# PERFORMANCE ASSESSMENT OF A JUZZLE-BASED GNSS SIMULATOR

Christophe Ouzeau, Joël Korsakissok

*Silicom, Toulouse, France*

*Email: couzeau@silicom.fr*

## BIOGRAPHY

Christophe OUZEAU graduated in 2005 with a master in astronomy at the Observatory of Paris. He started the same year his Ph.D. thesis on degraded modes resulting from the multi constellation use of GNSS, supported by DTI and supervised by ENAC. He is now a radio navigation engineer at Silicom, SGSO, Toulouse, France.

Joël KORSAKISSOK was graduated from SUPELEC engineer school in 1986, worked in Germany (Darmstadt) in the space domain until 1991 and for SILICOM since that year. He took the lead of the local SILICOM team in 1994. He manages nowadays 45 doctors and engineers, working for more than the half for space domain (telecommunication and GNSS).

## 1. INTRODUCTION

With the multiplication of navigation systems such as GPS, GALILEO, GLONASS and QZSS, and consequently, with an increasing number of various signals to be received and processed, it is felt by the GNSS industries and institutions as an important task to investigate new signals reception technologies.

Silicom, a company specialized in Telecom and Technological Engineering since 1983, is working in GNSS technology for many years now, especially on signal and physical layer.

Silicom is the major contributor of the Juzzle freeware environment since 2001, and offers service upon it (evolution, development of simulations, training, etc...)

Beside, Silicom, today main partner of the Juzzle project is the CNES, which help Silicom to invest in new functionalities year after year.

Juzzle (Java + Puzzle) is a simulation framework particularly aimed for interoperability, most used in the physical layer domain.

With the increasing demands coming from the GNSS world, the CNES has decided to develop a generic GNSS signal software simulator under Juzzle, while SILICOM completed it with all needed functionalities in order to obtain a “global GNSS constellation simulator” (i.e. orbitography, navigation messages, PVT, and so on). The result software is commercialized by SILICOM under CNES licence (© CNES Copyright 2007).

One of the objectives of Silicom is to provide a (quasi) real-time software signal generator at base band level and a real-time receiver. To this end, a parallelized structure of a signal generator is proposed, and a parallelized software receiver is targeted.

This publication is aimed at presenting the optimization strategy and effort done to achieve these goals.

This paper starts with an architecture description of the processing tools, including the orbitography, navigation message generation, and PVT calculation, at generation level as well as at reception level – however, physical layer is not addressed in this paper -.

Then, signals generation and reception performances are discussed and evaluation tests are performed to know if the compliance with actual needs is reached.

## 2. OBJECTIVES

The main objective of this paper is to present a software emitter and receiver simulator and to study the performances of such an algorithm in terms of calculation times and compliance with existing and new GNSS signals.

The main guideline is that the efforts must be done to converge towards a real time software solution. To this end, the simulator architecture is discussed and solutions are proposed to reduce, if possible, the simulations duration. This optimization can be done at the transmitter (satellite) level as well as at the receiver level.

The proposed simulator is based upon the Juzzle software tool, which is presented hereafter.

In the following, Tx refers to a satellite in view and Rx refers to the receiver.

## 3. JUZZLE PLATFORM

Juzzle provides a full integrated simulation environment for development and exploitation of simulators including a real-time simulation debug solution using graphical charts. It is also already interfaced to many systems and languages (for algorithm implementation) like Java, C, Matlab, FORTRAN, Ada, SystemC, Ptolemy, every module being assembled graphically in the environment. This means that, for instance, a simulation can mix modules written in Java, C and FORTRAN and use the graphical post-treatment of Matlab, all this being developed and executed in Juzzle. Juzzle also provides many tools for parametric or statistic case studies automation and grid computing. Juzzle is a freeware available at: http:\\www.juzzle.org.

