	6000	100	15	Altitude and velocity	Medium
At sea	500	25	5	Altitude and velocity	Medium
Airborne <1g	80000	100	6400	Altitude	Large

Platform	Max altitude [m]	Max horizontal velocity [m/s]	Max vertical velocity [m/s]	Sanity check type	Max position deviation
Airborne <2g	80000	250	10000	Altitude	Large
Airborne <4g	80000	500	20000	Altitude	Large
Wrist	9000	30	20	Altitude and velocity	Medium

Table 5: Dynamic platform model details

Applying dynamic platform models designed for high acceleration systems (e.g. airborne <2g) can result in a higher standard deviation in the reported position.

If a sanity check against a limit of the dynamic platform model fails, then the position solution becomes invalid. [Table 5](#) shows the types of sanity checks which are applied for a particular dynamic platform model.

2.2.2 Navigation input filters

The navigation input filters in the CFG-NAVSPG-* configuration group control how the navigation engine handles the input data that comes from the satellite signal.

Configuration item	Description
CFG-NAVSPG-FIXMODE	By default, the receiver calculates a 3D position fix if possible but reverts to 2D position if necessary (auto 2D/3D). The receiver can be forced to only calculate 2D (2D only) or 3D (3D only) positions.
CFG-NAVSPG-CONSTR_ALT, CFG-NAVSPG-CONSTR_ALTVAR	The fixed altitude is used if fixMode is set to 2D only. A variance greater than zero must also be supplied.
CFG-NAVSPG-INFIL_MINELEV	Minimum elevation of a satellite above the horizon to be used in the navigation solution. Low-elevation satellites may provide degraded accuracy, due to the long signal path through the atmosphere.
CFG-NAVSPG-INFIL_MINSVS, CFG-NAVSPG-INFIL_MAXSVS	Minimum and maximum number of satellites to use in the navigation solution. There is an absolute maximum limit of 32 satellites that can be used for navigation.
CFG-NAVSPG-INFIL_NCNOTHRS, CFG-NAVSPG-INFIL_CNOTHRS	A navigation solution will only be attempted if there is at least the given number of satellites with signals at least as strong as the given threshold.

Table 6: Navigation input filter parameters

If the receiver has only three satellites for calculating a position, the navigation algorithm uses a constant altitude to compensate for the missing fourth satellite. This is called a 2D fix. The constant altitude value is taken from the last successful 3D fix using a minimum of four available satellites.

u-blox receivers do not calculate any navigation solution with fewer than three satellites.

2.2.3 Navigation output filters

The result of a navigation solution is initially classified by the fix type (as detailed in the `fixType` field of UBX-NAV-PVT message). This distinguishes between failures to obtain a fix at all ("No Fix") and cases where a fix has been achieved, which are further subdivided into specific types of fixes (for example, 2D, 3D).

Where a fix has been achieved, the fix is checked to determine whether it is valid or not. A fix is only valid if it passes the navigation output filters as defined in CFG-NAVSPG-OUTFIL. In particular, both PDOP and accuracy values must be below the respective limits.

Important: Users are recommended to check the `gnssFixOK` flag in the UBX-NAV-PVT or the NMEA valid flag. Fixes not marked as valid should not be used.

UBX-NAV-STATUS message also reports whether a fix is valid in the `gpsFixOK` flag. These messages have only been retained for backwards compatibility and it is recommended to use the UBX-NAV-PVT message.

2.2.4 Odometer filters

2.2.4.1 Speed (3D) low-pass filter

The CFG-ODO-OUTL�VEL configuration item activates a speed (3D) low-pass filter. The output of the speed low-pass filter is available in the UBX-NAV-VELNED message (`speed` field). The filtering level can be set via the CFG-ODO-VELLPGAIN configuration item and must be between 0 (heavy low-pass filtering) and 255 (weak low-pass filtering).

- ☞ The internal filter gain is computed as a function of speed. Therefore, the level as defined in the CFG-ODO-VELLPGAIN configuration item defines the nominal filtering level for speeds below 5 m/s.

2.2.4.2 Course over ground low-pass filter

The CFG-ODO-OUTLPCOG configuration item activates a course over ground low-pass filter when the speed is below 8 m/s. The output of the course over ground (also named heading of motion 2D) low-pass filter is available in the UBX-NAV-PVT message (`headMot` field), UBX-NAV-VELNED message (`heading` field), NMEA-RMC message (`cog` field), and NMEA-VTG message (`cogt` field). The filtering level can be set via the CFG-ODO-COGLPGAIN configuration item and must be between 0 (heavy low-pass filtering) and 255 (weak low-pass filtering).

- ☞ The filtering level as defined in the CFG-ODO-COGLPGAIN configuration item defines the filter gain for speeds below 8 m/s. If the speed is 8 m/s or higher, no course over ground low-pass filtering is performed.

2.2.4.3 Low-speed course over ground filter

The CFG-ODO-USE_COG configuration item activates this feature and the CFG-ODO-COGMAXSPEED, CFG-ODO-COGMAXPOSACC configuration items are used to configure a low-speed course over ground filter (also named heading of motion 2D). This filter derives the course over ground from position at very low speed. The output of the low-speed course over ground filter is available in the UBX-NAV-PVT message (`headMot` field), UBX-NAV-VELNED message (`heading` field), NMEA-RMC message (`cog` field) and NMEA-VTG message (`cogt` field). If the low-speed course over ground filter is not configured, then the course over ground is computed as described in section [Freezing the course over ground](#).

2.2.5 Static hold

Static hold mode allows the navigation algorithms to decrease the noise in the position output when the velocity is below a predefined "Static Hold Threshold" level. This reduces the position wander caused by environmental factors such as multi-path and improves position accuracy especially in stationary applications. By default, static hold mode is disabled.

The CFG-MOT-GNSSSPEED_THRS configuration item defines the static hold speed threshold. If the speed drops below the defined "Static Hold Threshold", static hold mode is activated. Once static hold mode is active, the position output is kept static and the velocity is set to zero until there is evidence of movement again. Such evidence can be velocity, acceleration, changes of the valid flag (for example, position accuracy estimate exceeding the position accuracy mask, see also section [Navigation output filters](#)), position displacement, etc.

The CFG-MOT-GNSSDIST_THRS configuration item defines the static hold distance threshold. If the distance between the estimated position and the static hold position exceeds the defined threshold, static hold mode is suspended or deactivated until there is evidence of no movement.

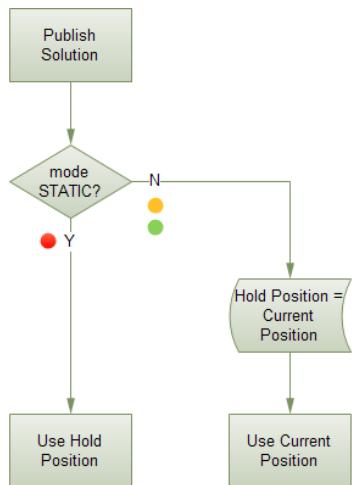


Figure 3: Position output in static hold mode

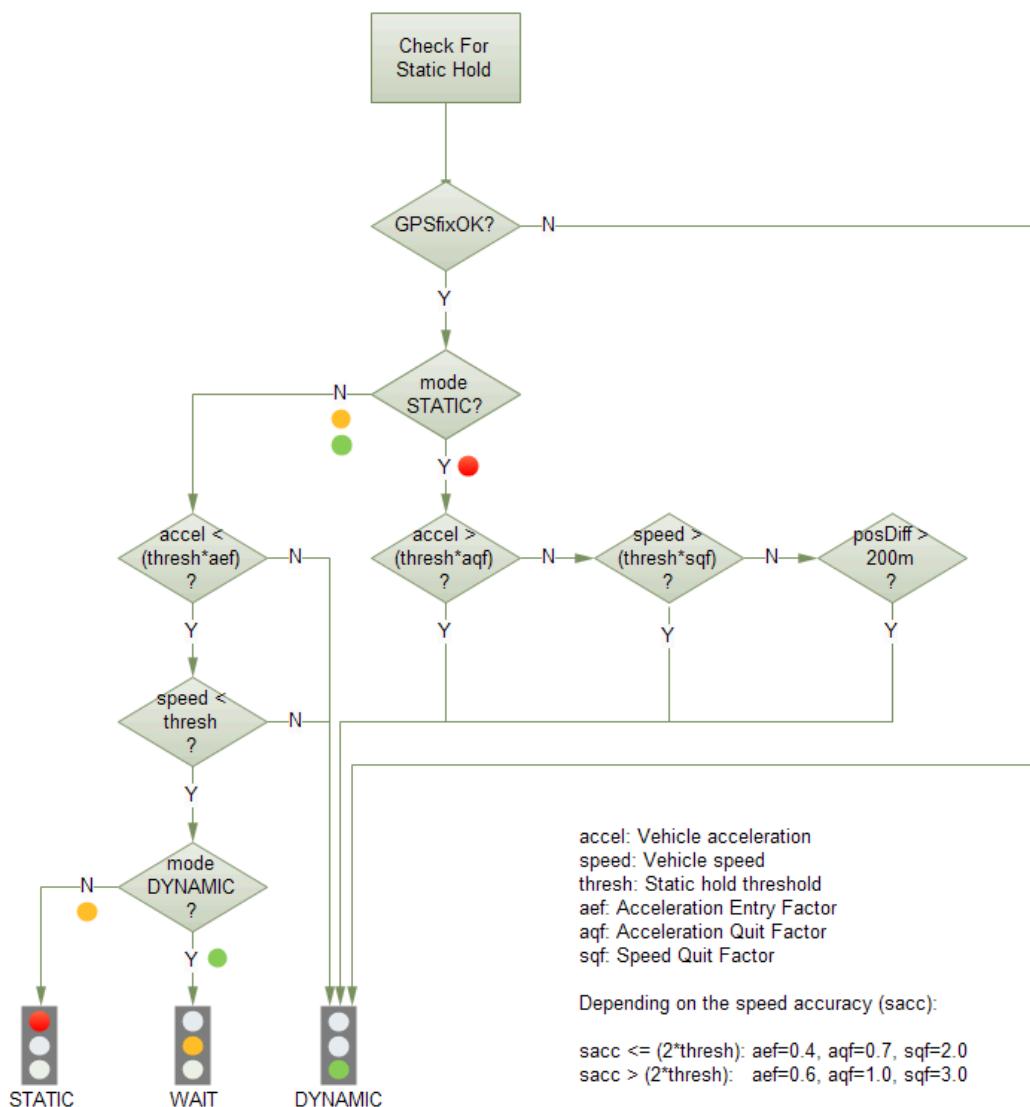


Figure 4: Flowchart of static hold mode

2.2.6 Freezing the course over ground

If the low-speed course over ground filter is deactivated or inactive (see section [Low-speed course over ground filter](#)), the receiver derives the course over ground from the GNSS velocity information. If the velocity cannot be calculated with sufficient accuracy (for example, with bad signals) or if the absolute speed value is very low (under 0.1 m/s) then the course over ground value becomes inaccurate too. In this case the course over ground value is frozen, that is, the previous value is kept and its accuracy degrades over time. These frozen values will not be output in the NMEA messages NMEA-RMC and NMEA-VTG unless the NMEA protocol is explicitly configured to do so (see NMEA protocol configuration in the applicable interface description [3]).

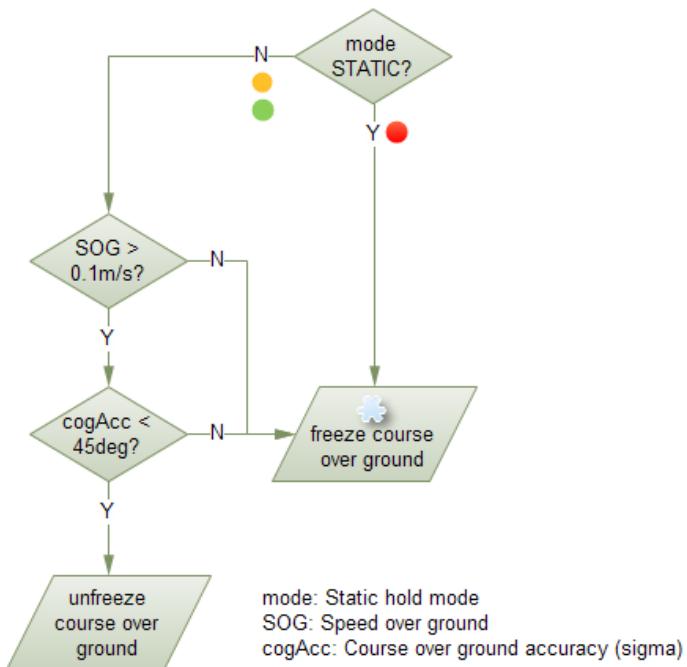


Figure 5: Flowchart of course over ground freezing

2.2.7 Super-Signal (Super-S) technology

In normal operating conditions, low signal strength (that is, signal attenuation) indicates possible degradation due to multi-path. The receiver trusts such signals less in order to preserve the quality of the position solution in poor signal environments. This feature can result in degraded performance in situations where the signals are attenuated for another reason, for example due to antenna placement. In this case, the weak signal compensation feature can be used to restore normal performance. There are three possible modes:

- Disabled: no weak signal compensation is performed
- Automatic: the receiver automatically estimates and compensates for the weak signal
- Configured: the receiver compensates for the weak signal based on a configured value

These modes can be selected using CFG-NAVSPG-SIGATTCOMP. In the case of the "configured" mode, the user should input the maximum C/N0 observed in a clear-sky environment, excluding any outliers or unusually high values. Choose the configured value carefully, as it can have a large impact on the receiver performance.

2.3 OTP memory configuration

SAM-M10Q contains a one-time programmable (OTP) memory. This is a non-volatile memory for storing configuration settings and ROM patches permanently in the device. The stored data cannot be modified after it has been initially programmed. The device applies the settings and ROM patches on the device startup.

As the space in the OTP memory is limited, only essential system configuration settings should be stored. The total space used for device configuration and ROM patches in the OTP memory must not exceed 69 bytes. Other settings can be stored in the BBR or sent from the host to the device on each device startup.

- Ensure that the final configuration stored and optional ROM patches do not require more than 69 bytes of OTP memory space.

3 Receiver functionality

This section provides a description of the receiver's functionality.

3.1 Augmentation systems

3.1.1 SBAS

SAM-M10Q is capable of receiving multiple SBAS signals concurrently, even from different SBAS systems (WAAS, EGNOS, etc.). SBAS signals are recommended to be used only for correction data. SBAS signals can also be used for navigation, however they have low weighting and therefore only a minor impact on the navigation solution.

For receiving correction data, SAM-M10Q automatically chooses the best SBAS satellite as its primary source. It will select only one since the information received from other SBAS satellites is redundant and could be inconsistent. The selection strategy is determined by the proximity of the satellites, the services offered by the satellite, the configuration of the receiver (test mode allowed/disallowed, integrity enabled/disabled) and the signal link quality to the satellite.

If corrections are available from the chosen SBAS satellite and used in the navigation calculation, the differential status will be indicated in several output messages such as UBX-NAV-PVT, UBX-NAV-STATUS, UBX-NAV-SAT, NMEA-GGA, NMEA-GLL, NMEA-RMC, and NMEA-GNS. The UBX-NAV-SBAS message provides detailed information about which corrections are available and applied. Refer to the interface description [3] for a detailed description of the messages.

The most important SBAS feature for accuracy improvement is the ionosphere correction parameters. The measured data from regional Ranging and Integrity Monitoring Stations (RIMS) are combined to make a Total Electron Content (TEC) map. This map is transferred to the receiver via SBAS satellites to allow a correction of the ionosphere delays on each received signal.

Message type	Message content	Source
0(0/2)	Test mode	All
1	PRN mask assignment	Primary
2, 3, 4, 5	Fast corrections	Primary
6	Integrity	Primary
7	Fast correction degradation	Primary
9	Satellite navigation (ephemeris)	All
10	Degradation	Primary
12	Time offset	Primary
17	Satellite almanac	All
18	Ionosphere grid point assignment	Primary
24	Mixed fast / long-term corrections	Primary
25	Long-term corrections	Primary
26	Ionosphere delays	Primary

Table 7: Supported SBAS messages

Each satellite serves a specific region and its correction signal is only useful within that region. Planning is crucial to determine the best possible configuration, especially in areas where signals from different SBAS systems can be received:

- **Example 1 - SBAS receiver in North America:** In eastern parts of North America, make sure that EGNOS satellites do not take preference over WAAS satellites. The satellite signals from the EGNOS system should be disallowed by using the PRN scan mask (configuration key CFG-SBAS-PRNSCANMASK).
 - **Example 2 - SBAS receiver in Europe:** Some WAAS satellite signals can be received in western parts as well as GAGAN SBAS satellites in some parts of Europe, therefore it is recommended that the satellites from all but the EGNOS system should be disallowed using the PRN scan mask.
-  Although u-blox receivers try to select the best available SBAS correction data, it is recommended to configure them to exclude unwanted SBAS satellites.

To configure the SBAS functionality, use the CFG-SBAS-* configuration group.

Parameter	Description
CFG-SIGNAL-SBAS_ENA	Enabled/disabled status of the SBAS subsystem
CFG-SBAS-USE_TESTMODE	Allow/disallow SBAS usage from satellites in test mode
CFG-SBAS-USE_RANGING	Use the SBAS satellites for navigation (ranging)
CFG-SBAS-USE_DIFFCORR	Combined enable/disable switch for fast, long-term, and ionosphere corrections
CFG-SBAS-USE_INTEGRITY	Apply integrity information data
CFG-SBAS-PRNSCANMASK	Allows selectively enabling/disabling SBAS satellites

Table 8: SBAS configuration parameters

-  When SBAS integrity data is applied, the navigation engine stops using all signals for which no integrity data is available (including all non-GPS signals). It is not recommended to enable SBAS integrity on borders of SBAS service regions in order not to inadvertently restrict the number of available signals.
-  SBAS integrity information is required for at least five GPS satellites. If this condition is not met, SBAS integrity data will not be applied.
-  When the receiver switches from a solution using correction data to a standard position solution, the reference frame of the output position will switch from that of the correction data to that of the standard position solution. For an SBAS solution, this reference frame will be aligned within a few cm of WGS84 (and modern ITRF realizations).

3.1.2 QZSS SLAS

QZSS SLAS (Sub-meter Level Augmentation Service) is an augmentation technology, which provides correction data for pseudoranges of GPS, QZSS, and other major GNSS satellites. The correction stream is transmitted on the L1S signal at the L1 frequency (1575.42 MHz).

For more information on QZSS SLAS, visit qzss.go.jp/en/.

Multiple QZSS SLAS signals can be received simultaneously. When receiving QZSS SLAS correction data, SAM-M10Q will autonomously select the best QZSS satellite. The selection strategy is determined by the quality of the QZSS L1S signals, the receiver configuration (test mode allowed or not), and the location of the receiver with respect to the QZSS SLAS coverage area. When outside of this coverage area, the receiver will likely fall back to using SBAS corrections.

If QZSS SLAS corrections are used in the navigation solution, the differential status will be indicated in several output messages such as UBX-NAV-PVT, UBX-NAV-STATUS, UBX-NAV-SAT, NMEA-GGA, NMEA-GLL, NMEA-RMC, and NMEA-GNS. The UBX-NAV-SLAS message provides detailed information about which corrections are available and applied. Refer to the interface description [3] for a detailed description of the messages.

Message type	Message content
0	Test mode
47	Monitoring station information
48	PRN mask
49	Data issue number
50	DGPS correction
51	Satellite health

Table 9: Supported QZSS L1S SLAS messages for navigation enhancement

Use the configuration key CFG-SIGNAL-QZSS_L1S_ENA to enable QZSS L1S signal. For further QZSS SLAS functionality, use the CFG-QZSS-USE_SLAS* configuration keys.

Parameter	Description
CFG-QZSS-USE_SLAS_DGNSS	Apply QZSS SLAS corrections
CFG-QZSS-USE_SLAS_TESTMODE	Allow the correction provided by QZSS satellites that are in test mode
CFG-QZSS-USE_SLAS_RAIM_UNCORR	If this configuration is set, the receiver will try to estimate the position by using only corrected measurements; if all corrected measurements are not available, it will not use any corrections. If this configuration is not set, the receiver will mix corrected and uncorrected measurements for the navigation solution.

Table 10: QZSS SLAS configuration parameters

- ☞ If the RAIM option is set, QZSS is the only GNSS time system that measurements can observe.

3.2 Communication interfaces and PIOs

SAM-M10Q supports communication over UART and I2C interfaces for communication with a host CPU. Each protocol can be enabled on several interfaces at the same time with individual settings for, for example, baud rate, message rates, and so on. In SAM-M10Q, several protocols can be enabled on a single interface at the same time.

3.2.1 UART

SAM-M10Q supports a Universal Asynchronous Receiver/Transmitter (UART) port consisting of an RX and a TX line. The UART can be used as a host interface which supports a configurable baud rate and protocol selection.

- ☞ Neither handshaking signals nor hardware flow control signals are supported.

The UART baud rate can be configured for selected speeds. Different rates than these speeds are not supported for transmission and reception.

- ☞ The UART RX interface will be disabled when more than 100 frame errors are detected during a one-second period. This can happen if the wrong baud rate is used or the UART RX pin is grounded. An error message appears when the UART RX interface is re-enabled at the end of the one-second period.

