A Seminar for GNSS Software Defined Receivers

Pocket SDR ver.0.14の開発とその応用

Development of Pocket SDR ver.0.14 and its Applications

Tomoji TAKASU

ttaka@gpspp.sakura.ne.jp, https://gpspp.sakura.ne.jp

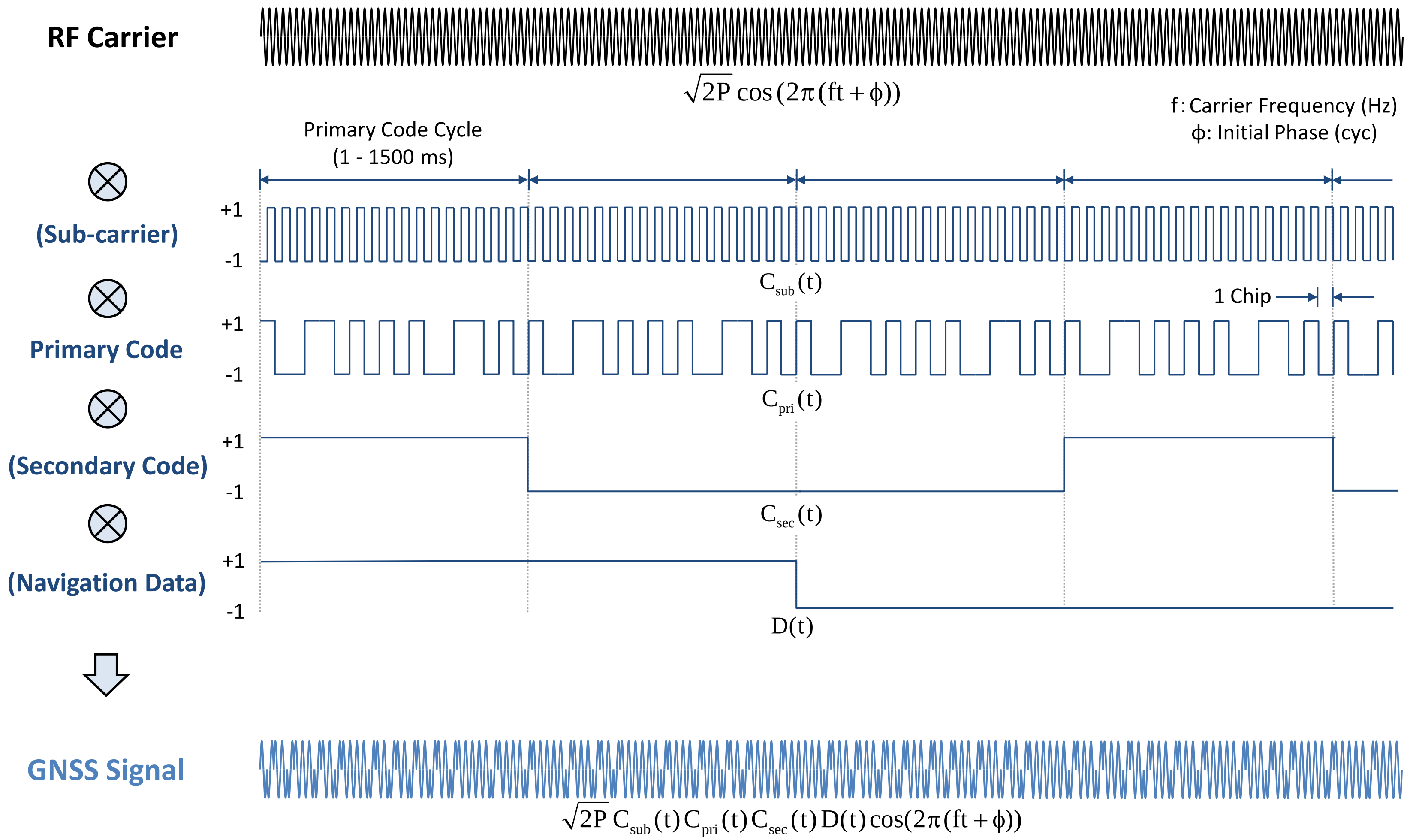
2025-3-12 @TUMSAT, Tokyo, Japan

Time Table

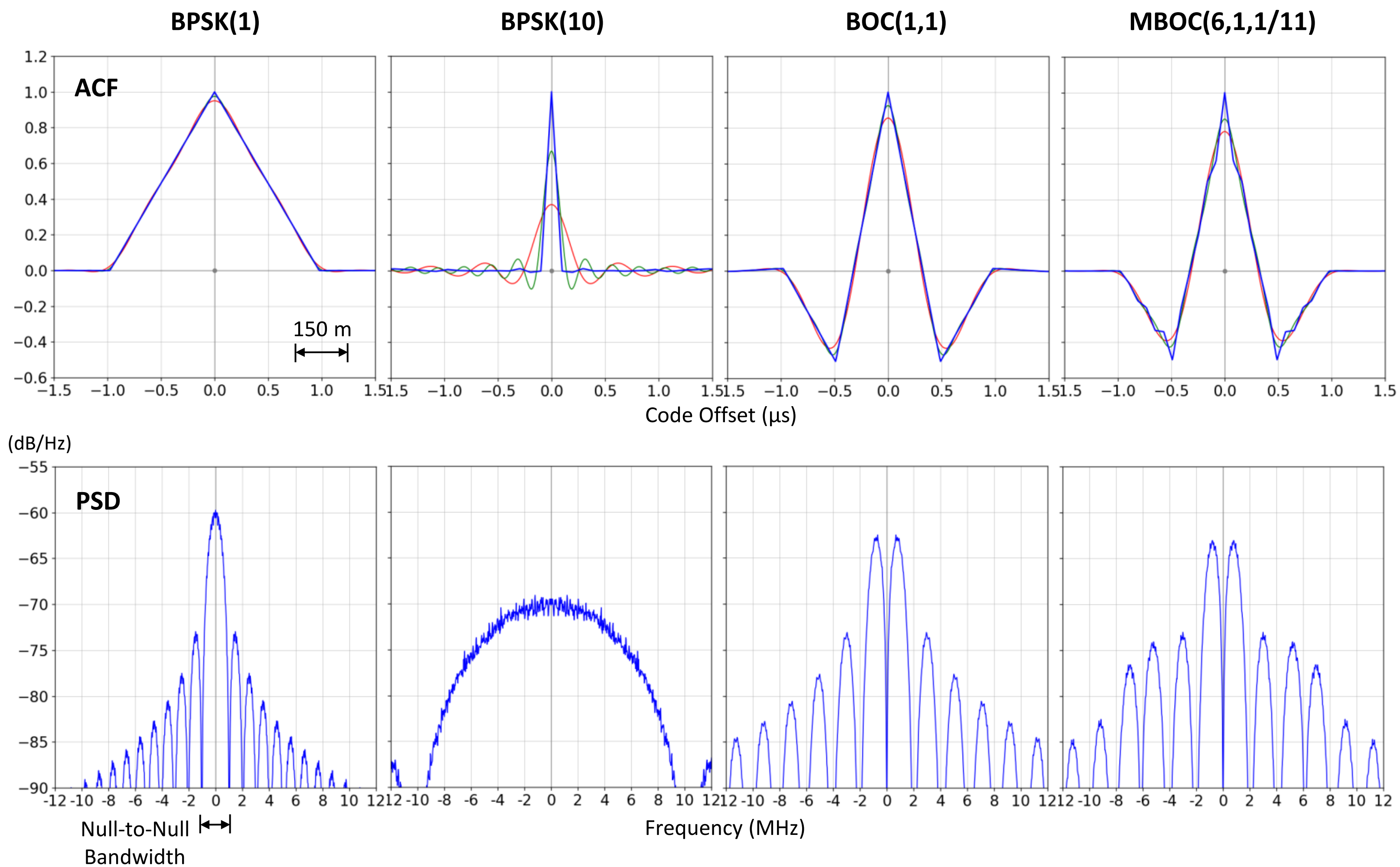
- 13:00 - 13:20 GNSS SDR
- 13:20 - 13:50 Pocket SDR FE
- 13:50 - 15:00 Pocket SDR S/W
- 15:10 - 16:30 Applications
- 16:30 - 17:00 Q&A, Free Discussion

