Generating a timing information (1-PPS) from a software defined radio decoding of GPS signals

D. Rabus[1], G. Goavec-Merou[1], G. Cabodevila[1], F. Meyer[3], J.-M Friedt[1]

[1]FEMTO-ST Institute, Time and Frequency Department, Besançon, France
[2]OSU THETA, Besançon, France

e-mail: jmfriedt@femto-st.fr
https://github.com/oscimp/gnss-sdr-1pps

May 28, 2021

https://github.com/topics/gnss-sdr

Software Defined Radio (SDR) implementation of a GNSS receiver

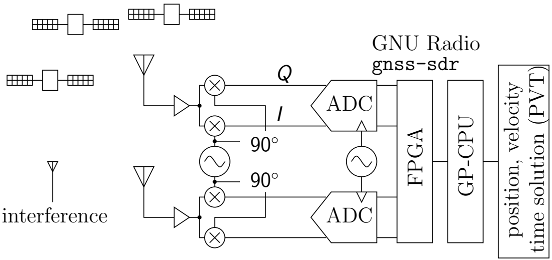
- SDR: minimize hardware to RF frontend (antenna, amplifier and frequency transposition)
- Process all radiofrequency signal as software on complex values I+jQ
- Gain access to the raw IQ stream for incoming plane wave physical characteristics
- Implement anti-jamming/anti-spoofing processing [1] in the SDR frontend (null-steering)
- Full control of the complete signal processing chain
- Steer local oscillator to match incoming timing information

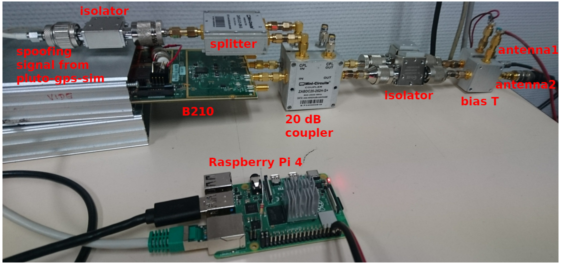
```
Current receiver time: 1 min 52 s
Position at 2001-Oct-12 06:25:15.500000 UTC using 4 observations is Lat = 47.251495868 [deg], Long = 5.993075579 [deg], Height = 360.117 [m]
init offset: 0.015487310819 [s]init LO frequ: 12.346229542937 [Hz]
Velocity: East: -0.282 [m/s], North: -1.510 [m/s], Up = -1.480 [m/s]
Position at 2001-Oct-12 06:25:16.000000 UTC using 4 observations is Lat = 47.251542192 [deg], Long = 5.993097609 [deg], Height = 377.010 [m]
Velocity: East: 0.249 [m/s], North: 0.922 [m/s], Up = 0.651 [m/s]
Current receiver time: 1 min 53 s
Position at 2001-Oct-12 06:25:16.500000 UTC using 4 observations is Lat = 47.251576137 [deg], Long = 5.993169968 [deg], Height = 373.085 [m]
Velocity: East: 0.289 [m/s], North: 0.988 [m/s], Up = 1.087 [m/s]
New GPS NAV message received in channel 0: subframe 5 from satellite GPS PRN 12 (Block IIR-M)
New GPS NAV message received in channel 2: subframe 5 from satellite GPS PRN 24 (Block IIF)
New GPS NAV message received in channel 3: subframe 5 from satellite GPS PRN 02 (Block IIR)
New GPS NAV message received in channel 4: subframe 5 from satellite GPS PRN 19 (Block IIR)
New GPS NAV message received in channel 5: subframe 5 from satellite GPS PRN 32 (Block IIF)
Position at 2001-Oct-12 06:25:17.000000 UTC using 4 observations is Lat = 47.251548923 [deg], Long = 5.993130850 [deg], Height = 363.682 [m]
Velocity: East: -0.154 [m/s], North: 0.483 [m/s], Up = 0.803 [m/s]
Current receiver time: 1 min 54 s
```

How to materialize timing information is 1-Pulse Per Second (1-PPS)?