Baud rate	Data bits	Parity	Stop bits
4800	8	none	1
9600	8	none	1
19200	8	none	1
38400	8	none	1
57600	8	none	1
115200	8	none	1

Baud rate	Data bits	Parity	Stop bits
230400	8	none	1
460800	8	none	1
921600	8	none	1

Table 11: Possible UART interface configurations

Allow a short time delay of typically 100 ms between sending a baud rate change message and providing input data at the new rate. Otherwise some input characters may be ignored or the port could be disabled until the interface is able to process the new baud rate.

If there is too much data for the interface's bandwidth, the output buffer will fill up. Once the buffer space is exceeded, new messages to be sent will be dropped. To prevent message loss, the baud rate and the number of enabled messages should be selected carefully.

3.2.2 I2C

An I2C interface is available for communication with an external host CPU in I2C Fast-mode. Backwards compatibility with Standard-mode I2C bus operation is not supported. The interface can be operated only in slave mode with a maximum bit rate of 400 kbit/s. The interface can make use of clock stretching by holding the SCL line LOW to pause a transaction. In this case, the bit transfer rate is reduced. The maximum clock stretching time is 20 ms.

The SCL and SDA pins have internal pull-up resistors which should be sufficient for most applications. However, depending on the clock speed of the host and the capacitive load on the I2C lines, additional external pull-up resistors may be necessary. The higher the speed and the capacitance load, the lower the pull-up resistor needs to be.

To poll or set the I2C address, use the CFG-I2C-ADDRESS configuration item (see the u-blox M10 interface description [3]). The CFG-I2C-ADDRESS configuration item is an 8-bit value containing the I2C address in the 7 most significant bits plus a 0 as the least significant bit. Thus, the default address becomes 0x84(1000 0100).

-  In designs where the host uses the same I2C bus to communicate with more than one u-blox receiver, each receiver's I2C address must be configured with a different value.

3.2.2.1 I2C register layout

As shown in Figure 6, there are 256 registers. The data registers 0 to 252, at addresses 0x00 to 0xFC, contain reserved information and must not be used. Hence, only the last three registers are left for communication. The registers 0xFD and 0xFE contain the currently available number of bytes to be read, while the register 0xFF buffers the message stream. The 0xFF address delivers a 0xFF byte value if there is no data awaiting for transmission, or all the bytes have been read.

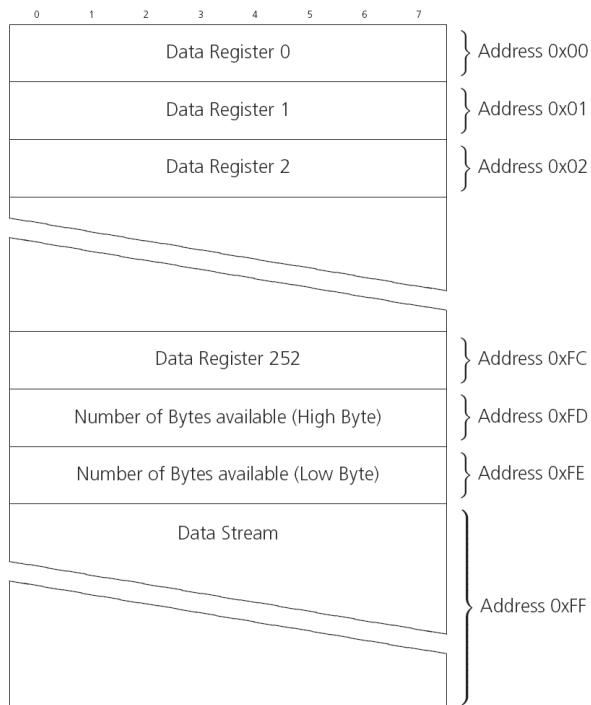


Figure 6: I2C register layout

3.2.2.2 Read access types

The host can choose one of the following two modes:

- Random read access: the master first reads the number of available bytes at the 0xFD and 0xFE before accessing the data at 0xFF.
- Current address read access: the master directly reads the data at the register 0xFF, without knowing first if there is any data waiting. If there is no data, the read result is a 0xFF byte value. This mode basically skips the first step of the "random read access", as it does not address to any particular register.

Figure 7 shows the format of the "random access" form of the request.

Following the start condition from the master, the 7-bit device address and the RW bit (which is a logic low for write access) are clocked onto the bus by the master transmitter. The receiver answers with an acknowledge (logic low) to indicate that it recognizes the address.

Next, the 8-bit address of the register to be read must be written to the bus (0xFD for u-blox receivers). Following the receiver's acknowledgment, the master again triggers a start condition and writes the device address, but this time the RW bit is a logic high to initiate the read access. Now, the master can read 1 to N bytes from the receiver. The receiver will first deliver the byte value at 0xFD, followed by the value at 0xFE. At this point the master knows the number of bytes waiting at the 0xFF register, and by acknowledging again, the data stream follows. The data transfer will stop once the master emits a not-acknowledge response or a stop condition is triggered after the last byte has been read.

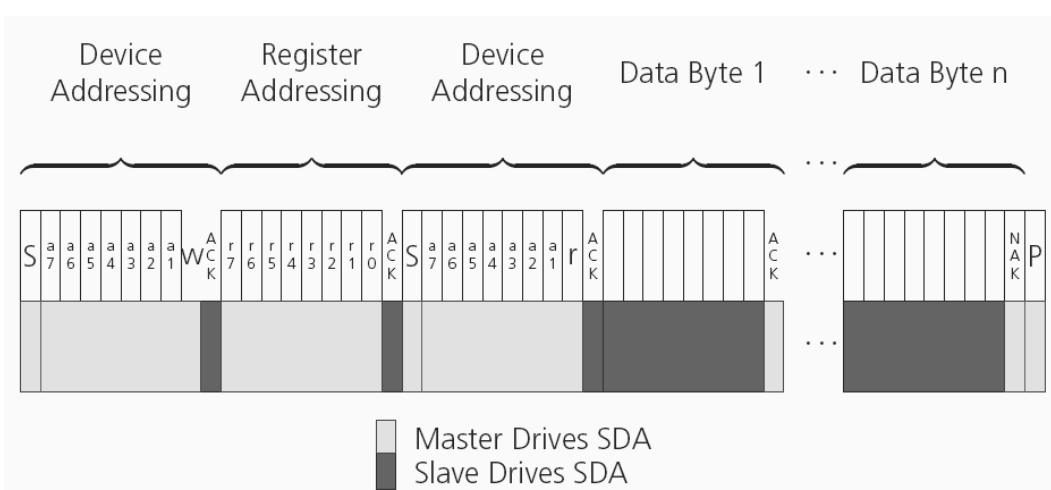


Figure 7: I2C random read access

If "current address" is used, an address pointer in the receiver is used to determine which register to read. This address pointer will increment after each read operation unless it is already pointing at register 0xFF, the highest addressable register, in which case it remains unaltered.

The initial value of this address pointer at startup is 0xFF, so by default all current address read operations will repeatedly read register 0xFF and receive the next byte of message data (or 0xFF value if no message data is waiting).

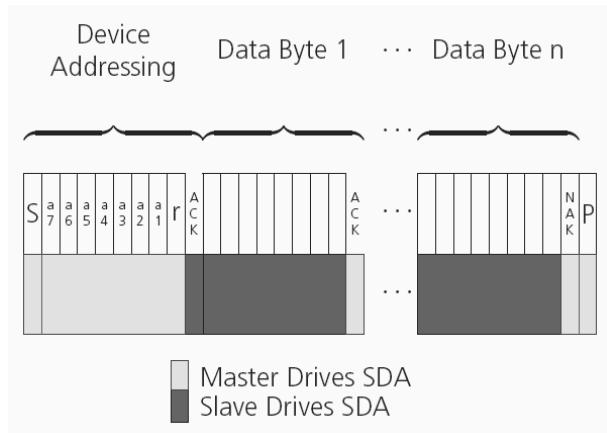


Figure 8: I2C current address read access

Only after addressing the slave, the receiver starts the data stream. If the master does not read data from the receiver for a certain timeout, the receiver assumes that the communication is broken and stops the data stream, preventing an overflow of the output buffer. This timeout is 1.5 seconds by default. However, it can be extended by setting the CFG-I2C-EXTENDEDTIMEOUT configuration item to true (see the SAM-M10Q interface description [3]). By disabling the timeout, the receiver will only interrupt the data stream when the buffer is full. The buffer can store up to 4 kB and the time for an overflow event depends on the number of messages enabled.

3.2.2.3 Write access

The receiver does not provide any write access except for writing UBX and NMEA messages to the receiver, such as configuration or aiding data. Therefore, the register set mentioned in section [I2C register layout](#) is not writeable.

Following the start condition from the master, the 7-bit device address and the RW bit (which is a logic low for write access) are clocked onto the bus by the master transmitter. The receiver answers with an acknowledge (logic low) response to indicate that it is responsible for the given address.

The master can write 2 to N bytes to the receiver, generating a stop condition after the last byte being written. To properly distinguish from the write access to set the address counter in random read accesses, the number of data bytes must be at least 2.

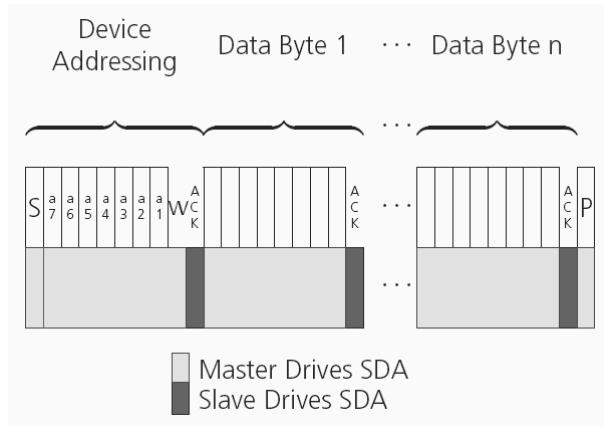


Figure 9: I2C write access

3.2.3 PIOs

This section describes the PIOs supported by SAM-M10Q. All PIO active voltage levels are related to the V_IO supply voltage. All the inputs have internal pull-up resistors in normal operation and can be left open if unused.

- ☞ When assigning a different function to a PIO, ensure that the default function is disabled where applicable. For example, disable the I2C interface with the CFG-I2C-ENABLED configuration key if I2C pins are used for antenna supervisor functions.

3.2.3.1 RESET_N

SAM-M10Q provides a RESET_N pin to reset the receiver. The RESET_N pin is input-only with an internal pull-up resistor to V_IO and should be left open for normal operation. Driving RESET_N low for at least 1 ms will trigger a reset of the receiver. The RESET_N complies with the V_IO level and can be actively driven high.

- ☞ Use RESET_N only in critical situations to recover the receiver. RESET_N resets the receiver and clears the BBR content including receiver configuration, real-time clock (RTC), and GNSS orbit data, triggering a cold start.
- ☞ No capacitor should be placed at RESET_N to GND, otherwise it could trigger a reset on every startup.

3.2.3.2 SAFEBOOT_N

The SAFEBOOT_N pin is for future service, updates and reconfiguration.

- ☞ The SAFEBOOT_N pin is internally connected to the TIMEPULSE pin through a 1 kΩ series resistor.

3.2.3.3 TIMEPULSE

The SAM-M10Q features one time pulse output at the TIMEPULSE pin. This can only be configured in PIO04. The details about this feature are explained in the section [Time pulse](#).

-  The TIMEPULSE and SAFEBOOT_N functions share the same internal IC function. If this pin is low at receiver startup, the receiver will enter safeboot mode. However, in normal operation the pin outputs the time pulse signal. Make sure there is no load on this pin that could pull it low at startup.

3.2.3.4 EXTINT

SAM-M10Q supports external interrupts at the EXTINT pin. The EXTINT pin has a fixed input voltage threshold with respect to V_IO. It can be used for functions such as accurate external [Frequency assistance](#), [Time assistance](#) and [Time mark](#) reporting.

The EXTINT pin enables [External control](#) for host-controlled on/off operation of the receiver, and as a wake-up source for power save mode on/off operation (PSMOO). If configured for host-controlled on/off operation, the internal pull-up is disabled. Make sure the EXTINT input is always driven within the defined voltage level by the host.

The EXTINT pin can also be configured for another functionality.

-  EXTINT functionality is only available at the EXTINT pin.

3.2.3.5 TX_READY

The receiver has an internal message buffer for storing bytes to be sent to the host application. TX-ready feature in TX_READY pin enables I2C to have an associated signal to indicate that the buffer has bytes to be transmitted. The host application can wait on the signal instead of polling the interface.

The TX-ready signal is enabled and assigned to a selected pin with CFG-TXREADY configuration items. The polarity of the signal (active-low or active-high) and the threshold for amount of bytes in the buffer must also be configured. When the number of bytes in the buffer reaches the threshold, the TX-ready signal becomes active. The signal stays active until all of the bytes in the buffer have been transferred. The receiver has additional small transmit buffer for each interface. Up to 16 bytes may still need to be transferred to the host after the TX-ready signal has become inactive.

The TX-ready signal can be assigned to any pin. Old function on the pin must be disabled before the assignment. Pin assignment can be verified with the UBX-MON-HW3 message.

The threshold value of the CFG-TXREADY-THRESHOLD is presented as 8-byte (64 bit) chunks. Divide the desired threshold value by 8. E.g., value of 128 sets the TX-ready signal activation threshold to 1024 bytes. The threshold should not be set above 2048 bytes (256 8-byte chunks). Otherwise, the TX-ready signal may not be activated in time, and some messages may get discarded.

3.2.3.5.1 Extended TX timeout

If the host does not communicate over I2C for more than 1.5 seconds, the receiver assumes that the host is no longer using this interface and no more packets are scheduled for this interface. This mechanism can be changed by enabling "extended TX timeouts" (configuration key CFG-I2C-EXTENDEDTIMEOUT). When enabled, the receiver delays idling the interface until the allocated and undelivered bytes for this interface reach 4 kB. This feature is especially useful when using the TX-ready feature with a message output rate of less than once per second, and fetching data only when available, determined by the TX_READY pin becoming active.

3.3 Forcing receiver reset

GNSS receivers typically make a distinction between cold, warm, and hot start based on the type of valid information the receiver has during the restart.

- **Cold start:** in the cold start mode, the receiver has no information from the last position (e.g. time, velocity, frequency etc.) at startup. Therefore, the receiver must search the full time and frequency space, and all possible satellite numbers. If a satellite signal is found, it is tracked to decode the ephemeris (18-36 seconds under strong signal conditions), while the other channels continue to search satellites. Once there is a sufficient number of satellites with valid ephemeris, the receiver can calculate position and velocity data. Other GNSS receiver manufacturers call this the Factory startup mode.
- **Warm start:** in the warm start mode, the receiver has approximate information for time, position, and coarse satellite position data (Almanac). In this mode, the receiver normally needs to download ephemeris after power-up before it can calculate position and velocity data. As the ephemeris data is usually outdated after 4 hours, the receiver typically starts with a warm start if it has been powered down for more than 4 hours. In this scenario, several augmentations are possible. See [Multiple GNSS assistance](#).
- **Hot start:** in the hot start mode, the receiver has been powered down only for a short time (4 hours or less), so that its ephemeris is still valid. Since the receiver does not need to download ephemeris again, this is the fastest startup method.

Using the UBX-CFG-RST message, you can force the receiver to reset and clear data, in order to see the effects of maintaining/losing such data between restarts. For this purpose, use the `navBbrMask` field in the UBX-CFG-RST message to initiate hot, warm, and cold starts, or a combination of startup modes.

The reset type can also be specified. This is not related to GNSS, but to the way the software restarts the system.

- **Hardware reset** uses the on-chip watchdog to electrically reset the chip. This is an immediate asynchronous reset. No stop events are generated.
- **Controlled software reset** terminates all running processes in an orderly manner. Once the system is idle, restarts the receiver operation, reloads its configuration and starts to acquire and track GNSS satellites.
- **Controlled software reset (GNSS only)** only restarts the GNSS tasks, without reinitializing the full system or reloading any stored configuration.
- **Hardware reset (after shutdown)** uses the on-chip watchdog to reset the receiver after shutdown.
- **Controlled GNSS stop** stops all GNSS tasks. The receiver is not restarted, but stops any GNSS-related processing.
- **Controlled GNSS start** starts all GNSS tasks.

[Table 12](#) below contains an overview of the different reset types and the data that is cleared.

Reset type	Clears RAM	Clears BBR
0x00 - Hardware reset (immediately), 0x04 - Hardware reset (after shutdown)	Yes	Yes
0x01 - Controlled Software reset	Yes	No
0x02 - Controlled Software reset (GNSS only), 0x08 - Controlled GNSS stop, 0x09 - Controlled GNSS start	No	No
RESET_N pin	Yes	Yes

[Table 12: Overview of the available reset types](#)

3.4 Security

The security concept of SAM-M10Q covers:

- the integrity of the receiver
- communication between the receiver and the GNSS satellites

Some security functions monitor and detect threats and report them to the host system. Other functions mitigate threats and allow the receiver to operate normally.

3.4.1 Configuration locking

The receiver configuration can be locked so that further changes are no longer possible. The configuration can be locked by setting the configuration item CFG-SEC-CFG_LOCK to "true". This item can be set in the same way as all other configuration items on various layers.

-  If the configuration lock is set in BBR, it remains locked until the BBR is cleared. To test the function, set it in the RAM configuration layer. After the receiver's power cycle, the information in this layer is lost and the configuration lock is not set anymore.

3.5 Power management

u-blox receivers support different operating modes. These modes represent strategies of controlling the acquisition and tracking engines to achieve either the best possible performance or good performance with reduced power consumption.

3.5.1 Continuous mode

SAM-M10Q uses dedicated signal processing engines optimized for signal acquisition and tracking. The acquisition engine actively searches for and acquires signals during cold starts or when insufficient signals are available during navigation. The tracking engine continuously tracks and downloads all the almanac data and acquires new signals as they become available during navigation. The tracking engine consumes less power than the acquisition engine.

The current consumption is lower when a valid position is obtained quickly after the start of the receiver navigation, the entire almanac has been downloaded, and the ephemeris for each satellite in view is valid. If these conditions are not met, the search for the available satellites takes more time and consumes more power.

3.5.2 Power save mode

Power save mode (PSM) allows a reduction in system power consumption by selectively switching parts of the receiver on and off. It is enabled with CFG-PM-OPERATEMODE and configured with items in the CFG-PM group.

Power save mode (PSM) has two modes of operation:

- **Power save mode cyclic tracking (PSMCT)** operation is used when position fixes are required in short periods of 0.5 s to 10 s.
- **Power save mode on/off (PSMOO)** operation is used for periods longer than 10 s, and can be in the order of minutes, hours, or days.

The mode of operation can be configured, and depending on the setting, the receiver demonstrates different behavior. In on/off operation the receiver switches between phases of startup/navigation and phases with low or almost no system activity (backup/sleep). In cyclic tracking the receiver does not shut down completely between fixes, but uses low-power tracking instead.

-  In PSMOO mode, the RAM memory is cleared during the off periods. Consequently, store the configuration in the BBR memory to maintain the settings.

GPS, GLONASS, BeiDou B1I, Galileo and QZSS signals are supported in power save mode. BeiDou B1C signal is not supported. The receiver is unable to download or process any SBAS data in power save mode and it is therefore recommended to disable SBAS.

- BeiDou B1C is not supported in power save mode.

3.5.2.1 Operation

PSM is based on a state machine with five different states: Inactive for update, Inactive for search, Acquisition, Tracking and Power optimized tracking (POT) state.

- Inactive states: most parts of the receiver are switched off.
- Acquisition state: the receiver actively searches for and acquires signals. Maximum power consumption.
- Tracking state: the receiver continuously tracks and downloads data. Less power consumption than in the acquisition state.
- POT state: the receiver repeatedly loops through a sequence of tracking (Track), calculating the position fix (Fix), and entering an idle period (Idle). No new signal is acquired and no data is downloaded. The power consumption is much lower than in the tracking state.

The PSM state machine is described in [Figure 10](#).

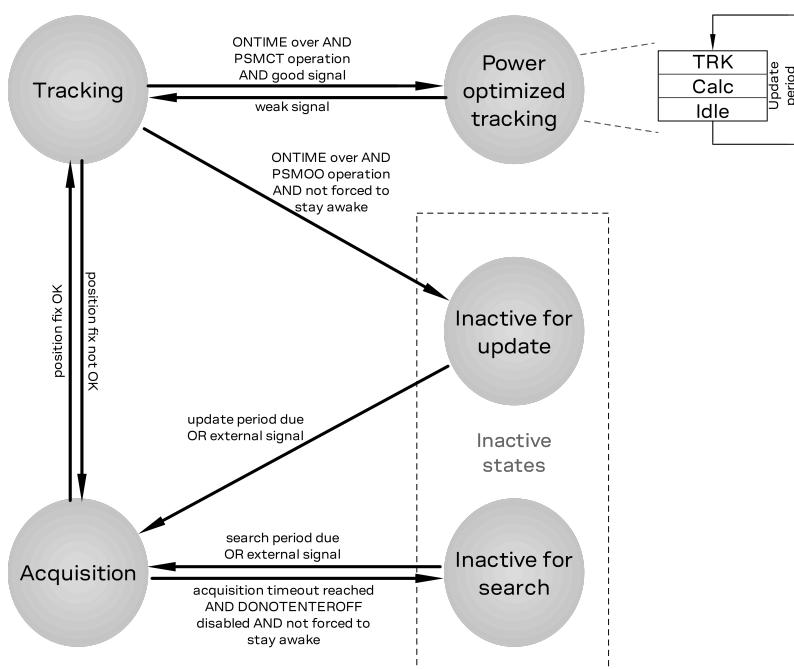


Figure 10: State machine

3.5.2.2 Acquisition timeout

The receiver has internal, external, and user-configurable mechanisms that determine the time to be spent in acquisition state. This logic is put in place to ensure good performance and low power consumption in different environments and scenarios. This collective logic is referred to as acquisition timeout.

