International Global Navigation Satellite Systems Association
IGNSS Conference 2016

Colombo Theatres, Kensington Campus, UNSW Australia
6 – 8 December 2016

An ultra-low-cost antenna array frontend for GNSS application

Thuan D Nguyen
Hanoi University of Science and Technology, Vietnam
thuan.nguyendinh@hust.edu.vn
Vinh T Tuan
Australian Centre for Space Engineering Research, School of Electrical Engineering and Telecommunication, UNSW
vinh.tran@student.unsw.edu.au
Tung H Ta
Hanoi University of Science and Technology, Vietnam
tung.tahai@hust.edu.vn
Letizia Lo Presti
Department of Electronics and Telecommunications, Politecnico di Torino, Italy
letizia.lopresti@polito.it

ABSTRACT

Antenna array frontends play a key role in grabbing GNSS digitalized signal and providing the input data to the signal processing stage. However, the existing antenna array frontends for GNSS application are often bulky and costly. In this paper, a low-cost and compact antenna array frontend is proposed. Besides, a possible solution to estimate the clock drift raised by the internal architecture of the hardware components is also introduced. The performance evaluation of the proposed frontend shows that the frontend is effectively applicable for GNSS applications.
KEYWORDS: antenna array frontend, RTL2832.

1. INTRODUCTION

In recent years, antenna array processing emerges as an effective technique for mitigating interference and spoofing. However, antenna array frontends are costly, especially for GNSS application. This is a barrier to widening studies on antenna array processing in GNSS field.
Several antenna array frontends for antenna array were proposed such as De Lorenzo *et al.*, (2007). However, the quantization bits of these frontends are limited to the maximum of 2 bits. Consequently, the frontends are less robust on mitigating strong interferences.
This paper proposes an ultra-low-cost antenna array frontend for GNSS application. In fact, the proposed solution is based on the combination of the RTL2832 dongles obtained from Nooelec. The operating frequency range of such dongles varies from 25 MHz to 1750 MHz covering the whole band of GNSS signals. Moreover, the quantization bits of the ADC embedded in the frontend can expand to 16 bits. Therefore, the proposed frontend is suitable for GNSS applications.

A software is also developed for this frontend. In addition to collecting signals, this software synchronizes received signals among dongles and estimates frequency difference between elements. Since each element of this frontend is a complete dongle with their own interface to the host computer, the signals from the elements are not received at the same time. Moreover, regardless of the use of a common clock for all elements, the tuned frequency of Local Oscillator (LO) is different in each element. Therefore, these issues must be addressed prior to the use of this frontend. A full explanation of the algorithm used in our software will be given in the next sections.
This design brings several benefits such as a large number of quantization bits, compactness, and scalability. Adding or removing an element can be performed with a minimum change of the system.

2. OVERVIEW OF ANTENNA ARRAY PROCESSING

The received signal at the antenna of a GNSS receiver, without a loss of generality, can be considered as the combination of line of sight (LOS) signals, non-line of sight (NLOS) signals, noise, and interferences. However, for simplicity, only LOS signals and noise are considered. Hence, this combination can be written in mathematical model as follows:

$$R_{L1}^{m}(t) = \sum_{k=1}^{N} S_{L1,k}^{m}(t) + \eta^{m}(t) \qquad (1)$$

where:
$R_{L1}^{m}(t)$ is the composite signal at the m^{th} element.
N is the number of LOS signals.
$S_{L1,k}^{m}(t)$ is the L1 LOS GPS signal of the k^{th} satellite at the m^{th} element.
$\eta^{m}(t)$ is the ambient noise at the m^{th} element.

To model antenna array signals, assume that a far-field signal impinges an antenna in the direction expressed by the azimuth and elevation angles (ϕ, θ), the unit vector of the incoming signal can be expressed as:

$$\boldsymbol{a}_s = [\sin(\theta)\cos(\phi) \quad \sin(\theta)\sin(\phi) \quad \cos(\theta)]$$

Assuming that the first element of the array is at the origin of the coordinate system (Figure 1). Hence, the delay between the m^{th} element and the first element are expressed as the propagation delay in the direction of the incoming signal from the origin to the wavefront passing through the m^{th} element (Figure 1) (Arnold *et al.*). The delay in meters is:

$$\Delta\rho^{m} = \boldsymbol{p}^{m} \cdot \boldsymbol{a}_s(\theta, \phi) = X^{m}\sin(\theta)\cos(\phi) + Y^{m}\sin(\theta)\sin(\phi) + Z^{m}\cos(\theta) \qquad (2)$$

where $\boldsymbol{p}^{m} = (X^{m}, Y^{m}, Z^{m})$ is the position of the m^{th} element.