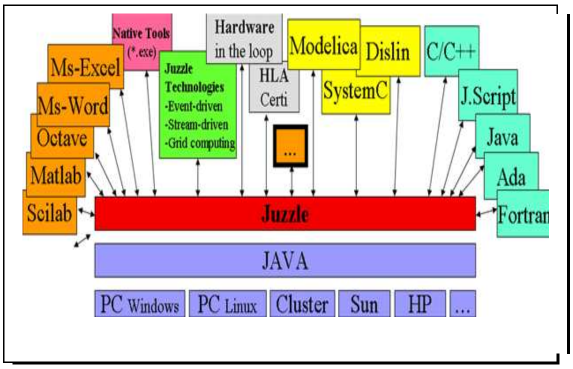
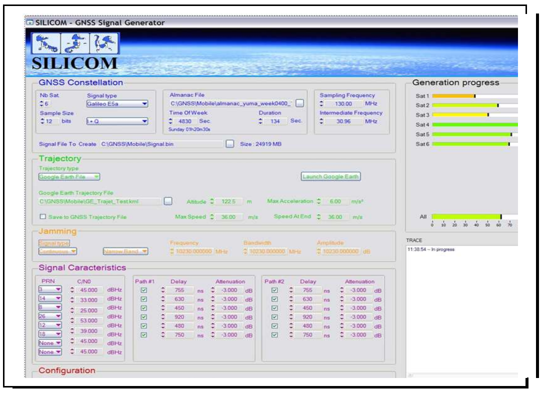


**Figure 1 : Juzzle simulation platform architecture**

## 4. JGNSS PRESENTATION

"Juzzle-GNSS" is a software simulator that allows to generate GNSS signals on the one hand (Tx generator) and to process it to provide PVT on the other hand (Rx simulator).



Nb Sat: 6
Signal type: Galileo E5a
Sample Size: 12 bits
I+Q
Almanac File: C:\GNSS\Mobile\almanac_yuma_week0400_
Time Of Week: 4830 Sec.
Sunday 01h20m30s
Duration: 134 Sec.
Sampling Frequency: 130.00 MHz
Intermediate Frequency: 30.96 MHz
Signal File To Create: C:\GNSS\Mobile\Signal.bin
Size : 24919 MB
Trajectory type: Google Earth File
Google Earth Trajectory File: C:\GNSS\Mobile\GE_Trajet_Test.kml
Altitude: 122.5 m
Max Acceleration: 6.00 m/s²
☐ Save to GNSS Trajectory File
Max Speed: 36.00 m/s
Speed At End: 36.00 m/s
Signal type: Continuous
Narrow Band
Frequency: 10230.000000 MHz
Bandwidth: 10230.000000 MHz
Amplitude: 10230.000000 dB
TRACE
11:38:54 -- In progress

| PRN | C/N0 | Path #1 | Delay | Attenuation | Path #2 | Delay | Attenuation |
|---|---|---|---|---|---|---|---|
| 3 | 45.000 dBHz | ☑ | 755 ns | -3.000 dB | ☑ | 755 ns | -3.000 dB |
| 14 | 33.000 dBHz | ☑ | 630 ns | -3.000 dB | ☑ | 630 ns | -3.000 dB |
| 8 | 25.000 dBHz | ☑ | 450 ns | -3.000 dB | ☑ | 450 ns | -3.000 dB |
| 26 | 53.000 dBHz | ☑ | 920 ns | -3.000 dB | ☑ | 920 ns | -3.000 dB |
| 12 | 39.000 dBHz | ☑ | 480 ns | -3.000 dB | ☑ | 480 ns | -3.000 dB |
| 18 | 45.000 dBHz | ☑ | 750 ns | -3.000 dB | ☑ | 750 ns | -3.000 dB |
| None | 45.000 dBHz | | | | | | |
| None | 45.000 dBHz | | | | | | |

**Figure 2: User Tx interface created by Silicom**

The software simulator is capable of generating GNSS signals coming from a satellite constellation and to process it via several signal processing channels.

### 4.1. Signal generation

The signal generation relies on:

- A complete orbitography simulation
- A complete signal generation, including the navigation messages, spreading codes and modulations
- A calculation of the resulting GNSS signal at any Earth point, taking into account multiple satellites, propagation effects, Doppler…

The signal generation toolbox currently provides a highly configurable model of satellite, covering various signal configurations for Galileo and GPS IIF block.