The configuration items related to acquisition timeout are described in section [Configuration](#).

Internal mechanisms:

- If the receiver is able to acquire weak signals but not of the quality needed to get a fix, it will transition to the "Inactive for search" state after the timeout configured in MAXACQTIME or earlier if too few signals are acquired.
- If the receiver is unable to acquire any signals or it acquires a small number of extremely bad signals (e.g., no sky view), it will transition to the "Inactive for search" state after the timeout configured in MAXACQTIME.

User-configurable mechanisms:

- MINACQTIME is the minimum time that the receiver will spend in the "Acquisition" state. MINACQTIME is applicable only when no or very poor GNSS signal is available.
- MAXACQTIME is the maximum time that the receiver will spend in the "Acquisition" state.
- DONOTENTEROFF forces the receiver to stay awake and in the "Acquisition" state even when a fix is not possible.

External mechanisms:

- The receiver is forced to stay awake if EXTINTWAKE is enabled and the EXTINT pin is set to "high". The receiver is forced to stay in the "Inactive for search/Fix" states if EXTINTBACKUP is enabled and the EXTINT pin is set to "low".
- The receiver is forced to stay awake if EXTINTINACTIVE is enabled and the EXTINT pin is toggled. If EXTINT pin state is not changed for a longer time than EXTINTINACTIVITY, the receiver enters the "Inactive for search/Fix" states.

3.5.2.3 Cyclic tracking

Power save mode cyclic tracking (PSMCT) operation is described in [Figure 11](#).

- PSMCT supports 1 Hz and 2 Hz navigation update rates. In addition, longer update periods from 2 s to 10 s are supported at 1 s steps.

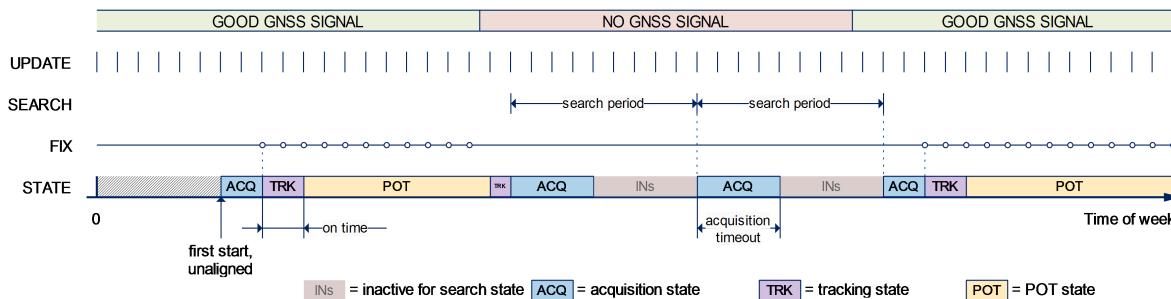


Figure 11: Cyclic tracking operation

- When the receiver is switched on, it first enters the "Acquisition" state. A larger number of signals tracked later helps the receiver to remain in the "POT" state if some signals get blocked and are lost. This may reduce the overall power consumption.
- If the receiver is able to acquire a valid position fix (one passing the navigation output filters) within the time given by the [Acquisition timeout](#), it switches to the "Tracking" state and the ONTIME starts. Otherwise it enters the "Inactive for search" state and restarts after the configured search period (minus a start-up margin).
- Once the ONTIME is over, the "POT" state is entered. Setting the ONTIME to zero causes the receiver to enter the "POT" state as soon as possible.
- In the "POT" state the receiver continues to output position fixes according to the CFG-RATE-*.
- If the signal becomes weak or is lost during the "POT" state, the "Tracking" state is entered.
- Once the signal is good again and the newly started ONTIME is over, the receiver will re-enter the "POT" state.

- If the receiver cannot get a position fix in the "Tracking" state, it enters the "Acquisition" state. Should the acquisition fail as well, the "Inactive for search" state is entered. If DONOTENTEROFF is enabled and no fix is possible, the receiver will remain in the "Acquisition" state until a fix is possible and it will never enter the "Inactive for search" state.

3.5.2.4 On/Off mode

Power save mode on-off (PSMOO) operation is described in [Figure 12](#).

PSMOO requires an RTC to maintain time.

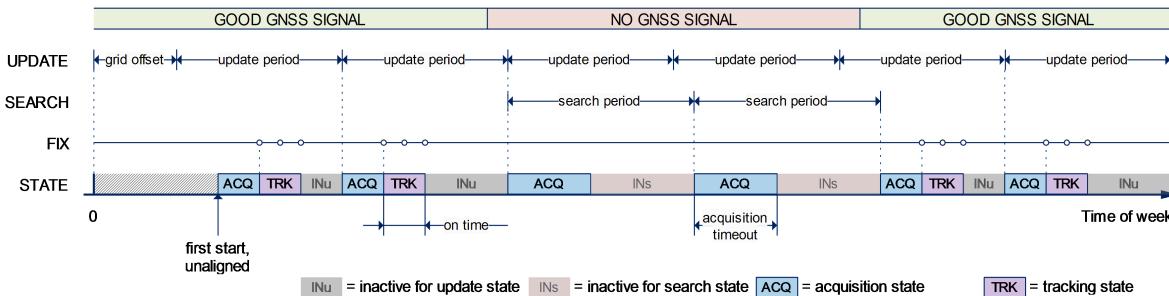


Figure 12: On/off mode operation

- When the receiver is switched on, it first enters the "Acquisition" state.
- If it is able to acquire a valid position fix (one passing the navigation output filters) within the time given by the [Acquisition timeout](#), it switches to the "Tracking" state and the ONTIME starts. Otherwise it enters the "Inactive for search" state and restarts after the configured search period (minus a startup margin).
- Once the ONTIME is over, the "Inactive for update" state is entered and the receiver restarts according to the configured update grid defined by GRIDOFFSET.
- If the signal is lost while in the "Tracking" state, the "Acquisition" state is entered. If the signal is not found within the acquisition timeout, the receiver enters the "Inactive for search" state. Otherwise the receiver will re-enter the "Tracking" state and stay there until the newly started ONTIME is over.

Entering the off state of the PSMOO operation clears the RAM memory including the receiver configuration. To maintain the configuration in PSMOO operation, store it on both RAM and battery-backed RAM (BBR) layers. Configuration in an optional flash memory is always maintained.

3.5.2.5 External control

The operation of power save mode can be controlled externally using the EXTINT pin. This external control allows the user to decide when to wake up the receiver to obtain a fix and when to force the receiver into backup mode to save power. Operating the receiver externally through the EXTINT pin will override internal functions that coincide with that specific operation.

Enabling EXTINTWAKE prevents the receiver from entering Inactive states for as long as the EXTINT pin is held "high". In PSMOO, the receiver will therefore always be in the "Acquisition" or the "Tracking" state. In PSMCT, the receiver can in addition be in the "POT" state. When EXTINT is set "low" the receiver will continue with its configured behavior.

Enabling EXTINTBACKUP forces the receiver to enter Inactive states for as long as the EXTINT pin is held "low" until the next wakeup event. Any wakeup event can wake up the receiver even if the EXTINT pin is held "low". In this case, the receiver only wakes up to read the configuration pins and then re-enters the Inactive state.

If both EXTINTWAKE and EXTINTBACKUP are enabled at the same time, the receiver PSM operation is completely under external control. Setting EXTINT "high" wakes up the receiver to get a position fix and setting "low" puts the receiver into backup mode.

 EXTINT pin control can also be used in continuous mode.

In PSMOO, an external wakeup source can be set. Any wake-up event will restart the receiver to try to obtain a position fix. Wakeup signals have no effect if the receiver is already in the "Acquisition", "Tracking", or "POT" state.

Setting the update period POSUPDATEPERIOD to zero causes the receiver to wait in the "Inactive for update" state until the host wakes it up. Setting the search period ACQPERIOD to zero causes the receiver to wait in the "Inactive for search" state indefinitely after an unsuccessful startup.

 External wake-up source is required when setting update or search period to zero.

The wake-up sources are:

- Rising or falling edge on the UART RX pin
- Rising or falling edge on the EXTINT pin
- Rising or falling edge on the SPI CS pin
- Rising edge on RESET_N pin

[Backup modes](#) can also be used to control the receiver state externally.

3.5.2.6 Configuration

Power save mode (PSM) is enabled and disabled with CFG-PM-OPERATEMODE and configured with items in the CFG-PM group listed in [Table 13](#).

 When using power save mode on/off (PSMOO) operation, set the OPERATEMODE as the last PSM configuration key to prevent the receiver entering the off state before all intended PSM configuration keys are set.

Config key	Description
OPERATEMODE	Receiver mode of operation
POSUPDATEPERIOD	Time between two position fix attempts in on/off power save mode
ACQPERIOD	Time between two acquisition attempts if the receiver is unable to get a position fix
GRIDOFFSET	Time offset of update grid with respect to start of week
ONTIME	Time the receiver remains in the "Tracking" state and produces position fixes
MINACQTIME	Minimum time the receiver spends in the "Acquisition" state
MAXACQTIME	Maximum time in the "Acquisition" state
DONOTENTEROFF	Receiver does not enter the "Inactive for search" state if it cannot get a position fix but keeps indefinitely attempting a position fix instead
WAITTIMEFIX	Wait for time fix before entering the "Tracking" state
UPDATEEPH	Enables periodic ephemeris update
EXTINTWAKE	Enables EXTINT pin control to force receiver on
EXTINTBACKUP	Enables EXTINT pin control to force receiver in backup
EXTINTINACTIVE	Enter a backup state if EXTINT pin is inactive longer time than specified by EXTINTINACTIVITY
EXTINTINACTIVITY	Specifies the inactivity period

Table 13: Power save mode configuration options in the CFG-PM group

OPERATEMODE The mode of operation to use mainly depends on the update period: For short update periods (in the range of a few seconds), cyclic tracking should be configured. For long update

periods (in the range of minutes or longer), only use on/off operation. See section [On/Off mode](#) and [Cyclic tracking](#) for more information on the two modes of operation.

POSUPDATEPERIOD, ACQPERIOD The update period POSUPDATEPERIOD specifies the time between successive position fixes. If no position fix can be obtained within the acquisition timeout, the receiver will retry after the time specified by the search period ACQPERIOD. Update and search periods are fixed with respect to an absolute time grid based on reference time standard (i.e., GPS time or UTC). They do not refer to the time of the last valid position fix or last position fix attempt. Where multiple GNSS can operate simultaneously, UTC time is used as the reference time standard.

The update period setting is ignored if the receiver is set into cyclic tracking mode. It only has an impact if the receiver is set to on/off mode. New settings are ignored if the update period or the search period exceeds the maximum number of milliseconds in a week. In that case the previously stored values remain effective.

GRIDOFFSET Once the receiver has a valid time, the update grid is aligned to the start of the week of the reference time standard (midnight between Saturday and Sunday). Before having a valid time, the update grid is unaligned. A grid offset shifts the update grid with respect to the start of the week of the reference time standard. The grid offset is not used in cyclic tracking operation.

ONTIME This specifies how long the receiver stays in the "Tracking" state before switching to the "POT" state in PSMCT or the "Inactive for update" state in PSMOO.

MINACQTIME The receiver tries to obtain a position fix for at least the time given by MINACQTIME. If the receiver determines that it needs more time for the given starting conditions then it will automatically prolong this time. If MINACQTIME is set to zero, the receiver determines the time. Once the MINACQTIME has expired, the receiver will terminate the acquisition state if either a fix is achieved or if the receiver estimates that any signals received are insufficient (too weak or too few) for a fix to be possible.

MINACQTIME is applicable only when no or very poor GNSS signal is available.

MAXACQTIME This defines the maximum time that the receiver will spend in the "Acquisition" state. If the receiver is unable to acquire a valid position fix within this maximum time, it will transition to the "Inactive for search" state (if DONOTENTEROFF is disabled). Subsequently, the receiver will attempt to acquire another position fix according to the search period ACQPERIOD. If MAXACQTIME is set to zero, the receiver will autonomously determine the maximum time to spend in the "Acquisition" state. Note that shorter settings (below about 45 s) will degrade an unaided receiver's ability to collect new Ephemeris data at low signal levels.

DONOTENTEROFF If this option is enabled, then when the receiver cannot get a fix it keeps attempting to acquire a position fix instead of entering the "Inactive for search" state. In other words, the receiver will never be in the "Inactive for search" state and therefore the search period ACQPERIOD and the minimum acquisition time MINACQTIME will be ignored.

WAITTIMEFIX A time fix is a fix type in which the receiver will ensure that the time is accurate and confirmed to within the limits set in CFG-NAVSPG. Enabling the WAITTIMEFIX option will force the receiver to stay in the "Acquisition" state until the time is known to be within the configured limits, then it will transition to the "Tracking" state. Take into account that enabling WAITTIMEFIX will delay the transition from the "Acquisition" state to the "Tracking" state by at least two extra seconds.

The quality of the position fixes can also be configured by setting the limits in the CFG-NAVSPG group. Setting harder limits in CFG-NAVSPG will typically prolong the time in the "Acquisition" state. When externally controlled, it is therefore necessary to ensure sufficient time for the receiver at startup. Refer to [Acquisition timeout](#) for more information. When internally controlled, the receiver

can make good judgment on the time needed in the "Acquisition" state and no further adjustments will be needed.

UPDATEEPPH To maintain the ability of a fast startup, the receiver needs to update its ephemeris data on a regular basis. This can be ensured by activating the update Ephemeris option UPDATEEPPH. The ephemeris data is updated approximately every 30 minutes. Refer to [Satellite data download](#) for more information.

3.5.2.7 Satellite data download

The receiver is not able to download satellite data (e.g. the ephemeris) while it is working in on/off or cyclic tracking operation. Therefore it has to temporarily switch to continuous operation for the time the satellites transmit the desired data. To save power the receiver schedules the downloads according to an internal timetable and only switches to continuous operation when data of interest is being transmitted by the satellites.

Each satellite transmits its own ephemeris data. Ephemeris data download is feasible when the corresponding satellite has been tracked with a sufficient C/N0 over a certain period of time. The download is scheduled in a 30-minute grid or immediately when fewer than a certain number of visible satellites have valid ephemeris data.

Almanac, ionosphere, UTC correction, and satellite health data are transmitted by all satellites simultaneously. Therefore these parameters can be downloaded when a single satellite is tracked with a sufficiently high C/N0.

Allowing more ephemerides to be downloaded before going into the "POT" or the "Inactive for update" state can help improve the quality of the fixes and reduce the number of wake ups needed to download ephemerides at the cost of extra time in the "Acquisition" state (only when an inadequate number of ephemerides are downloaded from tracked satellites).

3.5.3 Backup modes

A backup mode is an inactive state where the power consumption is reduced to a fraction of that in operating modes. The receiver maintains time information and navigation data to speed up the receiver restart after backup or standby mode.

The SAM-M10Q supports two backup modes: hardware backup mode and software standby mode.

3.5.3.1 Hardware backup mode

The hardware backup mode allows entering a backup state and resuming operation by switching the main power supplies on and off while maintaining a V_BCKP supply via, e.g. a battery. The receiver enters hardware backup mode if the VCC and V_IO supplies are removed.

V_BCKP must be supplied to maintain the backup domain (BBR and RTC) to allow better TTFF, accuracy, availability and power consumption at the next startup compared with a cold start. As V_IO is not supplied, the PIOs cannot be driven by an external host processor. If driving of the PIOs cannot be avoided, buffers are required for isolating the PIOs.

3.5.3.2 Hardware standby mode

In hardware standby mode, the VCC supply is removed. The V_IO is supplied and maintains the backup domain (BBR and RTC) and PIOs. As V_IO is supplied, the PIOs can be driven by an external host processor. No buffers are required for isolating the PIOs, which reduces cost.

 V_BCKP should be left open if not used.

3.5.3.3 Software standby mode

Software standby mode is entered using the UBX-RXM-PMREQ message. V_{IO} and VCC must be supplied, however VCC supply is internally disabled to save power. The V_{IO} supply maintains the battery-backed RAM (BBR), RTC, and PIOs.

Entering the software standby mode clears the RAM memory including the receiver configuration. To maintain the configuration, store it on both RAM and layers.

The software standby mode can be set for a specific duration, or until the receiver is woken up by a signal at a wake-up source defined in UBX-RXM-PMREQ. The possible wake-up sources are UART RX, EXTINT, and/or SPI CS pin. Refer to the interface description [3] for more information on the UBX-RXM-PMREQ message. A system reset with the RESET_N signal also terminates the software standby mode, clears the BBR content and restarts the receiver.

As V_{IO} is supplied, the PIOs can be driven by an external host processor. No buffers are required for isolating the PIOs, which reduces cost.

- ☞ The "force" flag must be set in UBX-RXM-PMREQ to enter software standby mode.
- ☞ V_BCKP should be left open if not used.

3.6 Time

Maintaining receiver local time and keeping it synchronized with GNSS time is essential for proper timing and positioning functionality. This section explains how the receiver maintains local time and introduces the supported GNSS time bases.

3.6.1 Receiver local time

The receiver is dependent on a local oscillator for both the operation of its radio parts and also for timing within its signal processing. No matter what nominal frequency the local oscillator has, u-blox receivers subdivide the oscillator signal to provide a 1-kHz reference clock signal, which is used to drive many of the receiver's processes. In particular, the measurement of satellite signals is arranged to be synchronized with the "ticking" of this 1-kHz clock signal.

When the receiver first starts, it has no information about how these clock ticks relate to other time systems; it can only count time in 1 millisecond steps. However, as the receiver derives information from the satellites it is tracking or from aiding messages, it estimates the time that each 1-kHz clock tick takes in the time base of the chosen GNSS system. This estimate of GNSS time based on the local 1-kHz clock is called receiver local time.

As receiver local time is a mapping of the local 1-kHz reference onto a GNSS time base, it may experience occasional discontinuities, especially when the receiver first starts up and the information it has about the time base is changing. Indeed, after a cold start, the receiver local time initially indicates the length of time that the receiver has been running. However, when the receiver obtains some credible timing information from a satellite or an aiding message, it jumps to an estimate of GNSS time.

3.6.2 GNSS time bases

GNSS receivers must handle a variety of different time bases as each GNSS has its own reference system time. What is more, although each GNSS provides a model for converting their system time into UTC, they all support a slightly different variant of UTC. So, for example, GPS supports a variant of UTC as defined by the US National Observatory, while BeiDou uses UTC from the National Time

Service Center, China (NTSC). While the different UTC variants are normally closely aligned, they can differ by as much as a few hundreds of nanoseconds.

Although u-blox receivers can combine a variety of different GNSS times internally, the user must choose a single type of GNSS time and, separately, a single type of UTC for input (on EXTINT pins) and output (via the TIMEPULSE pin) and the parameters reported in corresponding messages.

The CFG-TP-TIMEGRID_TP* configuration item allows the user to choose between any of the supported GNSS (GPS, GLONASS, BeiDou, etc.) time bases and UTC. Also, the CFG-NAVSPG-UTCSTANDARD configuration item allows the user to select which variant of UTC the receiver should use. This includes an "automatic" option which causes the receiver to select an appropriate UTC version itself, based on the GNSS configuration, using, in order of preference, USNO if GPS is enabled, SU if GLONASS is enabled, NTSC if BeiDou is enabled, NPLI if NAVIC is enabled, NICT when QZSS is enabled, finally, European if Galileo is enabled.

The receiver assumes that an input time pulse uses the same GNSS time base as specified for the time pulse output. So if the user selects GLONASS time for time pulse output, any time pulse input must also be aligned to GLONASS time (or to the separately chosen variant of UTC). Where UTC is selected for time pulse output, any GNSS time pulse input will be assumed to be aligned to GPS time.

- ☞ The receiver allows users to independently choose GNSS signals used in the receiver (using CFG-SIGNAL-*) and the input/output time base (using CFG-TP-*). For example it is possible to instruct the receiver to use GPS and GLONASS satellite signals to generate BeiDou time. This practice compromises time pulse accuracy if the receiver cannot measure the timing difference between the constellations directly and is therefore not recommended.
- ☞ The information that allows GNSS times to be converted to the associated UTC times is only transmitted by the GNSS at relatively infrequent periods. For example GPS transmits UTC(USNO) information only once every 12.5 minutes. Therefore, if a time pulse is configured to use a variant of UTC time, after a cold start, substantial delays before the receiver has sufficient information to start outputting the time pulse can be expected.

Each GNSS has its own time reference for which detailed and reliable information is provided in the messages listed in the table below.