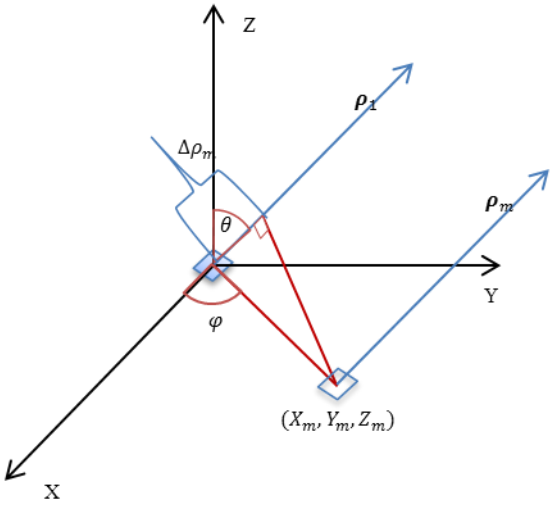


Figure 1 Geometry of antenna array

In general, the signal at the m^{th} element $\left(s^m(t)\right)$ is treated as a delayed version of the signal at the first element $\left(s^1(t)\right)$

$$s^m(t) = s^1(t - \tau^m) = s^1\left(t - \frac{\Delta\rho^m}{c}\right) \quad (3)$$

where c is the speed of light.
The phase difference between the two elements is thus used to verify the antenna array frontend. Particularly, if all dongles share a common antenna, the phase difference between every dongle with respect to the reference dongle must be the same value for all satellites.

3. PRACTICAL CONSIDERATION IN DESIGN A LOW-COST ANTENNA ARRAY

According to Arnold *et al.*, (2015) and Borio *et al.*, (2015), the combination of RTL2832U chipset and R802T2 turner was proved to satisfy the requirements of a GPS frontend. Thereby, those dongles are utilized to make a low-cost antenna array for GNSS application. The key in designing an antenna array frontend is the use of a common clock for Analog to Digital Converter (ADC) components. Therefore, to adapt the turner to the antenna array application, an external TCXO with 0.5ppm precision is then connected to all dongles (Figure 2).

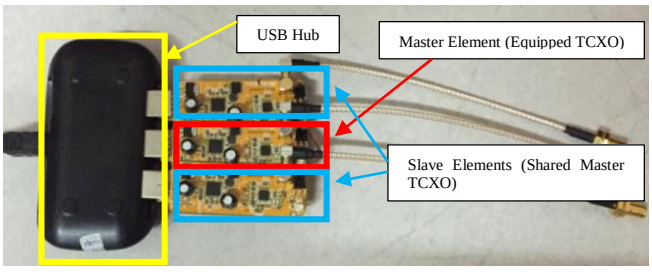


Figure 2. The 3-elements antenna array frontend modified from turner RTL2832Us

3.1. Synchronizing the data

To perform data synchronization, it is worthy to note that the time difference (in term of samples) among dongles is a constant, thanks to the use of a common clock.. In our solution,

the number of samples calculating from the start of the collecting process to each sub-frame are logged. Hence, the samples difference among dongles for the same satellite and the same sub-frame is exactly the desired value.
To implement this scenario, all dongles use the same antenna as Figure 2. The collected signals are processed independently. Relying on the absolute position of the navigation bits, the difference in samples between signals are the desired number.

3.2. The clock phase shift problem.

Before mitigating clock frequency shift, the phase difference between frontends must be evaluated. An architecture based on the receiver architecture proposed by (De Lorenzo *et al.*, 2007) is introduced and illustrated in Figure 3.

