EUREF-2003 Symposium, Toledo, 4-7 June 2003

POLARX2, A NEW GPS RECEIVER FOR GEODETIC APPLICATIONS

Andrew Simsky(Septentrio, Belgium a.simsky@septentrio.com*),*
Jean-Marie Sleewaegen (Septentrio, Belgium jm.sleewaegen@septentrio.com*),*
and
Carine Bruyninx (Royal Observatory of Belgium, C.Bruyninx@oma.be*)*

Table of Contents

Introduction

Septentrio Satellite Navigation, a company based in Leuven, Belgium, has recently released a new high-end PolaRx2 receiver. The PolaRx2 is a dual-frequency all-in-view GPS/SBAS receiver designed for precision-demanding geodetic and timing applications including reference networks. Development of the PolaRx2 is a response to the needs of the geodetic community to have a reliable low-noise geodetic-grade receiver at affordable cost. PolaRx2 is built around the proprietary GNSS chipset and implements the principles of flexible multi-process software architecture, which significantly reduces development effort.

TCP/IP connectivity is an innovative feature of the new receiver. It allows remote control and data logging by several users simultaneously. With the TCP/IP support, a new design of the reference network applications becomes possible. The network of receivers may be interconnected via the Internet so that there is no need to have controlling computers at each of the reference stations, and all the reference stations can be directly controlled from the central computer. Septentrio provides RxControl, a powerful graphical user interface, which allows to control the receiver either via a local PC or remotely through the network.

The purpose of this paper is to describe the architecture of PolaRx2, present the analysis of its performance characteristics and measurement noise, and demonstrate results for a sample geodetic application. Possible applications of the TCP/IP feature of the receiver are discussed.

Architecture of PolaRx2

All the essential building blocks of PolaRx2 are of proprietary Septentrio's design, including GreCo, the baseband processor, and GreFe, the analog Front-End chip. PolaRx2 is, in a nutshell, a Linux computer running embedded GNSS firmware. The design of the firmware is based on the principle of flexible multi-process software architecture: the software modules of PolaRx2 run as separate processes under Linux, exchanging data through a TCP/IP-based interface. This approach simplifies the development of new modules and user-side applications.

Hardware architecture

The layout or receiver hardware is presented in Figure 1. Received RF signal is digitized by 2 GreFe front-end chips (one for L1, another for L2). Tracking is performed by the GreCo baseband digital chip. The GreCo contains 48 hardware single-frequency channels, which are grouped in 16 dual-frequency channels. In PolaRx2, each dual-frequency channel consists of 3 single-frequency channels: for C/A code, P1, and P2 codes. L1 and L2 phase and Doppler measurements are also generated.