Time reference	Message
GPS time	UBX-NAV-TIMEGPS
BeiDou time	UBX-NAV-TIMEBDS
GLONASS time	UBX-NAV-TIMEGLO
Galileo time	UBX-NAV-TIMEGAL
QZSS time	UBX-NAV-TIMEQZSS
UTC time	UBX-NAV-TIMEUTC

Table 14: GNSS time messages

3.6.3 Navigation epochs

Each navigation solution is triggered by the tick of the 1 kHz clock nearest to the desired navigation solution time. This tick is referred to as a navigation epoch. If the navigation solution attempt is successful, one of the results is an accurate measurement of time in the time base of the chosen GNSS system, called GNSS system time. The difference between the calculated GNSS system time and receiver local time is called the clock bias (and the clock drift is the rate at which this bias is changing).

In practice the receiver's local oscillator is not as stable as the atomic clocks to which GNSS systems are referenced and consequently clock bias tends to accumulate. However, when selecting the next

navigation epoch, the receiver always tries to use the 1 kHz clock tick which it estimates to be closest to the desired fix period as measured in GNSS system time. Consequently, the number of 1 kHz clock ticks between fixes occasionally varies. This means that when producing one fix per second, there are normally 1000 clock ticks between fixes, but sometimes, to correct drift away from the GNSS system time, there are 999 or 1001 ticks.

The GNSS system time calculated in the navigation solution is always converted to a time in both the GPS and UTC time bases for output.

Clearly when the receiver has chosen to use the GPS time base for its GNSS system time, conversion to GPS time requires no work at all, but conversion to UTC requires knowledge of the number of leap seconds since GPS time started (and other minor correction terms). The relevant GPS-to-UTC conversion parameters are transmitted periodically (every 12.5 minutes) by GPS satellites, but can also be supplied to the receiver via the UBX-MGA-GPS-UTC aiding message. By contrast, when the receiver has chosen to use the GLONASS time base as its GNSS system time, conversion to GPS time is more difficult as it requires knowledge of the difference between the two time bases, but as GLONASS time is closely linked to UTC, conversion to UTC is easier.

When insufficient information is available for the receiver to perform any of these time base conversions precisely, predefined default offsets are used. Consequently, plausible times are nearly always generated, but they may be wrong by a few seconds (especially shortly after receiver start). Depending on the configuration of the receiver, such "invalid" times may well be output, but with flags indicating their state (e.g. the "valid" flags in UBX-NAV-PVT).

-  To support multiple GNSS systems concurrently, u-blox receivers employ multiple GNSS system times and/or receiver local times. For reporting GNSS system time or the receiver local time, users are recommended to use messages that report UTC time instead of using UBX messages. Other messages are retained only to support backwards compatibility.

3.6.4 iTow timestamps

All the main UBX-NAV messages (and some other messages) contain an iTOW field which indicates the GPS time at which the navigation epoch occurred. Messages with the same iTOW value can be assumed to have come from the same navigation solution.

Note that iTOW values may not be valid (i.e. they may have been generated with insufficient conversion data) and therefore it is not recommended to use the iTOW field for any other purpose.

-  The original designers of GPS chose to express time/date as an integer week number (starting with the first full week in January 1980) and a time of week (often abbreviated to TOW) expressed in seconds. Manipulating time/date in this form is far easier for digital systems than the more conventional year/month/day, hour/minute/second representation. Consequently, most GNSS receivers use this representation internally, only converting to a more conventional form at external interfaces. The iTOW field is the most obvious externally visible consequence of this internal representation.

If reliable absolute time information is required, it is recommended to use the UBX-NAV-PVT navigation solution message which also contains additional fields that indicate the validity (and accuracy in UBX-NAV-PVT) of the calculated times (see also the GNSS times section below for further messages containing time information).

3.6.5 Time validity

Information about the validity of the time solution is given in the following form:

- **Time validity:** Information about time validity is provided in the valid flags (e.g. validDate and validTime flags in the UBX-NAV-PVT message). If these flags are set, the time is known

and considered valid for use. These flags are shown in table GNSS times in section GNSS times above as well as in the UBX-NAV-PVT message.

- **Time validity confirmation:** Information about confirmed validity is provided in the `confirmedDate` and `confirmedTime` flags in the UBX-NAV-PVT message. If these flags are set, the time validity can be confirmed by using an additional independent source, meaning that the probability of the time to be correct is very high. Note that information about time validity confirmation is only available if the `confirmedAvai` bit in the UBX-NAV-PVT message is set.

- ☞ `validDate` means that the receiver has knowledge of the current date. However, it must be noted that this date might be wrong for various reasons. Only when the `confirmedDate` flag is set, the probability of the incorrect date information drops significantly.
- ☞ `validTime` means that the receiver has knowledge of the current time. However, it must be noted that this time might be wrong for various reasons. Only when the `confirmedTime` flag is set, the probability of incorrect time information drops significantly.
- ☞ `fullyResolved` means that the UTC time is known without full seconds ambiguity. When deriving UTC time from GNSS time the number of leap seconds must be known, with the exception of GLONASS. It might take several minutes to obtain such information from the GNSS payload. When the one second ambiguity has not been resolved, the time accuracy is usually in the range of ~20s.

3.6.6 UTC representation

UTC time is used in many NMEA and UBX messages. In NMEA messages it is always reported rounded to the nearest hundredth of a second. Consequently, it is normally reported with two decimal places (e.g. 124923.52). Although compatibility mode (selected using CFG-NMEA-COMPAT) requires three decimal places, rounding to the nearest hundredth of a second remains, so the extra digit is always 0.

UTC time is also reported within some UBX messages, such as UBX-NAV-TIMEUTC and UBX-NAV-PVT. In these messages date and time are separated into seven distinct integer fields. Six of these (year, month, day, hour, min. and sec.) have fairly obvious meanings and are all guaranteed to match the corresponding values in NMEA messages generated by the same navigation epoch. This facilitates simple synchronization between associated UBX and NMEA messages.

The seventh field is called nano and it contains the number of nanoseconds by which the rest of the time and date fields need to be corrected to get the precise time. So, for example, the UTC time 12:49:23.521 would be reported as: hour: 12, min: 49, sec: 23, nano: 521000000.

It is however important to note that the first six fields are the result of rounding to the nearest hundredth of a second. Consequently the nano value can range from -5000000 (i.e. -5 ms) to +994999999 (i.e. nearly 995 ms).

When the nano field is negative, the number of seconds (and maybe minutes, hours, days, months or even years) will have been rounded up. Therefore, some or all of them must be adjusted in order to get the correct time and date. Thus in an extreme example, the UTC time 23:59:59.9993 on 31st December 2011 would be reported as: year: 2012, month: 1, day: 1, hour: 0, min: 0, sec: 0, nano: -700000.

Of course, if a resolution of one hundredth of a second is adequate, negative nano values can simply be rounded up to 0 and effectively ignored.

Which master clock the UTC time is referenced to is output in the message UBX-NAV-TIMEUTC.

The preferred variant of UTC time can be specified using the CFG-NAVSPG-UTCSTANDARD configuration item. The UTC time variant configured must correspond to a GNSS that is currently enabled. Otherwise the reported UTC time will be inaccurate.

3.6.7 Leap seconds

Due to the slightly uneven spin rate of the Earth, UTC time gradually moves out of alignment with the mean solar time (that is, the sun no longer appears directly overhead at 0 longitude at midday). Occasionally, a "leap second" is announced to bring UTC back into close alignment with the mean solar time. Usually this means adding an extra second to the last minute of the year, but this can also happen on 30th June. When this happens, UTC clocks are expected to go from 23:59:59 to 23:59:60, and only then on to 00:00:00.

It is also possible to have a negative leap second, in which case there will only be 59 seconds in a minute and 23:59:58 will be followed by 00:00:00.

u-blox receivers are designed to handle leap seconds in their UTC output and consequently applications processing UTC times from either NMEA or UBX messages should be prepared to handle minutes that are either 59 or 61 seconds long.

Leap second information can be polled from the receiver with the message UBX-NAV-TIMELS.

3.6.8 Date ambiguity

Each navigation satellite transmits information about the current date and time in the data message. The time of week (TOW) indicates the elapsed number of seconds since the start of the week (midnight Saturday/Sunday). The week number (WN) indicates the elapsed number of weeks since the particular GNSS system was started. By combining these two values the current date and time can be known. Modern GPS satellites use a 13-bit value for the week number. As GPS system was started in 1980, it allows the week number to represent dates up to year 2137. Unfortunately, at the time when the commonly used GPS L1C/A data message was designed the signal had only 10 bits available for the week number. The top bits of the full week number had to be left out. The 10 bottom bits of the week number are not sufficient to yield a completely unambiguous date as every 1024 weeks (a bit less than 20 years), the transmitted week number value "rolls over" back to zero. Consequently, the information in GPS L1 message does not differentiate between, for example, 1980, 1999, or 2019. GPS L1 receivers must thus use additional methods to calculate the full week number.

Although BeiDou and Galileo have similar representations of time, they still transmit sufficient bits for the week number to be unambiguous for the foreseeable future (the first ambiguity will be in 2078 for Galileo, and not until 2163 for BeiDou). GLONASS presents the time and date in different way and transmits sufficient information to avoid any ambiguity during the expected lifetime of the system (the first ambiguous date will be in 2124). Therefore, the receiver regards the date information transmitted by GLONASS, BeiDou, and Galileo to be unambiguous and, where necessary, uses this information to resolve any ambiguity in the GPS date.

-  If the receiver is connected to a simulator, be aware that GPS time is referenced to 6th January 1980, GLONASS to 1st January 1996, Galileo to 22nd August 1999 and BeiDou to 1st January 2006; the receiver cannot be expected to work reliably with signals simulated before these dates.

3.6.8.1 GPS-only date resolution

If only GPS L1C/A signals are available, the receiver establishes the date by assuming that all week numbers must be at least as large as the reference rollover week number. The default value for the reference rollover week number is selected at the compile time of the receiver firmware and is

normally set to a value of a few weeks before the software is completed. The value can be overridden by CFG-NAVSPG-WKNROLLOVER configuration item.

The following example illustrates how this works:

Assume that the reference rollover week number set in the firmware at compile time is 2148 (which corresponds to a week in calendar year 2021, but is transmitted by the satellites as 100). In this case, if the receiver sees transmissions containing week numbers in the range of 100 ... 1023, they are interpreted as week numbers 2148 ... 3071 (calendar years 2021 ... 2038), whereas transmissions with week numbers from 0 to 99 are interpreted as week numbers 3072 ... 3171 (calendar years 2038 ... 2040).

-  It is important to set the reference rollover week number correctly when supplying the receiver with simulated signals, especially when the scenarios are in the past.

3.7 Time mark

The receiver can be used to provide an accurate measurement of the time at which a pulse was detected on the external interrupt pin. The reference time can be chosen by setting the time source parameter to UTC, GPS, GLONASS, BeiDou, Galileo, NAVIC or local time in the CFG-TP-* configuration group. The UTC standard can be set in the CFG-NAVSPG-* configuration group. The delay figures defined with CFG-TP-* are also applied to the results output in the UBX-TIM-TM2 message.

A UBX-TIM-TM2 message is output at the next epoch if

- The UBX-TIM-TM2 message is enabled, and
- a rising or falling edge was triggered since last epoch on the EXTINT pin.

The UBX-TIM-TM2 messages includes the time of the last time mark, new rising/falling edge indicator, time source, validity, number of marks and an accuracy estimate.

-  Only the last rising and falling edge detected between two epochs is reported since the output rate of the UBX-TIM-TM2 message corresponds to the measurement rate configured with CFG-RATE-MEAS (see [Figure 13](#) below).

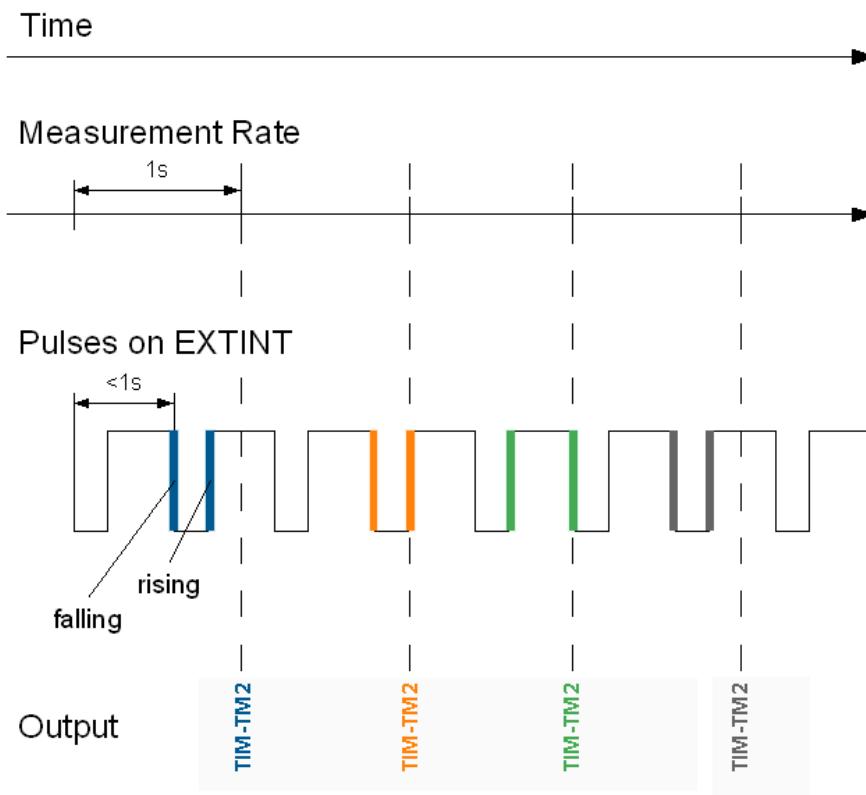


Figure 13: Time mark

3.8 Time pulse

The receiver includes a time pulse feature providing clock pulses with configurable duration and frequency. The time pulse function can be configured using the CFG-TP-* configuration group. The UBX-TIM-TP message provides time information for the next pulse and the time source.

Pulse Mode: Rising

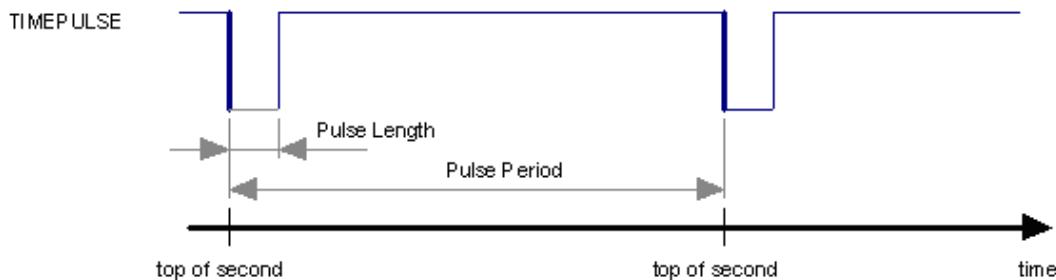
Pulse Mode: Falling


Figure 14: Time pulse

3.8.1 Recommendations

- The time pulse can be aligned to a wide variety of GNSS times or to variants of UTC derived from them (see the time bases section). However, it is strongly recommended that the choice of time base is aligned with the available GNSS signals (for example, to produce GPS time or UTC(USNO), ensure GPS signals are available, and for GLONASS time or UTC(SU) ensure the presence of GLONASS signals etc.). This involves coordinating the setting of CFG-SIGNAL-* configuration group with the choice of time pulse time base.
- When using time pulse for precision timing applications it is recommended to calibrate the antenna cable delay against a reference timing source.
- To get the best timing accuracy with the antenna, a fixed and accurate position is needed.
- If relative time accuracy between multiple receivers is required, do not mix receivers of different product families. If this is required, the receivers must be calibrated accordingly, by setting cable delay and user delay.
- The recommended configuration when using the UBX-TIM-TP message is to set both the measurement rate (CFG-RATE-MEAS) and the time pulse frequency (CFG-TP-*) to 1 Hz.

The sequential order of the signal present at the TIMEPULSE pin and the respective output message for the simple case of 1 pulse per second (1PPS) is shown in the following figure.

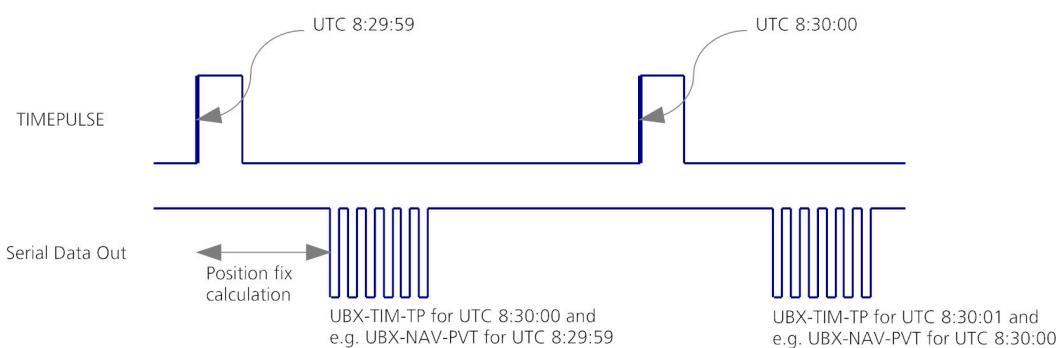


Figure 15: Time pulse and TIM-TP

3.8.2 Time pulse configuration

The time pulse (TIMEPULSE) signal has configurable pulse period, length and polarity (rising or falling edge).

It is possible to define different signal behavior (i.e. output frequency and pulse length) depending on whether or not the receiver is locked to reliable time source.

The configuration group CFG-TP-* can be used to change the time pulse settings, and includes the following parameters defining the pulse:

- **time pulse enable** - If this item is set, the time pulse is active.
- **frequency/period type** - Determines whether the time pulse is interpreted as frequency or period.
- **length/ratio type** - Determines whether the time pulse length is interpreted as length [us] or pulse ratio [%].
- **antenna cable delay** - Signal delay due to the cable between the antenna and the receiver.
- **pulse frequency/period** - Frequency or pulse time period when locked mode is not configured or not active.
- **pulse frequency/period lock** - Frequency or pulse time period for locked mode. In use as soon as the receiver has calculated a valid time from a received signal. Only used if the corresponding item is set to use another setting in locked mode.
- **pulse length/ratio** - Length or duty cycle of the generated pulse, specifies either time or ratio for the pulse to be on/off.
- **pulse length/ratio lock** - Length or duty cycle of the generated pulse for locked mode. In use as soon as the receiver has calculated a valid time from a received signal. Only used if the corresponding item is set to use another setting in locked mode.
- **user delay** - The cable delay from the receiver to the user device plus signal delay of any user application.
- **lock to GNSS freq** - If this item is set, uses the frequency gained from the GNSS signal information rather than the local oscillator's frequency.
- **locked other setting** - If this item is set, the alternative setting is used as soon as the receiver can calculate a valid time. This mode can be used, for example, to disable time pulse if the time is not locked, or to indicate a lock with different duty cycles.
- **align to TOW** - If this item is set, pulses are aligned to the top of a second.
- **polarity** - If set, the first edge of the pulse is a rising edge (pulse polarity: rising).
- **grid UTC/GNSS** - Selection between UTC (0), GPS (1), GLONASS (2), BeiDou (3), (4) Galileo and NAVIC (5) time grid. Also affects the time output by UBX-TIM-TP message.

The maximum pulse length cannot exceed the pulse period.

- The high and the low period of the output cannot be less than 50 ns, otherwise pulses can be lost.

3.8.2.1 Example

The example below shows the 1PPS TIMEPULSE signal generated on the time pulse output according to the specific parameters of the CFG-TP-* configuration group:

- **CFG-TP-TP1_ENA = 1**
- **CFG-TP-PERIOD_TP1 = 1 000 000 µs**
- **CFG-TP-LEN_TP1 = 100 000 µs**
- **CFG-TP-TIMEGRID_TP1 = 1 (GPS)**
- **CFG-TP-PULSE_LENGTH_DEF = 0 (Period)**
- **CFG-TP-ALIGN_TO_TOW_TP1 = 1**
- **CFG-TP-USE_LOCKED_TP1 = 1**
- **CFG-TP-POL_TP1 = 1**
- **CFG-TP-PERIOD_LOCK_TP1 = 1 000 000 µs**
- **CFG-TP-LEN_LOCK_TP1 = 100 000 µs**

The 1 Hz output is maintained whether or not the receiver is locked to GPS time. The alignment to TOW can only be maintained when GPS time is locked.

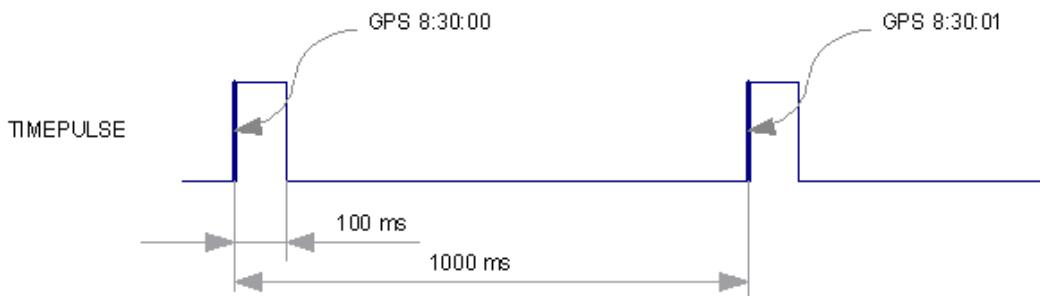


Figure 16: Time pulse signal with the example parameters

3.9 Time maintenance

Maintaining accurate time can improve the speed and performance of the receiver restart. Estimate of GNSS time can be maintained by a real-time clock, or it can be provided to the receiver by the host. Estimate of the clock drift of the receiver local oscillator or an external reference frequency can also be provided to improve the startup performance.