Our comparison study relies on the GPS L1 C/A, L5 and Galileo E5a signals of the Tx simulator. Those signals main characteristics are recalled hereafter [DR1] and the different signals generator modules are depicted.

### 4.2. The L1 C/A signal

The L1 C/A signal is generated according to the following scheme. The signal is composed of a navigation message, a spreading code sequence, and is BPSK-modulated. The following figure provides a description of the signal generation.

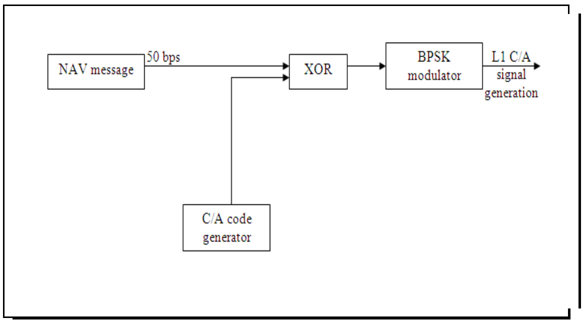


Figure 3: L1 C/A signal generation block diagram

The L1 C/A transmitted signal structure is given in the following formula:

- $s(t)$ (dimensionless): the L1 C/A transmitted signal

$$s(t) = A \cdot d(t) \cdot CA(t) \cdot \cos(2\pi f_{L1CA} t + \theta_0)$$

- $A$ (dimensionless): L1 C/A signal amplitude
- $d(t)$ (dimensionless): Data symbols
- $CA(t)$ (dimensionless): Civil Coarse/Acquisition PRN code chips
- $f_{L1CA}$ (Hz): L1 C/A carrier frequency, 1575.42 MHz
- $\theta_0$ (rad): initial carrier phase

### 4.3. The L5 signal

The transmitted signal is QPSK-modulated and is composed of a data and a pilot (data-free) channels that allows a carrier phase tracking at lower carrier to noise ratios than in the GPS L1 C/A case. This signal is based on a Code Division Multiple Access (CDMA) principle in order to distinguish the transmitting satellites. Each I and Q component is modulated with a specific bit train.

The in phase train is generated by the modulo-2 addition of a pseudorandom noise (PRN) ranging code, a synchronization sequence, called the Neuman-Hoffman (NH) code, and the navigation data message.

The quadraphase (or pilot) train is the modulo-2 sum of a different PRN code and a different NH synchronization sequence.

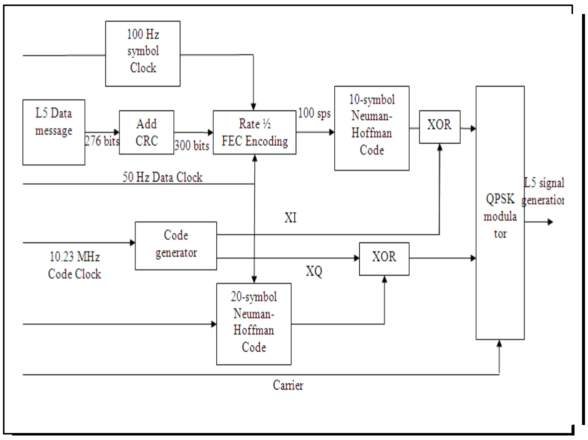


Figure 4: L5 signal generation block diagram

The L5 transmitted signal structure is given by the following formula:

- $s_{L5}$ (dimensionless): transmitted L5 signal

$$s_{L5}(t) = \sqrt{P}\begin{bmatrix} d_{L5}(t) \cdot XI(t) \cdot NH_{10}(t) \cdot \cos(2\pi f_{L5}(t) - \theta) \\ + XQ(t) \cdot NH_{20}(t) \cdot \sin(2\pi f_{L5}(t) - \theta) \end{bmatrix}$$

- P (W): Transmitted signal power
- $d_{L5}$ (dimensionless): CNAV navigation data stream
- $XI$ (dimensionless): P/NRZ/L materialization of the Inphase PRN code
- $NH_{10}$ (dimensionless): P/NRZ/L materialization of Neuman-Hoffman code used on the Inphase component
- $XQ$ (dimensionless): P/NRZ/L materialization of Quadraphase PRN code
- $NH_{20}$ (dimensionless): P/NRZ/L materialization of Neuman-Hoffman code used on the Quadraphase component