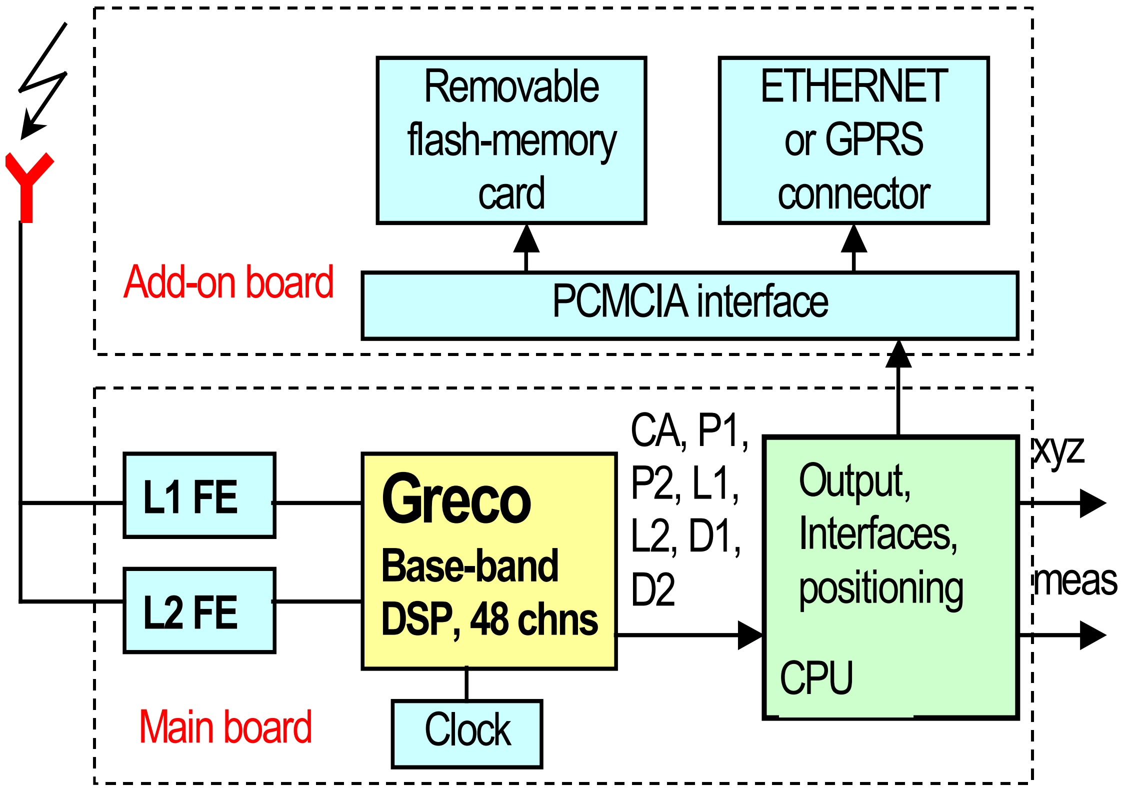

Figure 1. Hardware architecture of PolaRx2

The main CPU of PolaRx2 is a 486 MachZ processor running a Linux-based operating system. The CPU is connected to other elements of hardware via the bus of proprietary architecture (similar to PCI). The main board has an external connector into which an add-on board with a PCMCIA socket may be plugged. Currently there are 2 supported options for PCMCIA cards: Ethernet adaptor or a memory card for on-board logging. The possibility of the GPRS interface is also foreseen. PolaRx2 as an end-user product comes in a box, which contains both main and add-on boards.

Software architecture

Software architecture of PolaRx2 consists of several applications interconnected via an Inter-Process Communication (IPC) mechanism. Each application is running as a separate process under Linux. Because the IPC is based on the TCP/IP protocol, those separate applications could even be running on external computers, connected via Ethernet. This distributed processing approach is quite useful for development and debugging: the module under development may run on a developer's PC, while the rest of the firmware may run on the receiver board. Introduction of new applications is made easy with the unified interface of the IPC "software bus".

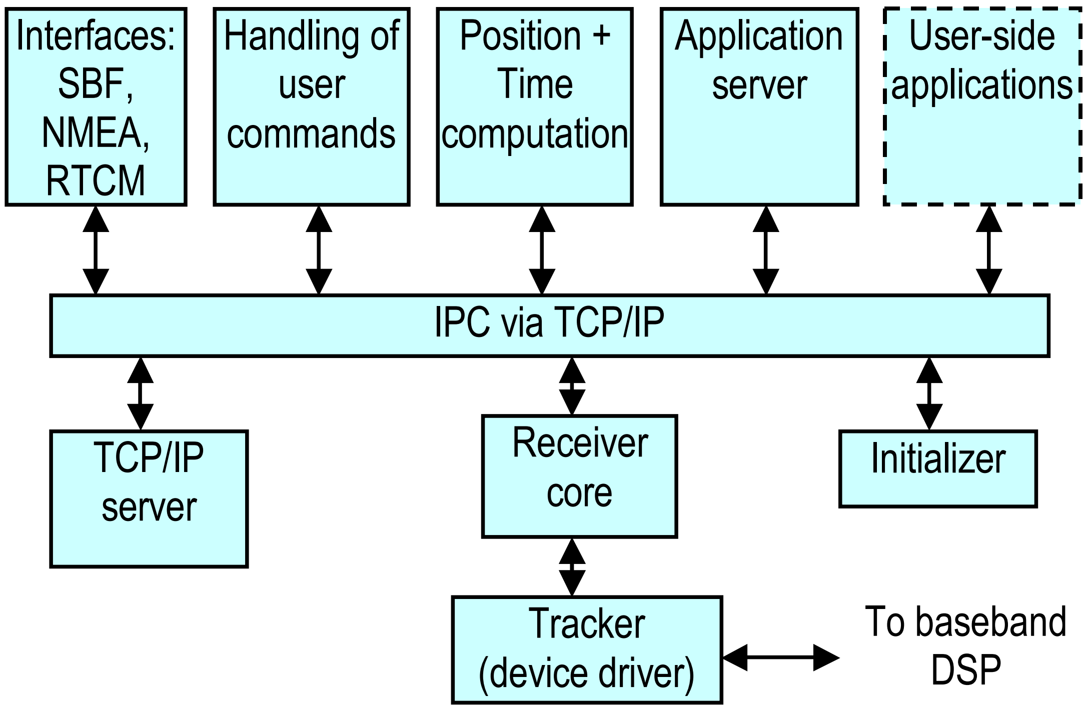

Figure 2. Software architecture of PolaRx2

The main building blocks of the PolaRx2 firmware are as follows.

Tracker (device driver)

The tracker acts as a device driver for the baseband DSP and performs the following tasks:

- Signal acquisition,
- Signal tracking (closing of PLLs, DLLs, implemented in the baseband DSP),
- Bit synchronization,
- Generation of raw measurements.

Application server

The application server takes care of correct delivery of messages between all the other applications. The messages are exchanged over the IPC software bus. The messages transmitted by IPC may include raw measurements, computed position, user commands, initialization strings, and various internal data.

Receiver core application

The receiver core is responsible for GNSS-specific tasks of the receiver. It communicates with the device driver to start acquisition of satellite signals and subsequently obtain raw measurements and the tracking status. The receiver core performs position/time computations and the generation of output data.