GNSS SDR

GNSS Signal Structure

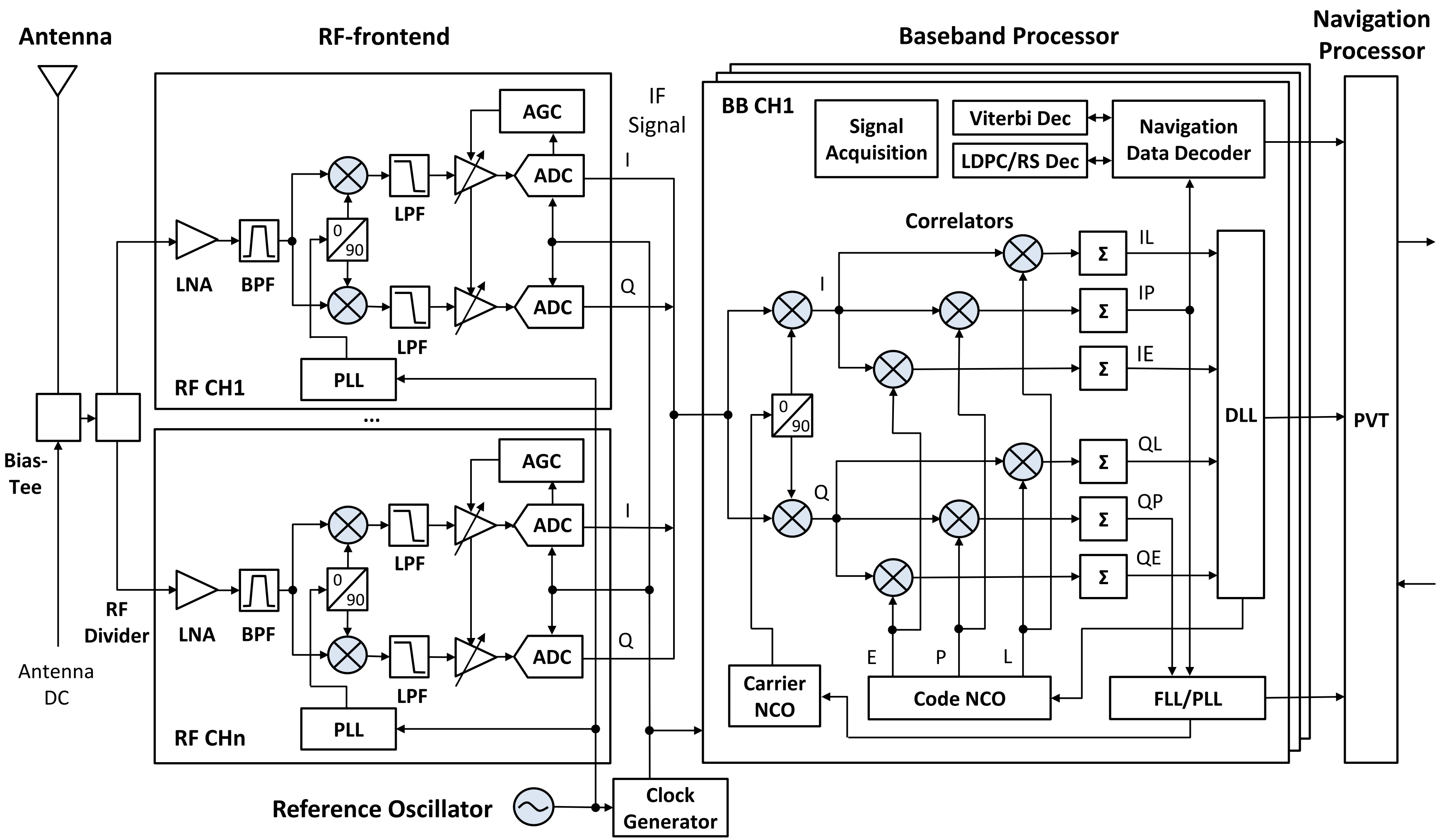


Modulation



BPSK: Binary Phase-Shift Keying, BOC: Binary Offset Carrier, MBOC: Multiplexed BOC,
ACF: Auto Correlation Function, PSD: Power Spectral Density, —— ACF with 4 MHz LPF, —— ACF with 8 MHz LPF

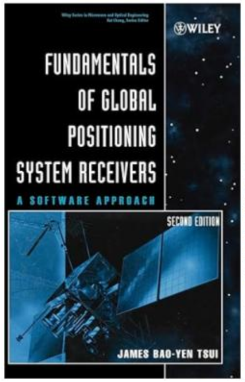
GNSS Receiver

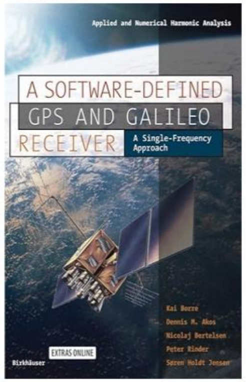


LNA: low noise amplifier, BPF: bandpass filter, LPF: lowpass filter, ADC: analog digital converter, AGC: automatic gain control, IF: intermediate frequency, PLL: phase lock loop, DLL: delay lock loop, FLL: frequency lock loop, NCO: numerical controlled oscillator, LDPC: low density parity check code, RS: Reed Solomon code, PVT: position, velocity and time

GNSS SDR

- **SDR (Software Defined Radio)**
  - RF Signal Processing by FPGA, DSP, GPP (CPU) or GPU instead of H/W circuits
  - RF-frontend: USRP (Universal Software Radio Peripheral) [1], RTL-SDR [2], ...
  - S/W: GNU Radio [3], LabVIEW [4], MATLAB/Simlink [5], ...
  - Applications: radio/TV, mobile comm. (WiFi, BT, LTE, ...), radar, measuring instrument, GNSS, ...
  - PROS: fast system development, high flexibility by S/W modification, easy upgrade by CPU replacement, good tool for research, education and training, ...
  - CONS: limited sampling rate, narrow BW, high power consumption, large size, heavy weight, ...
- **GNSS SDR (Software Defined Receiver)** [6]
  - GNSS Receiver implemented by SDR
  - RF-frontend H/W (+ FPGA) + S/W codes
  - Post-processing or real-time
  - Programming Languages: C/C++, MATLAB, python, ...
- **GNSS SDR Textbooks**
  - J. B-Y. Tsui, Fundamentals of Global Positioning System Receivers: A Software Approach, 2000, John Wiley & Sons [7]
  - K. Borre et al., A Software-defined GPS and Galileo Receiver: A Single-Frequency Approach, 2007, Birkhäuser [8]



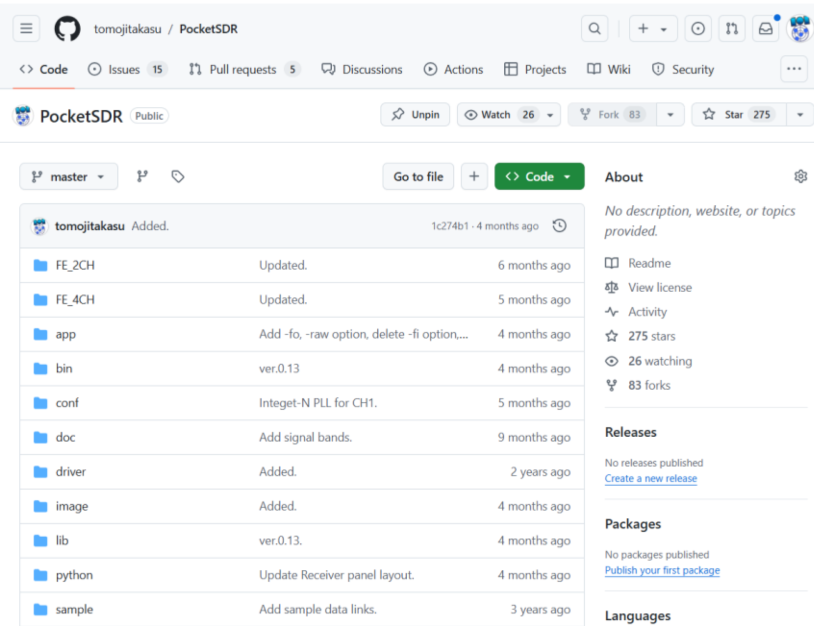


[1] https://www.ettus.com, [2] https://rtl-sdr.com/, [3] https://www.gnuradio.org, [4] https://www.ni.com/en/shop/labview.html, [5] https://www.mathworks.com, [6] https://en.wikipedia.org/wiki/GNSS_software-defined_receiver, [7] https://onlinelibrary.wiley.com/doi/book/10.1002/0471712582, [8] https://link.springer.com/book/10.1007/978-0-8176-4540-3