- $f_{L5}$ (Hz): L5 signal carrier frequency, 1176.45 MHz
- $\theta$ (rad): initial carrier phase

### 4.4. The E5a signal

The E5a signal is generated according to the following scheme:

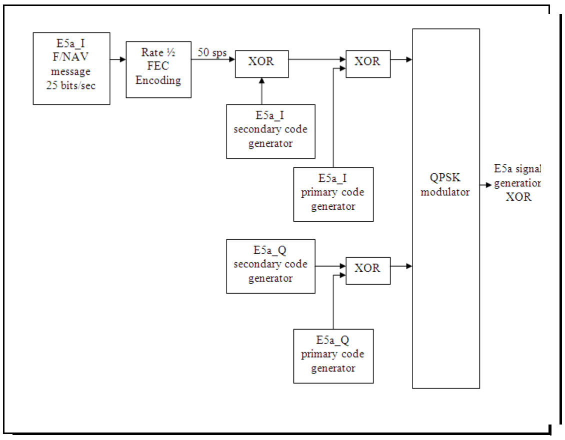


**Figure 5: E5a signal generation block diagram**

The E5a transmitted signal structure is that of a QPSK signal, as described in the equation below:

- $s_{E5a}$ (dimensionless): transmitted E5a signal

$$s_{E5a}(t)=\sqrt{P}\left[\begin{array}{l} d_{E5a\_I}(t)\cdot c_{E5a\_I}(t)\cdot c_{2E5a\_I}(t)\cdot\cos(2\pi f_{E5a}t+\theta_0) \\ -c_{E5a\_Q}(t)\cdot c_{2E5a\_Q}(t)\cdot\sin(2\pi f_{E5a}t+\theta_0) \end{array}\right]$$

- P (W): Transmitted signal power
- $d_{E5a\text{-}I}$ (dimensionless): F/NAV navigation data stream
- $c_{E5a\text{-}I}$ (dimensionless): Data channel primary ranging code
- $c_{E5a\text{-}Q}$ (dimensionless): Pilot channel primary ranging code
- $c_{2E5a\text{-}I}$ (dimensionless): Data channel secondary code
- $c_{2E5a\text{-}Q}$ (dimensionless): Pilot channel secondary code
- $f_{E5a}$ (Hz): E5a signal carrier frequency, 1176.45 MHz
- $\theta_0$ (rad): initial carrier phase

### 4.5. Signal propagation

Attenuation, propagation delay and Doppler shift are applied to the signals generated. Other phenomena include thermal noise and multipath, applying specular model or statistical model based on Rice and Rayleigh distributions.

### 4.6. RF stage

The RF conditioning simulator includes:

- Down-conversion
- Filtering
- Sampling and quantization

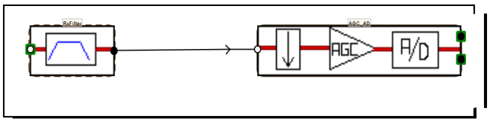


**Figure 6 : RF stage**

### 4.7. Digital signal processing

The digital signal processing relies on two main functionalities which are:

- The signal acquisition:

The acquisition algorithm is based on the well-known FFT technique. This step is represented hereafter:

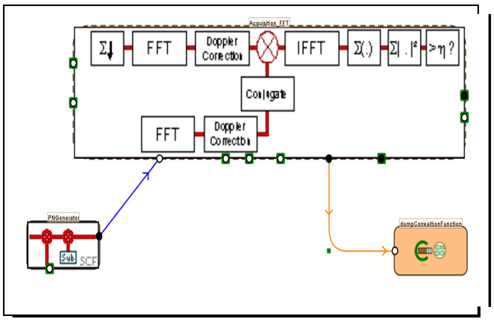


**Figure 7 : Acquisition architecture**

The acquisition proposed can be :

- Warm: the satellites to be tracked are automatically selected from the almanac Yuma file and the initial Rx time;
- Semi-automatic: the initial Rx time is unknown, the values of the tracked PRN have to be set at the Rx global controller level, they are seeked with the specified range Doppler values. The almanac file remains necessary to perform the PVT computing.

