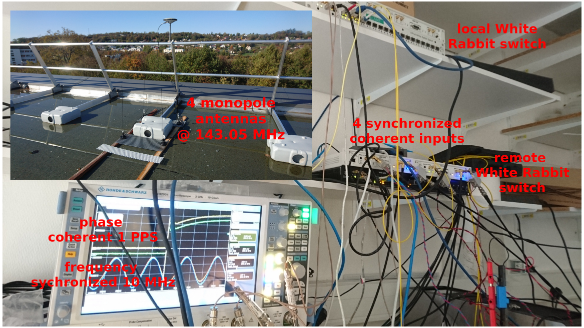
Software Defined Radio for two-way frequency and time transfer: wireless and wired demonstrations

G. Goavec-Merou, J.-M Friedt



GNURadio
THE FREE & OPEN SOFTWARE RADIO ECOSYSTEM

March 10, 2022

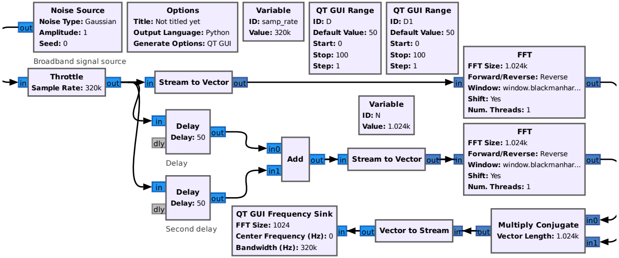
Time and frequency transfer

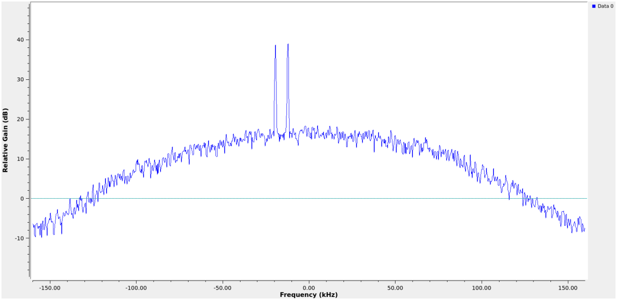
- Frequency = narrowband signal (pure sine wave)
- Time = broadband signal (pulse), resolution $1/B$ with B bandwidth
- Many ways of spreading spectrum:
  - most stupid ... a pulse (see below about T)
  - frequency steps over a given bandwidth (FSCW)
  - frequency sweep over a given bandwidth (chirps)
  - noise (pseudo-random sequence)
  - duration T: SNR gain from sliding average

$$\boxed{\text{Pulse Compression ratio} = T \times B}$$

For a digital signal:
bit rate B × signal duration T = number of bits N
Correlation calculation:
$xcorr(x,y)(\tau) = \int x(t) \cdot y^*(t+\tau)dt$ while $conv(x,y)(\tau) = \int x(t) \cdot y(\tau - t)dt$ and since $FT(conv(x,y)) = FT(x) \cdot FT(y)$, we use

$$\boxed{FT(xcorr(x,y)) = FT(x) \cdot FT^*(y)}$$



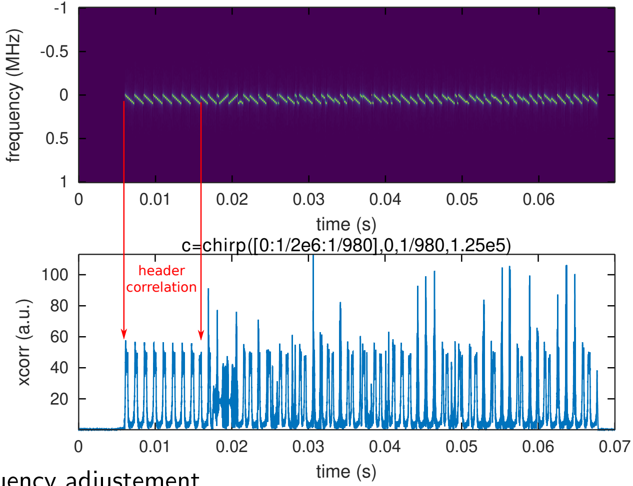


enseignement/efts/2104_xcorr38single_fft.grc

Time and frequency transfer over a terrestrial wireless network

LoRa: **Lo**ng **Ra**nge communication proprietary protocol (Semtech) achieved thanks to *narrowband* communication [1]

- Chirp based modulation: linear frequency sweep (Chirp Spread Spectrum – CSS) spreads the spectrum over $B =$ **125 kHz**, 250 or 500 kHz
- Chirp duration determined by Spreading Factor SF: higher SF = longer chirp T
- Message preamble: 8 up-chirps followed by 2 down-chirps for synchronization
- Payload: jump in frequency within each chirp is each symbol
- One **gateway** (master/server) acting as time server disseminates to multiple **endpoints** (slave/client)

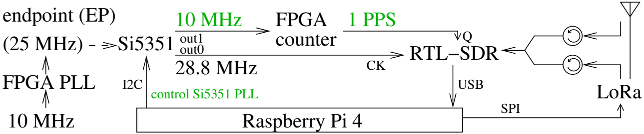


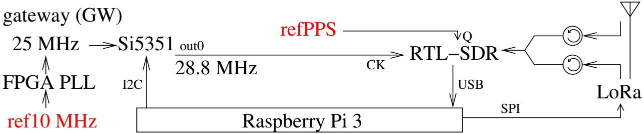
$\Rightarrow$ only preamble will be used for timing and frequency adjustement

[1] M. Knight, *Decoding the LoRa PHY*, 33c3 (2016) at `https://www.youtube.com/watch?v=NoquBA7IMNc` (40 min in the video for the analysis of Semtech's patent)

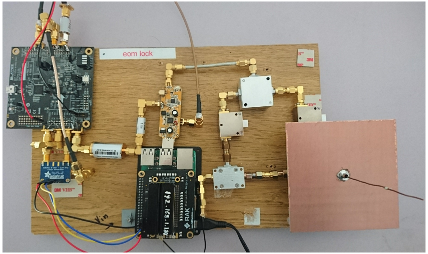
Two-way time transfer basics

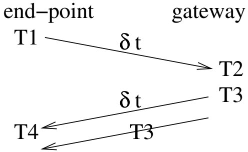
- clock on both ends are not running at the same rate
- clock on both ends are not phase synchronized (time delay)
- the only known timestamp is when sampling the incoming signal by the ADC
  $\Rightarrow$ the ADC clock **must be referenced** to the disciplined/reference oscillator
- physical layer (**OSI layer 0) for monitoring the chirps** using SDR receiver [a]
- communication layers (OSI layer >0) for sharing timestamps of 1-PPS generated by counting local oscillators
- Endpoint: feedback loop (PI control) to steer oscillator
- Endpoint initiates synchro., gateway timestamps **the same** message, returns an answer and then communicates the timestamp of the returned information.

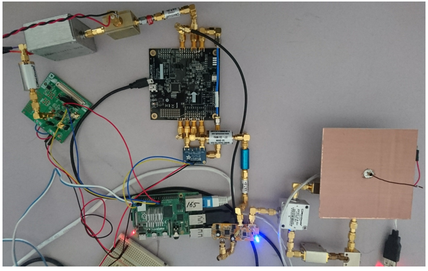




[a] W. Myrick, *RF Ranging with LoRa Leveraging RTL SDRs and GNU Radio*, GRCon (2018) at `https://www.youtube.com/watch?v=vBES6Baxc0M`

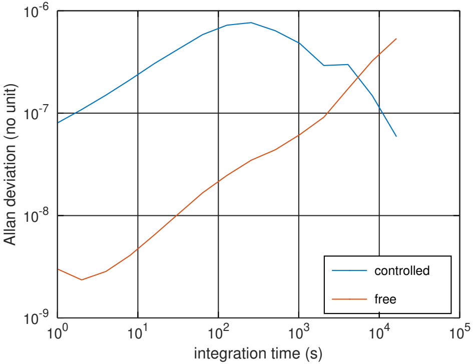


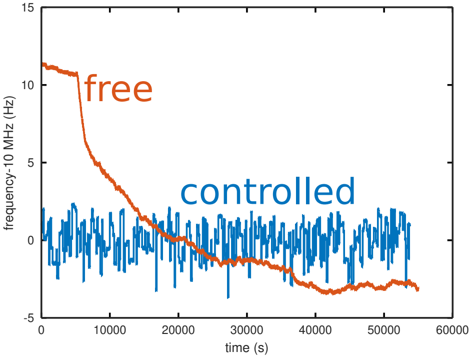




LoRa frequency transfer: results

- Poor short term performance due to poor frequency measurement capability
- Long term improvement of stability
- Intersection of Allan deviations provides feedback time constant
- Here δt is **assumed** null so that the outgoing and incoming message delay is used to calculate frequency offset
- Practically: both δt and frequencies will be off $\Rightarrow$ shorter time delay induces large frequency estimate uncertainty
- Increasing SF will increase the communication time and improve frequency estimate resolution [2]:

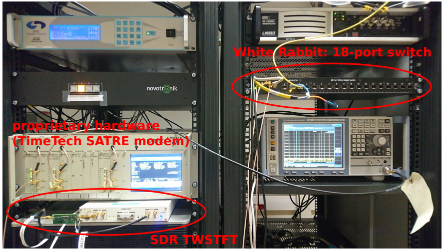




[2] *ThingSat: Internet of isolated objects by satellite* at https://gricad-gitlab.univ-grenoble-alpes.fr/thingsat/public and `https://www.csug.fr/main-menu/projects/thingsat/`

Time and frequency transfer over geostationnary satellite links

SATRE modem, sequel to MITREX modem (1980s) developed in Stuttgart, proprietary hardware manufactured and sold by Timetech (Germany)
No (publicly) available documentation on the link characteristics



- $16383 = 2^{14} - 1$ and there are "only" [a] 756 possible maximum length sequence generators
- truncation to 10000-long, but we do not know what is the starting point
- Bit rate of 2.5 Mb/s: code repeats every 4 ms (10 ms @ 1 Mb/s)
- All 0s is a forbidden state (XOR(0,0)=0) so we might start with all 1s
- Calculate all possible codes, interpolate to match sampling rate, store their Fourier transform ...
- ... and for all possible frequency offset identified by squaring the signal (cancelling BPSK modulation), matrix product of FFT(signal) with stored FFT of codes
- iFFT will display correlation peaks if patterns match.

