# ALS162 Time Signal
# SDR Receiver for GNU Radio

Henning Maier

henningM1r@gmail.com

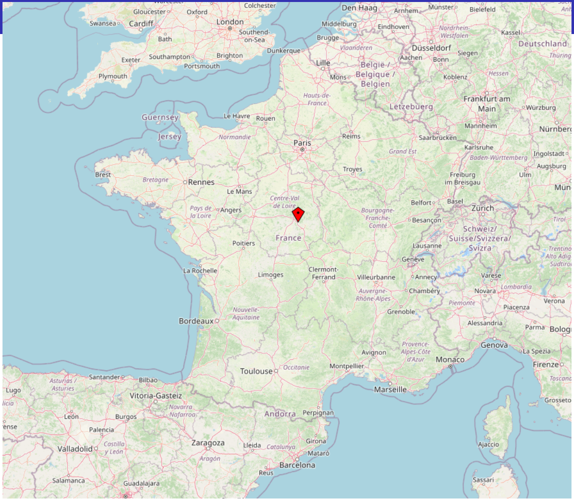
September 8, 2023

13$^{th}$ Annual GNU Radio Conference 2023, Tempe (AZ)

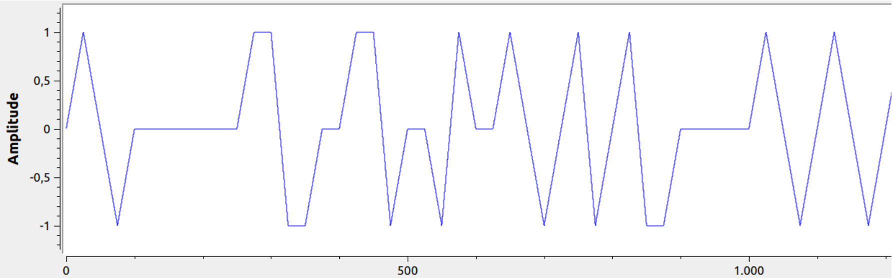
# Table of Contents

# ALS162 Transmitter

- French legal time signal from Allouis since 1980
- Caesium atomic clock
- LF carrier at 162 kHz
- Bandwidth $\sim \pm 250$Hz
- Tx-power $\sim$800kW
- Range $\sim$3.500km $\widehat{=}$ 2.175mi
- Phase modulated signal
- Tx is under maintenance each Tuesday morning



# ALS162 Transmitter Simulation




- Current time information (day, month, year, weekday, hour, minute, etc.) is encoded by 60 symbols per minute
- First second: waveform for symbol 0 (here: roughly at position 47 in minute)
- Next second: waveform for symbol 1 ...

# RF Channel Simulation

**Impairments**

- X(t) – ALS162 signal modulated to 162kHz
- A(t) – Attenuation: very slow fading
- I(t) – Independent impulsive interference
- N(t) – i.i.d. Gaussian white noise

$$Y(t) = A(t) \cdot X(t) + I(t) + N(t).$$

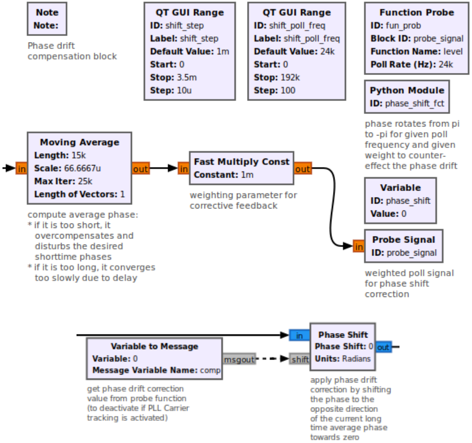
- Channel flowgraph & details skipped here

# ALS162 Receiver

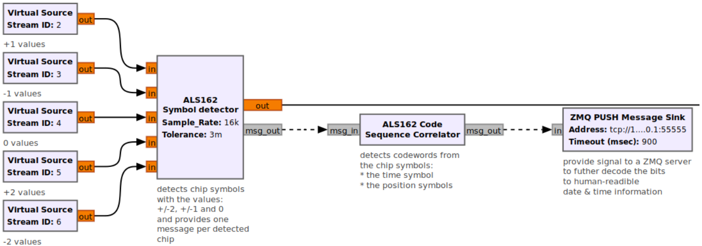
- Note: It is not claimed that this Rx is optimal!
- Main steps of the ALS162 Rx:
  1. ✓ **Demodulation** of RF signal to baseband signal
  2. ✓ **Downsampling** to reduce computational effort
  3. 🔍 **Phase shift correction** to stabilize drifting phase
  4. ✓ **Derivative filtering & moving average FIR filtering** for simpler decoding
  5. ✓ **Symbol level detection/quantization** $\approx$ quantize input to levels $0, \pm 1, \pm 2$
  6. 🔍 **Synchronization** of symbols to regular 25ms time-slices
  7. 🔍 **Code correlation** to detect time symbols & bit positions
  8. 🔍 **Symbol decoding** to get desired human-readible time information