Interface formats (SBF, NMEA, RTCM)

The receiver is using several digital formats to communicate data to and from the outside devices. SBF is an acronym for the Septentrio Binary Format, the compact binary format designed to output measurements, computed position and other user information. NMEA format is used mostly to output position and is widely accepted as an international standard for various external devices and s/w. RTCM is a standard format to transmit DGPS/RTK corrections data if the receiver is used as an RTK/DGPS rover or a base station. Though the RINEX format, widely used in geodesy, is not available directly from the receiver, the software converter from SBF to RINEX is provided.

TCP/IP server

PolaRx2 can communicate to the outside worlds via 2 serial ports and a number of TCP/IP channels. TCP/IP connectivity is implemented via Ethernet as a physical layer. Receiver s/w implements a TCP/IP server, which is able to respond to TCP/IP requests from outside users. Examples of applications based on the TCP/IP connectivity are presented in one of the following sections.

Position + time computation

PolaRx2 can compute position in 5 modes:

- Standalone dual-frequency,
- SBAS-corrected,
- Code-based DGPS,
- RTK with floating ambiguities,
- RTK with integer ambiguities.

Positioning modes are listed in the order of increasing precision

User-side applications

The currently implemented software architecture naturally allows to plug in user-side applications. The user applications have to implement the IPC interface in order to exchange data with other processes in the receiver. The current CPU load varies from 15-20% at the data rate of 1 Hz to 35-40% at the data rate of 10Hz. This means that a wide range of user-side applications is allowed. These applications may perform pre-processing of the data, integrity analysis, implementation of individual user interfaces etc.

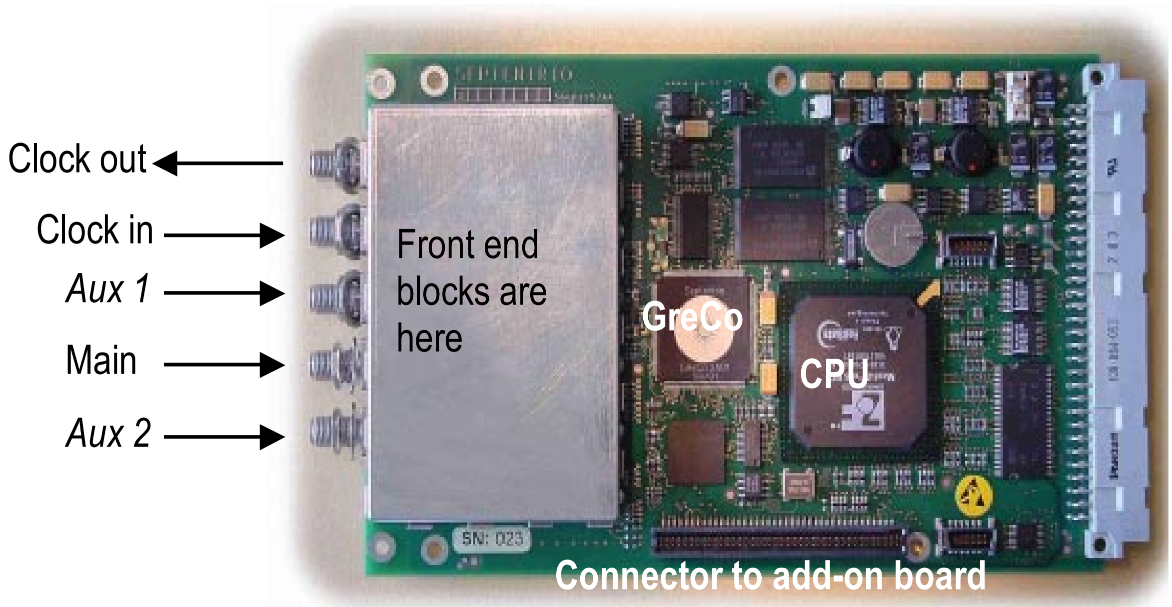

Figure 3. PolaRx2 is implemented as a EuroCard-sized board. Additional RF input sockets are stipulated for auxiliary antennas (multi-antenna version only) and external clock reference for time transfer applications.

Multi-antenna version (PolaRx2@)

PolaRx2 is implemented as a EuroCard-sized PCB board shown in Figure 3. This picture shows the multi-antenna version of the receiver (PolaRx2@), which is able of receiving and processing signals from 3 antennas simultaneously. PolaRx2@ is designed for attitude determination, deformation monitoring and other similar applications.

The hardware channels of PolaRx2 (the total of 48) are fully re-configurable through user commands. The hardware channels can be configured for dual-frequency and single-frequency tracking of GPS signals from each of the 3 antennas. The main antenna is foreseen to be either dual-frequency or single-frequency, while two auxiliary antennas can be only single-frequency. More details on the multi-antenna version of the receiver can be found in other Septentrio's publications (see for example [1]).

RxControl

Generally, PolaRx2 can be controlled through any communication software, which allows to send command strings and receive responses. However, we provide a dedicated GUI program, RxControl, which allows users to control the receiver and log data with the help of intuitive graphical interface.