- The signal tracking relies on the following functions:

- The correlators
- The integrators

- The discriminators
- The loop filters

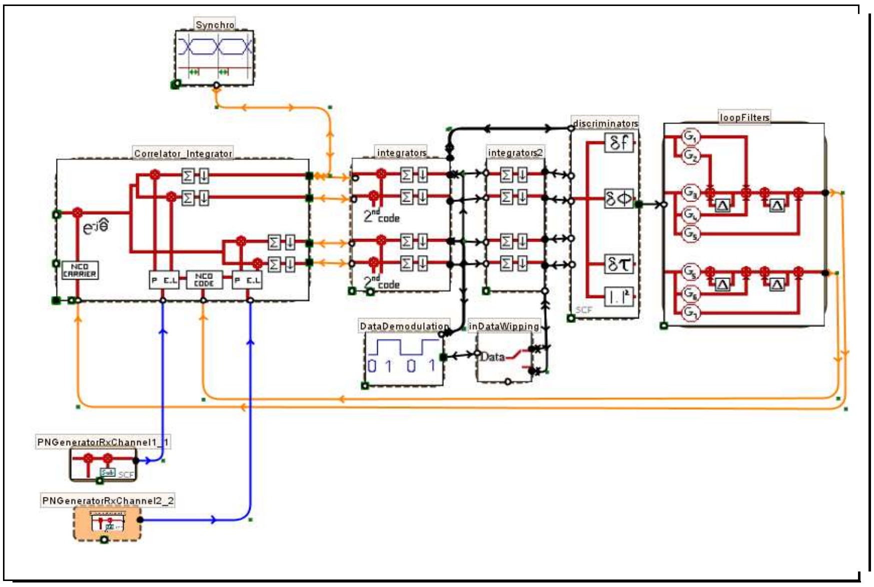


**Figure 8 : Tracking architecture**

- ➢ The bit, word and frame synchronization
- ➢ The navigation message extraction

For each satellite in view, the receiver uses one channel to process the incoming signal. The simulator can the provide for each channel, the autocorrelation functions, the tracking loops outputs, the carrier to noise estimations (used for tracking lock detection).

The parameters used for the simulations results presented hereafter are summarized in the following table:

| | |
|---|---|
| Simulator | JGNSS |
| Frequency band | GPS L1 C/A |
| Simulation type | Tx + Rx |
| FI | 20.45 MHz |
| Signal type | I and Q<br>Only a data channel |
| Generated signal duration | 10 sec |
| Nb of satellites | 6 |
| Selected PRN | 11/16/19/28/3/8 |
| Carrier to noise ratio for all satellites | 50 dB Hz |
| Integration time | 1 ms |
| Acquisition time | Warm |
| Doppler search range | [-5 kHz ; 5 kHz] |
| Number of coherent accumulations | 1 |
| Number of non-coherent accumulations | 5 |
| Initial receiver location | Latitude: 36°22’56’’<br>Longitude:127°21’50’’<br>Altitude: 100 m |

**Table 1: Simulator settings for the following PVT**

Hereafter are plotted some Rx simulator outputs, the x-axis unit is the number of samples; the y-axis unit depends upon the considered output and is given for each plot. Those outputs are provided for the PRN 11 as an example.

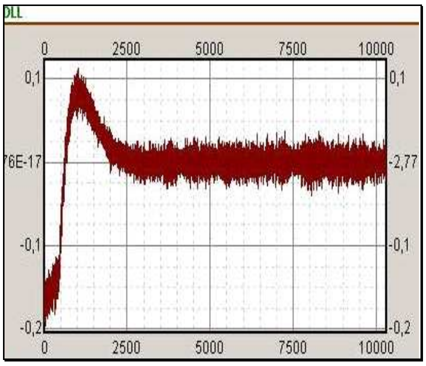


**Figure 9 : DLL output (chips)**

The DLL used is set according to the following characteristics :