OSS RT GNSS SDR

Software		GNSS-SDR	GNSS-SDRLIB	Pocket SDR
URL		https://gnss-sdr.org https://github.com/gnss-sdr	https://github.com/taroz/GNSS-SDRLIB	https://github.com/tomojitakasu/PocketSDR
Current Release		v0.0.19 (Jan 2024)	v2.0 Beta (Dec 2014)	ver. 0.13 (July 2024)
License		GPL-3.0	GPL-2 or later	BSD 2-clause
GitHub Stars/Forks		☆ 1.7k / 616	☆ 476 / 177	☆ 325 / 100
Processing Mode		Post Processing, Real-time	Post Processing, Real-time	Post Processing, Real-time
PVT		Single, PPP	-	Single
Output Format		KML, GeoJSON, GPX, RINEX, NMEA, RTCM3, Custom	RTCM3, RINEX	NMEA, RTCM3, CSV Log, Raw binary
GNSS Signals	GPS	L1C/A, L2C, L5	L1C/A	L1C/A, L1C, L2C, L5
	GLONASS	L1C/A, L2C/A	L1C/A	L1C/A, L2C/A, L1OC, L2OC, L3OC
	Galileo	E1B/C, E5a, E5b, E6B	E1B	E1B/C, E5a, E5b, E6B/C
	QZSS	-	L1C/A, L1SAIF, LEX	L1C/A, L1C/B, L1C, L1S, L2C, L5, L5S, L6D/E
	BDS	B1I, B3I	B1I	B1I, B1C, B2I, B2a, B2b, B3I
	NavIC	-	-	L1-SPS, L5-SPS (,S-SPS)
	SBAS	-	L1C/A	L1C/A, L5
Supported RF-frontends		USRP, UmTRX, HackRF One, Blade RF, RTL-SDR	SiGe GN3S sampler v2/v3, Stereo, Blade RF, RTL-SDR	Pocket SDR FE 2/4/8CH
OS		Linux, macOS	Windows, Linux	Windows, Linux, macOS, Raspberry Pi OS
Language		C++, C	C, C++/CLI	Python 3, C++, C
External Libraries		GNU Radio, Boost, FFTW3, VOLK, Armadillo, LAPACK/BLAS, glog, gflags, matio, pugiXML, Protocol Buffers, OpenSSL, RTKLIB	FFTW3, LIBFEC, RTKLIB, libusb	FFTW3, LIBFEC, LDPC-codes, RTKLIB, CyAPI or libusb-1.0

GitHub



History

- 2021-10-20 0.1 1st draft version
- 2021-10-25 0.2 Add Rebuild F/W and Write F/W Image to PocketSDR
- 2021-12-01 0.3 Add and modify Python scripts
- 2021-12-25 0.4 Add and modify Python scripts
- 2022-01-05 0.5 Fix several problems.
- 2022-01-13 0.6 Add and modify Python scripts
- 2022-02-15 0.7 Improve performance, Add some Python scripts.
- 2022-07-08 0.8 Add C-version of pocket_acq.py and pocket_trk.py.
- 2024-01-03 0.9 Add C-version of pocket_snap.py. pocket_trk supports multi-signal and multi-threading
- 2024-01-12 0.10 Support NavIC L1-SPS-D, L1-SPS-P, GLONASS L1OCd, L1OCp and L2OCp.
- 2024-01-25 0.11 Support decoding of GLONASS L1OCd NAV data Support NB-LDCP error correction for BDS B1C, B2a and B2b
- 2024-05-28 0.12 Performance optimized. Support PVT generation, RTCM3 and NMEA outputs
- 2024-07-04 0.13 GUI-based GNSS-SDR receiver AP added. Support macOS.

README License

Pocket SDR - An Open-Source GNSS SDR, ver. 0.13

Overview

Pocket SDR is an open-source GNSS (Global Navigation Satellite System) receiver based on the SDR (software defined radio) technology. It consists of RF frontend devices named "Pocket SDR FE", some utilities for the devices, and GNSS-SDR APs (application programs) written in Python, C, and C++. It supports almost all signals for GPS, GLONASS, Galileo, QZSS, BeiDou, NavIC, and SBAS.

The Pocket SDR FE device consists of 2 or 4 RF frontend channels, which support GNSS L1 band (1525 - 1610 MHz), or L2/L5/L6 band (1160 - 1290 MHz). The bandwidth of each RF channel covers up to 36 MHz. The sampling rate of the ADC can be configured up to 32 Msps (FE 2CH) or 48 Msps (FE 4CH).

The Pocket SDR also contains some utility programs for the Pocket SDR FE devices, to setup the devices, capture and dump the digitized IF (inter-frequency) data. These utilities support Windows, Linux, Raspberry Pi OS, macOS and other environments. The Pocket SDR also provides GNSS-SDR APs to show the PSD (power spectrum density) of captured IF data, search GNSS signals, track these signals, decode navigation data and generate PVT (position, velocity and time) solutions. The supported GNSS signals are as follows. For details on these signals and signal IDs used in the APs, refer Pocket SDR Signal IDs.

- **GPS**: L1C/A, L1C-D, L1C-P, L2C-M, L5-I, L5-Q
- **GLONASS**: L1C/A (L1OF), L2C/A (L2OF), L1OCd, L1OCp, L2OCp, L3OCd, L3OCp
- **Galileo**: E1-B, E1-C, E5a-I, E5a-Q, E5b-I, E5b-Q, E6-B, E6-C
- **QZSS**: L1C/A, L1C/B, L1C-D, L1C-P, L1S, L2C-M, L5-I, L5-Q, L5S-I, L5S-Q, L6D, L6E
- **BeiDou**: B1I, B1C-D, B1C-P, B2a-D, B2a-P, B2I, B2b-I, B3I
- **NavIC**: L1-SPS-D, L1-SPS-P, L5-SPS
- **SBAS**: L1C/A, L5-I, L5-Q

These utilities and APs are written in Python, C, and C++ by very compact way. They are easily modified by users to add user's unique algorithms.

https://github.com/tomojitakasu/PocketSDR

Pocket SDR FE

RF-Frontends for GNSS SDR

RF-frontend	Vendor	# of CH		# of PLL		RF Bands	Sampling Type/Rate	RF LSI	FPGA	Host I/F	Price
		RX	TX	RX	TX						
USRP N210 [1]	Ettus Research, NI	2	2	1	1	DC ~ 6 GHz	IQ, 14 bits ≤ 100 Msps	Daughterboard (Optional)	Spartan 3A-DSP 3400	Giga-Ether	$3,599
SiGe GN3S Sampler [2]	GNSS Lab Univ Colorado	1	-	1	-	GPS L1	I or IQ, 2 bits ≤ 16.368 Msps	SiGe 4120	-	USB 2.0	$496
Stereo [3]	NSL	2	-	2	-	GNSS L1/L2/L5/L6	I/IQ, 2/3 bits 26 Msps	MAX2769B + MAX2112 + ...	Spartan-6	USB 2.0	$800
RTL-SDR [4]	RTLSDR.com	1	-	1	-	500 kHz ~ 1.75 GHz	IQ, 8 bits ≤ 2.4 Msps	Realtek RTL2832U	-	USB 2.0	$20 - $50
Blade RF [5]	NuBand	2	2	1	1	300 MHz ~ 3.8 GHz	IQ, 12 bits ≤ 40 Msps	Lime Micro. LMS6002D	Cyclone IV E	USB 3.0	$520
Lime SDR [6]	Lime Microsystems	2	2	1	1	100 kHz ~ 3.8 GHz	IQ, 12 bits ≤ 61.44 Msps	Lime Micro. LMS7002M	Cyclone IV EP4CE40F23	USB 3.0	$299
NUT4NT [7]	Amungo Navigation	4	-	2	-	GNSS L1/L2/L5/L6	I, 2 bits ≤ 99 Msps	NTLab NT1065	Lattice ECP5	USB 3.0	$399
Pluto SDR [8]	Analog Devices (ADI)	1	1	1	1	325 kHz ~ 3.8 GHz	IQ, 12 bits ≤ 61.44 Msps	ADI AD9363	Zynq Z-7010	USB 2.0	$233
Pocket SDR FE 2/4/8CH	-	2/4/8	-	2/4/8	-	GNSS L1/L2/L5/L6	I/IQ, 2/3 bits ≤ 32/48 Msps	MAX2771 x 2/4/8	-	USB 2.0/3.0	$60/$130/ $200 *