[a] https://users.ece.cmu.edu/~koopman/lfsr/

Time and frequency transfer over geostationnary satellite links

SATRE modem, sequel to MITREX modem (1980s) developed in Stuttgart, proprietary hardware manufactured and sold by Timetech (Germany)

No (publicly) available documentation on the link characteristics [3] but

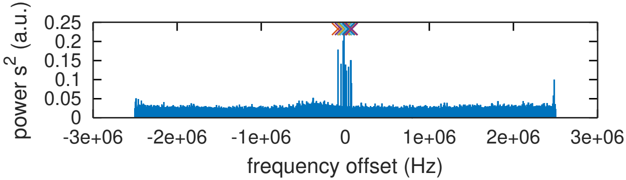
These constraints have been met by the MITREX MODEM. It is an interface between the 70-MHz IF of a standard earth station, a Time Interval Counter TIC and the time keeping hardware, i.e. the atomic clock. The hardware is housed in a 19-inch drawer and operates as a pseudorandom noise sequence (PN) encoder/decoder of the time signal. The PN-code is a truncated maximum length sequence of period 10.000, instead of the 16.383 chips. This period of 10.000 eases the overall system design, as it allows the use of even and decimal dividers for signal processing. The correlation features of the maximum length sequence are almost retained.

2.2 PN-Encoding/Decoding

- $16383 = 2^{14} - 1$ and there are "only" [a] 756 possible maximum length sequence generators
- truncation to 10000-long, but we do not know what is the starting point
- Bit rate of 2.5 Mb/s: code repeats every 4 ms (10 ms @ 1 Mb/s)
- All 0s is a forbidden state (XOR(0,0)=0) so we might start with all 1s
- Calculate all possible codes, interpolate to match sampling rate, store their Fourier transform ...
- ... and for all possible frequency offset identified by squaring the signal (cancelling BPSK modulation), matrix product of FFT(signal) with stored FFT of codes
- iFFT will display correlation peaks if patterns match.

[a] https://users.ece.cmu.edu/~koopman/lfsr/

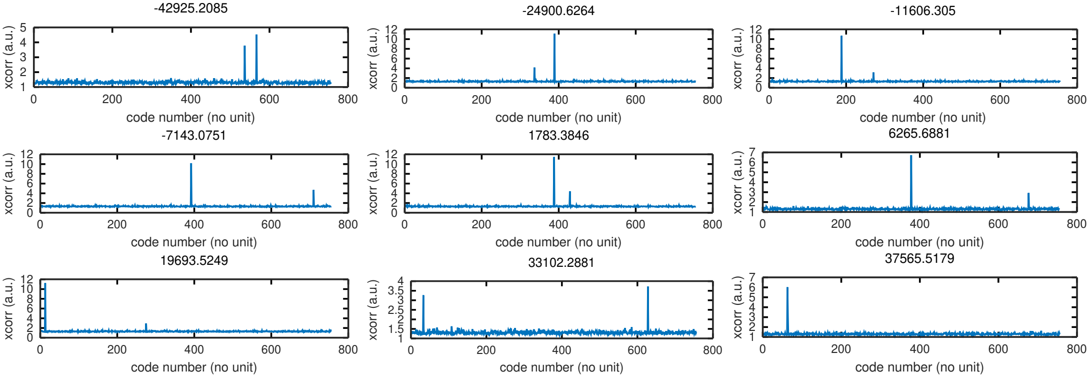
SATRE code reverse engineering



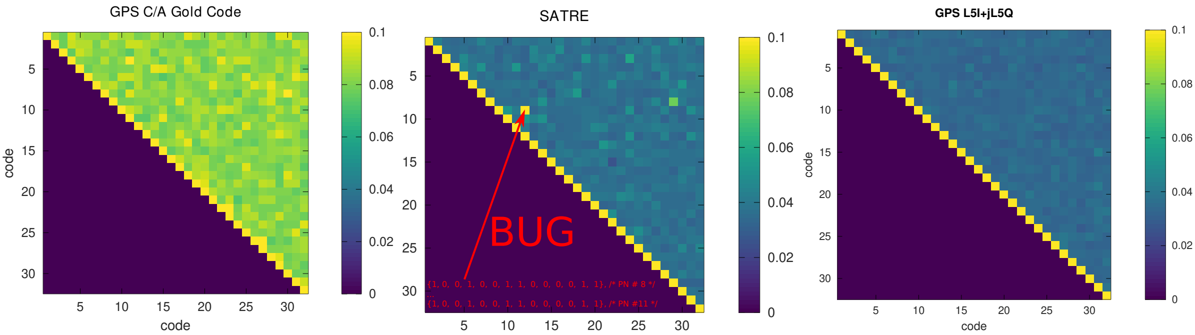
$$s(t) = \exp(j\delta\omega t + j \underbrace{\varphi}_{\{0;\pi\}}) \text{ for BPSK modulation}$$

$$\Rightarrow s^2(t) = \exp(j2\delta\omega t) \text{ since } 2 \times \{0;\pi\} = 0[2\pi]$$

For all possible frequenycy offsets, run all possible LFSR codes (indexed on the abscissa)



SATRE orthogonality analysis

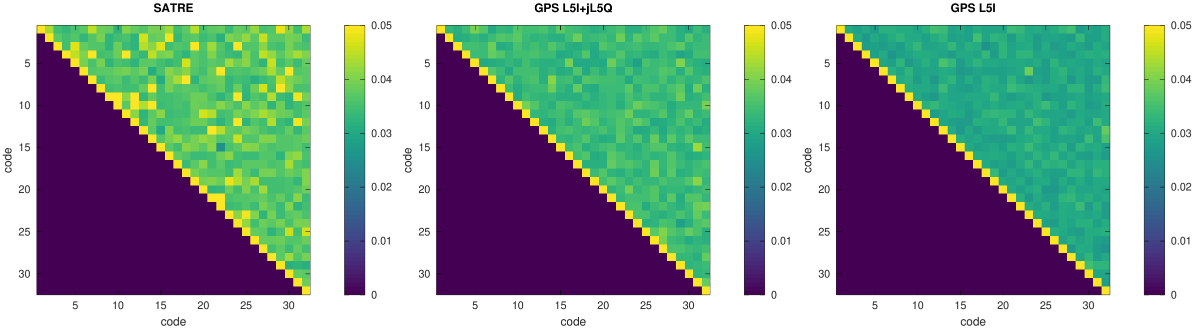


Yet another demonstration why proprietary software is a poor solution without outsider's scrutiny for correcting obvious errors.