- ➢ Dot Product discriminator

- 0.125 chip early-late spacing
- First order filter
- 1 Hz bandwidth

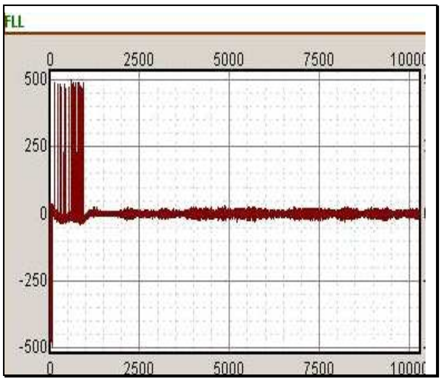


Figure 10 : FLL output (Hz)

The FLL used is set according to the following characteristics :

- Atan2 discriminator
- First order filter
- 5 Hz bandwidth

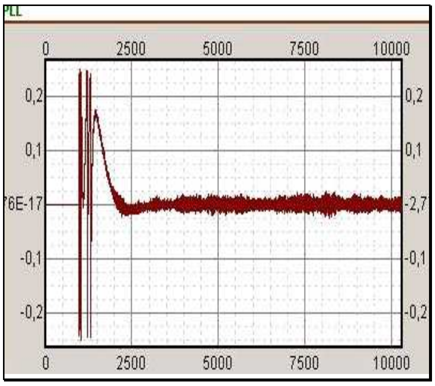


Figure 11 : PLL output (rad)

The PLL used is set according to the following characteristics :

- Atan discriminator
- Second order filter
- 10 Hz bandwidth

Thanks to the tracking step, the primary (and secondary when appropriate) codes are synchronized. The navigation message (if present) is then extracted.

The outputs for each channel (satellite) are then used to provide the pseudoranges needed to estimate the PVT.

### 4.8. PVT calculation

A Position, Velocity and Time algorithm has also been implemented. The results are the receiver longitude, latitude and altitude coordinates. Since the simulator allows choosing a static or dynamic receiver, the estimated position can be a point or a trajectory. An example of receiver coordinates estimation is provided hereafter, considering 6 satellites in view. The results are obtained by generating and processing the GPS L1 C/A signal. The first column presents the receiver location and the position estimation maximum variations. The simulator used is the JGNSS presented above with the settings presented in the table 1.

| | Position generated | Position Error in meters |
|---|---|---|
| **Latitude** | 36°22’56’’ | 5 meters variation |
| **Longitude** | 127°21’50’’ | 5 meters variation |
| **Altitude** | 100m | 9 meters variation |

**Table 2: Accuracy of the estimated receiver location by using the L1 C/A signal on 6 satellites in view**

## 5. OPTIMIZED Tx SIMULATOR ARCHITECTURE

An optimized Juzzle GNSS simulator version has been developed. In this version, the simulator is optimized for signal generation.

### 5.1. Buffer size optimization

In the simulator, a main function is automatically generated thanks to the Juzzle tool.

For each module implemented in the model (Data generator, navigation message, orbitography, signal modulation…), this main function realizes:

- A single call to a prepare function for the modules initialization
- Calls to a process function for each module
- A single call to a terminate function to close each module, when all processes are done

The main function also realizes a buffer management between the modules inputs and outputs.

Therefore, two main optimizations have been realized:

- A buffer management size optimization
- A buffer size adjustment to optimize the data flow

### 5.2. Calculations parallelization

- The calculations are distributed over several processors to optimize the signal generation, one PC per satellite is thus used for an optimized parallelized Tx structure as it is described in the next figures.