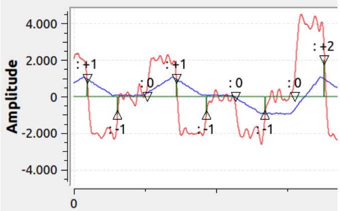
# ALS162 Receiver – Phase Shift Correction

- To compensate drifting phase at $\pm\pi$
- Compute average phase over a longer sliding window and substract weighted avg. phase via corrective feedback
- Alternatively: use *PLL Carrier Tracking* block with a very small bandwidth instead
- Note: Optimal parameters depend on your CPU's performance



# ALS162 Receiver – Synchronization & Correlation




- Count samples and detect 25ms time-slices within tolerance for values $0, \pm 1, \pm 2$
- Aggregate symbols to a sequence: $\cdots, +1, -1, 0, +1, -1, 0, -1, +2, \cdots$
- Correlate sequence to time symbols $0, 1$ and position symbols $0, 1, \cdots, 59$
- Provide detected time & position symbols as messages to decoder with ZeroMQ

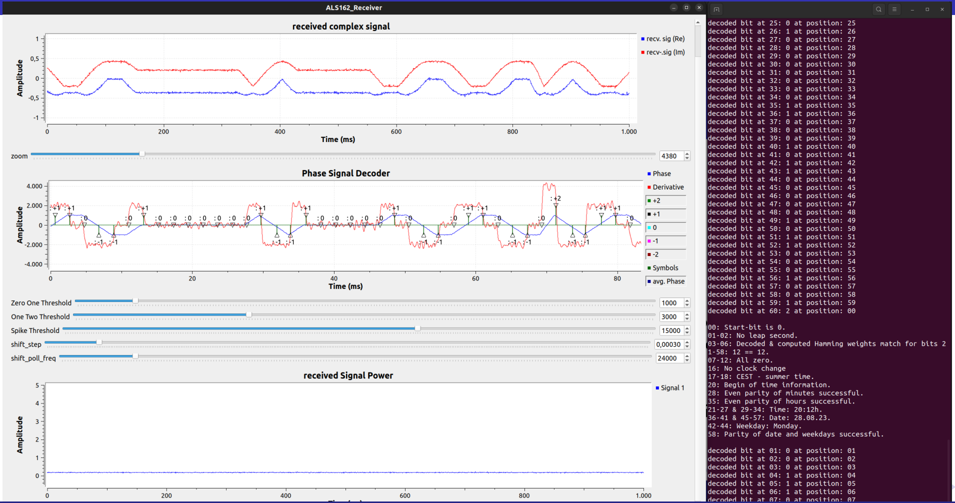
# ALS162 Receiver – Decoder

- Decode received bits each minute according to official ALS162 time code specification
- Use known bit positions and parity bits to detect and correct errors
- Display a human-readible time report each new minute

```
decoded bit at 19: 0 at position: 19
decoded bit at 20: 0 at position: 20
decoded bit at 21: 1 at position: 22
1 bit(s) lost before position: 22
decoded bit at 23: 1 at position: 23
decoded bit at 24: 1 at position: 24
```

```
00: Start-bit is 0.
01-02: No leap second.
03-06: ERROR: Bits 21-58 contain erroneous bits.
07-12: All zero.
16: No clock change
17-18: CEST - summer time.
Corrected single error at 27.
Corrected single error at 34.
21-27 & 29-34: Time: 12:37h.
36-41 & 45-57: Date: 27.08.23.
42-44: Weekday: Sunday.
58: Parity of date and weekdays successful.
# Bit errors: 3 => at positions: [21, 28, 34].
```

# ALS162 Receiver – GUI



# Insights

- This pretty basic time signal was a bit more challenging than initially expected
- Codewords for positions had to be "brute-forced"
- Indoor reception with antenna fixed at window is feasible (if within range)
- Proper placement and alignment of antenna is recommended
- Antenna should rather not swing in the wind
- Reception quality sometimes suffers from interference
- Using the derivative filter approach is certainly not optimal (but sufficient)
- Choice of phase correction parameters depend on CPU-load
- Testing such a Rx with Tx & channel simulation is recommended
- Using CI for unittesting & coverage with Docker container is recommended

# Thank You!

- For further details do not hesitate to take a look into the accompanying paper
- … or feel free to contact me.
- Repository: `https://github.com/henningM1r/gr_ALS162_Receiver`

# Backup: SDR Setup

- SDR with LF reception capability at 162kHz, i.e. within approx. 1-1000 kHz
- (Passive) loop antenna already works fine
- Ordinary PC/laptop with Linux OS or Windows OS works fine
- GNU Radio installation (3.10.1.1) was used here
- Other SDR equipment and antennas were not tested but might work

# Backup: Phase Drift

- Depicted phase signal suffers from phase drift
- To compensate, adjust parameters *shift_step* and *shift_poll_freq*
- Signal should rather oscillate around zero