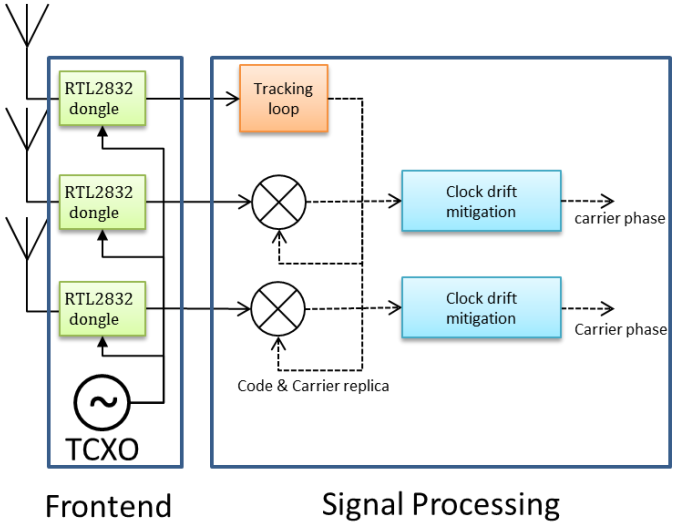


Figure 3. The architecture of the system to determine the phase offset

In this system, only the received signal at the first channel is acquired and tracked. The carrier and code yielded by this channel is then multiplied to the others. For simplicity, only a L1 signal of a single satellite is considered. The GNSS signal collected at the m^{th} element is as follows:

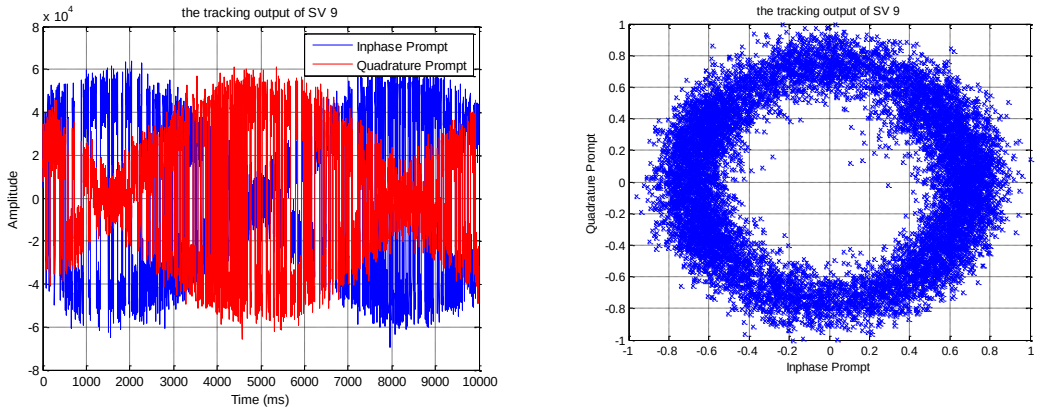
$$s_m(t) = \sqrt{P_m} C(t-\tau_m) D(t-\tau_m) \exp\left(j(2\pi f_m t + \Phi_m)\right) \tag{4}$$

where
P_m is the power of the received signal.
$C(t)$ is the CA code of the GPS signal.
$D(t)$ is the data of the GPS signal
τ_m is the code delay
f_m is the remain frequency after down converting to baseband.
Φ_m is the carrier phase of the received signal

The tracking output of the m^{th} channel before the clock mitigation block is defined as follows:

$$IQ_m(k) = \frac{1}{T_d} \int_{kT_d}^{(k+1)T_d} s_m(t) * C(t-\tau_1) \exp\left(-j(2\pi f_1 t + \Phi_1)\right) \mathrm{dt} \tag{5}$$

Clearly, the code and carrier of the first channel are always tuned to align with the incoming signal due to the tracking loop applied to this channel. Therefore, the carrier phase of this channel varies around zero and that of other channels represent the phase difference between them and the first one. This value can be considered unchanged in short time period because the satellite direction with respect to the receiver change slowly. In other words, the phase difference should fluctuate around a fixed value.
However, a non-zero value of the frequency error cause a rapid change in carrier phase (figure 4). Clearly, it is impossible to determine the phase difference between the two elements in the presence of the clock phase shift. Consequently, the antenna array is unusable if the frequency shift cannot be mitigated. Therefore, it requires a suitable solution to mitigate this clock phase drift.