* Parts only

USRP N210

GN3S Sampler

Stereo

Blade RF

Lime SDR

RTL-SDR

NUT4NT

Pluto SDR

[1] https://www.ettus.com/all-products/un210-kit/, [2] https://ccar.colorado.edu/gnss/, [3] https://gmvnsl.com/advanced-gnns-hw-sw/, [4] https://rtl-sdr.com/, [5] https://www.nuand.com/bladerf-1/, [6] https://limemicro.com/products/boards/limesdr/, [7] https://www.amungo-navigation.com/, [8] https://www.analog.com/en/resources/evaluation-hardware-and-software/evaluation-boards-kits/adalm-pluto.html

Pocket SDR FE

Device	FE 2CH	FE 4CH (ver.0.13 ~)	FE 8CH (ver.0.14 ~)	Notes
RF Input	SMA x 1	SMA x 1	SMA x 8	Antenna DC 4.2-5V, ≤ 100mA*1
# of RF CH	2 (CH1: L1, CH2: L2/L5/L6)	4 (CH1-4: L1 or L2/L5/L6)	8 (CH1-8: L1 or L2/L5/L6)	
LO Frequency	1525 - 1610 MHz (L1) or 1160 - 1290 MHz (L2/L5/L6)			
Sampling Type	I-sampling (2 or 3 bits) or IQ-sampling (2 bits)			
Sampling Rate	4, 6, 8, 12, 16, 24 or 32 Msps	4, 6, 8, 12, 16, 24, 32 or 48 Msps		
IF Filter	Bandpass or Lowpass, 3rd or 5th-order Butterworth, Bandwidth: 2.5, 4.2, 8.7, 16.4*, 23.4* or 36* MHz			* Lowpass Only
Host Interface	USB 2.0 (480 Mbps), micro-B (v.2.1),Type-C (v.2.3)	USB 3.0 (3.2 Gen 1, 5 Gbps), Type-C		
Output Format	RAW8	RAW16	RAW32	
Power	5V, 140 mA	5V, 240 mA	5V, 330 mA	USB Bus Power
Size (D x W x H)	67 x 25 x 15 mm	84 x 41 x 15 mm	77 x 110 x 30 mm	Including Case
Price	~ $60 *2	~ $130 *2 / $249 *4	~ $200 *2	

*1 Total ≤ 500mA (FE 8CH), *2 Parts only, *3 https://www.datagnss.com/collections/evk/products/pocketsdr-gnss-receiver

Prototype
v. 1.1 (2019.5)

FE 2CH v. 2.1 (2021.10) v. 2.0 (2021.10)
Prototype v. 1.4 (2021.9) v. 1.3 (2019.7)

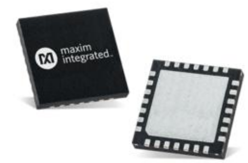
FE 2CH v. 2.1 (2021.10) v. 2.3 (2024.2)
FE 4CH v. 3.0 (2024.5)

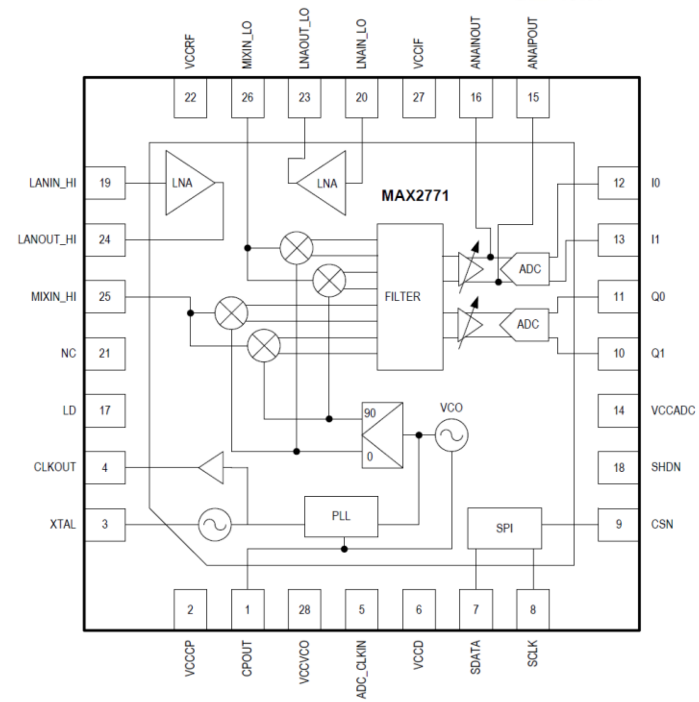
FE 8CH
v. 4.0
(2024.11)

MAX2771 (1/4)

Multiband Universal GNSS Receiver

- Multi-Constellation Support: GPS, Galileo, GLONASS, BeiDou, IRNSS, QZSS, SBAS
- Multiband Support: L1, L2, L5, E1, E5, E6, B1, B2, B3
- Programmable IF Bandwidths of 2.5MHz, 4.2MHz, 8.7MHz, 16.4MHz, 23.4MHz, 36MHz: Supports Wide-Band Carriers for Precision Applications (e.g., GPS L5, Galileo E5)
- Operates in Low IF or Zero IF Mode: Programmable IF Center Frequency
- Fractional-N Synthesizer with Integrated VCO Supports Wide Range of Reference Frequencies
- On-Chip LNAs to Support Multiple Bands
- 1.4dB Cascaded Noise Figure and 110dB of Cascaded Gain with Gain Control Range of 59dB from PGA
- Integrated Crystal Oscillator
- Supply Voltage Range: 2.7V to 3.3V
- 28-Pin, RoHS-Compliant, Thin QFN Lead-Free Package (5mm x 5mm)



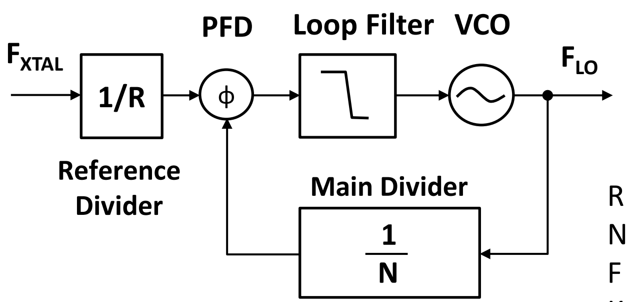


Block Diagram

ADI/maxim integrated, MAX2771 Multiband Universal GNSS Receiver Data Sheet, 2018

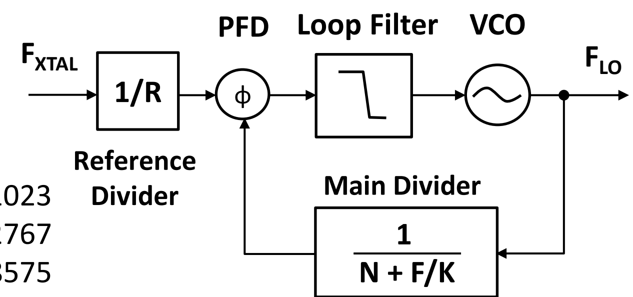
MAX2771 (2/4)