L5 codes from `https://github.com/danipascual/GNSS-matlab`

SATRE orthogonality analysis

Both SATRE and L5 ($\in \mathbb{C}$) 10000 bit long sequences: zoom on the color axis L5I$\in \mathbb{R} \downarrow$

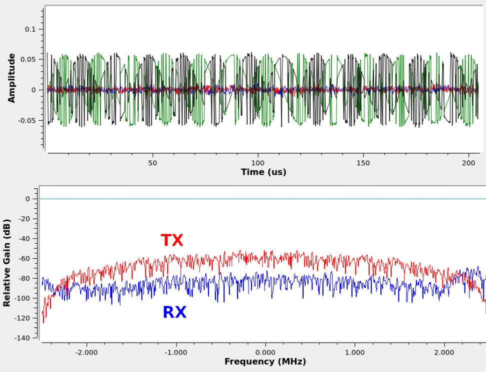


orthogonality: SATRE$=-11$ dB $\rightarrow$ L5 $\mathbb{C} = -13.7$ dB ; $\mathbb{R} = -14$ dB $\Rightarrow \simeq 3$ dB gain

Experimental setup

- Replacement of B210 (including its 77 dB gain) with X310 BasicRX receiver (external 1-PPS and external 10 MHz from stepper clocked by HM)
- Monitoring both SATRE TX and RX signals for differential time of flight + ranging
- $\simeq$ 32 dB gain on RX monitor (single MSA886 amplifier)
- Same processing chain:

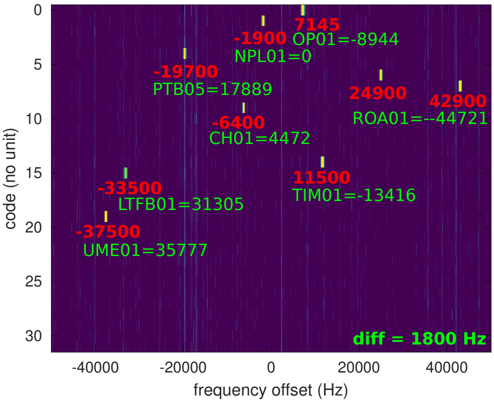
1. coarse frequency offset by squaring the signal to get rid of BPSK: carrier at $2\delta\omega$
2. correlation with known pseudo random pattern
3. search for correlation peaks and quantify sampling f_s rate error: measure phase at correlation peak maxima
4. linear fit of phase evolution and fine frequency offset correction ($xcorr(x, y \cdot \exp(j\varphi)) = \exp(j\varphi) \cdot xcorr(x, y)$)
5. atan(Q/I) provides π-rotation insensitive phase evolution, atan2(Q, I) provides BPSK message (timestamp)
6. 200 ns jumps at beginning of second readily detected on TX and RX
7. $\sigma_{\Delta t} \simeq$13 ns over 4 s $\rightarrow$ 2 ns over 3 min records **expected**



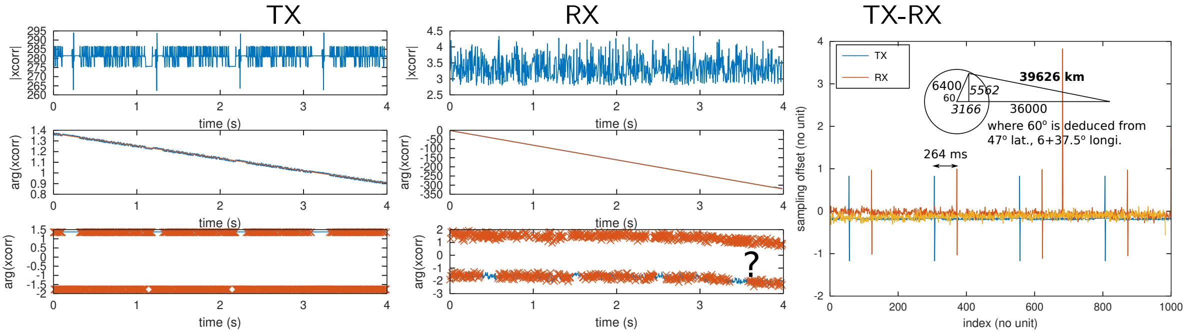
Experimental setup

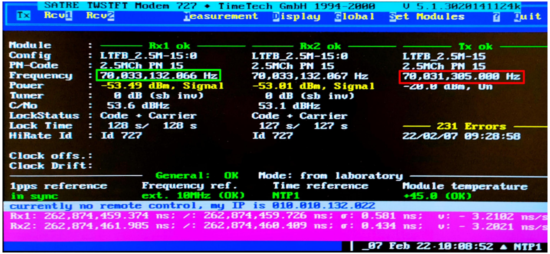
- Replacement of B210 (including its 77 dB gain) with X310 BasicRX receiver (external 1-PPS and external 10 MHz from stepper clocked by HM)
- Monitoring both SATRE TX and RX signals for differential time of flight + ranging
- $\simeq$ 32 dB gain on RX monitor (single MSA886 amplifier)
- Same processing chain:

1. coarse frequency offset by squaring the signal to get rid of BPSK: carrier at $2\delta\omega$
2. correlation with known pseudo random pattern
3. search for correlation peaks and quantify sampling f_s rate error: measure phase at correlation peak maxima
4. linear fit of phase evolution and fine frequency offset correction ($xcorr(x, y \cdot \exp(j\varphi)) = \exp(j\varphi) \cdot xcorr(x, y)$)
5. atan(Q/I) provides π-rotation insensitive phase evolution, atan2(Q, I) provides BPSK message (timestamp)
6. 200 ns jumps at beginning of second readily detected on TX and RX
7. $\sigma_{\Delta t} \simeq$13 ns over 4 s $\rightarrow$ 2 ns over 3 min records **expected**



Ranging measurement (OP: 39549 km=263.66 ms ; LTFB: 39642 km=264.28 ms)



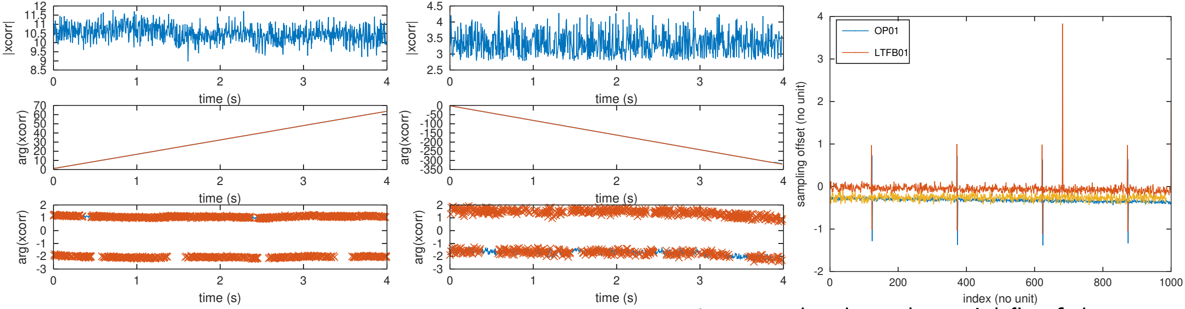


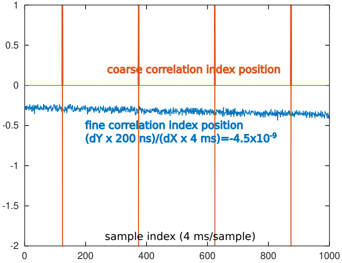
- Record both transmitted and received signal: two way time of flight between ground station and satellite
- Same processing steps on transmitted and received signals
- Frequency offset introduced by satellite uplink to downlink transposition?
- 262.87 ms two way trip delay consistent with our measurement and the distance to the satellite[4]
- 4 ms Pulse Repetition Interval (PRI): **how to measure 264 ms?**

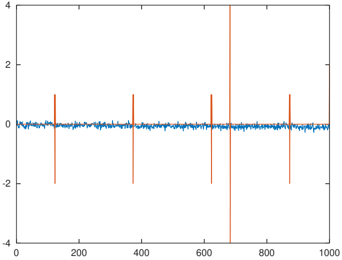
[4] `https://fr.acervolima.com/trouver-tous-les-angles-d-un-triangle-en-3d/`

1-PPS delay comparison

RX OP RX LTFB OP-LTFB



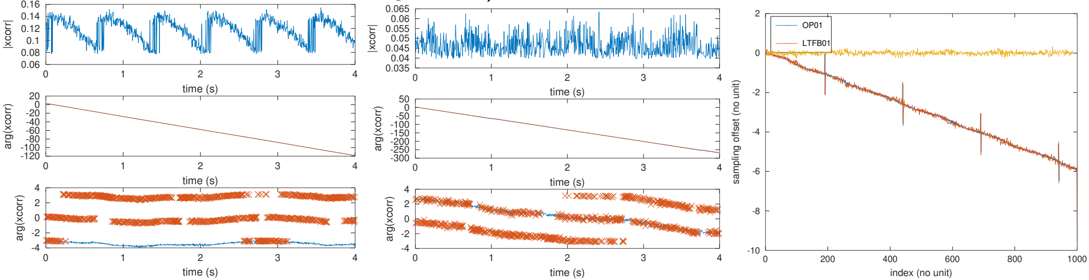




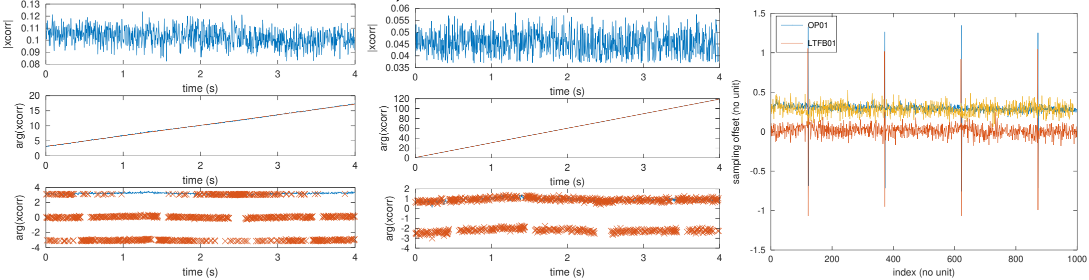
- second order polynomial fit of the correlation peak for sub-sampling period (200 ns at 5 MS/s) resolution identification of correlation peak position
- blue = offset of correlation peak max. to coarse maximum: slope = **freq offset**?
- red = coarse maximum position (200 ns jump at beginning of second)