RxControl is implemented as a Java application, which means that it can be installed on practically any computer platform on which Java virtual machine is available. The two most popular platforms are Microsoft Windows and Linux. We pay particular attention to implement all the user commands in RxControl. This means that for each receiver command, which can be sent via a command-line interface, there exists a graphical item in RxControl: a drop-down list, a check box or a button. All the user commands starting from the most essential logging functionality to the advanced channel reallocation commands are accessible in an easy and intuitive way.

Aside from providing intuitive graphical command interface, RxControl provides a number of screens with useful information related to the computed position, tracking of satellites, and other aspects of receiver operation. Some of these screens are shown in Figure 4.

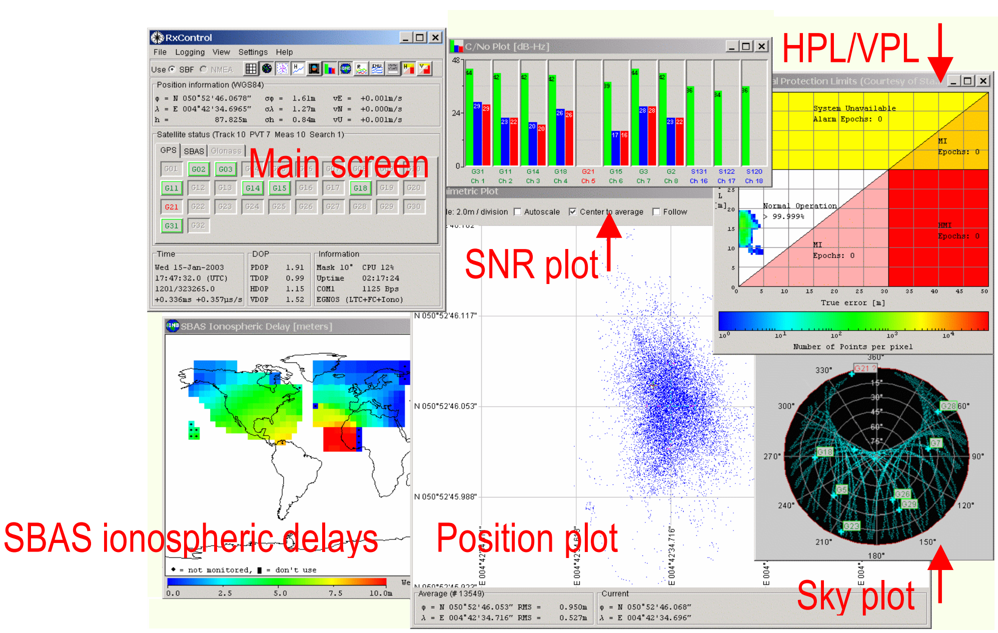

Figure 4. Sample screens of RxControl. Ionospheric delays and HPL/VPL are plotted according to SBAS data.

TCP/IP connectivity

Through its Ethernet interface, PolaRx2 may act as a node on a TCP/IP network. It may be assigned a TCP/IP address and would appear just as another computer on the internet/intranet. This means that the receiver can send and receive commands and transmit/receive data in the same manner through its COM ports and/or through its TCP/IP connections. The number of TCP/IP connections is currently limited by 7 (in principle this number is limited only by the CPU power).

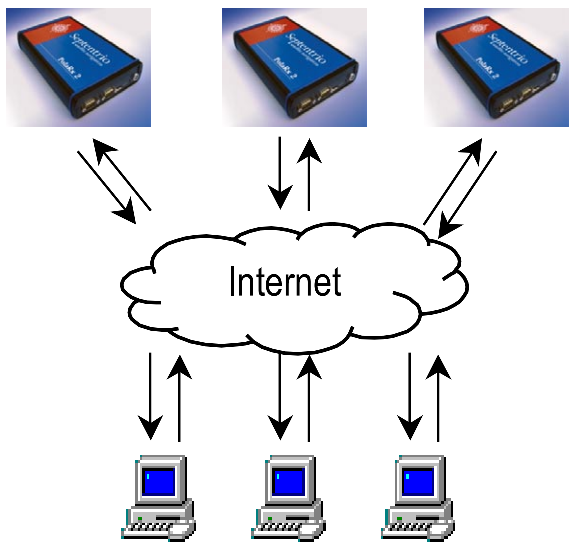

Figure 5. PolaRx2 in a TCP/IP network

The TCP/IP connectivity gives users the power to perform remotely all the same operations that are usually performed with GPS receivers via local PCs: send and receive commands, log data, change the settings of the receiver.

The TCP/IP connectivity allows remote multi-user access. For example, each member of a working team, who may be interested in using PolaRx2, may have independent access to the receiver from his office desk or even from home. This greatly facilitates development of applications with PolaRx2, testing and training. All the remote users must have RxControl installed on their computers. With the help of RxControl, each of them will be able to connect to the receiver, send and receive commands and log data. Each of the users may perform different tasks on the receiver, for example log different kinds of data.

The new concept of the data center application, which becomes possible with the advent of the TCP/IP connectivity, is of particular interest to the geodetic community.

