# Sub-picosecond Software Defined Radio receiver synchronization for multi-radiofrequency band time and frequency transfer

J.-M Friedt
*FEMTO-ST Time & Frequency, Besançon, France*
Email: jmfriedt@femto-st.fr

C. Plantard
*ESA ESTEC, Noordwijk, The Netherlands*
Email: Cedric.Plantard@esa.int

***Abstract*—Multichannel coherent Software Defined Radio Receivers are synchronized for simultaneously recording radiofrequency signals broadcasting time and frequency transfer over high and low L-bands by recording on a dedicated channel a pseudo random code sequence. Despite the narrow (5 MHz) band channel available for recording the digital stream, sub-picosecond synchronization is achieved thanks to the excellent signal to noise ratio of the reference channel recording. Demonstration of time and frequency comparison of Galileo E1 and E5 is achieved using this setup after processing the downlink signal with the free, opensource** `gnss-sdr`**.**

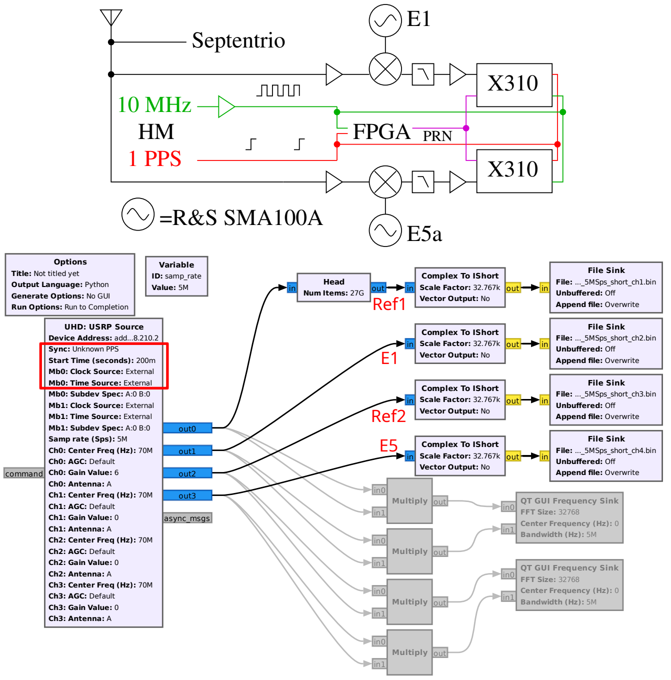
***Index Terms*—GNSS, Galileo, SDR,** `gnss-sdr`**, GNU Radio.**

In the context of Software Defined Radio (SDR), the complex mixer (IQ) output is digitized as early as possible to allow for recording and digital post-processing of the recorded samples. Despite tremendous computational capability, data storage and data communication improvements, radiofrequency links limit the available channel bandwidth for satellite communication to a few MHz to a few tens of MHz, well below the bandwidth that would be expected to achieve sub-nanosecond time-transfer resolution according to the RADAR range resolution equation which equates the timing resolution to the inverse of the signal bandwidth.

However, it is well known that by spreading the timing information over the duration of a pseudo-random sequence and assessing the time delay as a fitted cross-correlation peak for achieving super-resolution, a timing resolution improvement equal to the signal to noise ratio of the received signal can be achieved.

## I. SDR SYNCHRONIZATION

In this demonstration (Fig. 1), dual-coherent SDR receivers are assessed for time transfer, with one channel dedicated to synchronization by recording a pseudo-random number (PRN) sequence and the other channel recording the satellite downlink band. We exclusively focus on the Ettus Research X310 fitted with BasicRX daughter boards feeding each Analog to Digital Converter (ADC) input with the radiofrequency signal without ampification or frequency transposition, since other SDR platforms with advanced radiofrequency frontend capabilities exhibit excessive jitter [1] whose cause cannot be isolated due to lack of description of their internals.



**Options**
**Title:** Not titled yet
**Output Language:** Python
**Generate Options:** No GUI
**Run Options:** Run to Completion

**Variable**
**ID:** samp_rate
**Value:** 5M

**UHD: USRP Source**
**Device Address:** add...8.210.2
**Sync:** Unknown PPS
**Start Time (seconds):** 200m
**Mb0: Clock Source:** External
**Mb0: Time Source:** External
**Mb0: Subdev Spec:** A:0 B:0
**Mb1: Clock Source:** External
**Mb1: Time Source:** External
**Mb1: Subdev Spec:** A:0 B:0
**Samp rate (Sps):** 5M
**Ch0: Center Freq (Hz):** 70M
**Ch0: AGC:** Default
**Ch0: Gain Value:** 6
**Ch0: Antenna:** A
**Ch1: Center Freq (Hz):** 70M
**Ch1: AGC:** Default
**Ch1: Gain Value:** 0
**Ch1: Antenna:** A
**Ch2: Center Freq (Hz):** 70M
**Ch2: AGC:** Default
**Ch2: Gain Value:** 0
**Ch2: Antenna:** A
**Ch3: Center Freq (Hz):** 70M
**Ch3: AGC:** Default
**Ch3: Gain Value:** 0
**Ch3: Antenna:** A