SATRE code time difference analysis

SDR internal oscillator (free running TXCO)

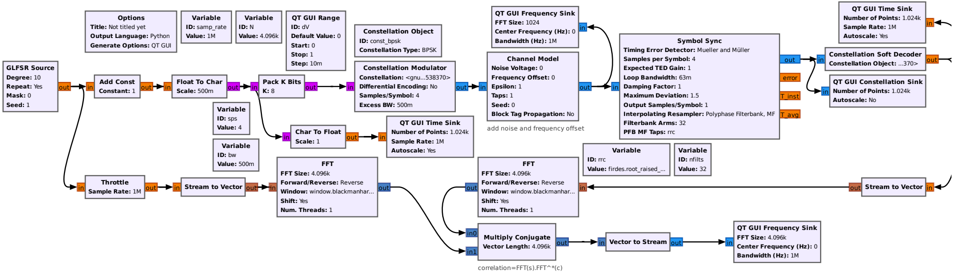


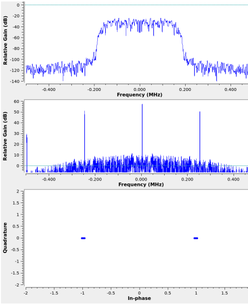
External oscillator (hydrogen maser)



Spectral efficiency: MITREX/SATRE using BPSK

How to squeeze as many bits as possible in a given analog bandwidth?
Shannon: $C = B\log_2(1 + SNR)$



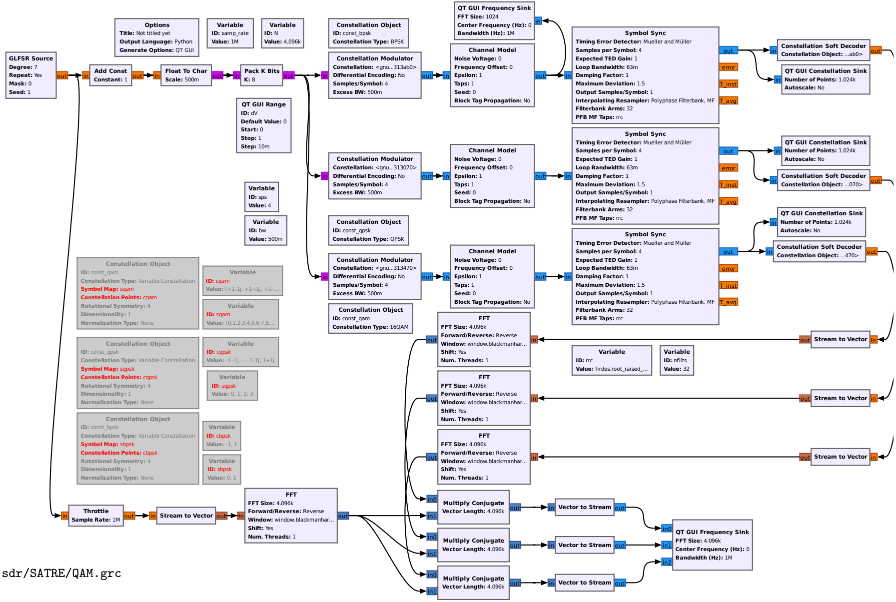


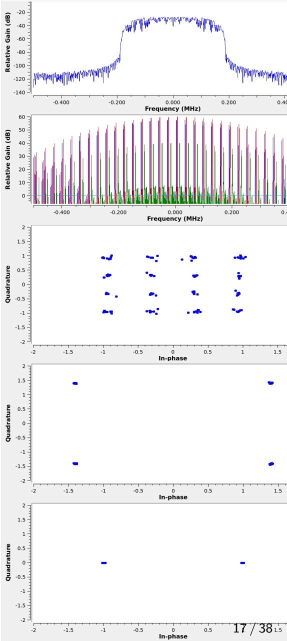
BPSK easy to implement: only $I \in \{\pm 1\}$ channel is needed since $Q = 0$

The more bits C in B, the sharper the cross-correlation and the better the time resolution...
... within allowed SNR [5] (constellation point clouds spreading with noise)

[5] Technical support of H. Boeglen, XLIM, Poitiers

From BPSK to QPSK and QAM ...

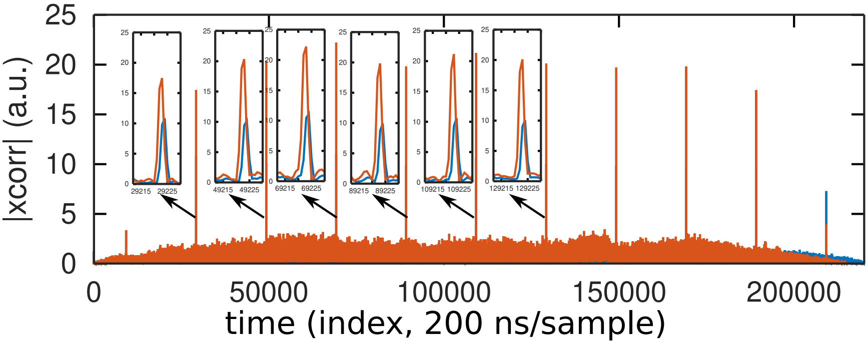




sdr/SATRE/QAM.grc

Work in progress

- Process complete sequence: 5 MS/s$\times$ 2 bytes/sample$\times$2 (complex)$\times$2 channels=7.2 GB/3 min
- FFT (correlation) must be split (900 complex Msamples for 3 minutes) or performed in the time domain (FPGA)



Work in progress to make sure that absolute correlation position is correct (red=split ; blue=single segment of data) – looks like one-index offset remains ...

- if frequency cannot be recovered from carrier, how to compare other than 1-PPS delay drift?

Development roadmap: short term

- emit GPS L1 codes: well known Gold codes, 1023 bit long, shorter Pulse Repetition Interval (PRI) but allows for assessing pseudo random sequence generation from FPGA synchronized in 1-PPS and 10-MHz
- replace GPS L1 with L5 (10-times longer) ⇒ recover MITREX/SATRE PRI and Pulse Compression Ration (PCR), still BPSK modulation
- in both cases, test and demonstrate link + metrologic characterization

Open development framework aimed at providing an opensource, openhardware (Ettus Research X310 + BasicRX) testbed:

`https://github.com/oscimp/gr-satre/`

Development roadmap: mid & long-term

Mid-term

- **increase** pseudo random sequence to $> 2^{21}$ bits to avoid range uncertainty...
- at the **expense** of the number of bits encoding time/date ...
- at the **expense** of large FFT for cross-correlation (not much of an issue on modern computers)
- improvement to BPSK (BOC similarity to frequency offset introduced on SATRE?)

Observatory	lat (deg)	lon (deg)	angle (deg)	delay (ms)
OP	48.84	2.34	59.645	262.23
LTFB	47.25	5.99	60.497	262.82
Greenwich	51.48	0	60.390	262.74
INRIM	45.02	7.64	60.092	262.54
PTB	52.3	10.46	65.826	**266.56**
SW Spain	36	-9	44.685	252.70
NW Finland	68	42	86.086	281.41

Long-term

- optimize usage of **analog bandwidth** [6] by sqeezing as much information as possible (QPSK, QAM...)
- waveform optimized for time and frequency recovery (Cyclic Prefix of DAB+?). See DVB-S for inspiration.

[6] determined by satellite service supplier

Beyond PRN for time and frequency transfer: DAB+ broadcast

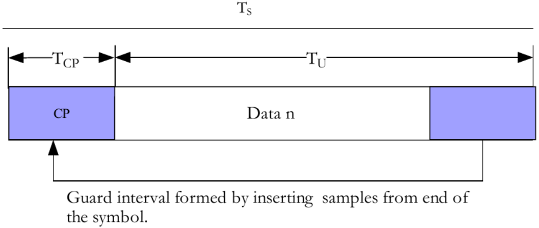


Figure 2.7: Guard interval and Cyclic Prefix

Where T_U is the OFDM symbol time without guard interval, T_{CP} is the duration of the copied information in the guard interval using cyclic prefix and T_S is the total OFDM symbol duration.

3.7.3 Cyclic prefix

This sub-block creates cyclic prefix as presented in section 2.7 in the second chapter. It takes a copy of 126 (mode II) samples corresponding to the guard interval period (see Table 2.1) from the end of every OFDM symbol and then copies them at the beginning of the symbol. This extends the length of the OFDM symbol to 638 samples length (see Table 2.1).

Petro Pesha Ernest, *DAB implementation in SDR*, Master thesis University of Stellenbosch (South Africa), Dec. 2005

Carrier Frequency Offset Estimation and Correction The Phase Reference symbol, the first, true OFDM symbol in each DAB frame, having strictly defined structure, is for us priceless. Apart from being the starting phase reference for all used sub-carriers, switched in phase, it can be also used for additional time OFDM symbol synchronization as well as for fractional carrier frequency offset estimation (CFO), i.e. value of carrier de-tuning from nominal value measured in a fraction of the FFT lag (e.g. $0.12345 \cdot f_0$, $f_0 = \frac{f_s}{N_{FFT}}$).