3.9.1 Real-time clock

The receiver contains a real-time clock (RTC). The RTC section is located in the backup domain and can keep time while the receiver is otherwise powered off. When the receiver powers up, it attempts to use the RTC to initialize receiver local time and in most cases this leads to considerably faster and more accurate first fixes.

3.9.2 Time assistance

The host can deliver time assistance to the receiver using UBX-MGA-INI-TIME_UTC or UBX-MGA-INI-TIME_GNSS for better startup performance.

The current GNSS time can be supplied to the receiver as a coarse value via the standard communication interfaces. This method suffers from communication latency and unpredictable

delays so the accuracy of the supplied time is poor. Accuracy of the supplied time can be improved greatly if the host system has a very good sense of the current time and can deliver an exactly timed pulse to the EXTINT pin. This pulse informs the receiver when the supplied time assistance data is to be applied.

UTC time leap seconds and GPS-to-UTC conversion parameters are transmitted periodically by GPS satellites, but that happens only every 12.5 minutes. The receiver can normally calculate the correct leap seconds value from other GNSS systems immediately, but in some situations that is not possible. If the leap seconds information or the difference of time between GPS and UTC system is important for the host application, the information can be supplied to the receiver via the UBX-MGA-GPS-UTC aiding message.

3.9.3 Frequency assistance

To supply hardware frequency assistance, connect a periodic rectangular signal with a frequency of up to 500 kHz to the EXTINT pin. The frequency can have an arbitrary duty cycle but the low/high phase duration must not be shorter than 50 ns. The applied frequency value must be submitted to the receiver using the UBX-MGA-INI-FREQ message.

Frequency assistance can improve the cold start speed in crystal-based designs. For TCXO-based designs, the frequency assistance has only minimal impact as the receiver is quick to acquire accurate frequency from satellite transmissions. A stable external reference frequency can be used to speed up receiver testing in production test setup. The host system may also be able to provide the reference frequency to improve the cold start speed.

3.9.4 Clock drift assistance

Estimate of the clock drift of the local oscillator can also be fetched from the receiver using the UBX-NAV-CLOCK message. This estimate can then be sent back to the receiver using the UBX-MGA-INI-CLKD message.

3.10 Protection level

3.10.1 Introduction

Critical applications need to know how much trust they can place in their GNSS receiver's output at any given moment. Computed by the GNSS receiver in real time, the protection level (PL) quantifies the reliability of the position information to allow systems to change their mode of operation and improve the efficiency and quality of the tasks being performed.

The GNSS receiver's protection level describes the maximum likely position error to a specified degree of confidence. For example, if a GNSS receiver determines its position with a 95% protection level of one meter, there is only a 5% chance that the reported position is more than one meter away from its true position. Like the accuracy estimate of the GNSS receiver, the protection level constantly fluctuates, influenced by all the common error sources that affect GNSS solutions. Unlike the accuracy estimate, the confidence level of the protection level is much higher and is validated against specific operating scenarios to ensure that the output bounds the true error.

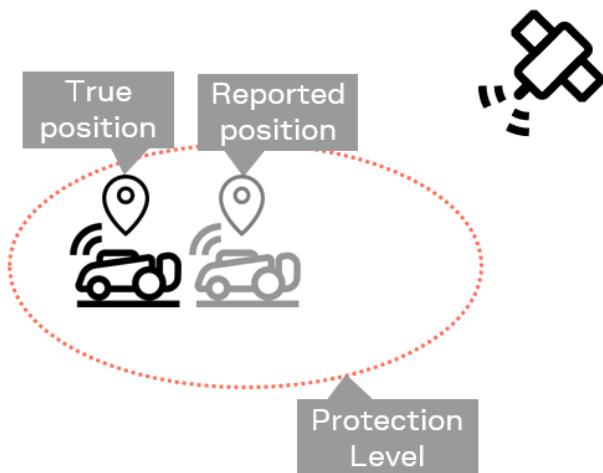


Figure 17: PL bounding true position error

3.10.2 Interface

The protection level bounds the true position error with a target misleading information risk (TMIR), for example 5% [MI/epoch] (read: 5% probability of having an MI per epoch). The target misleading information risk describes the probability per epoch of having misleading information (MI), meaning that it is not possible to bound the true position error because it is larger than the protection level (see Figure 18).

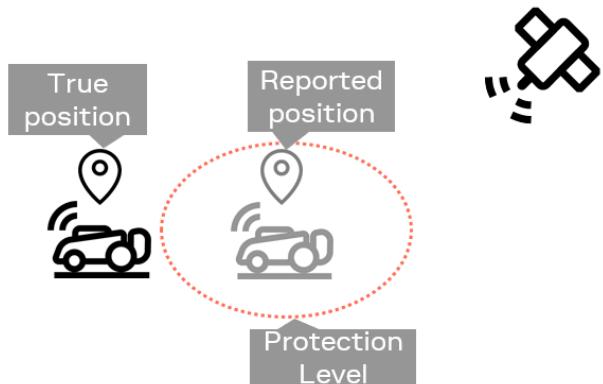


Figure 18: Misleading information

The output of the protection level is published through the UBX-NAV-PL message.

- ☞ TMIR is specified in one dimension for PL. It is not specified as a horizontal 2D or 3D value.
- ☞ The protection level values (UBX-NAV-PL.plPos1/2/3) are confidence intervals around the reported position (for example, UBX-NAV-PVT or UBX-NAV-HPPOSLLH).
- ☞ The target misleading information risk is provided in exponential notation (UBX-NAV-PL.tmirCoeff and UBX-NAV-PL.tmirExp), for example UBX-NAV-PL.tmirCoeff = 5 and UBX-NAV-PL.tmirExp = 0 results in 5e0 (= 5).
- ☞ The true position error is generally unknown, unless a very accurate and reliable truth positioning system is reporting an estimate for the true position.

When the GNSS environment deviates significantly from the normal mode of operation as compared to scenarios where the PL has been validated, a validity flag is set to false to indicate these

conditions. These conditions tend to be binary in nature, such as jamming has been detected, or the minimum number of satellites is being observed. UBX-NAV-PL reports a PL validity flag (see UBX-NAV-PL.pIPosValid), which indicates whether the PL is usable. .

- ☞ The protection level performance depends on many external and internal factors. Some external factors such as a harsh GNSS environment may lead to degraded PL performance.
- ☞ The protection level validity is not to be confused with misleading information and is independent of misleading information.

PL validity values	Description
UBX-NAV-PL.pIPosValid = 1	PL values are valid and can be used
UBX-NAV-PL.pIPosValid = 0	PL values are invalid and shall not be used

Table 15: Position PL validity

3.10.3 Expected behavior

For each navigation epoch and for each coordinate axis, a PL value is provided. For example, if the coordinate frame reported is North/East/Down, then the UBX-NAV-PL contents can be interpreted as follows:

PL values	Description
UBX-NAV-PL.pIPos1	1 stands for the north axis
UBX-NAV-PL.pIPos2	2 stands for the east axis
UBX-NAV-PL.pIPos3	3 stands for the down axis

Table 16: Position PL values

If the PL coordinate frame is set to invalid (UBX-NAV-PL.pIPosFrame = 0), then the PL values shall not be used. If the PL validity flag is cleared (UBX-NAV-PL.pIValid = 0), the PL values shall not be used. Both of these cases must be checked.

Only if the PL is set to valid (UBX-NAV-PL.pIPosValid), the PL values (UBX-NAV-PL.pIPos1/2/3) can be used and are reliable with respect to the target misleading information risk.

3.11 Multiple GNSS assistance (MGA)

The u-blox AssistNow services provide a proprietary implementation of an A-GNSS protocol compatible with u-blox GNSS receivers. The MGA services consist of AssistNow Online and Offline variants delivered by HTTP or HTTPS protocol.

When a client device makes an AssistNow request, the service responds with the requested data using standard UBX protocol MGA messages. These messages are ready for direct transmission from the client to the receiver port without requiring any modification.

AssistNow Online optionally provides immediate satellite ephemerides, health information and time aiding data suitable for GNSS receiver systems with direct internet access.

The AssistNow Offline service benefits u-blox GNSS receivers that only have occasional internet access. Users request data from the service by specifying the time period for which they want coverage (1 to 5 weeks). The data downloaded from the service is organized by date and encoded into a sequence of UBX MGA messages. In addition there is an MGA feature called AssistNow Autonomous which does not need an internet connection and runs entirely on the receiver.

Table 17 below contains an overview of the different MGA services u-blox provides. Refer to the SAM-M10Q datasheet for the supported GNSS signals by each MGA service [1].

Requirements	AssistNow Online	AssistNow Offline	AssistNow Autonomous
Requires external flash memory	No	Optional	Optional
Requires internet connection	Permanently	Sporadically	No
Amount of internet data	Medium	High	None
Ephemeris in data	Yes	No	No
Almanac in data	Yes	Yes	Yes

Table 17: AssistNow service overview

3.11.1 Authorization

To use the AssistNow services, customers will need to obtain an authorization token from u-blox. Go to <https://www.u-blox.com/en/solution/services/assistnow> or contact your local technical support to get more information and to request access to the service.

3.11.2 Preserving MGA and operational data during power-off

The time-to-fix after a receiver power interruption is dependent on the amount of operational data available at startup. Satellite broadcast information and an estimate of accurate time can be fetched from the AssistNow service. In addition, the following techniques can restore the data that was stored prior to powering down.

- **Battery-backed RAM:** The receiver operational state stored in this RAM can be maintained during power outages by connecting the V_BCKP pin to an independent supply, e.g. a battery. This is a recommended method as it will maintain all MGA-related information, any user configuration, calibration data, and an estimate of time via the real-time clock. See [V_BCKP](#) for more information.
- **Database dump:** The receiver can be made to dump the state of its navigation database in the form of a sequence of UBX messages reported to the host; these messages can be stored by the host and then sent back to the receiver when it has been restarted. See the description of the UBX-MGA-DBD messages in the SAM-M10Q interface description [3] for more information.

3.11.3 AssistNow offline

AssistNow Offline is a feature that combines special firmware in u-blox receivers and a proprietary service run by u-blox. It is targeted at receivers that only have occasional internet access and so cannot use AssistNow Online. AssistNow Offline speeds up time to first fix (TTFF), typically to considerably less than 10 s.

 AssistNow Offline currently supports GPS and GLONASS. u-blox intends to expand the AssistNow Offline Service to support other GNSS (such as BeiDou and Galileo) in due course.

The AssistNow Offline Service uses a simple, stateless, HTTP interface. Therefore, it works on all standard mobile communication networks that support internet access, including GPRS, UMTS and Wireless LAN. No special arrangements need to be made with mobile network operators to enable AssistNow Offline.

Users of AssistNow Offline are expected to download data from the AssistNow Offline Service, specifying the time period they want covered (1 day to 5 weeks) and the types of GNSS. This data must be uploaded to a u-blox receiver, so that it can estimate the positions of the satellites when no better data is available. Using these estimates will not provide as accurate a position fix as if current ephemeris data is used, but it will allow a much faster TTFF in nearly all cases.

The data obtained from the AssistNow Offline Service is organized by date, normally a day at a time. Consequently, the longer the requested coverage time, the more data there is to handle. Similarly, each different GNSS requires its own data and in extreme cases, several hundred kilobytes of data

will be provided by the service. This amount can be reduced by requesting lower resolution, but this will have a small negative impact on both position accuracy and TTFF. See the section on [Offline Service Parameters](#) for details of how to specify these options.

The downloaded AssistNow Offline data is encoded in a sequence of UBX-MGA-ANO messages, one for every satellite for every day of the period covered. For example, data for all GPS satellites for 4 weeks will involve in excess of 900 separate messages, taking up around 70 kb which would not fit into the available BBR memory in the receiver.

If the receiver has no flash storage, either a smaller amount of data must be requested, or the host system must store the AssistNow Offline data until the receiver needs it and then upload only the appropriate part for immediate use. See the section on [host-based AssistNow Offline](#) for further details.

3.11.3.1 Service parameters

The information exchange with the AssistNow Offline Service is based on the HTTP protocol. Upon reception of an HTTP GET request, the server will respond with the required messages in binary format or with an error string in text format. After delivery of all data, the server will terminate the connection.

The HTTP GET request from the client to the server should contain a standard HTTP query string in the request URL. The query string consists of a set of "key=value" parameters in the following form:

key=value;key=value;key=value;

The following rules apply:

- The order of keys is not important.
- Keys and values are case-sensitive.
- Keys and values must be separated by an equals character ('=').
- Key-value pairs must be separated by semicolons (';').
- If a value contains a list, each item in the list must be separated by a comma (',').

The following table describes the keys that are supported.

Key name	Unit/range	Necessity	Description
token	String	Mandatory	The authorization token supplied by u-blox when a client registers to use the service.
gnss	String	Mandatory	A comma-separated list of the GNSS for which data should be returned. The currently supported GNSS are: gps and glo.
period	Numeric [weeks]	Optional	The number of weeks into the future the data should be valid for. Data can be requested for up to 5 weeks into the future. If this value is not provided, the server assumes a period of 4 weeks.
days	Numeric [days]	Optional	The number of days into the future the data should be valid for. Can be used instead of the period parameter when data for less than one week is desired.
resolution	Numeric [days]	Optional	The resolution of the data: 1=every day, 2=every other day, 3=every third day. If this value is not provided, the server assumes a resolution of 1 day.

Table 18: AssistNow Offline parameter keys

Thus, as an example, a valid parameter string would be:

token=XXXXXXXXXXXXXXXXXXXXXX;gnss=gps,glo;

3.11.3.2 Time, position and almanac

While AssistNow Offline can be used on its own, it is expected that the user will provide estimates of the receiver's current position, the current time and ensure that a reasonably up-to-date almanac

is available. In most cases this information is likely to be available without the user needing to do anything. For example, where the receiver is connected to a battery backup power supply and has a functioning real-time clock (RTC), the receiver will keep its own sense of time and will retain the last known position and any almanac. However, should the receiver be completely without power before startup, then it will greatly improve TTFF if time, position and almanac can be supplied in some form.

Almanac data has a validity period of several weeks, so it can be downloaded from the AssistNow Online service at roughly the same time the AssistNow Offline data is obtained. It can then be stored in the host for uploading on receiver startup, or it can be transferred to the receiver straight away and preserved there (provided suitable non-volatile storage is available).

Where a receiver has a functioning RTC, it should be able to keep its own sense of time, but where no RTC is fitted (or power is completely turned off), providing a time estimate via the UBX-MGA-INI-TIME_UTC message will be beneficial to lower the time to first fix and make use of the AssistNow Offline data.

Similarly, where a receiver has effective non-volatile storage, the last known position will be recalled, but if this is not the case, then providing a position estimate via one of the UBX-MGA-INI-POS_XYZ or UBX-MGA-INI-POS_LLH messages will improve the TTFF (details can be found in the SAM-M10Q interface description [3]).

Where circumstances prevent providing all three of these pieces of data, providing some is likely to be better than none at all, as this helps to lower the TTFF.

3.11.3.3 Host-based AssistNow Offline

Host-based AssistNow Offline involves AssistNow Offline data being stored until it is needed by the host system in whatever memory it has available.

The host system must download the data from the AssistNow Offline service when an internet connection is available, but retain it until the time the u-blox receiver needs it. At this point, the host must upload just the relevant portion of the data to the receiver, so that the receiver can start using it. This is achieved by parsing all the data and uploading to the receiver only those UBX-MGA-ANO messages with a timestamp nearest to the current time. As each item is a complete UBX message, it can be sent directly to the receiver with no extra packaging. The host can control the message flow if required, but in most cases this is likely to prove unnecessary.

When parsing the data obtained from the AssistNow Offline service, the following points should be noted:

- The data is made up of a sequence of UBX-MGA-ANO messages.
- Customers should not rely on the messages all being of fixed size, but should read their length from the UBX header to work out where the message ends (and where the next begins).
- Each message indicates the satellite for which it is applicable through the svld and gnssld fields.
- Each message contains a timestamp within the year, month and day fields.
- Midday (UTC) on the day indicated should be considered to be the point at which the data is most applicable.
- The messages will be ordered chronologically, earliest first.
- Messages with the same timestamp will be ordered by ascending gnssld and then ascending svld.

3.11.3.3.1 Host-based procedure

The typical sequence for host-based AssistNow Offline is as follows:

- The host downloads a copy of the latest data from the AssistNow Offline service and stores it locally.
- Optionally it may also download a current set of almanac data from the AssistNow Online service.
- The host wants to use the u-blox receiver.
- If necessary it uploads any almanac, position estimate and/or time estimate to the receiver.
- It scans through AssistNow Offline data looking for entries with a timestamp that most closely matches the current (UTC) date and time.
- The host sends each such UBX-MGA-ANO message to the receiver.

Note that when data has been downloaded from the AssistNow Offline service with the (default) resolution of one day, the means for selecting the closest matching timestamp is simply to look for messages with the current (UTC) date.

3.11.4 AssistNow autonomous

The assistance scenarios covered by *AssistNow Online* and *AssistNow Offline* require an online connection and a host that can use this connection to download aiding data and provide this to the receiver when required.

The *AssistNow Autonomous* feature provides a functionality similar to *AssistNow Offline* without the need for a host and a connection. Based on a broadcast ephemeris downloaded from the satellite (or obtained by *AssistNow Online*), the receiver can autonomously (i.e. without any host interaction or online connection) generate an accurate satellite orbit representation ("AssistNow Autonomous data") that is usable for navigation much longer than the underlying broadcast ephemeris was intended for. This makes downloading new ephemeris or aiding data for the first fix unnecessary for subsequent startups of the receiver.

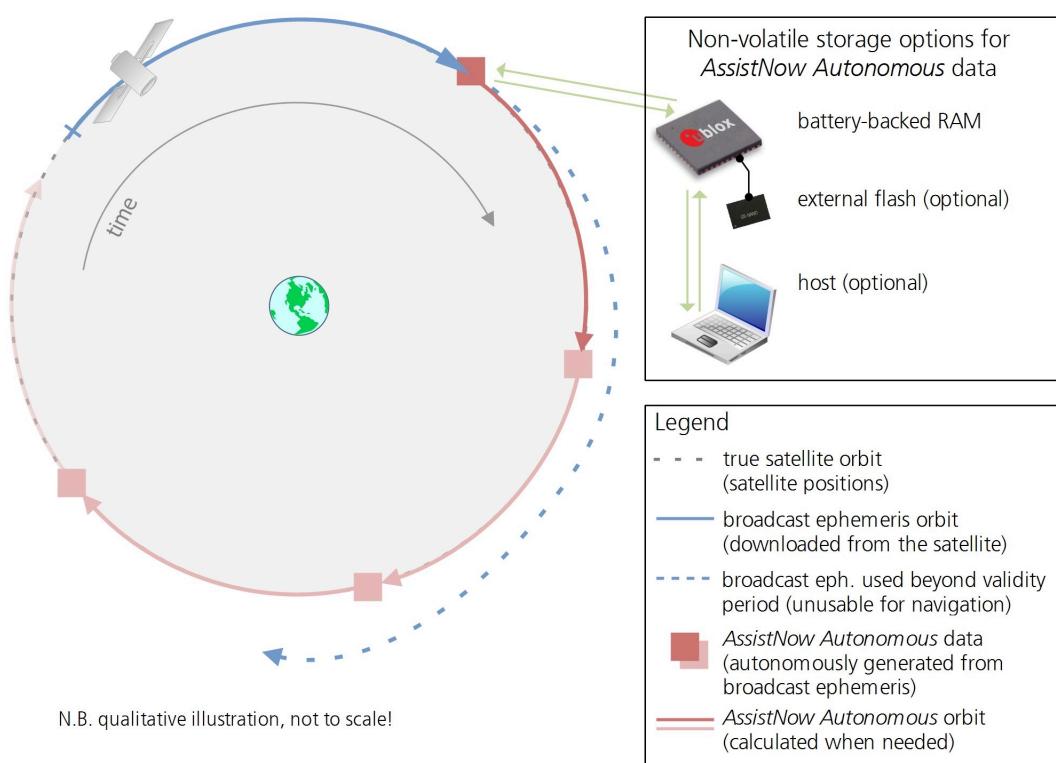
 The *AssistNow Autonomous* feature is disabled by default. It can be enabled using the CFG-ANA-USE_ANA configuration item.

3.11.4.1 Concept

The figure below illustrates the *AssistNow Autonomous* concept in a graphical way. Note that the figure is a qualitative illustration and is not to scale.