Data center application

The classical data center application implies a network of reference stations, each of which includes a GPS receiver and a controlling PC. Each PC collects data from its own receiver, performs required pre-processing and transmits the data via FTP to the central server. This architecture is shown in Figure 6.

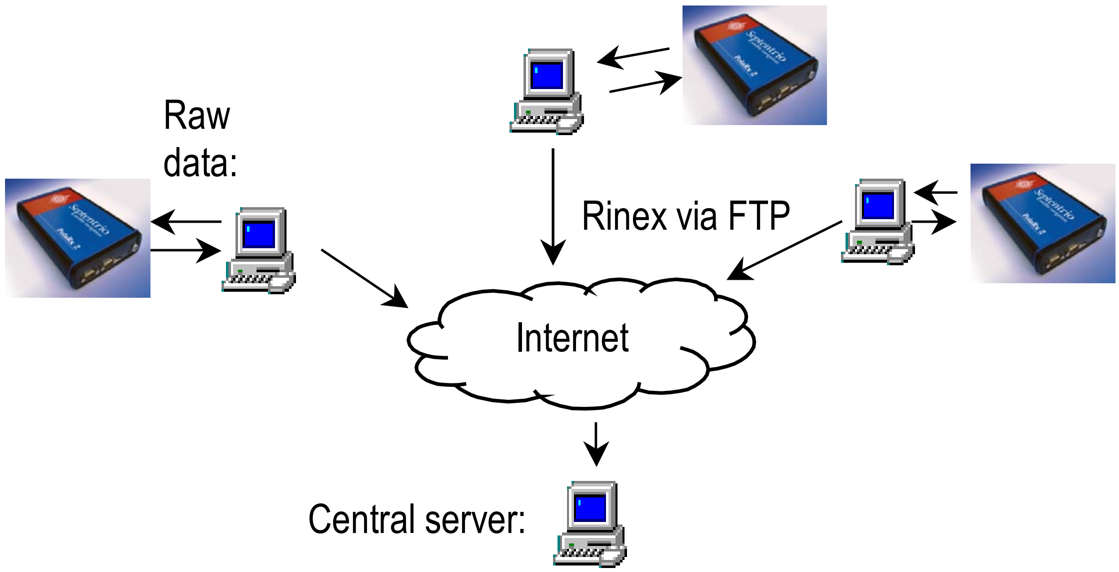

Figure 6. Classical architecture of the data center application.

This architecture has obvious drawbacks as follows:

- Extra costs of hardware/software and higher power consumption due to the use of a separate PC with each receiver.
- Each reference station requires on-site maintenance by a qualified personnel; whatever needs to be changed in the data collection settings, can be done only on-site.
- Communication from reference stations to the data center is only one-way. The central server is only receiving data, but has no control over receivers and the data collection process.

Of course, PolaRx2 can work within the framework of the classical architecture. RxControl, running on the controlling PC, will perform conversion to RINEX and transmission of the files via FTP. However, with the TCP/IP feature an alternative implementation becomes possible, in which all the network is controlled from the central server (Figure 7). For continuous control of the receivers and data logging, the central server must run as many instances of RxControl as required to connect to all the receivers. Each instance of RxControl performs dedicated control of its receiver exactly in the same manner as if it were running on a local PC. Raw data is transferred directly to the central computer and is converted and stored internally. This architecture allows complete control over the whole network from only one computer and by only one operator.

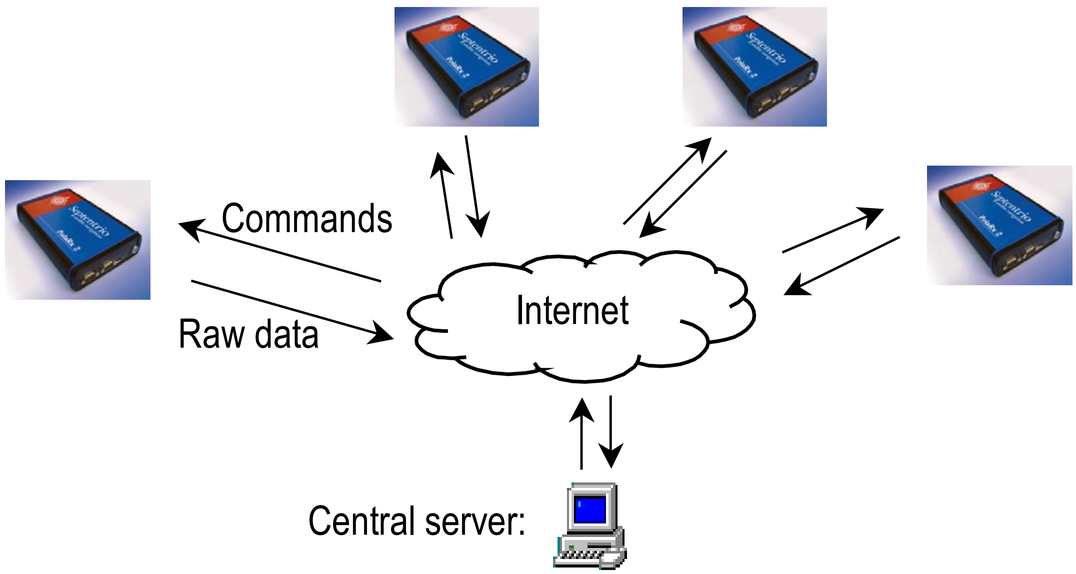

Figure 7. New architecture of the data center application based on the TCP/IP connectivity of PolaRx2. The central server runs multiple instances of RxControl.