After simplifying assumptions that: (1) the channel impulse response is constant and short in comparison to the CP length, (2) noise is absent, and (3) all operations performed in the receiver are ideal inverses of operations done in the transmitter, then the last samples of the received signal $x(t)$ and the cyclic prefix should be almost the same. Because the same were transmitted in signal $s(t)$, this is the CP concept. However, when the frequency f_{down} of signal down-conversion to the base-band, done in the receiver, is different than the frequency f_{up} of signal up-conversion from the base-band to intermediate or target frequency, done in the transmitter, we have a problem. In such case, the received signal is equal to:

$$x(n) = s(n) e^{j2\pi \frac{f_{up}-f_{down}}{f_s} n} = s(n) e^{j2\pi \frac{\Delta f}{f_s} n}. \qquad (23.2)$$

Let us assume that we multiply last samples of an OFDM symbol by corresponding samples of the CP, but with complex conjugation. In ideal case, the same samples are multiplied, only delayed by the OFDM symbol length N equal to the FFT length. If f_s denotes sampling frequency, $f_0 = f_s/N$ is a fundamental FFT frequency and a frequency shift error is equal to $\Delta f = \Delta k \cdot f_0$, where $\Delta k \in \mathbb{R}$, we should obtain

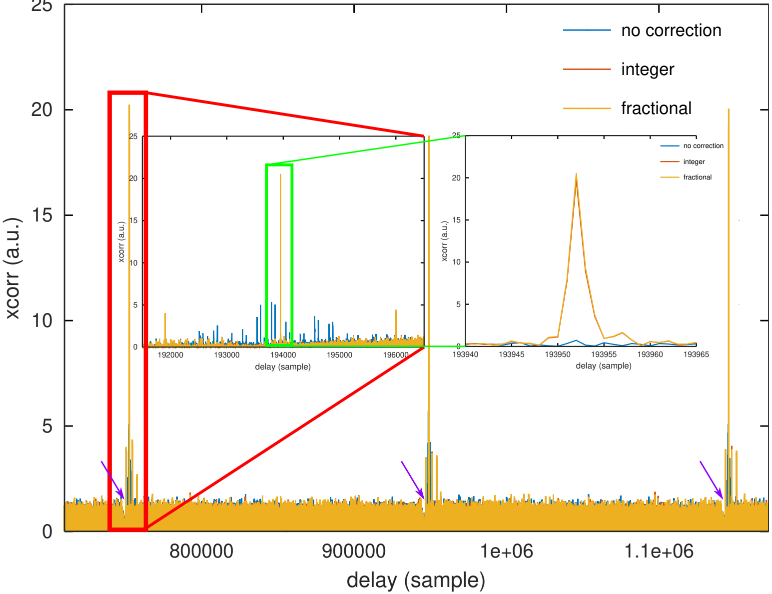
$$z(n) = x(n) \cdot x^*(n-N) = |s(n)|^2 \cdot e^{j2\pi \frac{\Delta f}{f_s} n} e^{-j2\pi \frac{\Delta f}{f_s}(n-N)} = |s(n)|^2 \cdot e^{j2\pi \frac{\Delta f}{f_s} N} =$$
$$= |s(n)|^2 e^{j2\pi \frac{\Delta f}{f_0}} = |s(n)|^2 \cdot e^{j2\pi \frac{\Delta k f_0}{f_0}} = |s(n)|^2 \cdot e^{j2\pi(\Delta k)}. \qquad (23.3)$$

Δk, being a real-value number, denotes the overall carrier frequency offset errors expressed in multiplicity of fundamental DFT frequency f_0. For example, $\Delta k = 3.456$ tells us that carrier frequency is shifted up $3.456 \cdot f_0$ Hz, i.e. 3 DFT bins (integer shift) plus 0.456 of the DFT bin (fractional shift). In general, when Δk has integer part k_Δ and fractional part ε_Δ:

$$\Delta k = k_\Delta + \varepsilon_\Delta, \qquad (23.4)$$

T.P. Zieliński, *Starting Digital Signal Processing in Telecommunication Engineering – A Laboratory-based Course*, Springer (2021)

DAB+ OFDM decoding



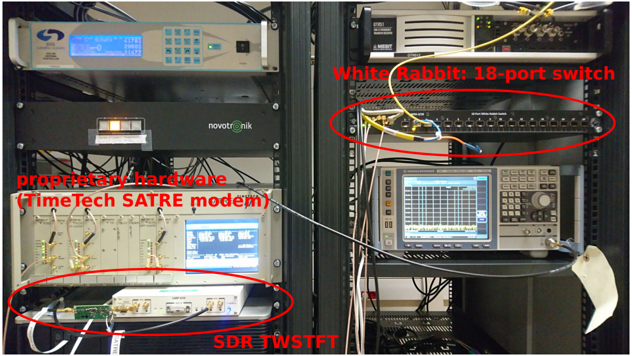
OFDM [a] requires time and frequency synchronization to be decoded:

- Similar to PRN, Phase Reference (PR) symbol sequence starts a new frame, *but*
- $\int PR_{TX}(t)PR_{RX}(t+\tau)\cdot \exp(j\delta\omega t)dt = 0$ if $\delta\omega \neq 0$
- need to find $\delta\omega$ frequency difference between TX and RX LO
- Cyclic Prefix provides same symbol (phase sequence) separated by N samples so that $\int x(n)\cdot x(n+N) = |x(n)|^2 \times \exp(j\underbrace{\delta\omega/fs}_{wanted}\times N)$
- correlation peak position = time delay

[a] spreading bits over multiple subcarriers affected differently by multipath interferences

Time and frequency transfer over optical fibers

- Opensource/openhardware White Rabbit PTPv2 protocol developed by CERN, licensed to various hardware manufacturers (including those implementing undisclosed modifications, e.g. Seven Solutions in Spain)
- Output is 1-PPS and 10 MHz: how to use?
- 18-port switch available for distribution to the laboratory experiments (coherent synthesis or acquisition, sub-100 ps time variations over the digital network, running through the Observatory to university)

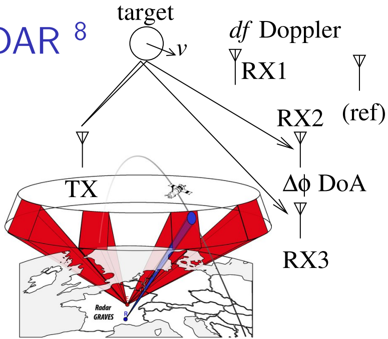


→ How to **generate** time and frequency coherent radiofrequency signals from the incoming 1-PPS and 10 MHz? → How to **collect** time-synchronized samples over the coherent network?

Dedicated Analog Devices chips: network clock distribution AD9548

- White Rabbit is implemented on dedicated PCI boards fitted with several local oscillators...
- ... accepting daughter boards (digital input/output, ADC).
- White Rabbit Distributed Trigger (WRTD) for launching acquisition on all ADCs on the WR network.

Practical demonstration: distributed passive RADAR [8]



Principle of multistatic passive RADAR

- RADAR: RAdiofreqeuency Detection And Ranging provides target range (time of flight resolution inverse of bandwidth) and velocity (Doppler shift)
- multistatic: separate emitter and receiver(s) for spatial diversity ...
- ... requires time and frequency synchronization of all components.

Application to GRAVES [7]:

- Continuous Wave (CW) illumination of space over the French metropolitan territory for satellite detection
- bistatic RADAR with 143.05 MHz emitter located north-easth of France and receiver located south of France
- $>$400 kW emitted power makes planes easily detected as frequency shift from carrier

$$\Delta f = 2f_0\frac{v}{c} \simeq v \text{ @ } 143.05 \text{ MHz}$$

- subsonic planes at $v < 280$ m/s exhibit frequency shift up to 300 Hz at most

Multistatic RADAR: complement velocity (CW source) with direction of arrival (phase analysis)

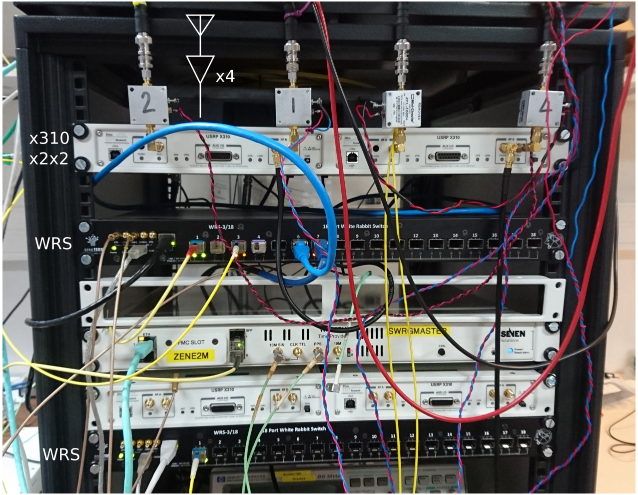
$\Rightarrow$ requirement on phase synchronization: 3° phase @ 143.05 MHz is $\frac{3}{360\times 143.05}$ μs=60 ps within the capability of WR