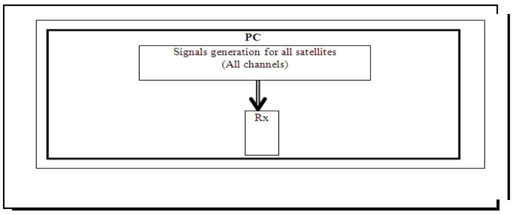


Figure 12 : Signals generation initial strategy

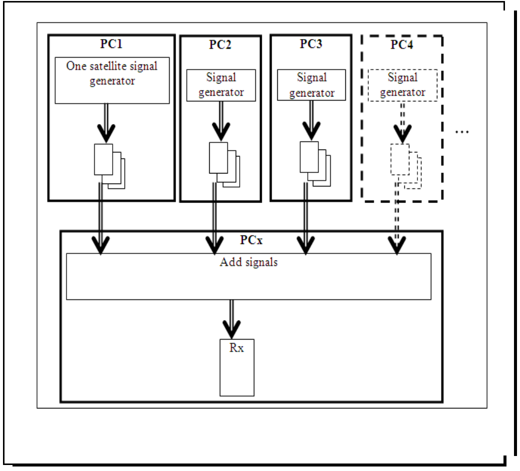


Figure 13 : New paralellized signal generation strategy

### 5.3. Satellites channels

In the optimized version, the number of satellites channels is configurable, they are automatically enabled or disabled and the software can simulate up to 12 channels whereas the first version only provided 6 channels.

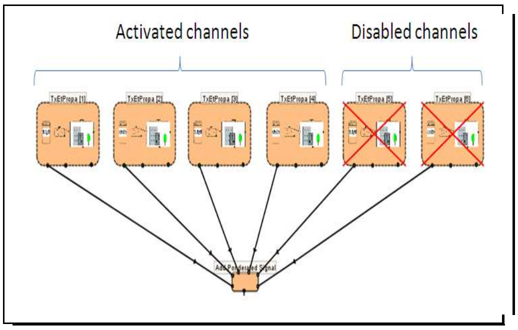


Figure 14 : JGNSS first version satellites (Tx) channels

## 6. PERFORMANCES ASSESSMENT

To analyse the receiver performances, several tests were performed to study the influence of the signals characteristics, the settings and the parallelized architecture on the simulation duration.

### 6.1. Signals characteristics impact on the simulation duration

To compare the times of simulation, the simulator used is the non-optimized JGNSS. The I and Q signal components were generated. For the L1 C/A signal, as it is mentioned in section 4 of this paper, there is only a data channel whereas, for the GPS L5 and Galileo E5a, both the data and pilot channels presented in section 4 are generated, which means the generation is more complex than for the L1 C/A signal case. As a consequence, a longer simulation time is required. Indeed, for instance, the L5 signal generation is 1.6 times longer than the L1 C/A signal one. The E5a generation duration is comparable to the L5 one (E5a generation is 1.04 times longer).

### 6.2. Performances for both non-optimized and optimized simulator versions

Simulations were launched by using JGNSS on a PC which characteristics are: Intel Xeon E5405, CPU 4 Core 2 GHz, RAM: 8 Go. The simulation duration is compared between the first simulator version and the optimized simulator version. Tests were conducted for various numbers of satellites, sampling frequencies, signal duration.

The first set of tests is made by simulating only one satellite in view. Indeed, the objective here is to notice the improvements brought by the buffer sizes management, without parallelization.

| Simulator | JGNSS | |
|---|---|---|
| **Frequency band** | GPS L1 C/A | |
| **Simulation type** | Tx | |
| **Sampling frequency (MHz)** | 102.3 | 10 |
| **Nb of satellites** | 1 | 1 |
| **Generated signal duration** | 1 sec | 1 sec |
| **Simulation duration** | 15’18’’ | 5’04’’ |

**Table 3: Simulation duration performance test for JGNSS**

| Simulator | Optimized JGNSS | |
|---|---|---|
| **Frequency band** | GPS L1 C/A | |
| **Simulation type** | Tx | |
| **Sampling frequency (MHz)** | 102.3 | 10 |
| **Nb of satellites** | 1 | 1 |
| **Generated signal duration** | 1 sec | 1 sec |
| **Simulation duration** | 1’42’’ | 46’’ |
| **Gain factor w.r.t the non-optimized version** | **9** | **6.6** |

**Table 4: Simulation duration performance test for the optimized JGNSS version**

The previous tables show the performance tests performed by using the two versions of the simulator. The optimized version clearly improves the simulation time for one satellite channel.