[1] W. Feng, J.-M Friedt, G. Goavec-Merou, F. Meyer, *Software Defined Radio Implemented GPS Spoofing and Its Computationally Efficient Detection and Suppression*, IEEE Aerospace and Electronic Systems Magazine **36** (3), March 2021 and `https://github.com/oscimp/gnss-sdr`

Problem statement: SDR reception

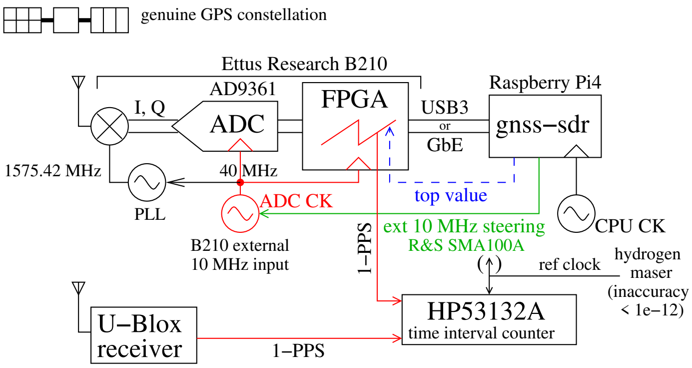




- Two-channel Ettus Research B210/ AD9361 coherent radiofrequency frontend
- Complex frequency transposition
  $I = s(t) \cdot \cos(\omega_{LO} t)$; $Q = s(t) \cdot \sin(\omega_{LO} t)$
- ADC samples at $f_s \geq 1.023$ MS/s both channels ($B \geq 2.046$ MHz)
- FPGA pre-processing and continuous datastream transfer to general purpose central processing unit (GP-CPU) running free, opensource implementation of Global Navigation Satellite System (GNSS) decoder `gnss-sdr`
- compatibility with embedded applications, e.g. Raspberry Pi4 single board computer[2], 1.5 GHz quad-core in `performance` mode
- output: Position, Velocity, Timing (PVT) solution

[2] G. Goavec-Merou, J.-M Friedt, *Porting GNU Radio to Buildroot: application to an embedded digital network analyzer*, FOSDEM 2021, and `gnss-sdr` at `https://github.com/oscimp/oscimp_br2_external`

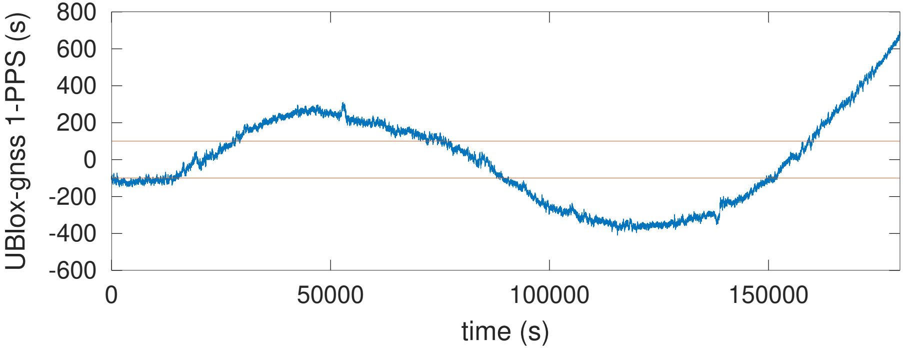
Problem statement: sample timing



40 MHz is 4-times multiplication of reference 10 MHz input of B210
HM always clocks the HP53132A counter and might clock the B210

- All communication between FPGA and GP-CPU hardware asynchronous
- Only ADC sampling time is known **assuming** continuous datastream at f_s
- Control the clock timing the AD9361 frontend ADC, which also clocks the FPGA
- Counter in the FPGA from 0 to $f_s - 1$
- Characterization: HP53132A time- interval counter clocked by Hydrogen Maser (HM) and reference 1-PPS from U-Blox NEO-M8P hardware receiver