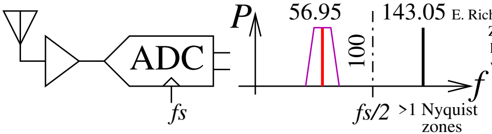
[7] A. Jouade & A. Barka, *Massively Parallel Implementation of FETI-2LM Methods for the Simulation of the Sparse Receiving Array Evolution of the GRAVES Radar System for Space Surveillance and Tracking*, IEEE Access 99 (2019)

[8] Seminar recorded for INSA Lyon: `https://youtu.be/66M6pT6G4tM`

Aliasing for distributed RADAR

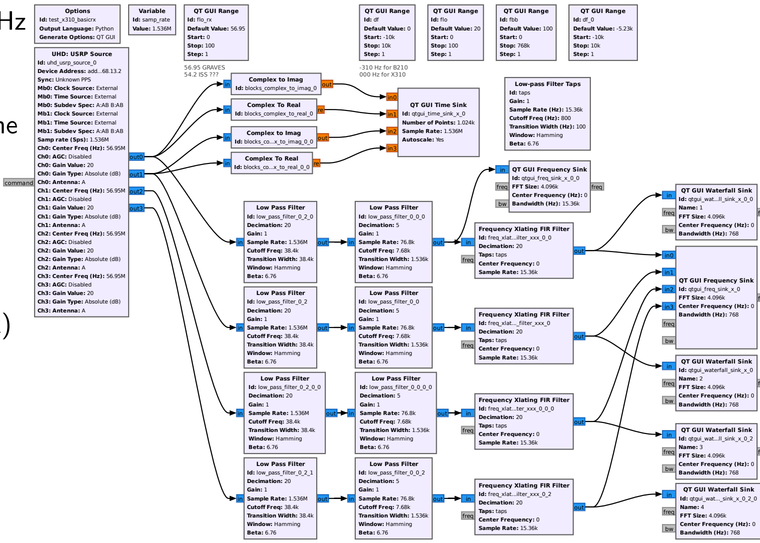
- Ettus Research X310: 200 MS/s ADC
- BasicRX daughter board: 1–250 MHz passive balun feeding the ADC
- Sub-sampling the 143.05 MHz at 200 MS/s: uses second Nyquist zone and aliased signal appears at $|100 - 43.05| = 56.95$ MHz
- Multiple USRP sources: all X310 must be controlled by the same computer running GNU Radio (connected to the same subnetwork)
- From `samp_rate`=1.536 MS/s to 768 Hz: cascaded FIR filters decimating by 20, 5 and 20 respectively ending with an 800 Hz cutoff frequency
- *assumes* no signal in aliases of 143.05 MHz $\pm$ 800 Hz





Aliasing for distributed RADAR

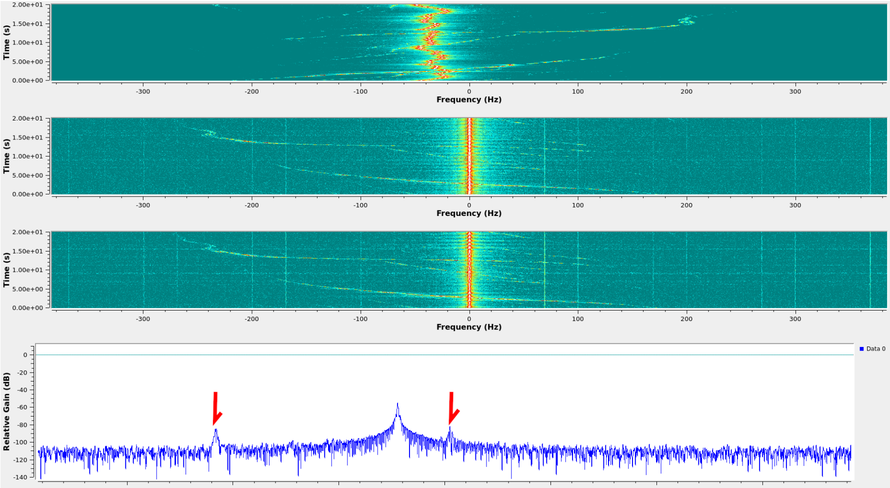
- Ettus Research X310: 200 MS/s ADC
- BasicRX daughter board: 1–250 MHz passive balun feeding the ADC
- Sub-sampling the 143.05 MHz at 200 MS/s: uses second Nyquist zone and aliased signal appears at $|100 - 43.05| = 56.95$ MHz
- Multiple USRP sources: all X310 must be controlled by the same computer running GNU Radio (connected to the same subnetwork)
- From `samp_rate`=1.536 MS/s to 768 Hz: cascaded FIR filters decimating by 20, 5 and 20 respectively ending with an 800 Hz cutoff frequency
- *assumes* no signal in aliases of 143.05 MHz $\pm$ 800 Hz



Impact of local oscillator stability

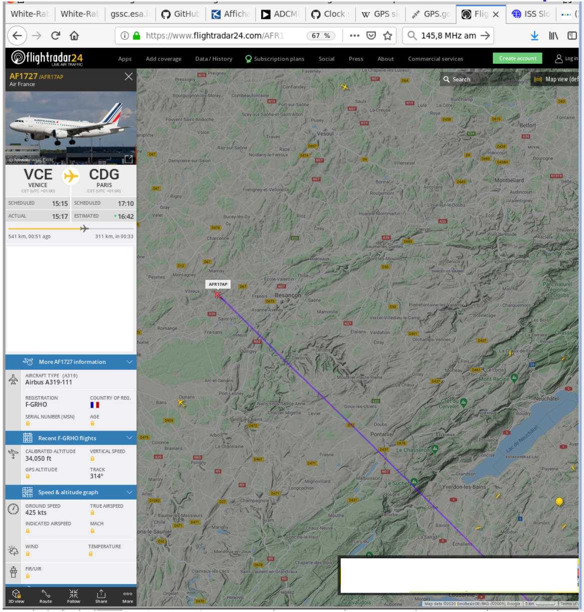
Need for stable local oscillator: 300 Hz @ 143 MHz = 2 ppm long term stability

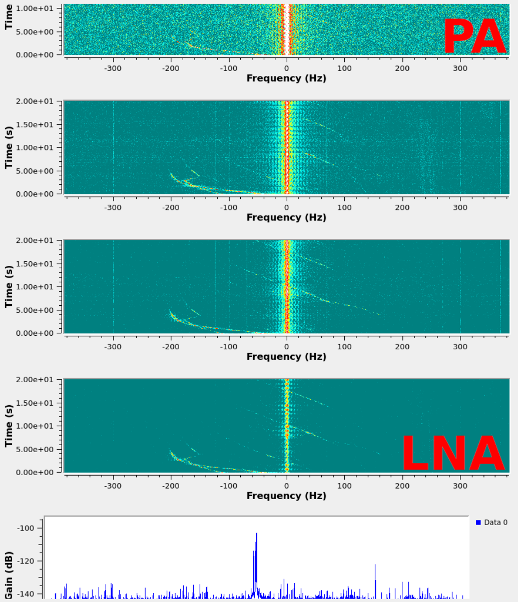
Top: X310 internal oscillator ; middle & bottom: hydrogen maser external reference



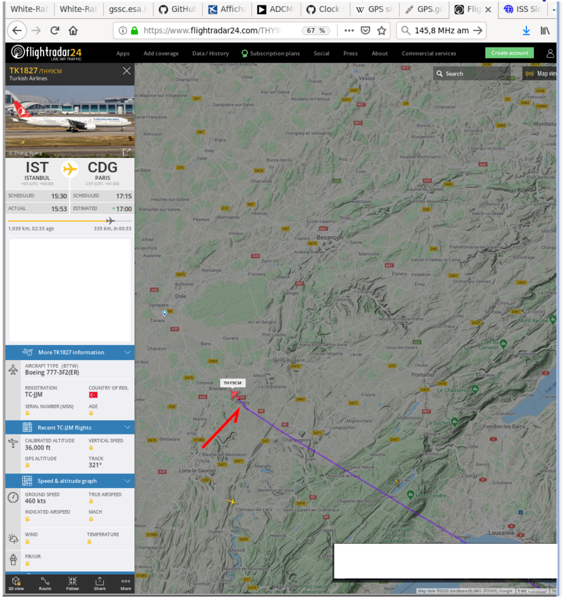
- each moving target is identified with a different Doppler shift after FFT of the recorded signal
- low streaming datarate: 600 S/s sufficient, 1 kS/s safe