- A broadcast ephemeris downloaded from the satellite is a precise representation of a part (for GPS nominally four hours) of the satellite's true orbit (trajectory). It is not usable for positioning beyond this validity period because it diverges dramatically from the true orbit afterwards.
- The *AssistNow Autonomous orbit* is an extension of one or more broadcast ephemerides. It provides a long-term orbit for the satellite for several revolutions. Although this orbit is not perfectly precise, it is a sufficiently accurate representation of the true orbit to be used for navigation.
- The *AssistNow Autonomous data* is automatically and autonomously generated from downloaded (or assisted) ephemerides. The data is stored automatically in the on-chip battery-backed memory (BBR). Optionally, the data can be backed up in external flash memory or on the host. The number of satellites for which data can be stored depends on the receiver configuration and may change during operation.
- If no broadcast ephemeris is available for navigation, *AssistNow Autonomous* automatically generates the required parts of the orbits suitable for navigation from the stored data. The data is also automatically kept current in order to minimize the calculation time once the navigation engine needs orbits.
- The operation of the *AssistNow Autonomous* feature is transparent to the user and the operation of the receiver. All calculations are done in the background and do not affect the normal operation of the receiver.

- The *AssistNow Autonomous* subsystem automatically invalidates data that has become too old and that would introduce unacceptable positioning errors. This threshold is configurable.
- The prediction quality will be automatically improved if the satellite has been observed multiple times. However, this requires the availability of a suitable flash memory. Improved prediction quality also extends the maximum usability period of the data.
- AssistNow Autonomous* considers GPS, GLONASS, Galileo and BeiDou satellites only. It will not consider satellites on orbits with an eccentricity of >0.05 (e.g., Galileo E18). For GLONASS support, a suitable flash memory is mandatory because a single GLONASS broadcast ephemeris contains information only for approximately 30 minutes. This is not long enough to extend it in a usable way. Orbit information of each GLONASS satellite must be collected at least for four hours to generate data.



3.11.4.2 Interface

Several UBX protocol messages provide interfaces to the *AssistNow Autonomous* feature:

- The CFG-ANA-USE_ANA item is used to enable or disable the *AssistNow Autonomous* feature. When enabled, the receiver will automatically produce *AssistNow Autonomous* data for newly received broadcast ephemerides and, if that data is available, automatically provide the navigation subsystem with orbits when necessary and adequate.
 - The CFG-ANA-* configuration group also allows for a configuration of the maximum acceptable orbit error. See the next section for an explanation of this feature. It is recommended to use the firmware default value that corresponds to a default orbit data validity of approximately three days (for GPS satellites observed once) and up to six days (for GPS and GLONASS satellites observed multiple times over a period of at least half a day).
-  If the receiver uses flash memory, disabling the *AssistNow Autonomous* feature will delete all previously collected satellite observation data from the flash memory.
- The UBX-NAV-AOPSTATUS message provides information on the current state of the *AssistNow Autonomous* subsystem. The status indicates whether the *AssistNow Autonomous* subsystem is currently idle (or not enabled) or busy generating data or orbits. Hosts should

monitor this information and only power off the receiver when the subsystem is idle (that is, when the `status` field shows a steady zero).

- The UBX-NAV-SAT message indicates the use of *AssistNow Autonomous* orbits for individual satellites.
- The UBX-NAV-ORB message indicates the availability of *AssistNow Autonomous* orbits for individual satellites.
- The UBX-MGA-DBD message provides a means to retrieve the *AssistNow Autonomous* data from the receiver in order to preserve the data in power-off mode where no battery backup is available. Note that the receiver requires the absolute time (i.e. full date and time) to calculate *AssistNow Autonomous* orbits. For the best performance, it is therefore recommended to supply this information to the receiver using the UBX-MGA-INI-TIME_UTC message in this scenario.

3.11.4.3 Benefits and drawbacks

AssistNow Autonomous can provide quicker startup times by lowering the TTFF, provided that data is available for enough visible satellites. This is particularly true under weak signal conditions where it might not be possible to download broadcast ephemerides at all and therefore, no fix would be possible without *AssistNow Autonomous* (or A-GNSS). It is however required that the receiver roughly knows the absolute time, either from an RTC or from time-aiding (see the [Interface](#) section), and that it knows which satellites are visible, either from the almanac or from tracking the respective signals.

The *AssistNow Autonomous* orbit (satellite position) accuracy depends on various factors, such as the particular type of satellite, the accuracy of the underlying broadcast ephemeris, or the orbital phase of the satellite and Earth, and the age of the data (errors add up over time).

AssistNow Autonomous will typically extend a broadcast ephemeris from three up to six days. The CFG-ANA-ORBMAXERR item allows changing this threshold by setting the «maximum acceptable modeled orbit error» (in meters). Note that this number does not reflect the true orbit error introduced by extending the ephemeris. It is a statistical value that represents a certain expected upper limit based on a number of parameters. A rough approximation that relates the maximum extension time to this setting is: $\text{maxError [m]} = \text{maxAge [d]} * f$, where the factor f is 30 for data derived from satellites seen once and 16 for data derived for satellites seen multiple times during a long enough time period (see the [Concept](#) section).

There is no direct relation between (true and statistical) orbit accuracy and positioning accuracy. The positioning accuracy depends on various factors, such as the satellite position accuracy, the number of visible satellites, and the geometry (DOP) of the visible satellites. Position fixes that include *AssistNow Autonomous* orbit information may be significantly worse than fixes using only broadcast ephemerides. Therefore, it might be necessary to adjust the limits of the navigation output filters (CFG-NAVSPG-OUTFIL_XXXX).

Unknown future events form a fundamental deficiency of any system and can prevent precise satellite orbit predictions. Hence, the receiver will not be able to know about satellites that will have become unhealthy, have undergone a clock swap, or have had a maneuver. This means that the navigation engine might rarely mistake a wrong satellite position as the true satellite position. However, provided that there are enough other good satellites, the navigation algorithms will eventually eliminate a defective orbit from the navigation solution.

The repeatability of the satellite constellation is a potential pitfall for the use of the *AssistNow Autonomous* feature. For a given location on Earth, the (GPS) constellation (geometry of visible satellites) repeats every 24 hours. Hence, when the receiver «learned» about a number of satellites at some point in time, the same satellites will in most places *not* be visible 12 hours later, and the

available *AssistNow Autonomous* data will not be of any help. However, after another 12 hours, usable data would be available because it was generated 24 hours ago.

The longer a receiver observes the sky, the more satellites it will have seen. At the equator, and with full sky view, approximately ten (GPS) satellites will show up in a one-hour window. After four hours of observation approx. 16 satellites (i.e. half the constellation), after 10 hours approx. 24 satellites (2/3rd of the constellation), and after approx. 16 hours the full constellation will have been observed (and *AssistNow Autonomous* data generated). Lower sky visibility reduces these figures (i.e. the number of satellites seen). Further away from the equator, the numbers improve because the satellites can be seen twice a day. For example, at 47 degrees north the full constellation can be observed in approx. 12 hours with full sky view.

The calculations required for *AssistNow Autonomous* are carried out on the receiver. This requires energy and users may therefore occasionally see increased power consumption during short periods (several seconds, rarely more than 60 seconds) when such calculations are running. Ongoing calculations will automatically prevent the power save mode from entering the power-off state. The power-down will be delayed until all calculations are done.

- ☞ AssistNow Autonomous should be enabled if the system has sporadic access to the AssistNow Offline service. In this case, the receiver will intelligently choose the more reliable orbit predictions for each satellite. This way the autonomous prediction can provide performance improvements if the offline data becomes old or gets outdated.

3.12 Data batching

3.12.1 Introduction

The data batching feature allows position fixes to be stored in the RAM of the receiver to be retrieved later in one batch. Batching of position fixes happens independently of the host system, and can continue while the host is powered down.

Table 19 lists all the batching-related messages:

Message	Description
UBX-MON-BATCH	Provides information about the buffer fill level and dropped data due to overrun
UBX-LOG-RETRIEVEBATCH	Starts the batch retrieval process
UBX-LOG-BATCH	A batch entry returned by the receiver

Table 19: Batching-related messages

3.12.2 Setting up the data batching

Data batching is disabled per default and it has to be configured before use via the CFG-BATCH-* configuration group.

The feature must be enabled and the buffer size must be set to greater than 0. It is possible to set up a PIO as a flag that indicates when the buffer is close to filling up. The fill level when this PIO is asserted can be set by the user separately from the buffer size. The notification fill level must not be larger than the buffer size.

If the host does not retrieve the batched fixes before the buffer fills up, the oldest fix will be dropped and replaced with the newest.

The RAM available in the chip limits the size of the buffer. To make the best use of the available space, users can select what data they want to batch. When batching is enabled, a basic set of data is stored and the configuration flags EXTRAPVT and EXTRAODO can be used to store more

detailed information about the position fixes. However, enabling the EXTRAPVT and EXTRAODO flags reduces the number of fixes that can be batched.

The receiver will reject the configuration if it cannot allocate the required buffer memory. To ensure robust operation of the receiver the limits in [Table 20](#) are enforced:

EXTRAPVT	EXTRAODO	Maximum number of epochs
0	0	600
0	1	443
1	0	309
1	1	261

Table 20: Maximum number of batched epochs

- ☞ It is recommended to disable all periodic output messages when using data batching. This improves system robustness and also helps ensure that the output of batched data is not delayed by other messages.
- ☞ The buffer size is set up in terms of navigation epochs. This means that the time that can be covered with a certain buffer depends on the navigation rate. This rate can be set via the "CFG-RATE-" configuration group.

3.12.3 Retrieval

UBX-LOG-RETRIEVEBATCH message starts the process which allows the receiver to output batch entries. Batching must not be stopped for readout because all batched data is lost when the feature is disabled.

Batched fixes are always retrieved starting with the oldest fix in the buffer and progressing towards newer ones. There is no way to skip certain fixes during retrieval.

When a UBX-LOG-RETRIEVEBATCH message is sent the receiver transmits all batched fixes. It is recommended to send a retrieval request with `sendMonFirst` set. This way the receiver will send a UBX-MON-BATCH message first that contains the number of fixes in the batching buffer. This information can be used to detect when the u-blox receiver finishes sending data.

Once retrieval has started, the receiver will first send UBX-MON-BATCH message if `sendMonFirst` option was selected in the UBX-LOG-RETRIEVEBATCH message. After that, it will send UBX-LOG-BATCH messages with the batched fixes.

To maximize the speed of transfer, it is recommended that a high communication data rate is used.

- ☞ The receiver will discard retrieval request while processing a previous UBX-LOG-RETRIEVEBATCH message.
- ☞ The receiver does **not** acknowledge the reception of UBX-LOG-RETRIEVEBATCH message. The receiver responds with UBX-MON-BATCH (optional) and UBX-LOG-BATCH messages.

3.13 CloudLocate

In CloudLocate setup, the host processor of the customer application fetches a set of satellite signal measurements from the receiver and sends those to the u-blox CloudLocate service for position calculation. The CloudLocate service uses these measurements and current assistance data to calculate the receiver position. This data is then provided to the customer enterprise cloud for further use. Power saving up to 90% is possible compared to a cold start scenario.

The receiver starts to collect measurements as soon as it finds any satellite signals. It does not need to wait for a position fix for this. Collecting the measurements takes only a short time, so the application can quickly turn off the receiver or put it into a backup state.

3.13.1 CloudLocate measurements

The satellite signal measurements can be requested from the receiver either as a complete or compact raw measurement message.

The complete raw measurement message (UBX-RXM-MEASX) provides measurements for all visible satellites. The customer application can wait until the number of satellites in the message is sufficient and then send the message to the CloudLocate service. The amount of data in a MEASX message for five satellites is about 170 bytes. This increases by 24 bytes for each additional satellite.

Compact raw messages (UBX-RXM-MEAS50, UBX-RXM-MEAS20, UBX-RXM-MEAS12C, and UBX-RXM-MEAS12D) can be used when the amount of data to be sent to the cloud needs to be minimized. The data can be reduced to 50, 20, or even 12 Bytes, and the time for the receiver to stay on shall be limited to the minimum as well. These messages contain the measurement data in compressed format and use satellites only from a limited set of GNSS constellations. With the default settings, these messages contain measurement data only for a small number of satellites.

The raw measurement messages are enabled with the configuration keys in the CFG-MSGOUT configuration group. For example, setting the configuration key CFG-MSGOUT-UBX_RXM_MEAS50_UART1 to value 1 with UBX-CFG-VALSET message enables output of the UBX-RXM-MEAS50 message in the UART1 port for each navigation epoch.

The UBX-RXM-MEASX message can be sent with UBX header and checksum, and the message can be either in binary format or as encoded text. With the compact raw measurement messages, only the payload portion of the message is sent. Encoding the data would increase the size, so the compact messages should be sent in binary format.

In adverse conditions, satellite signal reception may take a long time or may not be possible. In such a situation, the payload of a compact raw measurement message is empty. The host application can wait until the payload contains data and only then switch off the receiver. In the case of using UBX-RXM-MEASX, the payload always has some data. The host application must check if the message contains enough information to calculate a position. For example, the application can wait for a confirmation from the customer enterprise cloud.

For more information on the raw measurement messages and on using the CloudLocate setup, see the Interface description [3] and the [u-blox website CloudLocate documentation](#).

4 Hardware integration

This section explains how the receiver can be integrated into an application design.

4.1 Power supply

The SAM-M10Q has the following power supply pins: **VCC**, **V_IO** and **V_BCKP**.

Power supply at VCC and V_IO must be present for normal operation. These two pins can either be connected together or supplied independently by the application. Power supply at V_BCKP is optional. If present, it enables the hardware backup mode when V_IO and VCC supplies are off.

Refer to the SAM-M10Q Data sheet [1] for absolute maximum ratings, operating conditions, and power requirements.

4.1.1 VCC

VCC provides power to the core and RF domains. Consequently, it always needs to be supplied to start up the receiver or for operation in continuous mode. The VCC pin supplies power to the core via an internal DCDC converter for efficient power consumption.

It is recommended to connect the VCC and V_IO pins together. If independent supplies are used for VCC and V_IO, turn off the V_IO supply first or simultaneously with the VCC supply.

- ☞ Do not add series resistance greater than $0.2\ \Omega$ on the supply line to avoid voltage ripple due to the dynamic current conditions.

4.1.2 V_IO

V_IO supplies all the digital IOs, clock, and the backup domain. The current drawn at V_IO depends on the activity and loading of the PIOs and the main oscillator.

A power interruption at V_IO will erase the battery-backed RAM (BBR) unless there is an external supply connected to V_BCKP.

It is recommended to connect the VCC and V_IO pins together. If independent supplies are used for VCC and V_IO, turn off the V_IO supply first or simultaneously with the VCC supply.

4.1.3 V_BCKP

Power supply at V_BCKP is optional. If the power supply at V_IO is interrupted, but the V_BCKP pin is supplied, the receiver enters the hardware backup mode. In this mode, the RTC time and the GNSS orbit data in the BBR are maintained. Valid time and GNSS orbit data at startup improves positioning performance by enabling hot starts, warm starts, and AssistNow Autonomous. This ensures faster TTFF when V_IO is supplied again. To make these features available, connect an independent power supply to V_BCKP to ensure backup domain supply when V_IO is not supplied.

Designs using an external battery as a power source at the V_BCKP pin must consider the battery capacity. The GNSS satellite ephemeris data is typically valid for up to 4 hours for hot starts, and up to a few days for warm starts and AssistNow (Offline and Autonomous).

- ☞ Avoid high resistance on the V_BCKP line. During the switch to V_BCKP supply, a short current adjustment peak may cause a high voltage drop at the pin.
- ☞ If the hardware backup mode is not used, leave the V_BCKP pin open.

4.2 RF interference

The received GNSS signal power at the antenna is very low compared to other wireless communication signals. The nominal -130 dBm received GNSS signal strength is below the thermal noise floor, making a GNSS receiver susceptible to interference from nearby RF sources of any kind.

As an example, cellular applications emit signals with power levels of approximately +30 dBm, while the GNSS signal is less than -128 dBm when reaching the antenna. By simply comparing these numbers it is obvious that interference issues must be seriously considered during the design phase.

4.2.1 In-band interference

Although the radio communications standards prevent intentional RF signal sources from interfering the GNSS frequencies, many devices emit RF power into the GNSS band at levels much higher than the GNSS signal itself.

One reason is that the frequency band above 1 GHz is not well regulated with regards to EMI, and even if permitted, signal levels are much higher than the GNSS signal power. In particular, all types of digital equipment, such as PCs, digital cameras, LCD screens, etc. tend to emit a broad frequency spectrum up to several GHz of frequency. Also wireless transmitters may generate spurious emissions that fall into the GNSS band.

 The section [Layout](#) defines measures against in-band interference during the design phase of the application.

4.2.2 Out-of-band interference

Out-of-band interference is caused by signal frequencies that are different from the GNSS carrier frequency. The main sources are wireless communication systems such as LTE, GSM, CDMA, WCDMA, Wi-Fi, BT, etc. Typically, these systems may emit their specified maximum transmit power in close proximity to the GNSS receiving antenna, especially if such a system is integrated with the GNSS receiver. Even at reasonable antenna selectivity, destructive power levels may reach the RF input of the GNSS receiver. In addition, larger signal interferers may generate intermodulation products inside the GNSS receiver front-end that fall into the GNSS band and contribute to in-band interference.

4.2.3 Spectrum analyzer

The UBX-MON-SPAN message can be enabled in u-center to provide a low-resolution spectrum analyzer sufficient to identify noise or jammers in the reception band. Once enabled, u-center includes a real-time chart that is updated once per second with the message data. See [Figure 19](#) for an example.

The shape of the spectrum as well as any RF interference in the form of a spur or intermodulation product can be easily analyzed with this graphical tool. The vertical axis compares the amplitude difference in dB for each frequency. A good spectrum shape is characterized by an even noise floor along with the GNSS band. For example, if any unwanted spur stands out, the vertical axis gives a rough approximation of the power level in dB compared to the noise floor.

Next to the chart, the center frequency, span, and resolution values set for the spectrum, and the PGA value are also displayed. The PGA value represents the internal gain set by the receiver, which depends on the external amplification of the GNSS input signal.

The vertical discontinuous lines in the chart area represent the offset to the center frequency in MHz. This helps to estimate the frequency of any spurious emission seen.

In addition, u-center includes three functions commonly found in any spectrum analyzer. These features support the RF front-end design and help to spot out any jammer presence during the application operation.

- View / hold: if selected, the current spectrum shape freezes in a colored line. This allows for a comparison between the time the spectrum was frozen and the real-time spectrum. This is particularly helpful in assessing the impact of running other onboard components.
- Average: if selected, a colored line shows the averaged spectrum for each frequency. This supports the analysis over time and obtaining a less noisy shape.
- Max hold: if selected, a colored line shows the maximum amplitude measured at each frequency. This option helps to spot out any jammer over a period of time.

[Figure 19](#) shows the spectrum view in u-center with the view/hold option selected. The reddish line represents the frozen spectrum before modifying the external gain, while the black line represents the current measurement.

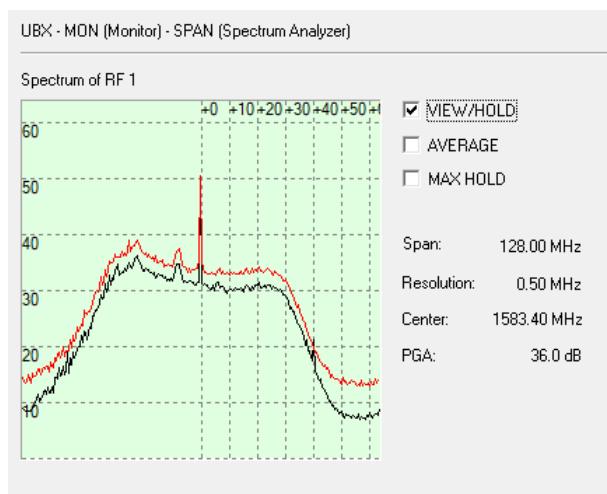


Figure 19: Spectrum analyzer view in u-center with the option view/hold selected

- ☞ By changing the number of constellations enabled, the span widens or narrows. This has a direct impact on the spectrum resolution, as the number of frequencies measured is fixed and equals to 256. For further details about this message and how to calculate each frequency, see the interface description [3].
- ☞ A big spur may be visible in the center frequency. The signal comes internally from the receiver and it does not cause any degradation in the performance.

4.3 RF design

4.3.1 Integrated antenna

SAM-M10Q antenna module is designed with an integrated multi-GNSS ceramic patch antenna. The antenna is right-hand circular polarized (RHCP) and has a peak gain of 3 dBic.

4.3.1.1 Antenna radiation pattern

SAM-M10Q antenna radiation pattern in free space is shown in [Figure 20](#). The antenna is measured on a solid $50 \times 50 \text{ mm}^2$ ground plane. The antenna RHCP gain pattern is shown in terms of elevation angle θ . The direction $\theta = 0^\circ$ is in the direction normal to the ground plane. The pattern is omnidirectional over the azimuth angle ϕ .