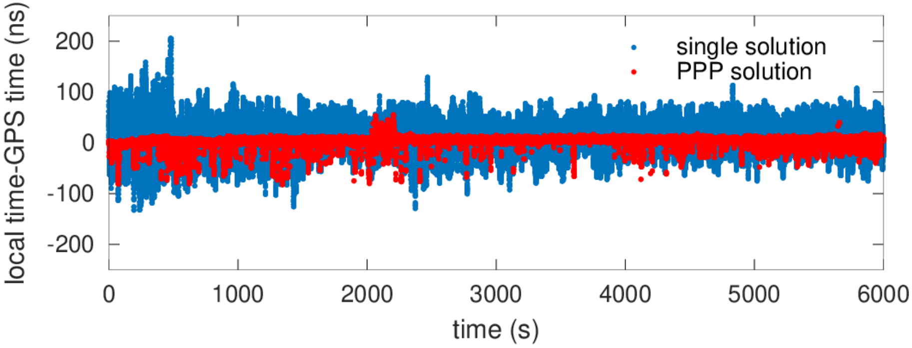
1-PPS output: open loop



- 50 hour long measurement
- time difference between a U-Blox NEO-M8P and the 1-PPS counter implemented in the FPGA
- GP-CPU running `gnss-sdr` and logging the time-difference
- B210 clocked by a *free running* Rohde & Schwarz SMA100A synthesizer set to nominal 10 MHz,
- +11.1 ns/s=ppb drift removed, only the residue is displayed.

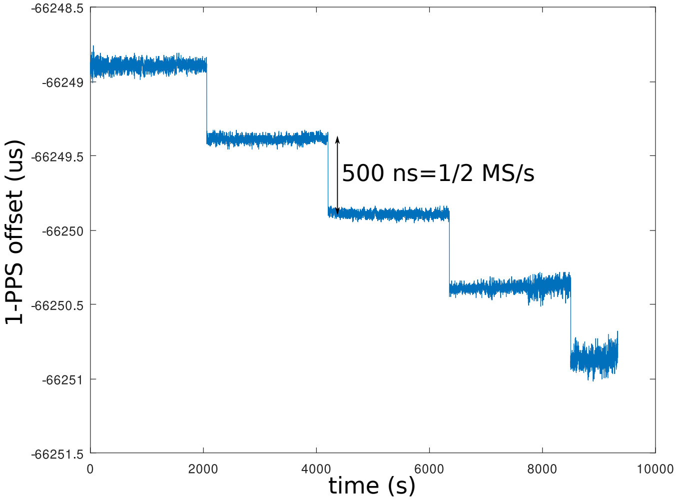
⇒ **objective: steer reference clock to keep 1-PPS within ±100 ns boundary**

1-PPS output: closed loop



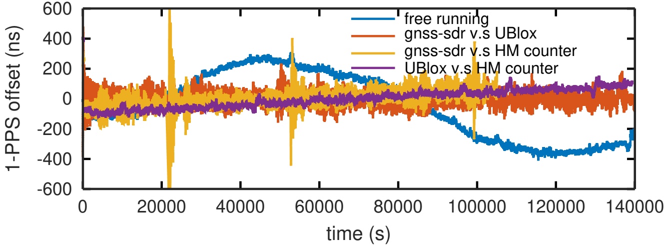
- Control SMA100A frequency with time-offset information from `gnss-sdr` PVT solution
- comparison between the "single solution" and the PPP solution
- sampling period is 1-chip or 20 ms, control loop every 500 ms
- select a sampling rate f_s fractional integer of the 40 MHz clock (1.25 MHz=integer division by 32, 2 MHz=integer division by 20, 1.125 MHz=fractional division by 320/9) to avoid drifting 1-PPS

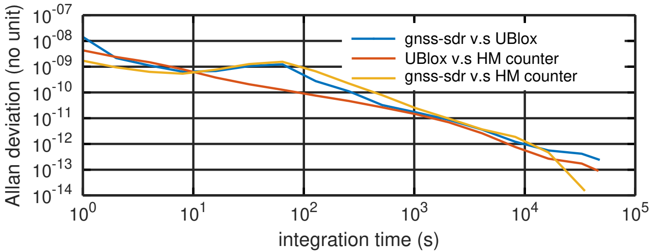
1-PPS output: closed loop issue



- Source → **Conditioner** → Acquisition/Tracking/Telemetry → Observables → PVT
- Useless `resampler` from f_s to f_s: misses one sample every 2^{32}
- Make sure to `Pass_Through` to avoid $1/f_s$ jumps every $2^{32}/f_s = 2147.5$ s at 2 MS/s
- (sampling rate 1.125 MS/s $\Rightarrow 2^{32}/(1.125 \cdot 10^6) = 3817$ s shift by $1/1.125 = 0.888\ \mu$s)