**Head**
**Num Items:** 27G

**Complex To IShort**
**Scale Factor:** 32.767k
**Vector Output:** No

**File Sink**
**File:** ..._5MSps_short_ch1.bin
**Unbuffered:** Off
**Append file:** Overwrite

**Complex To IShort**
**Scale Factor:** 32.767k
**Vector Output:** No

**File Sink**
**File:** ..._5MSps_short_ch2.bin
**Unbuffered:** Off
**Append file:** Overwrite

**Complex To IShort**
**Scale Factor:** 32.767k
**Vector Output:** No

**File Sink**
**File:** ..._5MSps_short_ch3.bin
**Unbuffered:** Off
**Append file:** Overwrite

**Complex To IShort**
**Scale Factor:** 32.767k
**Vector Output:** No

**File Sink**
**File:** ..._5MSps_short_ch4.bin
**Unbuffered:** Off
**Append file:** Overwrite

**QT GUI Frequency Sink**
**FFT Size:** 32768
**Center Frequency (Hz):** 0
**Bandwidth (Hz):** 5M

**QT GUI Frequency Sink**
**FFT Size:** 32768
**Center Frequency (Hz):** 0
**Bandwidth (Hz):** 5M

Fig. 1. Experimental setup: an FPGA generates a PRN sequence setting the BPSK modulation on a 70 MHz carrier and feeds the reference channel of each Ettus Research X310 SDR platform. The second channel of each SDR receiver records the baseband signal of E1 and E5a bands offset by 70 MHz. The DDC of all SDR receivers are set to 70 MHz. The BasicRX daughter board neither amplifies nor frequency transposes the recorded signal but directly feeds the ADCs.

An FPGA is programmed to generate a 100 kchip-long PRN sequency Binary Phase Shift Keying (BPSK) modulating a 70 MHz carrier digitally generated and output as a GPIO toggling prior to bandpass filtering with a Surface Acoustic Wave bandpass filter. All X310 Digital Down Conververtes (DDC) are set to a local oscillator frequency of 70 MHz, while the external GNSS radiofrequency signal is frequency transposed by setting the local oscillator to $1575.42 - 70 = 1505.42$ MHz for Galileo E1 and $1176.45 - 70 = 1106.45$ MHz for Galileo E5a.

## II. GNSS SIGNAL RECORDING

All signals are recorded on Linux EXT2 formatted filesystems since storing on an unformatted raw disk partition did

PRN generation in the FPGA was developed by G. Goavec-Merou, now at Enjoy Digital (France). The free and opensource SDR community is acknowledged for developing and maintaining the software frameworks allowing for this work to be completed, including authors and maintainers of GNU Radio, `rinex-cli` and `gnss-sdr`. The digital electronics workpackage of the Oscillator Instability Measurement Platform (OscIMP) PIA grant prompted these investigations and supported financially hardware acquisition. F. Vernotte (FEMTO-ST) produced the Allan Time Deviation plot using SigmaTheta.

not lead to reduced loss of data. All data are post-processed for generating 30 s RINEX files using `gnss-sdr` v0.0.19 since available computational power does not allow for real time processing of these constellation signals. Simultaneously, the RINEX file generated by a Septentrio PolaRx5TR is used for pseudo-range comparison and timing analysis. While `gnss-sdr` provides similar $C/N_0$ carrier to noise density ratio than the hardware receivers, the number of tracked satellites is systematically lower with the former than the latters (Fig. 2).

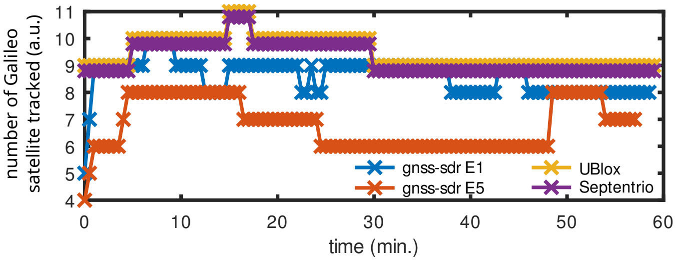


Fig. 2. Number of satellites tracked by the Septentrio PolaRx5TR receiver (“Septentrio”), a UBlox-Zed F9P (“UBlox”) and `gnss-sdr` under the same configuration.

## III. GNSS SIGNAL PROCESSING AND TIME AND FREQUENCY ANALYSIS

The PRN reference channel correlation peak is extracted for each channel individually with respect to a local copy of the code sequence, and the correlation peak is oversampled and fitted by a parabolic function for super resolution improving the timing resolution over the sampling period by a factor equal to the signal to noise-ratio. Fig. 3 demonstrates how the X310 clock fluctuates over time (top and middle charts) but the relative difference between the two input channels exhibit sub-ps standard deviation despite a 5 MHz sampling frequency (200 ns sampling period).