TCP/IP connectivity opens a Pandora box of new opportunities not only for geodetic networks, but also for real-time RTK/DGPS networks. Reference receivers themselves can broadcast real-time RTK/DGPS corrections via the Internet.

Measurement noise of PolaRx2

The measurement noise of PolaRx2 is presented in Table 1.

Measurements	STD of tracking noise at 45dB-Hz, 1 Hz
C/A code ranges	0.15 m (0.3 m with APME)
P1 and P2 code ranges	0.1 m (no Xcorrelation with C/A code)
L1 phase	0.2 mm (full wavelength)
L2 phase (semicodeless tracking)	0.4 mm (static mode), 1.0 mm(kin. mode)
D1 & D2 (Doppler shifts)	0.5 mm/sec (2.5 mHz)

Table 1. Measurement noise in PolaRx2. Presented values are STD of noise for the signal power of 45dB-Hz at a data rate of 1Hz.

C/A code may be tracked with or without APME, the multipath mitigation technique based on the real-time analysis of the shape of the auto-correlation peak[2]. Because APME is based on the estimation of the multipath error, it contains the random error of this estimation and the tracking noise of C/A code measurements with APME is increased. More details on the multipath performance of PolaRx2 will be provided in one of the following sections.

The tracking of P1 and P2 is highly independent from the tracking of C/A code, which can be proven by the low values of cross-correlation coefficients for the random noise between all the pairs of the three code ranges. The values in Table 2 have been computed from double-differenced residuals in post-processing of zero-baseline observation data.

code	**STD, meters**	**Correlation coefficients**		
		C/A	**P1**	**P2**
C/A	0.1	1	9.6E-4	7.2E-4
P1	0.11	9.6E-4	1	8.8E-3
P2	0.12	7.2E-4	8.8E-3	1

Table 2. Correlation coefficients for the random noise of the three code ranges available in PolaRx2. Low values of correlation coefficients prove independence of tracking for all the three pairs of codes.

Phase noise at various data rates

The tracking noise of L1 phase depends upon the measurement data rate due to the mechanism of internal smoothing of phase measurements. This mechanism not only reduces the random tracking noise, but also is important for internal control of the measurement quality and outlier detection. The L1 phase measurements are generated through the following steps:

- Raw measurements are sampled at internal rate of 20Hz (interval of 50 ms).
- For output, measurements are fitted to a quadratic polynomial to reduce the measurement noise.
- Fitting interval is the same as output data interval, but ≤ 5 seconds.

As a result, the measurement noise is a function not only of the signal-to-noise ratio, but also of the data rate:

$$STD, mm = \frac{32.1 \cdot 10^{-\frac{CN_0}{20}}}{\sqrt{data\ \ interval, \sec}}$$

The above formula is obtained through the noise theory and corresponds to the red line in Figure 8. It is apparently in good agreement with experimental values of phase noise at different data rates, obtained through averaging of double-differenced residuals for zero-baseline data sets. The plot shows that for the data intervals longer than 5 seconds the phase noise remains constant – in agreement with the above statement that the fitting interval is never increased above 5 seconds. Therefore the data rate for the data interval of 30 seconds would be the same as for the data rate of 5 seconds.