1-PPS output: closed loop result



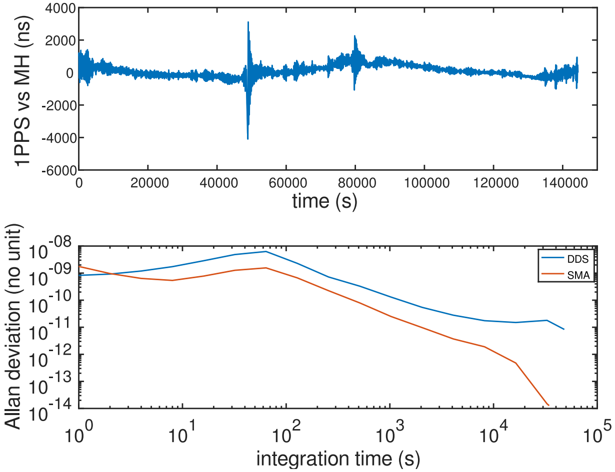


- Long term stability assessment
- Drift of the HM with respect to 1-PPS GNSS visible at $\geq X \cdot 10^4$ s
- Convert time interval x phase-time fluctuation to[3] fractional-frequency fluctuation $y = \dot{x}$
  `SigmaTheta-4.1/1col2col file`
  `SigmaTheta-4.1/X2Y file_2col`
- Allan Time Deviation (MDEV[4]) from 10^{-8} at 1 s (10 ns @ 1 s) $\searrow 1/\tau$
  `SigmaTheta-4.1/MDev file_2col.ykt`
- Control loop time constant visible $\tau \in [10 - 100]$ s

[4] E. Rubiola, *Phase Noise and Frequency Stability in Oscillators*, Cambridge Univ. Press (2009), p.8

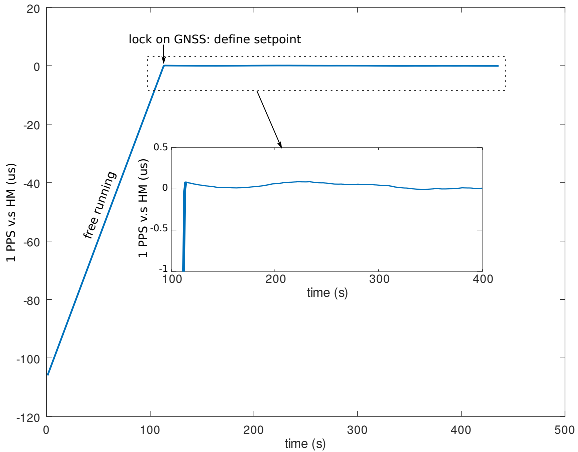
[4] F. Vernotte *& al.*, SigmaTheta at `https://theta.obs-besancon.fr/spip.php?article103&lang=en`

1-PPS output: OCXO→TCXO



- Replace SMA100A OCXO clocked synthesizer with TCXO clocked AD9959 Direct Digital Synthesizer (DDS)
- Short term performance degraded by drift
- 32-bit resolution → 18.6 mHz DDS frequency resolution when clocked by a 20 MHz TCXO internally multiplied by 4
- close to SMA100A 10 mHz frequency resolution

1-PPS output: control loop initialization



- Excessive frequency offset between TCXO frequency and nominal frequency ...
- ... prevents initial fast convergence and leads to excessive oscillations.
- Initial coarse estimate of the frequency offset by *observing* the frequency drift once the first PVT solutions are found before closing the control loop

Conclusion

- Bridge the virtual output of SDR & application to GNSS decoding with hardware 1-PPS output
- Only the ADC timestamp provides accurate timing propagated to the asynchronous processing chain
- $\rightarrow$ need to steer the clock controlling the ADC ...
- ... which also happens to control the FPGA in which the counter is implemented.
- Demonstrated 1-PPS frequency control to 10^{-8}@1 s & $\searrow 1/\tau$ consistent with hardware receiver performance
- Sample loss will hardly affect *frequency control*

Perspective: 1-PPS phase (time-offset) control even when ADC samples are lost

Fork of `gnss-sdr` with 1-PPS support (SMA100A) available at `https://github.com/oscimp/gnss-sdr-1pps`

Acknowledgement: the **free, opensource software/gateware** development community:

- `gnss-sdr` at `github.com/gnss-sdr/` by the Centre Tecnològic Telecomunicacions Catalunya (CTTC, Spain): C. Fernández-Prades and J. Arribas are acknowledged for fruitful discussions
- GNU Radio at `github.com/gnuradio/`
- Ettus Research FPGA `github.com/ettusresearch/fpga` and USRP Hardware Driver (UHD) Software at `github.com/ettusresearch/uhd`
- Buildroot at `https://buildroot.org/`