Frequency (PLL) Synthesizer



Integer-N PLL

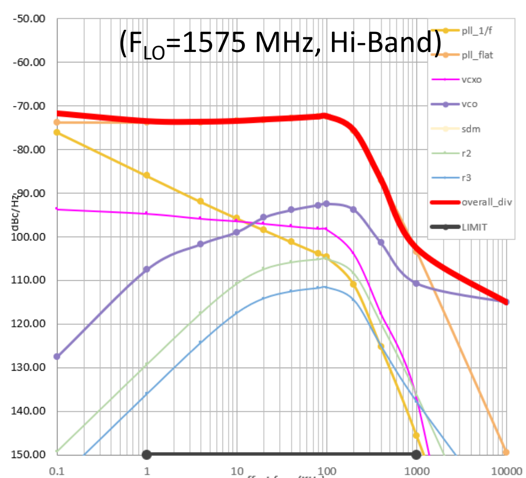
R = 1 ~ 1023
N = 36 ~ 32767
F = 0 ~ 1048575
K = 1048576 (2^{20})



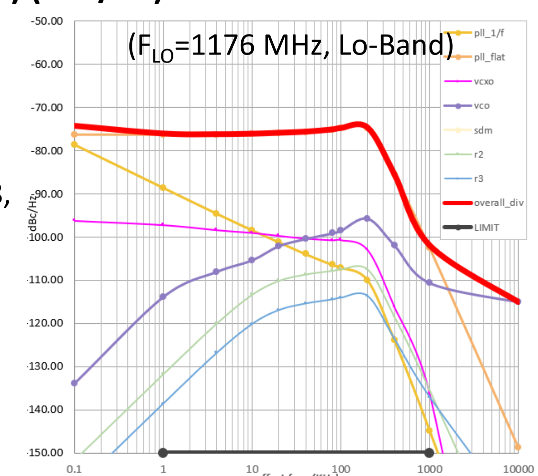
Fractional-N PLL

PFD: Phase Frequency Detector, VCO: Voltage Controlled Oscillator

PLL Phase Noise (Simulated) (dBc/Hz)



F_{XTAL}=24 MHz, RDIV=8,
Loop Filter RC:
C1=15 pF, R2=15 KΩ,
C2=750 pF



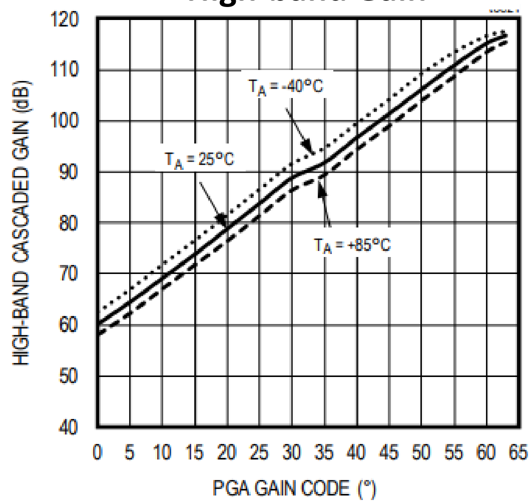
https://www.analog.com/media/en/designtools/calculators/max2771-pll-simulator.xlsx

maxim integrated, MAX2771 PLL Loop Filter Calculator User Guide, 2018

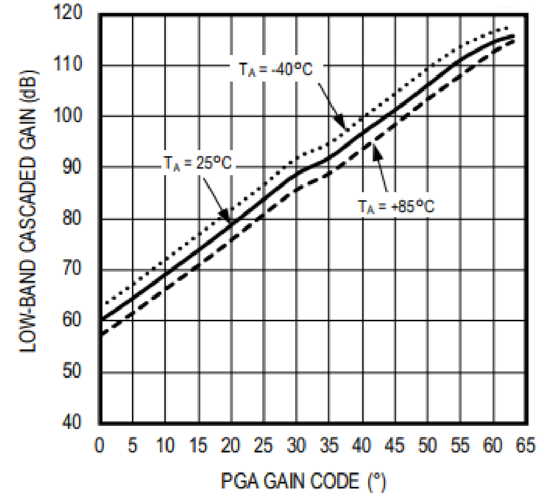
MAX2771 (3/4)

LNA

High-band Gain



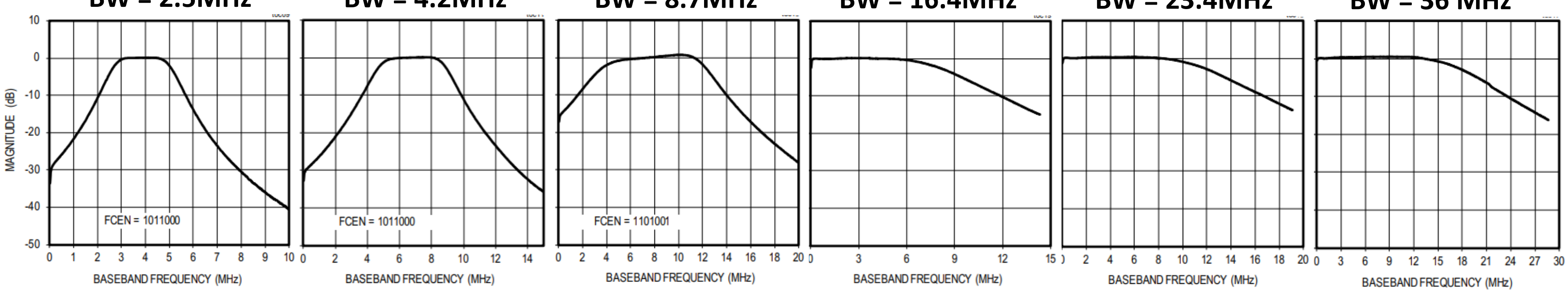
Low-band Gain



IF Filter

Filter Response (3rd-order Butterworth)

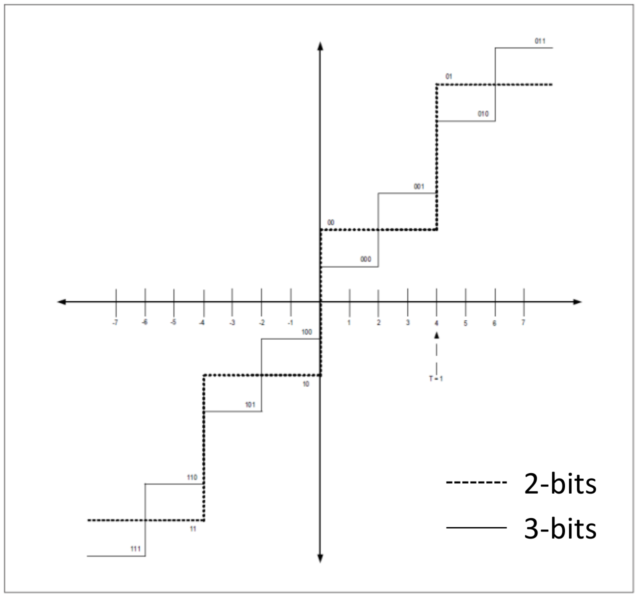
BW = 2.5MHz | BW = 4.2MHz | BW = 8.7MHz | BW = 16.4MHz | BW = 23.4MHz | BW = 36 MHz



MAX2771 (4/4)

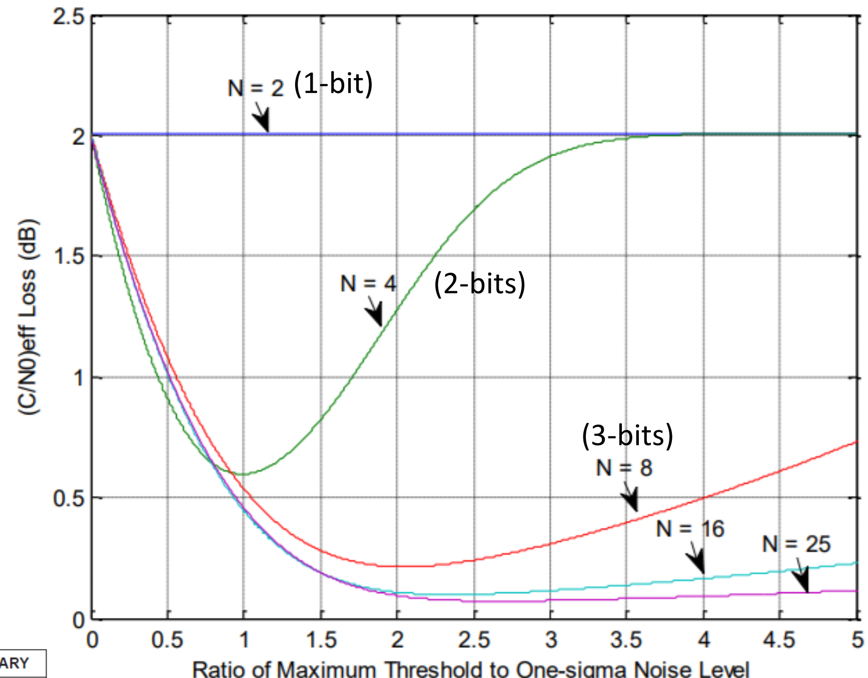
ADC

ADC Quantization Levels [1]