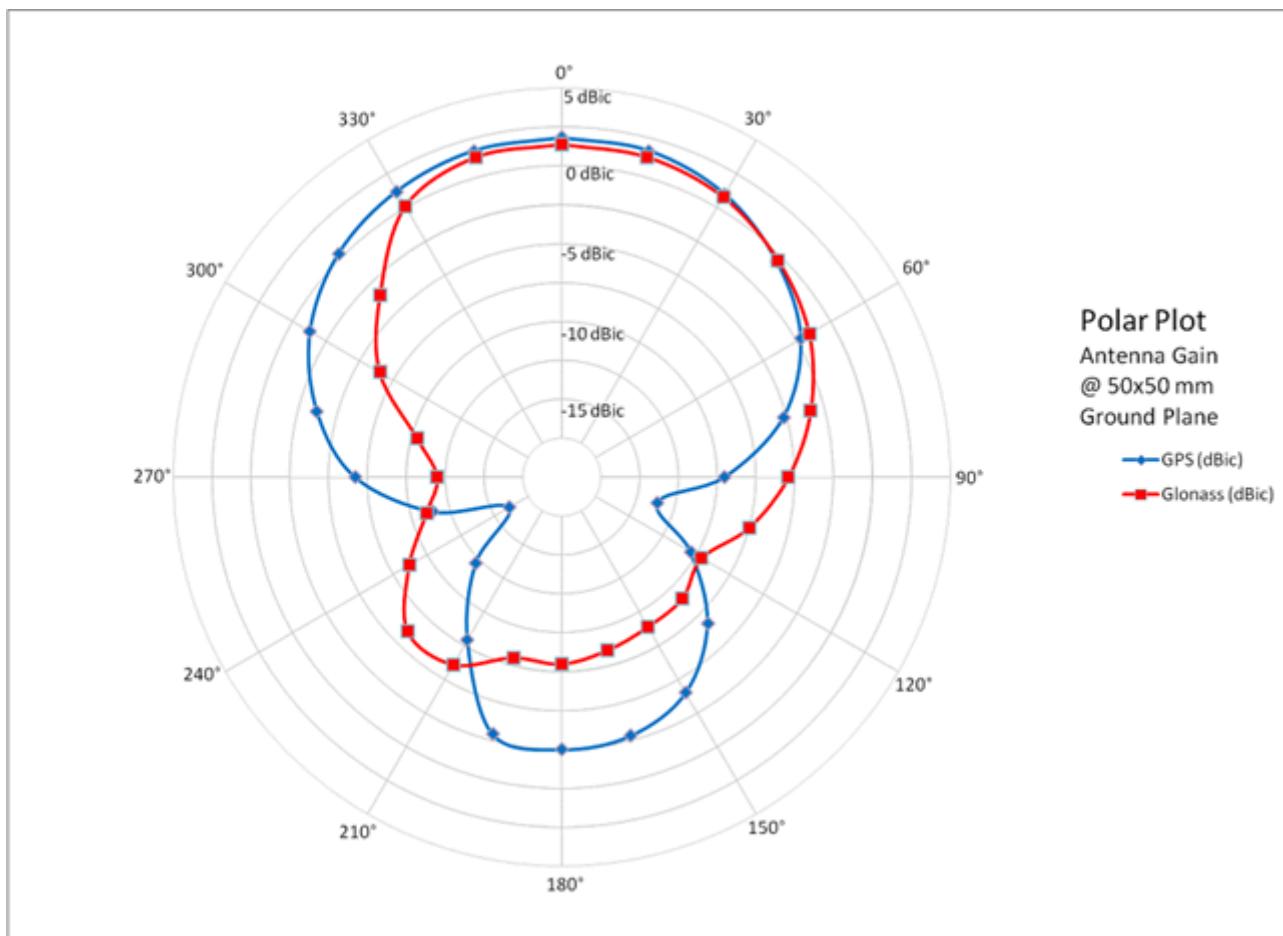


Figure 20: SAM-M10Q antenna RHCP gain pattern in free space at GPS and GLONASS bands. Measured on a 50 x 50 mm² ground plane.

The ground plane of the application design forms a part of the antenna. For best performance, place the SAM-M10Q antenna module in the middle of the ground plane. A solid ground plane around the antenna is required for free flow of the antenna currents. The size and shape of the ground plane as well as discontinuities (e.g. holes or signal traces) in the ground plane may affect the radiation pattern. The optimal radiation pattern is achieved with a 50 x 50 mm² ground plane. A larger ground plane can be used, but on a small ground plane the antenna gain and radiation efficiency are reduced. Significant degradation in performance occurs with a ground plane smaller than 40 x 40 mm².

The ceramic patch antenna is insensitive to surroundings and has a high tolerance against frequency detuning. It is however recommended to place any tall components ($h > 3$ mm) at least 10 mm away from the antenna. An enclosure or a plastic cover should have a minimum distance of 5 mm to the antenna.

The close proximity of human body can have a detrimental effect on the antenna performance due to the high permittivity and electrical loss of body tissue. The antenna operating frequency can be

detuned and the radiation efficiency significantly reduced. Placing the ground plane between the body and the antenna minimizes the body effects on antenna performance.

4.3.2 RF front-end

GNSS receivers operate with very low signal levels, ranging from -130 dBm to approximately -167 dBm. This alone is a challenge for the GNSS application design. Out-of-band sources of interference such as GSM, CDMA, WCDMA, LTE, Wi-Fi, or Bluetooth wireless systems with a much higher signal level require additional specific measures. The goal of the RF front-end design is to receive the in-band signal with minimum loss and added noise while suppressing the out-of-band interference.

The SAM-M10Q RF front-end is designed for high immunity against RF interference. The integrated RF circuit between the antenna input terminal and the receiver RF input includes a pass-band SAW filter followed by a low-noise amplifier (LNA). The SAW filter rejects out-of-band blocking signals and the LNA amplifies the GNSS signal enhancing the sensitivity of the receiver. Refer to the [Block diagram](#) for an overview of the RF front-end. The SAM-M10Q offers high GNSS performance for designs with moderate or high RF interference levels.

4.3.2.1 Out-of-band blocking immunity

Out-of-band RF interference may degrade the quality and availability of the navigation solution. Out-of-band immunity limit describes the maximum power allowed at the receiver RF input with no degradation in performance. Minor violation of the immunity limit may reduce C/N0 of the received signals but does not necessarily affect the overall receiver performance. However, a significant violation may reduce receiver sensitivity or cause a complete loss of signal reception. The severity of the interference depends on the repetition rate, frequency, signal level, modulation, and bandwidth of the signal.

A typical out-of-band immunity limit at SAM-M10Q antenna feed point is 0 dBm at 400 - 1460 MHz and 1710 - 3300 MHz. The immunity decreases closer to the GNSS in-band. The limit is defined at room temperature using a test signal with 64QAM modulation and 10 MHz bandwidth similar to an LTE signal.

- ☞ If the out-of-band immunity limit is exceeded, it is recommended to verify that the receiver performance is not affected or is at an acceptable level in the presence of interference.

4.4 Layout

When integrating a GNSS receiver into a PCB, the placement of the components, as well as grounding, shielding, and interference from other digital devices are crucial issues that need to be considered very carefully.

SAM-M10Q GNSS patch antenna module is intended to be placed in the middle of a 50 x 50 mm GND size board. A larger or a smaller ground plane can also be used. Note that when using a smaller ground plane than 40 x 40 mm, the performance may decrease significantly. It is recommended not to place anything closer than 10 mm to each edge of SAM-M10Q.

- ☞ For applications using cellular antennas, increase the distance between both antennas as much as possible.

Figure 21 shows an example of a PCB design. The yellow area represents the solid ground plane in the top layer. The module is placed in the middle of the board and no signal traces are allowed below. If necessary, keep at least 20 mm distance from the module edge when swapping any signal from the top to other layers. The supply and digital lines of the module are short on the top layer and routed to other layers.

The GND plane below the module is filled with GND vias to increase GND reference and to tie separate ground plane areas together.

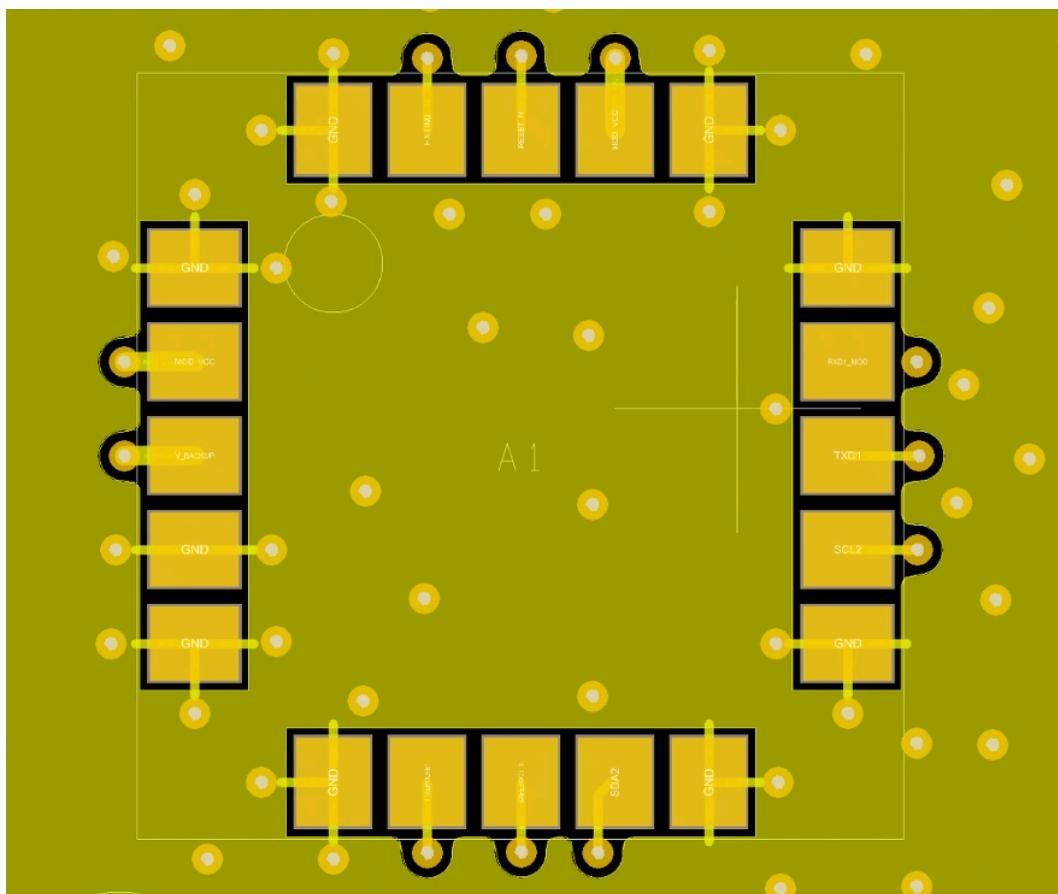


Figure 21: Example of recommended PCB layout (top layer)

Note that all the GND pads can be connected to the GND plane with airgaps, working as thermal reliefs during the soldering process, as shown in Figure 22.

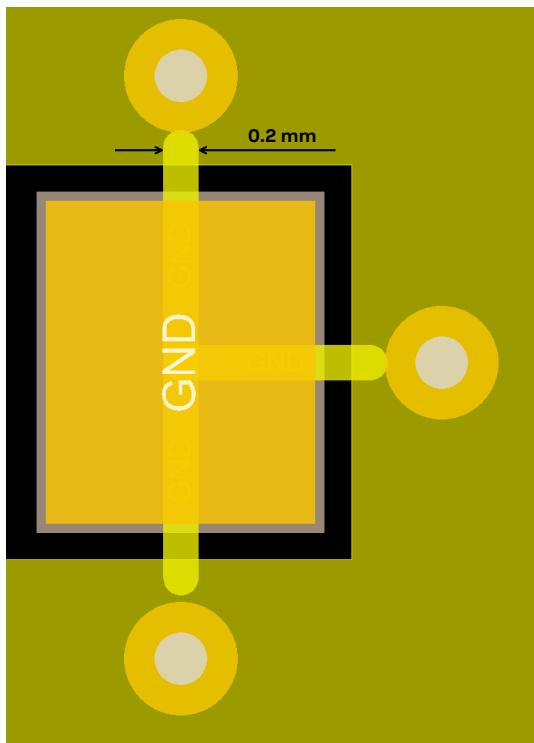


Figure 22: Example of thermal relief in a GND pad

Care must also be exercised with placing the receiver in proximity to circuitry that can emit heat. Temperature-sensitive components inside the module, like TCXOs, are sensitive to sudden changes in ambient temperature which can adversely impact satellite signal tracking. Sources can include co-located power devices, cooling fans or thermal conduction via the PCB.

The GND planes can conduct heat to other elements, but they can act as heat dissipators as well. Increasing the number of GND vias helps to decrease sudden temperature changes.

-  High temperature drift and air vents can affect the GNSS performance. For best performance, avoid high temperature drift and air vents near the module.

4.4.1 Package footprint, copper and solder mask

Figure 23 shows the footprint of the SAM-M10Q form factor. The suggested solder mask opening is 0.1 mm wider than the pad dimensions, as shown in Figure 24.

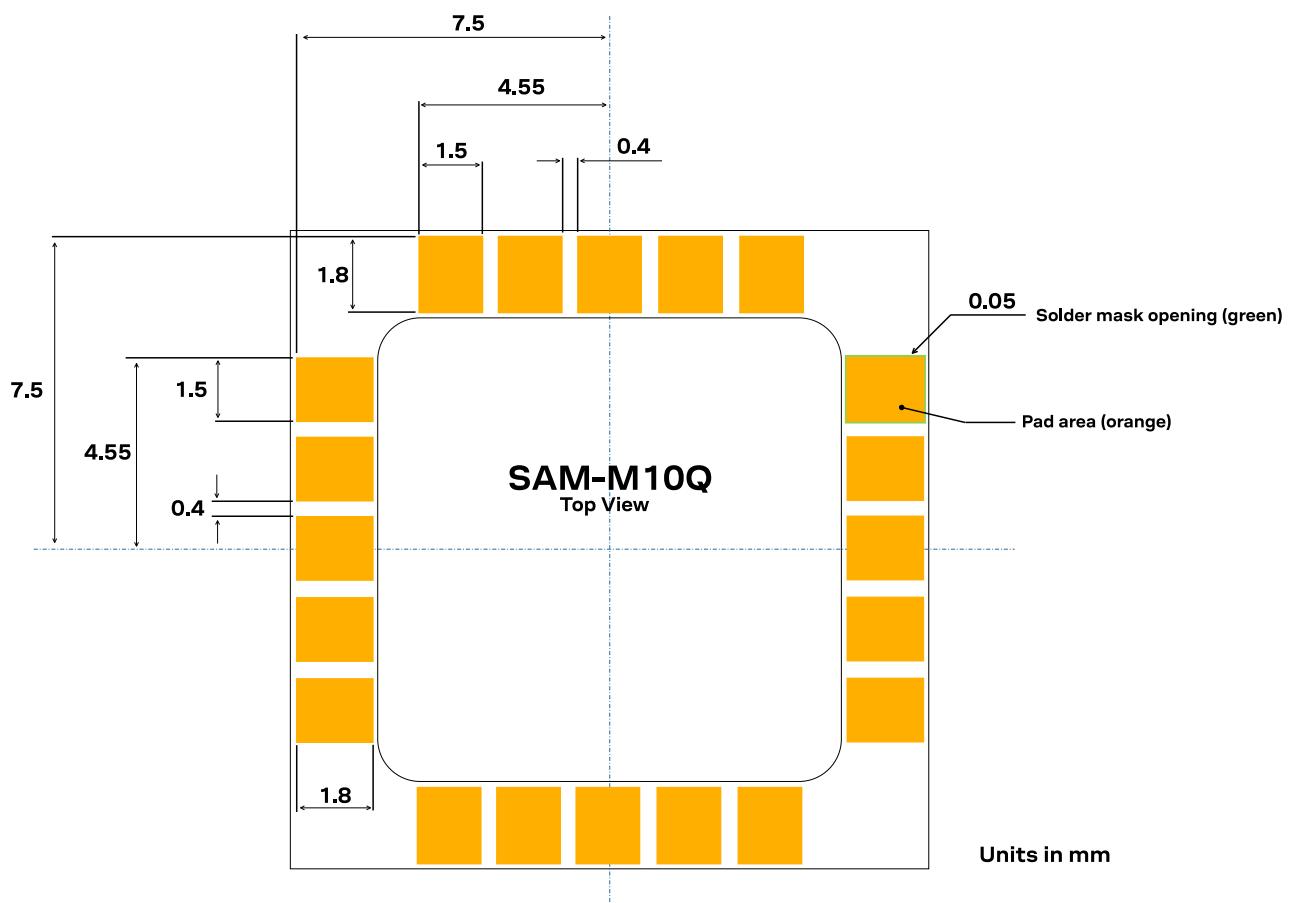


Figure 23: SAM-M10Q footprint and solder mask opening

Figure 24 shows the paste mask dimensions for each pad. Recommended stencil thickness is 120 μm .

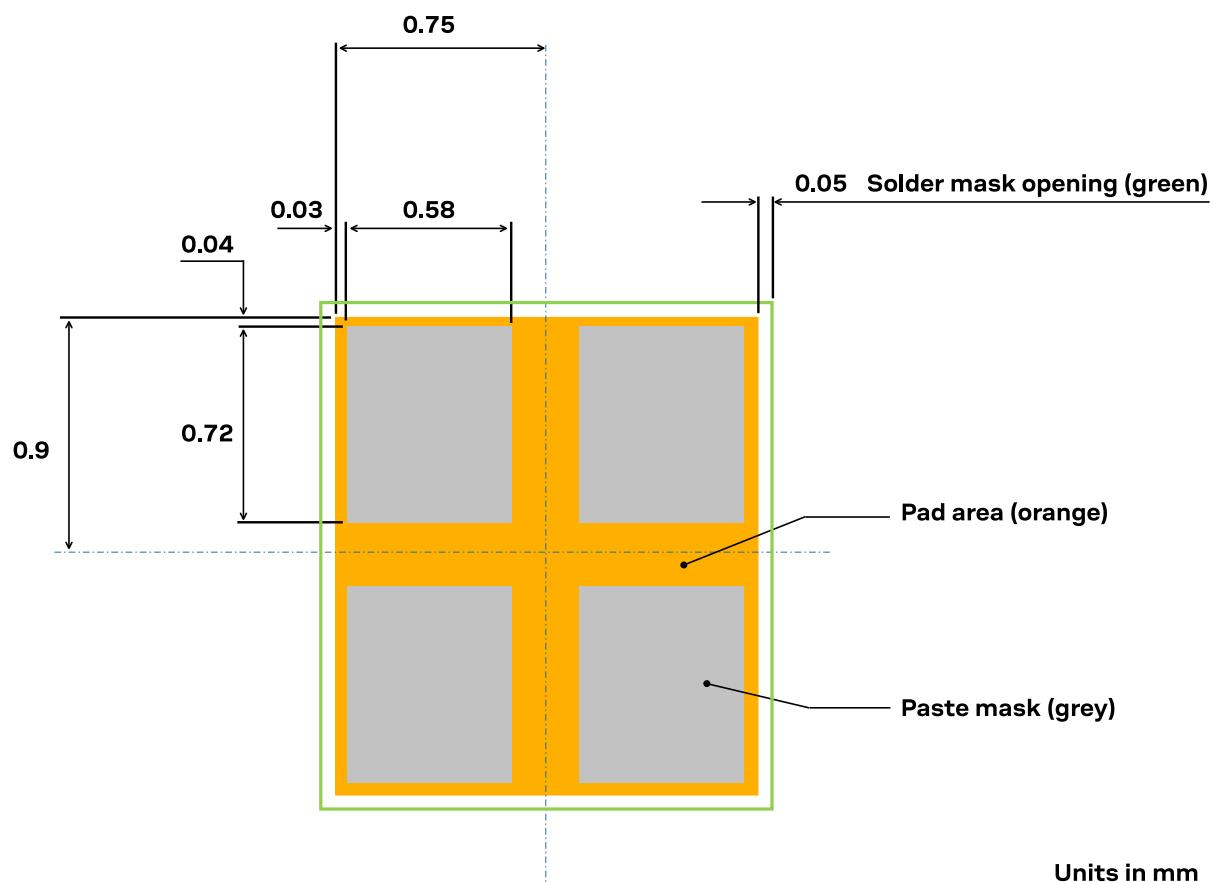


Figure 24: Paste mask detail for each pad

 These are only recommendations and not specifications. The exact geometry, distances, stencil thicknesses, and solder paste volumes must be adapted to the customer's specific production processes (for example, soldering).

5 Product handling

5.1 Safety

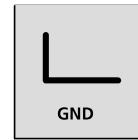
5.1.1 ESD precautions

 u-blox chips and modules contain highly sensitive electronic circuitry and are electrostatic sensitive devices (ESD). Observe precautions for handling! Failure to observe these precautions can result in severe damage to the component!

- Unless there is a galvanic coupling between the local GND (that is, the work desk) and the PCB GND, then the first point of contact when handling the PCB must always be between the local GND and PCB GND.
- Before mounting an antenna patch, connect ground of the device.



Connect first
to local GND

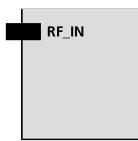


GND

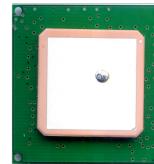
- When handling the RF pin, do not come into contact with any charged capacitors and be careful when contacting materials that can develop charges (for example, patch antenna ~10 pF, coax cable ~50-80 pF/m or soldering iron).



ESD Sensitive!



- To prevent electrostatic discharge through the RF input, do not touch any exposed antenna area. If there is any risk that such exposed antenna area is touched in a non-ESD protected work area, implement proper ESD protection measures in the design.



- When soldering RF connectors and patch antennas to the receiver's RF pin, make sure to use an ESD-safe soldering iron (tip).



ESD safe only!