(A)Tracking output (B)Scatter plot

Figure 4. The tracking output (A) and scatter plot (B) of the PRN22 satellite before mitigating clock drift

3.3 Clock phase shift mitigation solution

According to [9], the impact of the delay due to element spacing on code phase is much smaller than carrier phase. In the other words, the difference between τ_m and τ_1 are negligible. Therefore, the tracking output now becomes:

$$IQ_m(k) \approx \frac{1}{T_d}\int_{kT_d}^{(k+1)T_d} \exp\big(j(2\pi\Delta f t + \Delta\phi)\big) \tag{6}$$

$$= \frac{1}{j2\pi\Delta f T_d}\exp(j(\Delta\Phi + 2\pi k\Delta f T_d))\,(\exp(j2\pi\Delta f T_d) - 1) \tag{7}$$

$$= \frac{1}{j2\pi\Delta f T_d}\exp(j(\Delta\Phi + 2\pi k\Delta f T_d))\,(\exp(j2\pi\Delta f T_d) - 1) \tag{8}$$

$$= \frac{\exp(j\pi\Delta f T_d)\,(\exp(j\pi\Delta f T_d) - \exp(-j\pi\Delta f T_d))}{2\pi\Delta f} \tag{9}$$

$$= \exp(j\Delta\Phi)\exp(j\pi\Delta f T_d)\,\mathrm{sinc}(\pi\Delta f T_d) \tag{10}$$

$$= \exp\left(j(\Delta\Phi + \pi\Delta f T_d(2k+1))\right) \quad (11)$$

Since $\Delta f T_d$ is pretty small, $\text{sinc}(\pi\Delta f T_d) \approx 1$.
Therefore:

$$\angle(IQ_m(k)) = \Delta\Phi_{\text{m}} + \pi\Delta f T_d(2k+1) \bmod 2\pi \quad (12)$$

Clearly, if the frequency error $\Delta f \neq 0$, the phase error will be rotated as shown in Figure 4.
From (12), it derives the following formula:

$$\angle(IQ_m(k)) - \angle\left(IQ_m(k-1)\right) = 2\pi\Delta f T_d \quad (13)$$

To estimate such value, the following discriminator is used:

$$\Delta f = \frac{\angle(IQ_m(k)) - \angle\left(IQ_m(k-1)\right)}{2\pi T_d}$$

The loop-filter used to estimate this drift is presented in Figure 5.

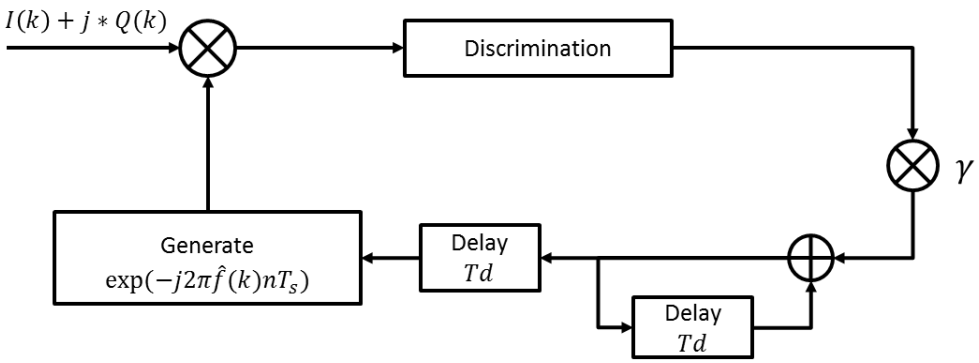


Figure 5. The loop filter using for estimating the clock drift

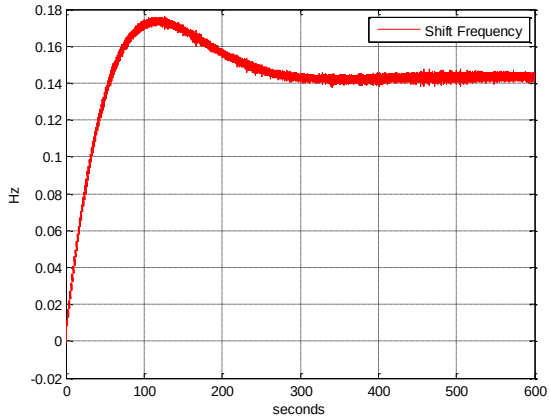


Figure 6. The estimated frequency shift using the loop filter.

As shown in Figure 6, the steady state of the loop filter presents the frequency shift of the m^{th} element with respect to the first element. Because of the small value of this quantity, the loop filter takes a long time to reach the steady state.
The tracking output after mitigating the clock drift is shown in the Figure 7. It can be seen that the phase offset of the second channel oscillate a fixed value. This presents the success of the both data synchronization and clock shift mitigation.