Output Data Format [1]

INTEGER VALUE	SIGN/MAGNITUDE			UNSIGNED BINARY			TWO'S COMPLEMENT BINARY		
	1b	2b	3b	1b	2b	3b	1b	2b	3b
7	0	01	011	1	11	111	0	01	011
5	0	01	010	1	11	110	0	01	010
3	0	00	001	1	10	101	0	00	001
1	0	00	000	1	10	110	0	00	000
-1	1	10	100	0	01	011	1	11	111
-3	1	10	101	0	01	010	1	11	110
-5	1	11	110	0	00	001	1	10	101
-7	1	11	111	0	00	000	1	10	100



SNR Losses due to N-Quantization (BW = 10 * chip rate) [2]

[1] ADI/maxim integrated, MAX2771 Multiband Universal GNSS Receiver Data Sheet, 2018

[2] C.J.Hegarty, Analytical Model for GNSS Receiver Implementation Losses, Navigation, 2011

Reference Oscillator

TCXO (Temperature Compensated Crystal Oscillator)

TG2520SMN-24.0000M-MCGNNM3

(1) (2) (3) (4)(5)(6) (7)

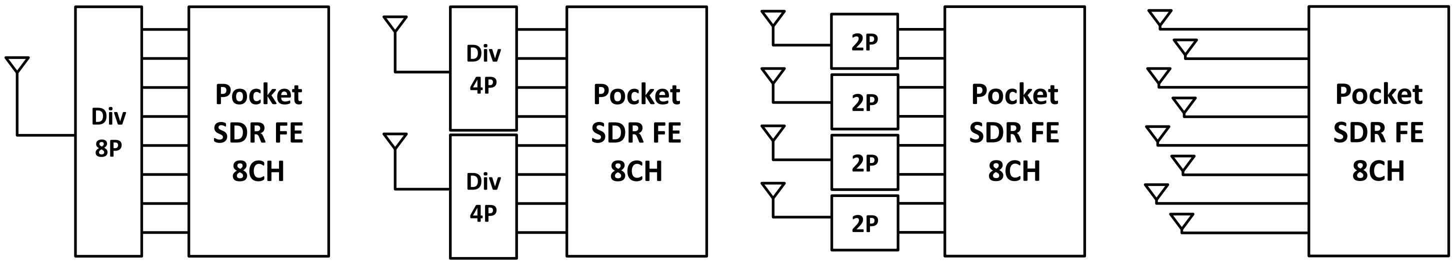
(1) Model (TG2016: 2.0 x 1.6 mm, TG2520: 2.5 x 2.0 mm)
(2) Output (S: Clipped sine wave)
(3) Frequency
(4) Supply voltage (E: 1.8V, M: 2.8 ~ 3.3V)
(5) Freq./temp. charac. (C: ±0.5 x 10^{-6}, F: ±2.0 x 10^{-6} MAX)
(6) Operating temperature: (G: -40 C ~ +85 C)
(7) VC function (N: non, B: 0.9V, C: 1.4V, D:1.5V, E: 1.65V)

Item	Symbol	VC-TCXO	TCXO	Conditions / Remarks
Output frequency range	f_0	10 MHz to 55MHz		
		16, 16.368, 16.369, 19.2, 20, 24, 25, 26, 27, 27.6, 30, 32, 38.4, 40, 48, 50, 52 MHz		Standard frequency
Supply voltage	Vcc	1.8 V ±0.1 V / 2.8 V ±5 % / 3.0 V ±5 % / 3.3 V ±5 %		Supply voltage range :1.7 V to 3.63 V
Storage temperature	T_stg	-40 °C to +90 °C		Storage as single product.
Operating temperature	T_use	G: -40 C to +85 C		
Frequency tolerance	f_tol	±1.5 × 10^{-6} Max.		After reflow, +25 C
Frequency/temperature characteristics	f_0-Tc	C: ±0.5 × 10^{-6} Max. / G: -40 C to +85 C F: ±2.0 × 10^{-6} Max. / G: -40 C to +85 C		Standard stability version
Frequency/load coefficient	f_0-Load	±0.1 × 10^{-6} Max.		10 kΩ // 10 pF ±10 %
Frequency/voltage coefficient	f_0-Vcc	±0.1 × 10^{-6} Max.		Vcc ± 5 %
Frequency aging	f_age	±0.5 × 10^{-6} Max.		+25 C, First year, 10MHz, 12 MHz≤ f_0 ≤20 MHz, 24 MHz≤ f_0 ≤40 MHz
		±1.5 × 10^{-6} Max.		+25 C ,First year, 10 MHz< f_0 <12 MHz, 20 MHz< f_0 <24 MHz, 40 MHz< f_0 ≤55 MHz
Current consumption	Icc	1.5 mA Max.		10 MHz≤ f_0 ≤26 MHz
		1.8 mA Max.		26 MHz< f_0 ≤40 MHz
		2.0 mA Max.		40 MHz< f_0 ≤50 MHz
		2.1 mA Max.		50 MHz< f_0 ≤55 MHz
Input resistance	Rin	500 kΩ Min.	-	Vc - GND (DC)
Frequency control range	f_cont	±8.0 × 10^{-6} to ±12.0 × 10^{-6}	-	B: Vc =0.9 V ±0.6 V (Vcc =1.8 V) or C: Vc =1.4 V ±1.0 V (Vcc =2.8 V) or D: Vc =1.5 V ±1.0 V (Vcc =3.0 V) or E: Vc =1.65 V ±1.0 V (Vcc =3.3 V)
Frequency change polarity	-	Positive polarity	-	
Symmetry	SYM	45 % to 55 %		GND level (DC cut)
Output voltage	V_{PP}	0.8 V Min.		Peak to Peak
Start-up time	t_str	1.0 ms Max.		T=0 at 90% Vcc
Output load condition	Load_R	10 kΩ		DC cut capacitor = 0.01 µF
	Load_C	10 pF		

Seiko Epson, VC-TCXO/TCXO High Stability / Low Noise TG2016SMN/TG2520SMN Data Sheet

Pocket SDR FE 8CH (1/3)