5.1.2 Safety precautions

The SAM-M10Q modules must be supplied by an external limited power source in compliance with the clause 2.5 of the standard IEC 60950-1. In addition to external limited power source, only Separated or Safety Extra-Low Voltage (SELV) circuits are to be connected to the module including interfaces and antennas.

 For more information about SELV circuits see section 2.2 in Safety standard IEC 60950-1.

5.2 Packaging

The SAM-M10Q modules are delivered as hermetically sealed, reeled tapes in order to enable efficient production, production lot set-up and tear-down. For more information, see the u-blox packaging information reference [4].

5.2.1 Reels

SAM-M10Q modules are deliverable in quantities of 250 pieces on a reel. They are shipped on reel type A, as specified in the u-blox Packaging information reference [4].

5.2.2 Tapes

Figure 25 shows the feed direction and illustrates the orientation of the components on the tape.

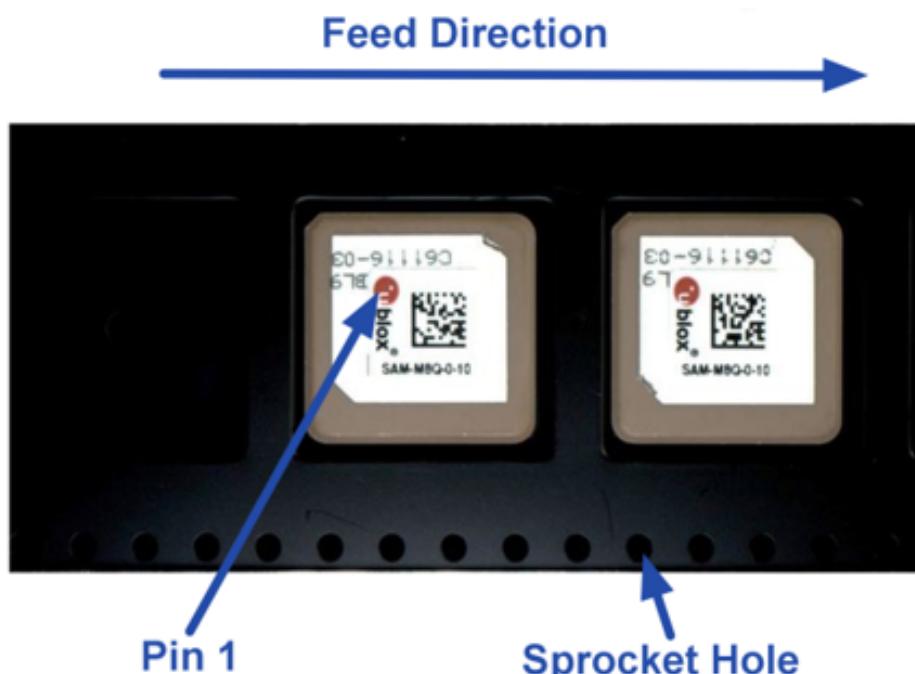


Figure 25: Orientation of the components on the tape

Pin 1 is indicated by the u-blox logo. The feed direction into the pick and place pick-up is from the reel (located on the left of the figure) towards right, and the tape is fed to the right.

The dimensions of the tape are specified in Figure 26 (measurements in mm).

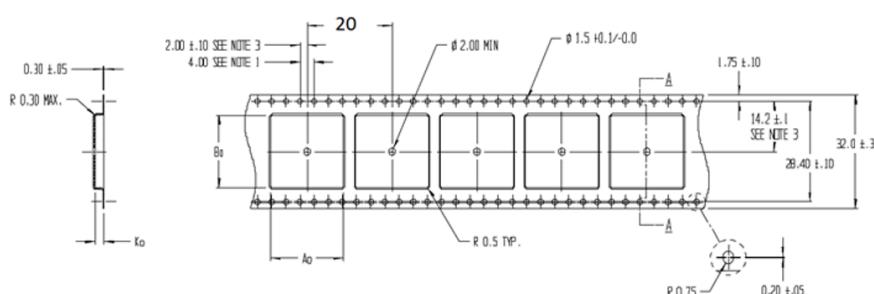


Figure 26: Tape dimensions (mm)

Parameter	Value	Units
Ao	16.2	mm
Bo	16.2	mm
Ko	7.0	mm

Table 21: SAM dimensions

5.2.3 Moisture sensitivity level

The moisture sensitivity level (MSL) for SAM-M10Q modules is specified in the table below.

Package	MSL level
Antenna module (professional grade)	4

Table 22: MSL level

For MSL standard see IPC/JEDEC J-STD-020, and J-STD-033 that can be downloaded from www.jedec.org. For more information regarding moisture sensitivity levels, labeling, storage and drying, see the u-blox packaging information reference [4].

5.3 Soldering

Reflow soldering procedures are described in the IPC/JEDEC J-STD-020 standard.

-  When populating the modules, make sure that the pick and place machine is aligned to the copper pins of the module and not to the module edge.

Soldering paste

Use of “no clean” soldering paste is highly recommended, as it does not require cleaning after the soldering process. The paste in the example below meets these criteria.

- Soldering paste: OM338 SAC405 / Nr.143714 (Cookson Electronics)
- Alloy specification: Sn 95.5 / Ag 4 / Cu 0.5 (95.5% tin/ 4% silver/ 0.5% copper)
- Melting temperature: 217 °C
- Stencil: The exact geometry, distances, stencil thicknesses and solder paste volumes must be adapted to the customer's specific production processes.

Reflow soldering

A convection-type soldering oven is highly recommended over the infrared-type radiation oven. Convection-heated ovens allow precise control of the temperature, and all parts will heat up evenly, regardless of material properties, thickness of components and surface color.

As a reference, see “IPC-7530 Guidelines for temperature profiling for mass soldering (reflow and wave) processes”, published in 2001.

Preheat phase

During the initial heating of component leads and balls, residual humidity will be dried out. Note that the preheat phase does not replace prior baking procedures.

- Temperature rise rate: max 3 °C/s. If the temperature rise is too rapid in the preheat phase, excessive slumping may be caused
- Time: 60 – 120 s. If the preheat is insufficient, rather large solder balls tend to be generated. Conversely, if performed excessively, fine balls and large balls will be generated in clusters
- End temperature: 150 – 200 °C. If the temperature is too low, non-melting tends to be caused in areas containing large heat capacity

Heating - reflow phase

The temperature rises above the liquidus temperature of 217 °C. Avoid a sudden rise in temperature as the slump of the paste could become worse.

- Limit time above 217 °C liquidus temperature: 40 – 60 s
- Peak reflow temperature: 245 °C

Cooling phase

A controlled cooling prevents negative metallurgical effects of the solder (solder becomes more brittle) and possible mechanical tensions in the products. Controlled cooling helps to achieve bright solder fillets with a good shape and low contact angle.

- Temperature fall rate: max 4 °C/s

To avoid falling off, the modules should be placed on the topside of the board during soldering.

The final soldering temperature chosen at the factory depends on additional external factors such as the choice of soldering paste, size, thickness and properties of the base board, etc. Exceeding the maximum soldering temperature in the recommended soldering profile may permanently damage the module.

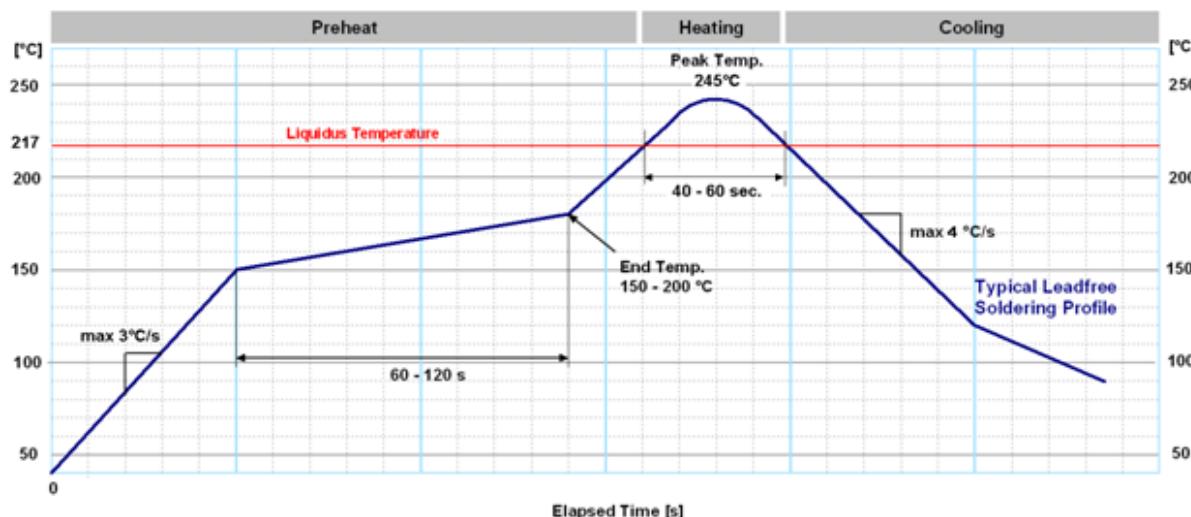


Figure 27: Soldering profile

Modules **must not** be soldered with a damp heat process.

Optical inspection

After soldering the module, consider optical inspection.

Cleaning

Do not clean with water, solvent, or ultrasonic cleaner:

- Cleaning with water will lead to capillary effects where water is absorbed into the gap between the baseboard and the module. The combination of residues of soldering flux and encapsulated water leads to short circuits or resistor-like interconnections between neighboring pads.
- Cleaning with alcohol or other organic solvents can result in soldering flux residues flowing underneath the module, into areas that are not accessible for post-cleaning inspections. The solvent will also damage the sticker and the printed text.
- Ultrasonic cleaning will permanently damage the module, in particular the quartz oscillators.

The best approach is to use a “no clean” soldering paste to eliminate the cleaning step after the soldering.

Repeated reflow soldering

- ⚠ Repeated reflow soldering processes or soldering the module upside down are not recommended.

A board that is populated with components on both sides may require more than one reflow soldering cycle. In such a case, the process should ensure the module is only placed on the board submitted for a single final upright reflow cycle. A module placed on the underside of the board may detach during a reflow soldering cycle due to lack of adhesion.

The module can also tolerate an additional reflow cycle for rework purposes.

Wave soldering

Base boards with combined through-hole technology (THT) components and surface-mount technology (SMT) devices require wave soldering to solder the THT components. Only a single wave soldering process is encouraged for boards populated with modules.

Rework

We do not recommend using a hot air gun because it is an uncontrolled process and can damage the module.

- ⚠ Use of a hot air gun can lead to overheating and severely damage the module. Always avoid overheating the module.

After the module is removed, clean the pads before reapplying solder paste, placing and reflow soldering a new module.

- ⚠ Never attempt a rework on the module itself, e.g. by replacing individual components. Such actions will immediately void the warranty.

Conformal coating

Certain applications employ a conformal coating of the PCB using HumiSeal® or other related coating products. These materials affect the RF properties of the GNSS patch antenna module.

- ⚠ Conformal coating of the module will void the warranty.

Casting

Casting materials affect the RF properties of the GNSS patch antenna, including resonant frequency shifts, and are not recommended for antenna modules.

- ⚠ Casting will void the warranty.

Use of ultrasonic processes

Some components on the module are sensitive to ultrasonic waves. Use of any ultrasonic processes (cleaning, welding etc.) may cause damage to the receiver.

- ⚠ u-blox offers no warranty against damages to the module caused by ultrasonic processes.

Oxidation of patch antenna

The patch antenna is metalized by silver paste and thus tends to oxidize and changes color. This is normal and is not a case for warranty.

Appendix

A Migration

u-blox is committed to ensure that products in the same form factor are backwards compatible over several technology generations. This section describes important differences to consider when migrating from u-blox M8 to u-blox M10.

A.1 Hardware changes

SAM-M10Q is pin-to-pin compatible with the SAM-M8Q module. Table 23 lists the key **hardware-related changes**.

Feature	Change	Action needed / Remarks
VCC, V_IO	New power supply ramp requirements. The V_IO supply voltage must not be higher than VCC + 0.3 V. The absolute maximum rating for the minimum voltage at V_IO and VCC is -0.3 V.	There is no change in the nominal supply voltage range 2.7 V - 3.6 V. Refer to the minimum and maximum VCC and V_IO ramp requirements in the SAM-M10Q data sheet [1].
V_BCKP supply range	SAM-M10Q: 1.65 V - 3.6 V SAM-M8Q: 1.4 V - 3.6 V	Hardware change is required when migrating from a design with less than 1.65 V V_BCKP supply.
Backup current	The hardware backup and software standby current in SAM-M10Q is higher than in SAM-M8Q module.	Check the backup battery capacity for suitability for the target design requirements.
SAW-LNA RF front-end	Internal component changes.	Improved out-of-band immunity.
SAFEBOOT_N, TIMEPULSE	The SAFEBOOT_N pin is internally connected to TIMEPULSE pin through a 1 kΩ series resistor.	Do not drive the TIMEPULSE pin low at startup because it will put the receiver in safeboot mode.
RESET_N	Resetting the receiver clears the BBR content. SAM-M10Q: time information, GNSS orbit data and receiver configuration stored in BBR cleared SAM-M8Q: time information cleared	TTFF after resetting SAM-M10Q with the RESET_N pin is similar to performing a cold start.
Digital IO	Digital IO pins must not be driven if VCC and V_IO are not supplied. The electrical specification for digital IO pins has changed. The voltage levels are defined at 1 mA current (exception: TIMEPULSE pin 2 mA).	Do not drive IO pins when VCC and V_IO are not supplied. Otherwise permanent damage may result. If driving the IO pins cannot be avoided, buffers are required to isolate the pins. Additional external pull-up resistors may be required to increase the drive strength of the digital IO output pin or adjust the load accordingly.

Table 23: SAM-M10Q hardware features

Refer to the SAM-M10Q data sheet [1] for product specification.

A.2 Software changes

Table 24 presents a summary of the key **software-related changes** between u-blox M10 and u-blox M8.

Feature	Change	Action needed / Remarks
Signals		

Feature	Change	Action needed / Remarks
Default GNSS configuration	SAM-M10Q: GPS, Galileo, GLONASS, BeiDou B1C, QZSS and SBAS. SAM-M8Q: GPS, GLONASS, QZSS and SBAS.	Code change (optional)
BeiDou B1C	New signal. BeiDou satellite IDs up to 63 supported. AssistNow and power save mode not supported.	Code change (optional)
SBAS	New SBAS PRN selection: 123, 126-129, 131, 133, 136-138.	Code change (optional)
QZSS L1S	SLAS corrections are now applied for navigation.	Code change (optional)
QZSS IMES	Not supported in current firmware.	Code change (optional)
Protocols		
NMEA	Supports NMEA V4.11, V4.10, V4.0, V2.3, and V2.1. NMEA V4.11 is enabled by default. NMEA GSV messages are grouped separately in u-blox M10 as compared to u-blox M8 in the case of zero signal C/No level. In M10, if the signal's C/No level is 0, the FW will set the "signal ID" to zero as an unknown signal for not being able to track them. FW will then create separate GSV messages, grouping the untracked signals with zero C/No and other messages for grouping the tracked signals with non-zero C/No. For the untracked signals message group, the C/No field is marked as empty (i.e.,). There is no configuration to revert this to the u-blox M8 format.	Code change (optional)
RTCM	Not supported in current firmware.	Code change
General		
Configuration concept	New configuration scheme using UBX-CFG-VALSET and UBX-CFG-VALGET messages.	Code change
Navigation update rate	Navigation update rate up to 18 Hz for single GNSS, 10 Hz for 3 GNSS, and 5 Hz for 4 GNSS constellations.	Code change (optional)
Super-S	New feature. Improves performance under weak signal conditions. Enabled by default in current firmware.	-
Power save mode (PSM)	PSM configuration options have changed: PSMCT: 4 Hz navigation rate not supported. PSMCT period is configured with CFG-RATE configuration group. BeiDou B1C not supported in power save modes.	Code change (new configuration messages)
Altitude limit	The maximum altitude limit increased to 80 000 m.	-
Features		
AssistNow	Simultaneous operation of AssistNow Online, Offline and Autonomous. New AssistNow Offline data download options are available. The time period can be limited to a resolution of days to reduce the data package size.	Code change (optional)
CloudLocate	New feature. Position calculation in the cloud provided by the u-blox CloudLocate service. Extends the life of energy-constrained IoT applications. Low payload messages supported.	Code change (optional)
Geofencing	Not supported in current firmware.	Code change
Data logging	Not supported in current firmware.	Code change
Galileo return link message	Galileo search and rescue (SAR) return link message UBX-RXM-RLM.	Code change (optional)
RF spectrum view	New message. UBX-MON-SPAN shows in-band RF spectrum around the GNSS band. This can be used to identify potential in-band RF interference sources in the design.	Code change (optional)
Location batching	New feature that can be used to reduce power consumption by saving navigation solutions on the receiver for up to 10 minutes (at 1 Hz) and polling the solutions afterwards to a host MCU.	Code change (optional)

Feature	Change	Action needed / Remarks
Protection level	New feature. Real-time position accuracy estimate with 95% confidence level with the message UBX-NAV-PL.	Code change (optional)
Security		
Message integrity	New feature. Authentication of data output based on private/public key pair.	Code change (optional)
Unique chip identifier	A unique chip identifier output in boot screen and in the UBX-SEC-UNIQID message.	Code change (optional))
Configuration lock	New security feature that is enabled with CFG-SEC-CFG_LOCK message for locking the receiver configuration.	Code change (optional)

Table 24: SAM-M10Q software features

Refer to the firmware Release notes [2] and the Interface description [3] for more information on supported features and messages in u-blox M10 receiver.

B Reference designs

B.1 Typical design

Here are some key features for a SAM-M10Q typical design:

- VCC and V_IO are connected together to a single supply as shown in Figure 28.
- V_BCKP supply is optional. If present, the hardware backup mode is supported. This mode maintains the time and GNSS orbit data in the battery-backed RAM memory if the main supply is switched off.
- If there is no backup supply, time aiding with the UBX-MGA-INI-TIME_UTC message (optionally with a timing signal at the EXTINT pin) and the GNSS orbit data from the AssistNow services or stored on the host controller can be used to reduce the TTFF.
- UART and I2C communication interfaces are available.
- For an absolute minimum design using UART, other PIOs (RESET_N, EXTINT, TIMEPULSE, SDA, SCL, SAFEBOOT_N) can be left open.

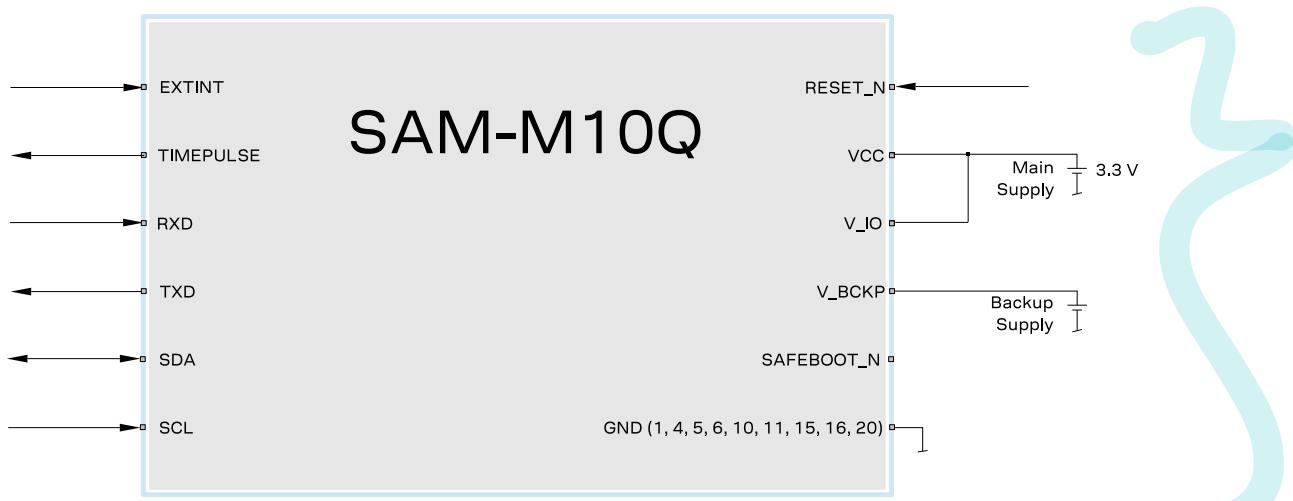


Figure 28: Typical design

Related documents

- [1] SAM-M10Q Data sheet, [UBX-22013293](#)
- [2] u-blox M10 SPG 5.10 Release notes, [UBX-22001426](#)
- [3] u-blox M10 SPG 5.10 Interface description, [UBX-21035062](#)
- [4] u-blox Packaging information reference, [UBX-14001652](#)

 For regular updates to u-blox documentation and to receive product change notifications please register on our homepage <https://www.u-blox.com>.

Revision history

Revision	Date	Name	Status / comments
R01	29-Aug-2022	jesk, rmak, imar, msul	Initial release. Product status is available in the data sheet.[1].
R02	01-Jun-2023	msul, imar, rmak, mban	Added sections <ul style="list-style-type: none">• High performance navigation update rate configuration• OTP memory configuration

Contact

u-blox AG

Address: Zürcherstrasse 68
 8800 Thalwil
 Switzerland

For further support and contact information, visit us at www.u-blox.com/support.