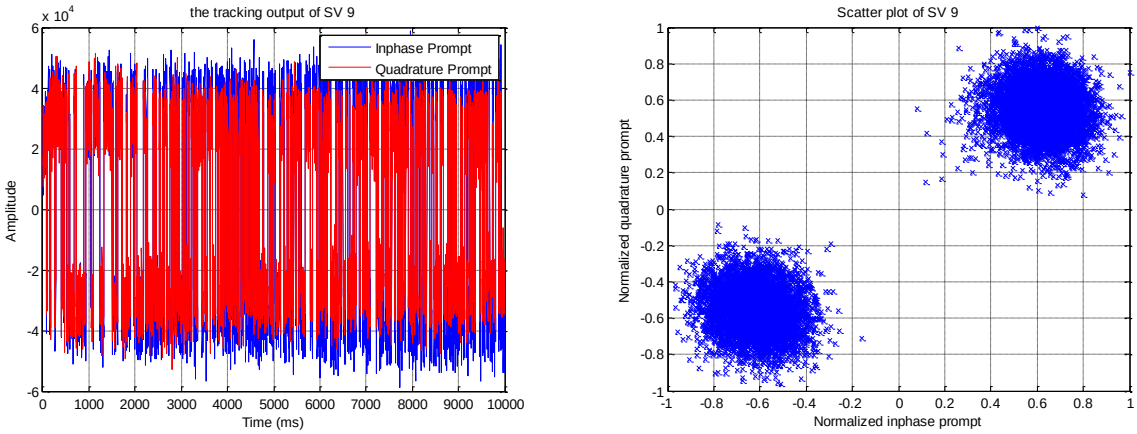


Figure 7. The scatter plot of the signal after mitigating clock phase shift

4. Antenna Array Frontend Verification

Experiments are conducted with our simulator and to verify: phase difference between frontends, the gain using beamforming algorithm (3 elements frontend).

4.1. Phase difference between frontends

The first experiment is conducted to verify the reliability of the antenna array frontend.
The three elements of the frontend are connected to a splitter as shown in Figure 8. A simulated signal is then transmitted to the frontend. This simulated signal was generated using the simulator (Thuan Nguyen Dinh *et al.*, 2015). This simulator allows us easily controlling external factors (e.g. multipath, interference) which may corrupt the received signal.

The simulated satellite signals are transmitted from the same source, thus the phase difference among the elements now depends only on the cable length and internal architecture of each element. It is obvious that this delay is the same for all satellite. Therefore, the phase difference must be comparable to all satellites in view.

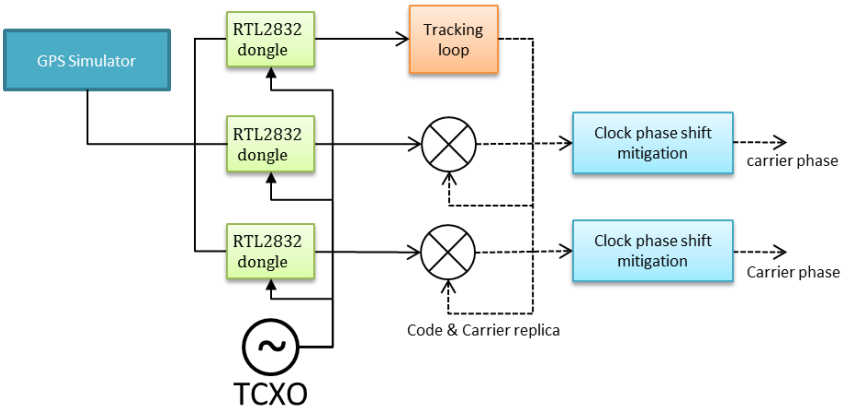


Figure 8. The setup of the verification of the frontend using a GPS simulator

The experimental results illustrated in Figure 9 presents the carrier phase consistency of all satellites, it means that the proposed antenna array frontend works properly as expected.