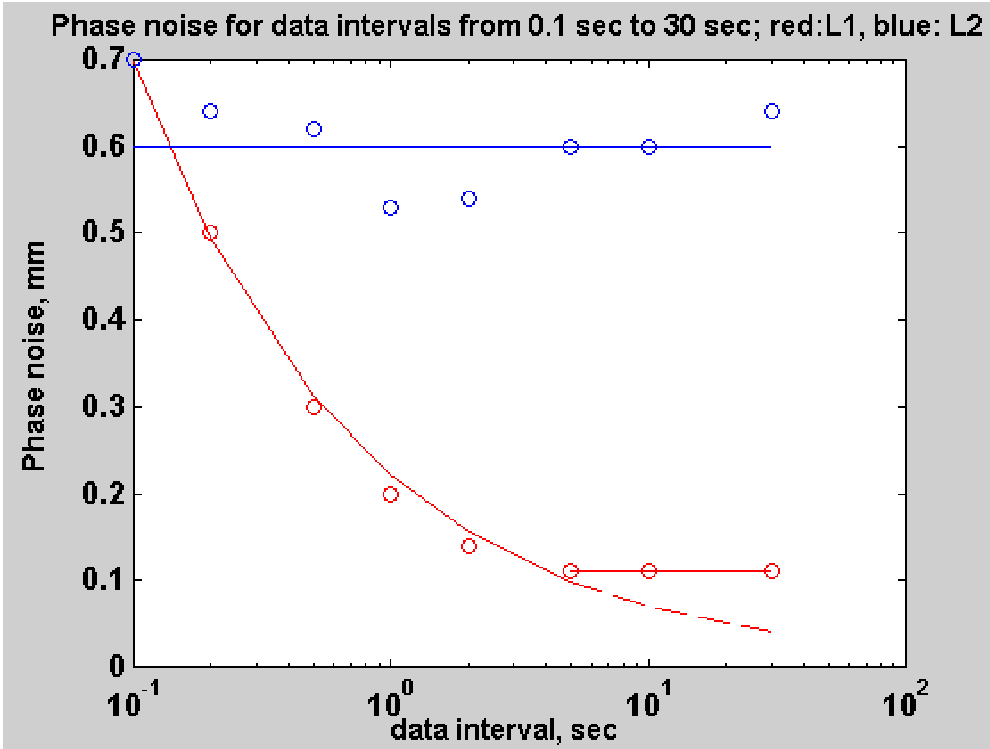

Figure 8. Tracking noise of L1 and L2 phases for data intervals from 0.1 to 30 seconds at a signal-to-noise ratio of 43 dB-Hz, representative of an average power level for actual GPS data.

The left-side limit of the plot shows the noise of raw un-smoothed measurements, which is about 0.7 mm. This limit corresponds to the data rate of 10Hz.

The smoothing of phase is useful for many applications with low-noise requirements. One remarkable consequence of low-noise phase measurements is the significant reduction of the noise of Doppler measurements and the ensuing reduction of the noise of velocity estimates. It can be shown that the actual phase noise of PolaRx2 at the data rate of 1Hz corresponds to exceptionally low velocity noise of about 2-3mm/sec.

However, for many reference station applications, the uniformity of data is the highest priority. In this case smoothing is undesirable because smoothed phase contains smoothed short-term satellite clock errors and should not be combined with non-smoothed measurements from other receivers on the network. Therefore the above-described phase-smoothing mechanism in PolaRx2 is optional. It may be switched on/off by user commands.

For the semi-codeless tracking of L2, PolaRx2 provides a possibility to vary the PLL bandwidth in order to adjust it to the needs of the applications. For static applications the narrowest setting will result in the lowest value of L2 noise. For kinematic applications, where acquisition threshold is the highest priority, the PLL bandwidth should be increased up to the “kinematic” setting thus increasing the noise value by a factor of about 2.

Multipath mitigation in PolaRx2

Table 3 contains the values of average multipath noise for Septentrio's antenna site on the rooftop of the company's office in Leuven (see Figure 9) for two types of antennas. The choke-ring antenna provides high degree of multipath suppression due to elevation-dependent gain pattern. This type of antenna is designed for static geodetic applications. The Sensor Systems antenna, on the other hand, is an omni-directional antenna, which is designed for aviation applications and does not suppress multipath.

	APME	C/A code, m	P1 code, m	P2 code, m
AOA choke ring	YES	0.35	0.40	0.45
AOP choke ring	NO	0.45	0.40	0.45
Sensor Systems	YES	0.50	0.50	0.55
Sensor Systems	NO	0.75	0.50	0.55

Table 3. STD of the multipath combination for all the three code ranges.

APME (A-Priori Multipath Estimation) is Septentrio's multipath mitigation technology based on the analysis of the shape of the auto-correlation peak [2]. APME affects only C/A code. Table 3 shows that the reduction of multipath by APME is more significant for the Sensor Systems antenna, which is natural because there is a lot more multipath for this antenna. In this case multipath is reduced by about 30%. For the choke-ring antenna the reduction of multipath is only 20% because the antenna itself already suppresses much of the multipath. The remaining multipath noise (35 cm) comes close to the tracking noise, which means that most of the multipath is suppressed. P1 and P2 codes are not affected by APME, but their multipath is intrinsically low due to the shorter chip length.

APME is a bias-free multipath mitigation method. This means that it does not introduce any additional delays to C/A code pseudoranges on top of hardware delays in satellites and receivers. This is not a trivial statement because many multipath mitigation methods do introduce PRN-dependent biases, which may become a distortion for the network processing, when receivers of different kinds (and with different biases of C/A code) are combined. A common reason for these biases is that the shapes of autocorrelation peaks for some PRNs slightly deviate from the nominal triangle. Therefore the normal shape of the peak is taken for multipath, which causes the bias. In APME all the known effects of this nature are compensated, and the output C/A code measurements are bias-free. Please refer to our future publication [3] for more details.

Performance of PolaRx2 in a geodetic network

In order to directly test the new receiver in the context of an actual geodetic application, a joint experiment was organized together with the Royal Observatory of Belgium. We set up an improvised reference site on the rooftop of Septentrio's office building using a standard AOA choke-ring antenna (Figure 9).

Figure 9. Septentrio reference site nicknames PLRX.