- **Configurable by External RF-dividers**

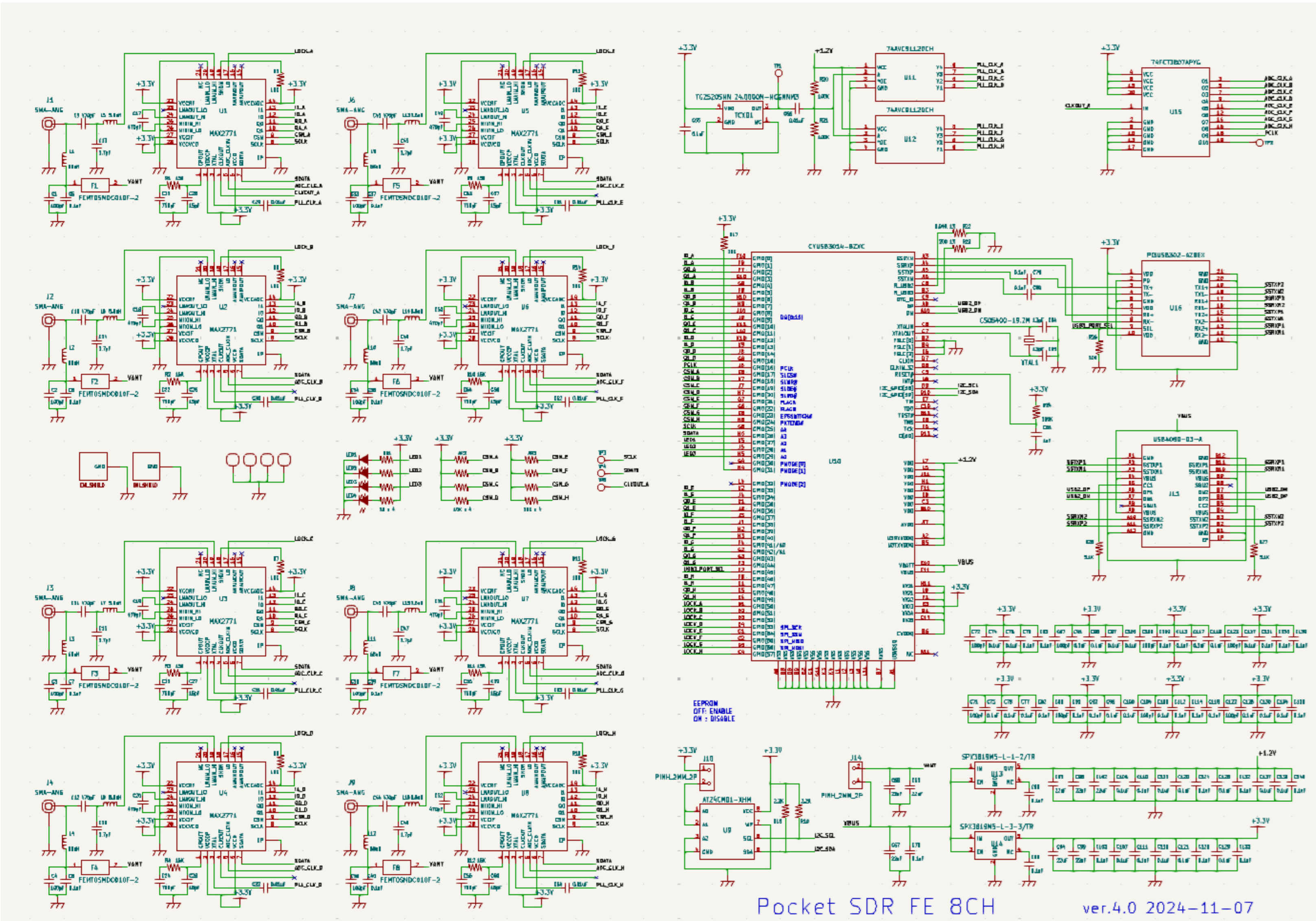


All L-band GNSS receiver (L1+B1+G1+L2+G2+L5+ E5b/B2+B3/L6)	GNSS-compass (qual-freq.)	4-elements Antenna-array (dual-freq.)	8-elements Antenna-array (single-freq.)

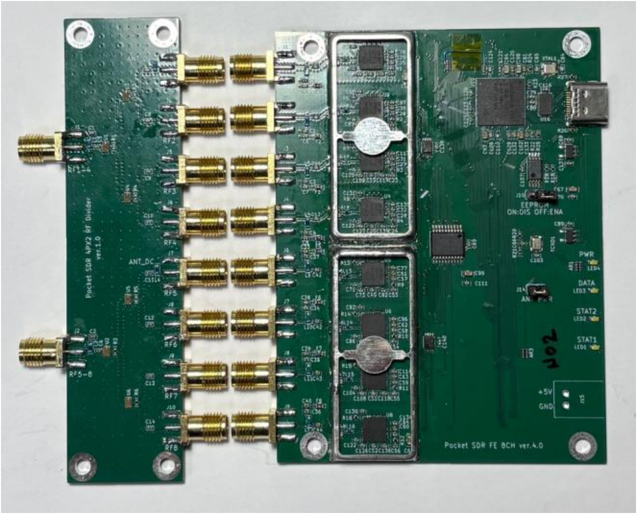
- **Parts Cost**

	A	B	C	E	F	G	H	J	K	L
1	No	Reference	Value	Qty	Assy	Supplier	Supplier Part #	Price	Total	URL
38	37	J11	USB Type-C Receptacle USB 3.2	1	1	GCT	USB4080-03-A	226.0	226.0	https://www.digikey.jp/
39	38	J12,J13	2-PIECE SHIELD Frame	2	0	Masach	MS448-10F	234.0	468.0	https://www.digikey.jp/
40	39	J12-S,J13-S	2-PIECE SHIELD Cover	2	0	Masach	MS448-10C	230.0	460.0	https://www.digikey.jp/
41										
42		Subtotal		233	217				26836.5	
43										
44	41	PCB	FR-4, 4-layer	1	0	JLCPCB		784.5	784.5	5pcs + metal mask
45	42	CASE BODY	MX3 mobile case	1	0	Takachi	MX3-11-8SBP	781.0	781.0	
46	43	CASE PANEL	SLA(Resin), Black, Sanding-Ger	1	0	JLC3DP		777.0	777.0	2pcs (F+B)
47										
48		Subtotal							2342.5	
49										
50		Total							29179.0	(Yen)

Pocket SDR FE 8CH (2/3)

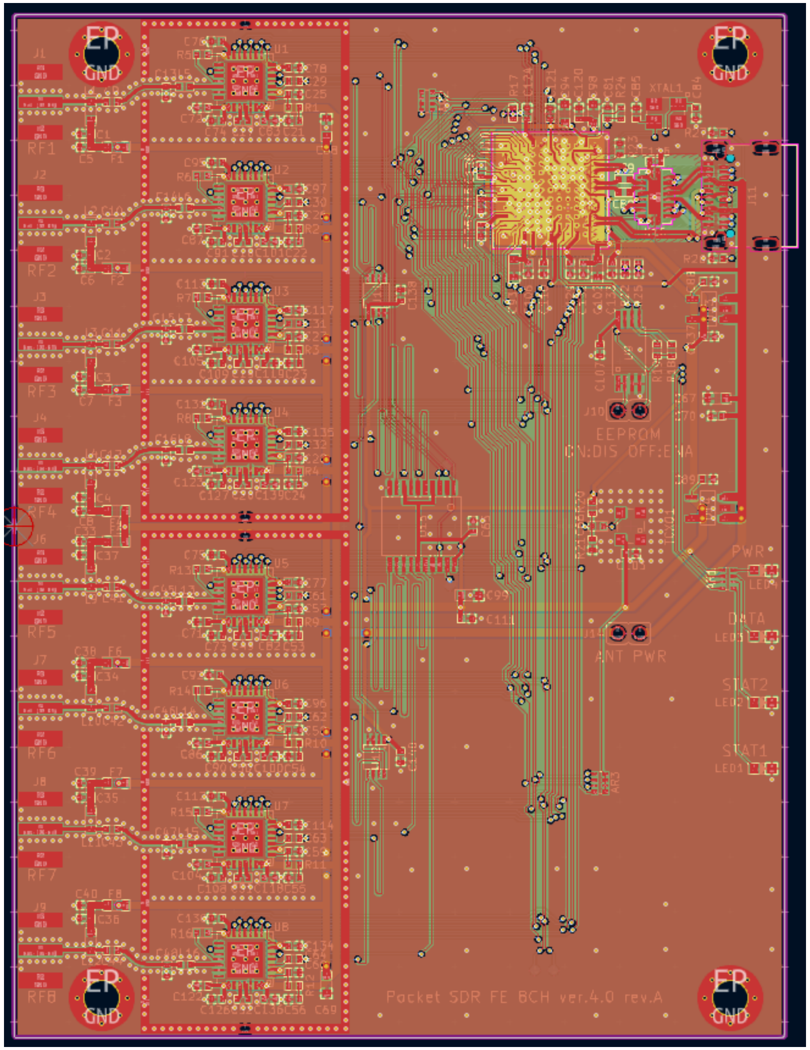


Pocket SDR FE 8CH (3/3)