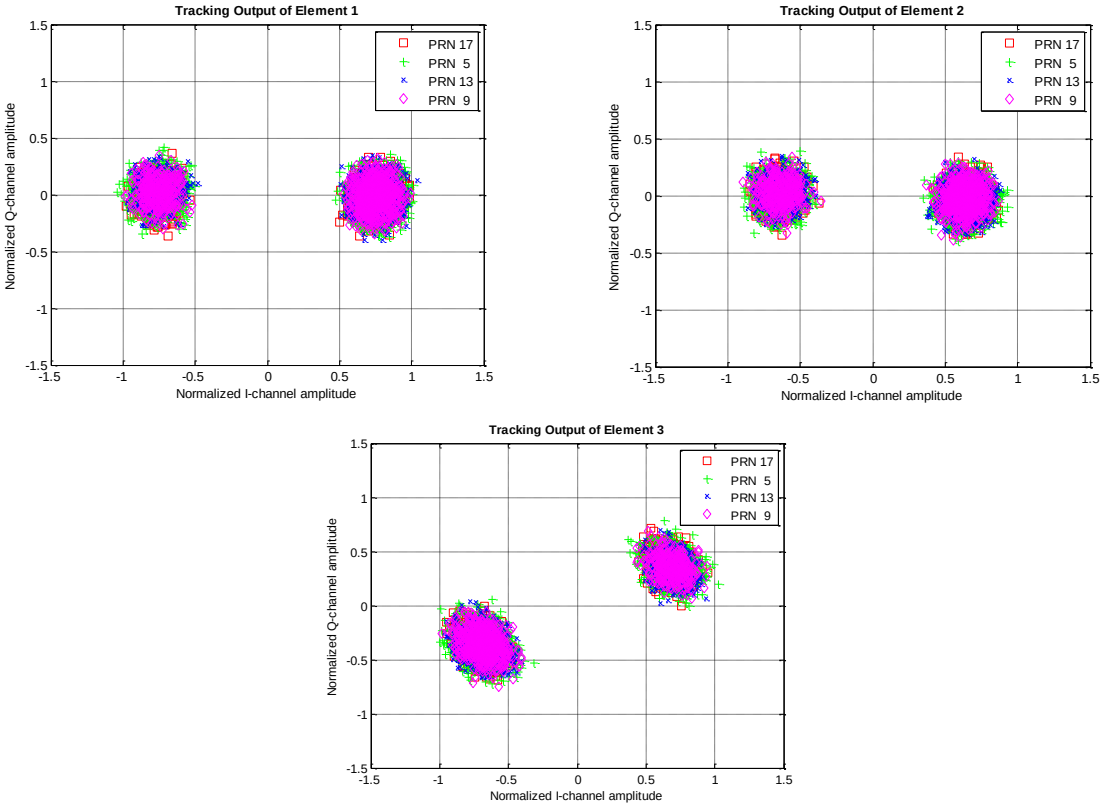


Figure 9. Scatter plot of satellites in view

4.2. Carrier to noise ratio (C/N_0) improvement

The setup of this experiment is the same as the previous experiment where all elements of the frontend are fed by the same signal.

The analytic model of the received signal from 3 elements can be expressed as follows (Thuan Nguyen Dinh *et al.*, 2015):

$$r_1(nT_s) = g_1\sqrt{P_s}C(nT_s - \tau)D(nT_s - \tau)\exp(j2\pi fnT_s)\exp(j\Phi_1) + \eta_1(nT_s) \quad (14)$$

$$r_2(nT_s) = g_2\sqrt{P_s}C(nT_s - \tau)D(nT_s - \tau)\exp(j2\pi fnT_s)\exp(j\Phi_2) + \eta_2(nT_s) \quad (15)$$

$$r_3(nT_s) = g_3\sqrt{P_s}C(nT_s - \tau)D(nT_s - \tau)\exp(j2\pi fnT_s)\exp(j\Phi_3) + \eta_3(nT_s) \quad (16)$$

where:

g_i is the gain of the i^{th} element

η_i is the noise of the i^{th} element

P_s is the received signal power of the kth satellite at the i^{th} element.

τ is the propagation delay of the considered satellite.

f is the Doppler frequency of the considered satellite.

Φ_i is the initial carrier phase of the considered satellite at the i^{th} element.

T_s is the sampling period.

The beamed signal is formed as follows:

$$r(nT_s) = r_1(nT_s) + r_2(nT_s)\exp(j(\Phi_1 - \Phi_2)) + r_3(nT_s)\exp(j(\Phi_1 - \Phi_3)) \quad (17)$$

$$= (g_1 + g_2 + g_3)\sqrt{P_s}C(nT_s - \tau)D(nT_s - \tau)\exp(j2\pi fnT_s)\exp(j\Phi_1) + \eta(nT_s) \quad (18)$$

If the noise η_1, η_2, and η_3 have the same power, the power of the synthetic noise $\eta(nT_s)$ are 3 times of the component noise power. The C/N_0 of the beamed signal can be expressed as follows:

$$\gamma = \frac{\left(\sqrt{\gamma_1} + \sqrt{\gamma_2} + \sqrt{\gamma_3}\right)^2}{3} \quad (19)$$

where γ is the C/N_0 of the beamed signal and γ_i is the C/N_0 of the i^{th} element of the frontend.