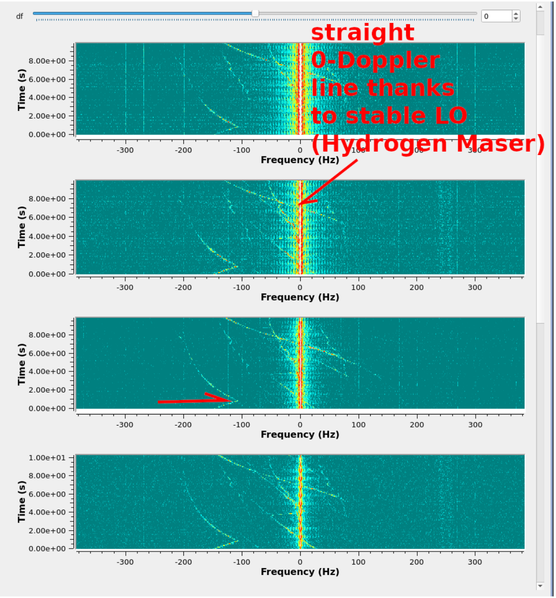
Results: 2 km ENSMM-UFR ST optical fiber between WRS





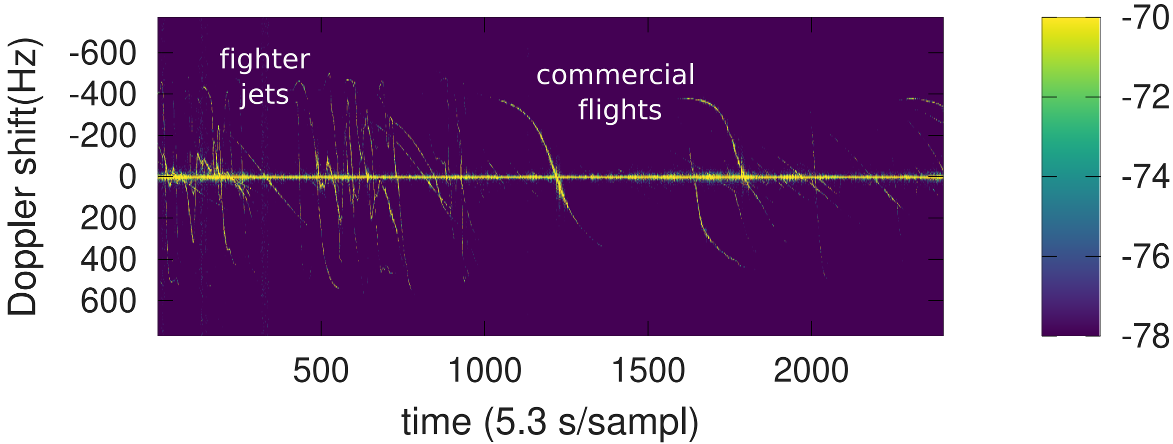
Results: 2 km ENSMM-UFR ST optical fiber between WRS



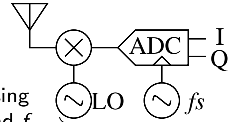


Results: 2 km ENSMM-UFR ST optical fiber between WRS

Fighter jet signature clearly visible: sharp direction change + Doppler ≥300 Hz



Initial architecture



Software Defined Radio (SDR) based multistatic RADAR system

- classical architecture: frequency transposition from radiofrequency band to baseband using a phase locked loop (PLL) generated local oscillator (LO) requires sycnhronizing LO and f_s $\rightarrow$
- ... and analog to digital Conversion (ADC) of the baseband signal, here at sampling frequency $f_s \gtrsim 600$ Hz
- SPEC[9] board allows for 100 MS/s acquisition in a White Rabbit synchronized board when fitted with 4-channel ADC (direction of arrival at each receiver station)
- replace dedicated PC with embedded board: Raspberry Pi4 OEM version (Compute Module 4) provides 1-line PCIe interface compatible with SPEC

Problem: phase synchronization of spatially distributed PLL LOs even if ADCs are time and frequency synchronized

Solution: aliasing

ADC is time (1-PPS) and frequency (10 MHz) synchronized: functional demonstration of consistent Doppler shift and phase difference from colocated receivers.
Buildroot [10] with latest Linux kernel is used for Raspberry Pi 4 binary image cross-compilation:

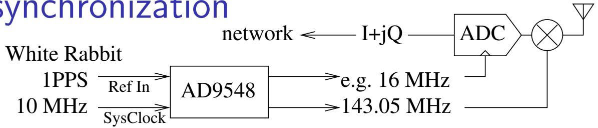
can White Rabbit operate in such an environment

---

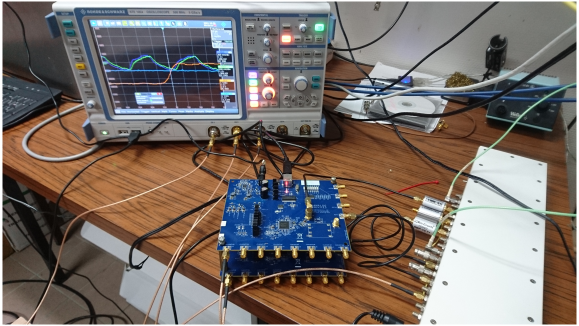
[9] Simple PCI Express Carrier

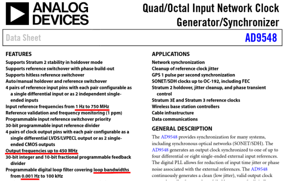
[10] G. Goavec-Merou, J.-M Friedt, *Porting GNU Radio to Buildroot: application to an embedded digitalnetwork analyzer*, FOSDEM 2021 at `http://jmfriedt.free.fr/fosdem2021_buildroot.pdf`

Frequency transposition: local RF synchronization



- LO and sampling frequency synchronization: two synchronizations needed for a fully coherent system
- challenge of generating the time-synchronized radiofrequency local oscillator
- Analog Device [a] AD9548 "Quad/Octal Input Network Clock Generator/Synchronizer"...
- ... uses 1-PPS input as reference and 10-MHz input as system clock...
- ... to output $f_0 \in [0 : 450]$ MHz and its integer fractions.




ANALOG DEVICES

Quad/Octal Input Network Clock Generator/Synchronizer

Data Sheet AD9548

FEATURES

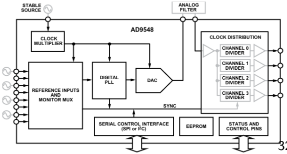
Supports Stratum 2 stability in holdover mode
Supports reference switchover with phase build-out
Supports hitless reference switchover
Auto/manual holdover and reference switchover
4 pairs of reference input pins with each pair configurable as a single differential input or as 2 independent single-ended inputs
Input reference frequencies from 1 Hz to 750 MHz
Reference validation and frequency monitoring (1 ppm)
Programmable input reference switchover priority
30-bit programmable input reference divider
4 pairs of clock output pins with each pair configurable as a single differential LVDS/LVPECL output or as 2 single-ended CMOS outputs
Output frequencies up to 450 MHz
30-bit integer and 10-bit fractional programmable feedback divider
Programmable digital loop filter covering loop bandwidths from 0.001 Hz to 100 kHz

APPLICATIONS

Network synchronization
Cleanup of reference clock jitter
GPS 1 pulse per second synchronization
SONET/SDH clocks up to OC-192, including FEC
Stratum 2 holdover, jitter cleanup, and phase transient control
Stratum 3E and Stratum 3 reference clocks
Wireless base station controllers
Cable infrastructure
Data communications

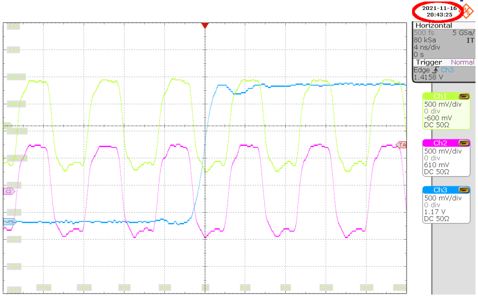
GENERAL DESCRIPTION

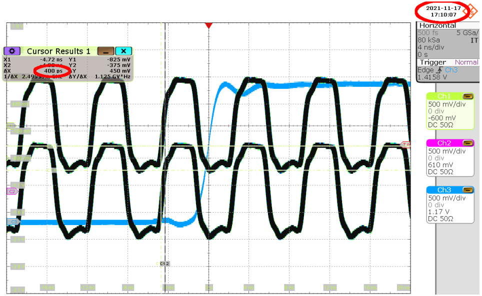
The AD9548 provides synchronization for many systems, including synchronous optical networks (SONET/SDH). The AD9548 generates an output clock synchronized to one of up to four differential or eight single-ended external input references. The digital PLL allows for reduction of input time jitter or phase noise associated with the external references. The AD9548 continuously generates a clean (low jitter), valid output clock

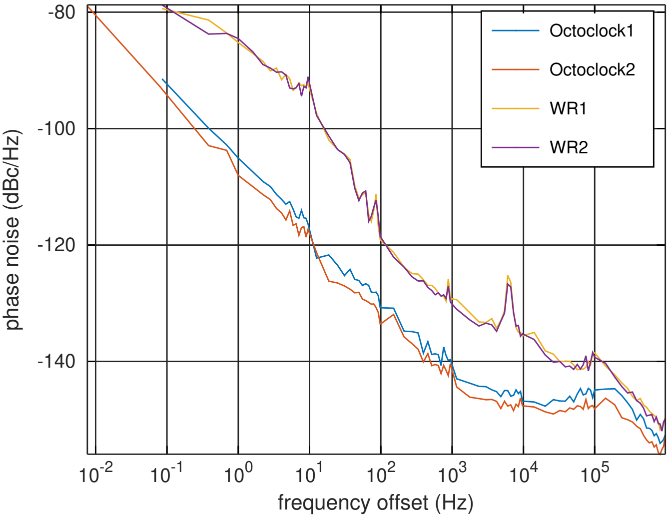


[a] K. Gentile, *The AD9548 as a GPS Disciplined Stratum 2 Clock*, Application Note AN-1002, Analog Devices (2009)