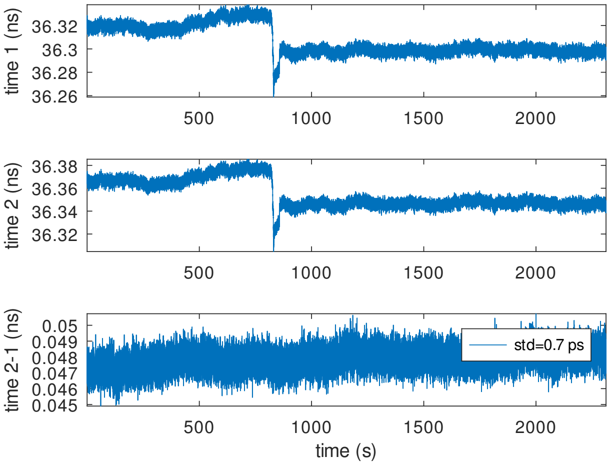


Fig. 3. Timestamp of two Ettus Research X310 SDR platforms (top, middle) fitted with BasicRX frontend boards, and time difference (bottom) with sub-ps standard deviation over 30 minutes despite sampling at 5 MS/s (200 ns sampling period).

Subtracting the pseudo-range information [2] collected from the RINEX files results in the time difference estimate between the Septentrio GNSS receiver and the SDR collected samples (Fig. 4), both connected to the same antenna, with results summarized as a function of sampling bandwidth in Tab. I.

TABLE I

STANDARD DEVIATION ON THE CODE PSEUDO-RANGE DIFFERENCE BETWEEN SEPTENTRIO HARDWARE RECEIVER AND `gnss-sdr`. EACH COLUMN: STANDARD DEVIATION FOR ALL PSEUDO-RANGE DIFFERENCES/MEDIAN VALUE OF EACH 30” EPOCH.

| sampling freq. $f_s$ | $\sigma_{L1}$ (ns) | $\sigma_{E1}$ (ns) | $\sigma_{E5}$ (ns) |
|---|---|---|---|
| 5 MHz | 23.6/6.7 | 16.5/3.2 | 13.3/5.3 |
| 10 MHz | 21.6/7.1 | 13.8/4.2 | 12.7/2.2 |
| 12.5 MHz | 16.7/6.4 | 19.3/5.1 | 15.1/4.0 |

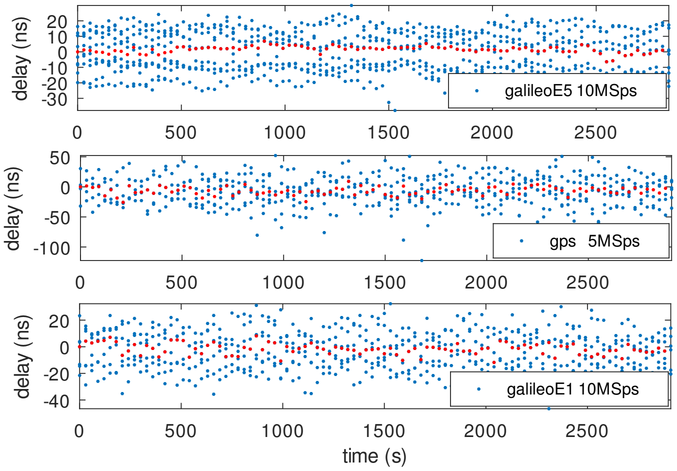


Fig. 4. Common-clock Septentrio hardware receiver to `gnss-sdr` generated pseudo-range estimates from the SDR samples relative time diffference, for Galileo E1 and E5 and GPS. Blue are all pseudo-ranges, red is median value for each epoch.

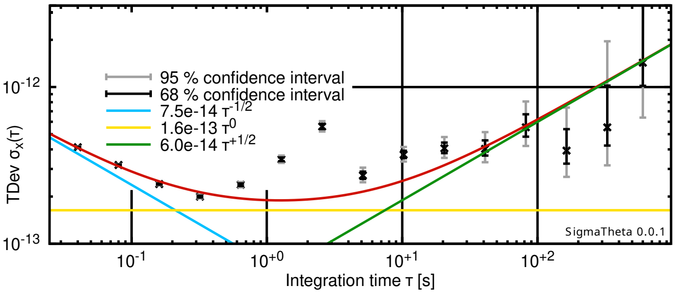


Fig. 5. Time deviation of the delay between SDR channels.

## IV. CONCLUSION

We have demonstrated that X310 SDR platforms fitted with BasicRX connecting the radiofrequency inputs to the analog to digital converters can be synchronized to sub-ps (Fig. 5) with a 5 MHz wide channel, while the second channel is used to record the baseband signal using a real external mixer for frequency transposition of any radiofrequency signal, including L-band GNSS. The principle was demonstrated when comparing time differences between E1 and E5a time transfer signals.

## REFERENCES

[1] J.-M. Friedt and G. Goavec-Merou, “Time of flight measurement with sub-sampling period resolution using software defined radio,” in *13th GNU Radio Conference*, 2023, https://pubs.gnuradio.org/index.php/grcon/article/view/142.

[2] GeoRust RINEX Team, “Rinex: analysis and processing,” 2023, https://georust.org.