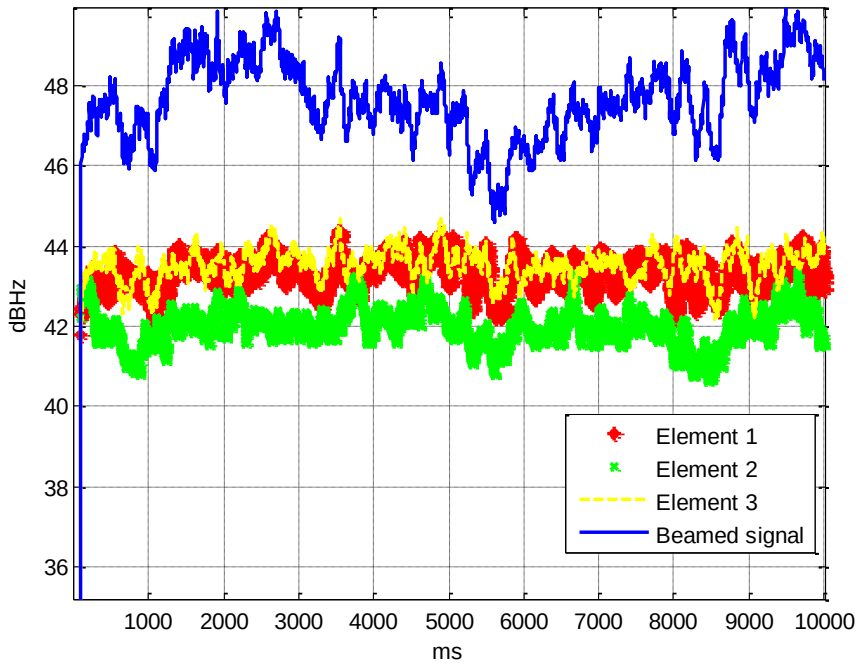


Figure 10. C/N_0 of the satellite PRN 09 for the received signal at every element and beamed signal

Experiment results illustrated in figure 10 show that the C/N_0 of the beamed signal complies with (19).

5. CONCLUSIONS

The paper presented the practical consideration in designing an antenna array for GNSS application. The result shown in this paper is a very promising for not only GNSS application but also the other field.
In the future, we will use such antenna array frontend to suppress interference, point to the source of the interference and spoofing.

REFERENCES

Arnold, Anthony D., and Konstanty Bialkowski (2015). Distributed Open Source Software-Defined GPS *Proceedings of the ASWEC 2015 24th Australasian Software Engineering Conference. ACM*.

Borio, D., Gioia, C., Dimc, F., Bazec, M., Fortuny, J., Baldini, G., & Basso, M.(2015) An Experimental Evaluation of the GNSS Jamming Threat.

Parkinson, Bradford W., and James J. Spilker (1996). *Progress In Astronautics and Aeronautics*: Global Positioning System: Theory and Applications.

De Lorenzo, David S. (2007) Navigation accuracy and interference rejection for GPS adaptive antenna arrays, *Diss. Stanford University*

Backén, Staffan, Dennis M. Akos, and Magnus L. Nordenvaad (2008) Post-processing dynamic GNSS antenna array calibration and deterministic beamforming, *Proceedings of the 21st International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2008)*, Savannah, GA, USA. Vol. 1619.

Backén, Staffan, and Dennis Akos (2006) Antenna array calibration using live GNSS signals.

Proceedings of the third ESA Workshop on Satellite Navigation User Equipment Technologies: NAVITEC.

Borre, K., Akos, D. M., Bertelsen, N., Rinder, P., & Jensen, S. H. (2007) A software-defined GPS and Galileo receiver: a single-frequency approach. *Springer Science & Business Media*.

Thuan Nguyen Dinh, Tung Ta Hai, Letizia Lo Presti (2015) A software based multi-IF output simulator *IS-GNSS*, Kyoto, Japan, November 16-19, 2015

Nooelec, http://www.nooelec.com/store/sdr/sdr-receivers/nesdr-mini2-rtl2832u-r820t2.html