This site, nicknamed PLRX, was operating during 2 months in the same way as if it were a real reference site. The data was collected continuously at a data rate of 30 sec. Daily RINEX files were automatically FTPed to the data center at the ROB. The ROB operates one of the EPN analysis centers and contributes to the weekly EUREF combined solution by analyzing daily an EPN network of about 30 stations. The data collected by the PLRX site were integrated, without any special tuning, into this routine EPN data analysis.

The results of this experiment demonstrated about the same level of noise of the network positional solution, which is typical for other EPN sites of good standing. The noise of the positional components is presented in Figure 9. The shift in the East component of the position corresponds to an actual shift of the antenna.

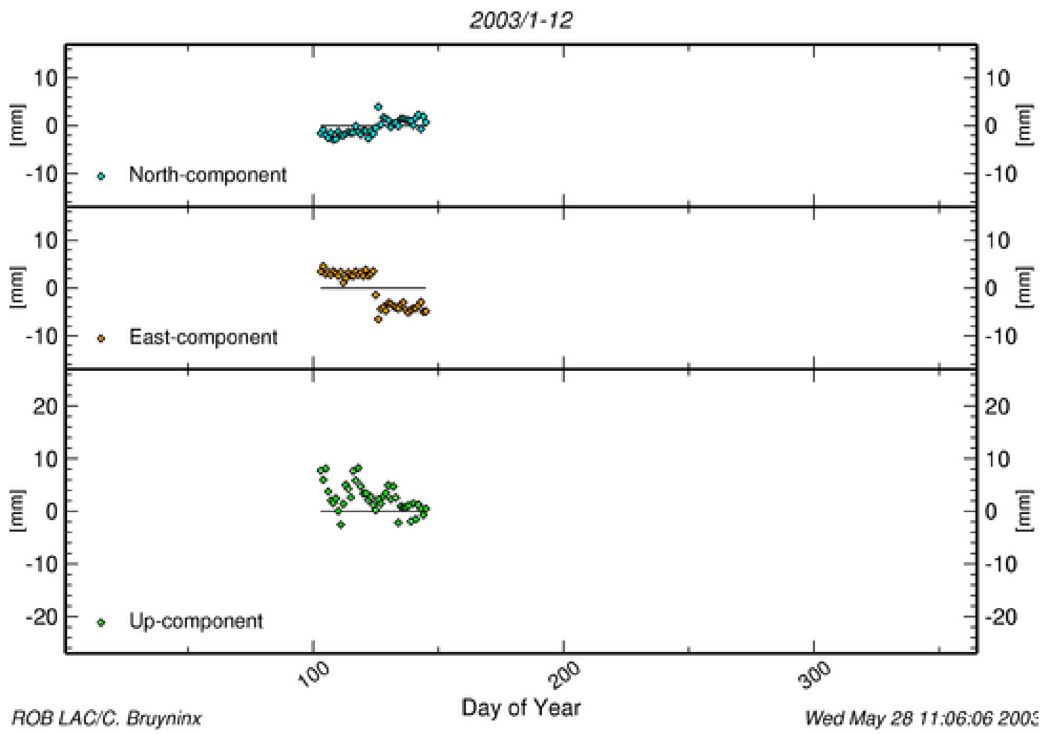

Figure 10. Position time series computed for the PLRX site at the ROB.

The results obtained by the processing of PLRX data have confirmed that performance of PolaRx2 is comparable to the performance of well-reputed geodetic receivers. Most of the positional noise can be attributed to phase multipath from the rooftop structures and adjacent buildings (see Figure 9).

Further testing of PolaRx2 at the ROB was focused on time transfer applications [4].

Conclusions.

PolaRx2 is a new low-noise geodetic dual-frequency receiver designed at Septentrio around the proprietary chipset and the principle of flexible multi-process software architecture. Due to its TCP/IP feature, PolaRx2 can be used in reference network applications of a non-traditional design, where all the reference stations are controlled directly from the central server without the use of controlling PCs at the reference stations. The phase tracking implemented in PolaRx2 includes optional measurement smoothing and reduces the noise level and cycle slip rate to a minimum. Performance of PolaRx2 as a reference station in a test EPN application was comparable to the performance of other well-reputed receivers used by the EUREF.

References

1. Simsky A. *Light-weight position/attitude determination system based on a single-board multi-antenna GNSS receiver*. Presented at the UVStech meeting, December 3-5, 2003, Brussels.
2. Sleewaegen, J.- M. and Boon, F. *Mitigating short-delay multipath: a promising new technique*, Proceedings of ION GPS-2001, Sept. 2001.
3. Simsky, A. and Sleewaegen, J.-M. *C/A code bias in high-end receivers*. Submitted for ENC-GNSS 2004. Rotterdam, 16-19 May 2004.
4. Defraigne, P., Bruyninx, C., Roosbeek F. *Initial testing of a new GPS receiver, the PolaRx2, for time and frequency transfer using dual frequency codes and carrier phases*. Presented at the 35th Annual PTTI Meeting December 2-4 2003, San Diego, CA, USA.