Distributed coherent radiofrequency signal synthesis





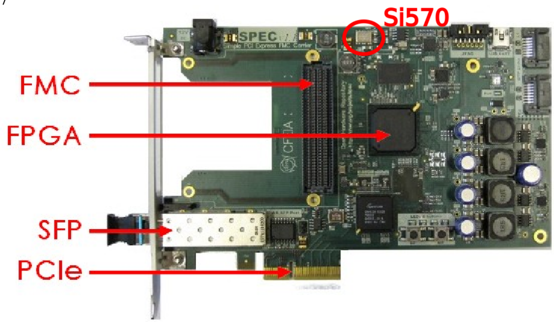


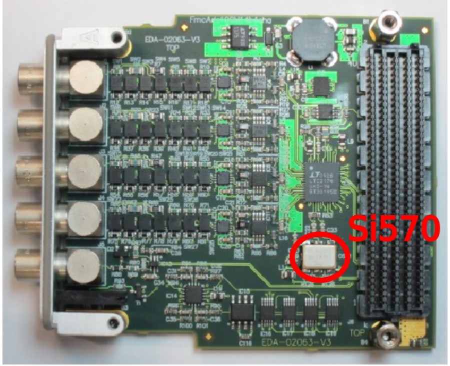
WRTD on latest Linux kernels [11]?

WRTD (White Rabbit Trigger Distribution) provides exactly the targeted functionalities ... on paper

- diverse source of repositories for the various parts of the project
- runs on CentOS7 with 3.10 kernel (released 2014, last update 2020)

First development: porting WRTD to a current (summer 2021: 5.12) Debian/stable distribution





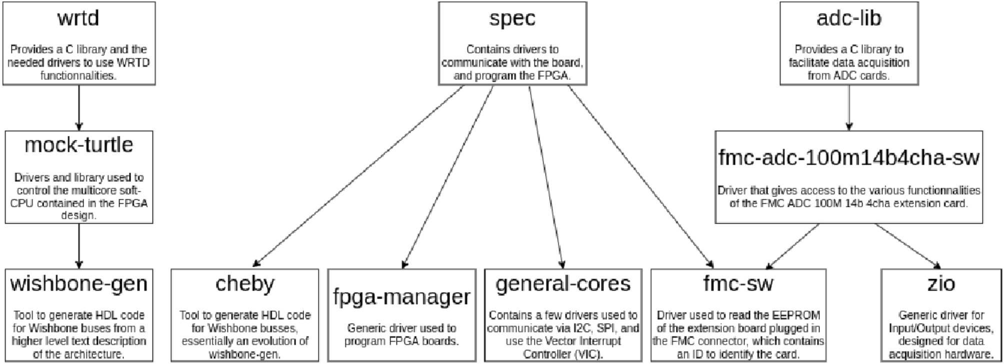
Kernel development = multiple reboots when kernel panics ⇒ `qemu` for fast reboot when memory is corrupted

[11] M. Marion, G. Goavec-Merou, J.-M. Friedt, *Porting SPEC drivers to current Linux kernels and embedded ARM boards for WRTD applications*, 11th White Rabbit Workshop, (6-8 October 2021) at `https://ohwr.org/project/white-rabbit/wikis/Oct2021Meeting`

WRTD on latest Linux kernels [11]?

WRTD (White Rabbit Trigger Distribution) provides exactly the targeted functionalities ... on paper

- diverse source of repositories for the various parts of the project
- runs on CentOS7 with 3.10 kernel (released 2014, last update 2020)

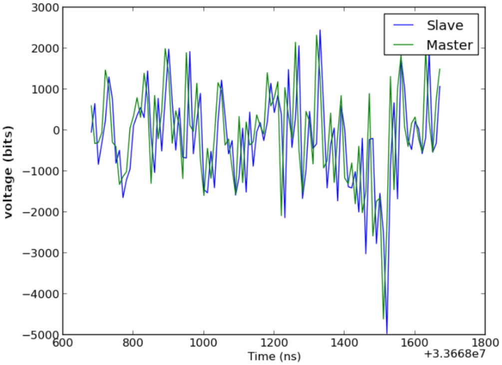


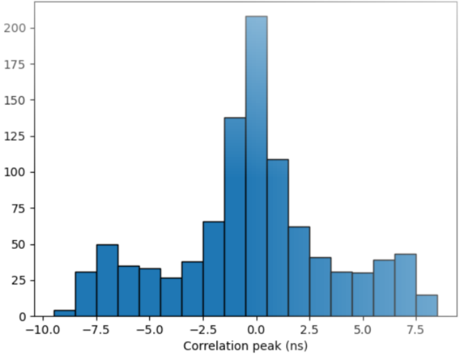
Kernel development = multiple reboots when kernel panics ⇒ qemu for fast reboot when memory is corrupted

[11] M. Marion, G. Goavec-Merou, J.-M. Friedt, *Porting SPEC drivers to current Linux kernels and embedded ARM boards for WRTD applications*, 11th White Rabbit Workshop, (6-8 October 2021) at `https://ohwr.org/project/white-rabbit/wikis/Oct2021Meeting`

Functional WRTD on x86 & ARM-based computers

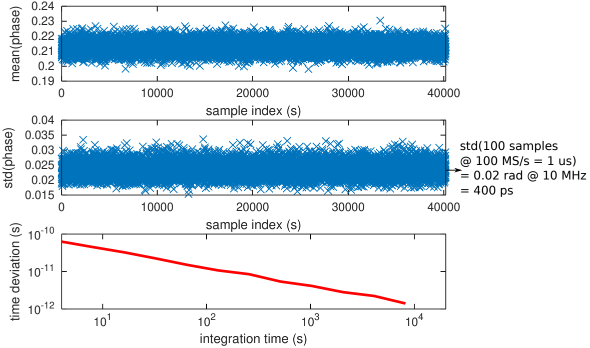
Functional acquisition demonstration but ... random phase between multiple channels collected by distributed boards (cross-correlation maximum of broadband noise)

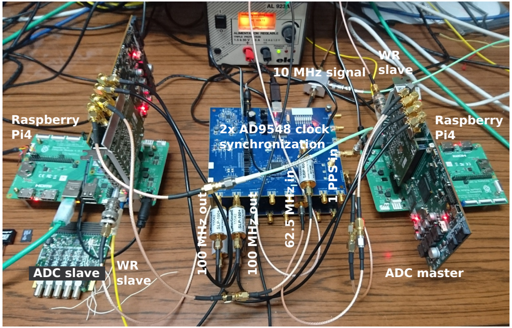




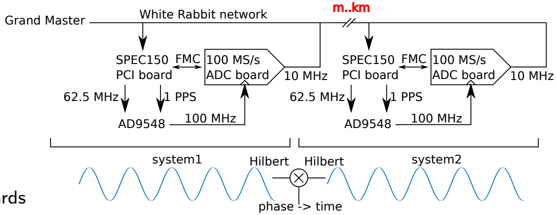
The ADC (Si570) clock **is not synchronized** on the White Rabbit clock: although trigger time is common to all boards, the sampling rate and phase within 10-ns time slot is random

White Rabbit synchronized distributed ADC conversion of RF signals





- White Rabbit (WR) time and frequency dissemination
- analog to digital conversion daughter boards clocked by free running oscillators ! replaced with ...
- ... clock distribution system for converting 1-PPS and 10-MHz (White Rabbit internal 62.5 MHz) to 100 MHz ADC clock.
- **Demonstration:** acquisition of WR 10 MHz by multiple boards
- phase to time delay conversion: **1 ps** at 10^4 s integration time
- working with Raspberry Pi (Compute Module 4) embedded boards



Conclusion & perspectives

Conclusions

- LoRa: monitoring the physical layer of any broadband communication system allows for timestamping and possibly synchronizing
- Synchronization is a core issue for any modern digital communication scheme (e.g. OFDM in DAB+ or DVB) ...
- ... to be applied for TWSTFT aimed at replacing SATRE with an opensource/openhardware SDR solution.
- WRTD functional on kernel $\geq$ 5.12 for x86, ARM & qemu
- demonstration of remote trigger of analog acquisitions ...
- ... and disciplined oscillator for clocking ADC

Resources:

- detailed documentation of this work at `https://github.com/oscimp/WRTD-FMC-ADC`
- `https://github.com/oscimp/oscimp_br2_external` with WRTD for Buildroot in `packages`

Work in progress:

- discipline ADC Si570 on White Rabbit clock (generalize soft-PLL in the FPGA?)
- datastream over White Rabbit dark fibers (functional with WRS ... to be verified with SPEC)
- identify missing samples (timestamp?)