Further investigations have been performed by comparing the performances of JGNSS and its optimized version for three satellites in view with four processors (three for all the satellites channels and one for the annex tasks). Obviously, by setting the number of satellites to three it is not possible to calculate the PVT. The priority in this paragraph is to present the gain brought by the Tx simulator parallelization. Since, during the tests we only disposed of a quad core computer, the tests were performed with three satellites, which is sufficient to estimate the simulation time gain. Nevertheless, tests over more processors are targeted (8 processors). Then, a PVT will be calculated with the optimized architecture and with low simulation times.

| Simulator | JGNSS |
|---|---|
| **Frequency band** | GPS L1 C/A |
| **Simulation type** | Tx |
| **Sampling frequency (MHz)** | 10 |
| **Nb of satellites** | 3 |
| **Generated signal duration** | 1 sec |
| **Simulation duration** | 9’34’’ |

**Table 3: Simulation duration performance test for JGNSS**

| Simulator | Optimized JGNSS |
|---|---|
| **Frequency band** | GPS L1 C/A |
| **Simulation type** | Tx |
| **Sampling frequency (MHz)** | 10 |
| **Nb of satellites** | 3 |
| **Generated signal duration** | 1 sec |
| **Simulation duration** | 1’53’’ |
| **Gain factor w.r.t the non-optimized version** | **5** |

**Table 4: Simulation duration performance test for the optimized JGNSS version**

## 7. CONCLUSION

The simulator architecture allows taking into account all the GNSS signals at the Tx level (modulations, FEC, navigation messages…) as well as at the Rx level (number of correlators, discriminators, integration times, loop filters…). For instance, the simulator can generate GPS L1 C/A signal as well as the Galileo E5 signal and the Rx is set accordingly.

The signal generator is built for up to 12 satellites simulated. The Rx simulator can process up to 12 channels. The software architecture has been optimized so as to provide an improved calculation cost. Indeed, the parallelized Tx structure allows decreasing significantly the simulations durations.

The particular cases of GPS L1 C/A, GPS L5 and Galileo E5a signals are assessed in this paper since their characteristics are of interest for performances comparisons (Presence of pilot channels or not, secondary code, FEC, …)

As it is mentioned in this paper, the performances presented here depend upon the simulator settings. Amongst others, the sampling frequency selection plays a role in the calculation duration, and depends upon the generated signal (see table 3, 4 and section 6.1). The simulation time decreases with the signal complexity. Indeed, the L1 C/A generator is simpler than the L5 generator for instance, since this last one is composed of data and pilot channels for instance as it can be noticed on figure 4.

The signal duration and the number of satellites processed are of importance to compare the performances of the first simulator and the optimized simulator architectures. The number of processors used is four, so the improvement brought by the parallelized simulator architecture can be seen noticed for three satellites in view (three processors for each satellite channel and a processor for the remaining calculations and the annex tasks).

As a consequence, the results presented in this paper are comparative and present the improvements brought by the optimizations. The improvements brought by the simulator optimization are of interest since they reduce significantly the simulation durations.

## 8. FUTURE WORKS

Since the objective is to provide a quasi-real-time simulator, a parallelized Rx structure is also targeted.

Many new features are targeted for the simulator to comply with the actual needs in the GNSS world. Amongst others, the propagation model will be completed (ionosphere crossing code delay, multipath, interferences generator). A pseudorange control module will therefore complete the algorithms (cycle slip detectors etc…). Then the PVT module will be completed with a complete Kalman-based algorithm. Future works will also include a parallelized dual frequency Tx and Rx to estimate the generated ionospheric code delay.

A SBAS augmentation and an integrity monitoring modules are also in the scope of the developments.

## ACKNOWLEDGEMENTS

This work has been performed thanks to the whole Silicom GNSS team. JGNSS is under CNES Copyright 2007 ©.

## BIBLIOGRAPHY

[DR1] Galileo OS SIS ICD, Signal In Space Interface Control Document, Draft 1, European Space Agency, February 2008.

[DR2] G.Artaud, G.Menard, L.Ries, J. Dantepal, JL. Issler « Juzzle Software receiver », Navitec 2006, 11-14 december 2006, Noordwijk, the Netherlands.