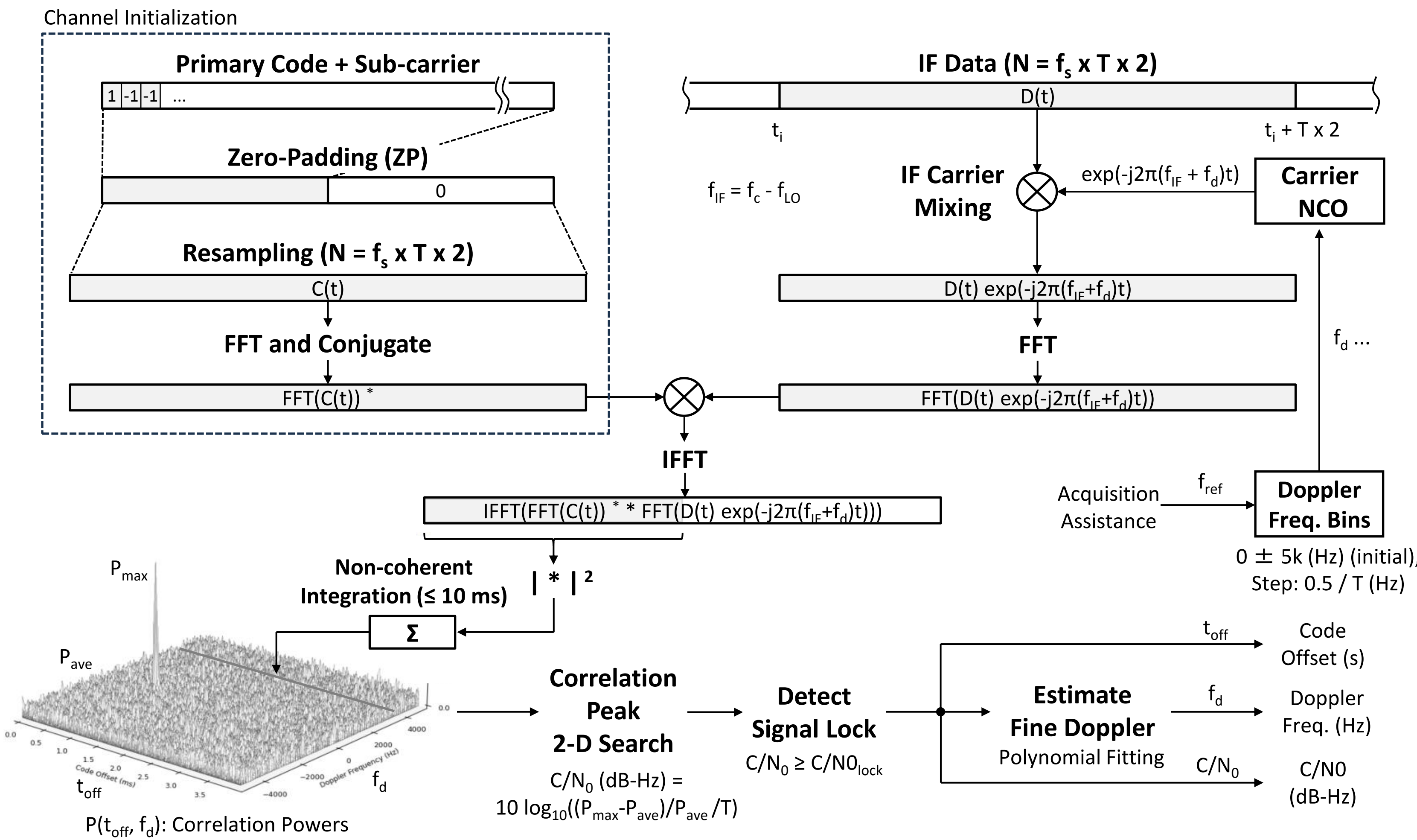
4Px2 Divider　　Pocket SDR FE 8CH

Pocket SDR FE 8CH-2ANT



Pocket SDR S/W

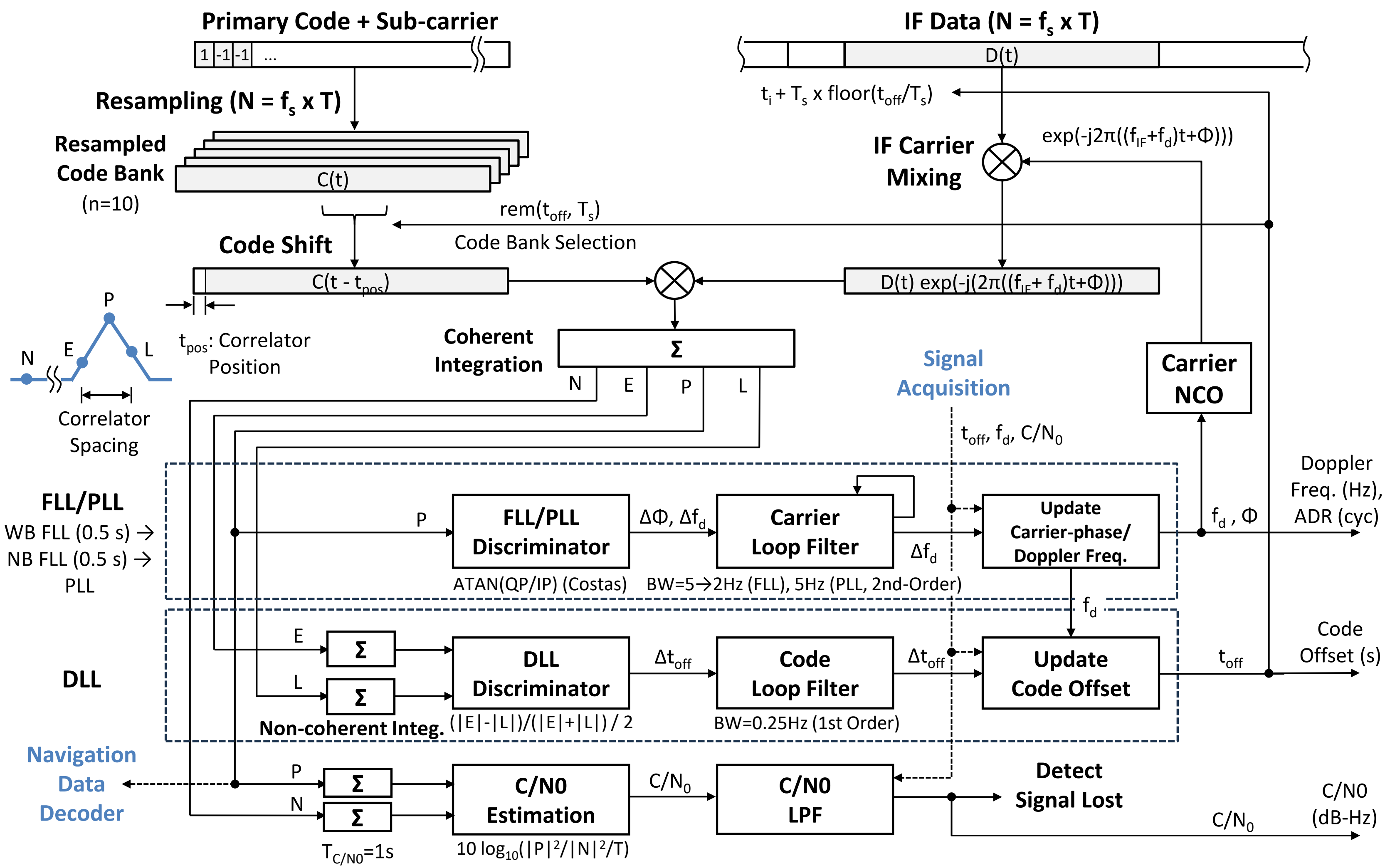
Signal Acquisition



f_s: Sampling Rate (sps), T: Primary Code Cycle (s), f_{IF} : IF Carrier Freq. (Hz), f_d: Doppler Freq. (Hz), t_{off} : Code Offset (s), $C/N0_{lock}$: Lock Threshold (dB-Hz)

f_c: Carrier Freq. (Hz), f_{LO}: Local Oscillator (PLL) Freq. (Hz)

Signal Tracking



E/P/L/N: Early/Prompt/Late/Noise Correlation, f_s: Sampling Rate (sps), T_s: Sampling Cycle (s), T: Primary Code Cycle (s), f_{IF} : IF Carrier Freq. (Hz), f_d: Doppler Freq. (Hz), t_{off} : Code Offset (s), Φ: Carrier-phase (cyc), Δf_d: Doppler Error (Hz), Δt_{off}: Code Offset Error (s), ΔΦ: Carrier-phase Error (cyc)

Navigation Data Decoder

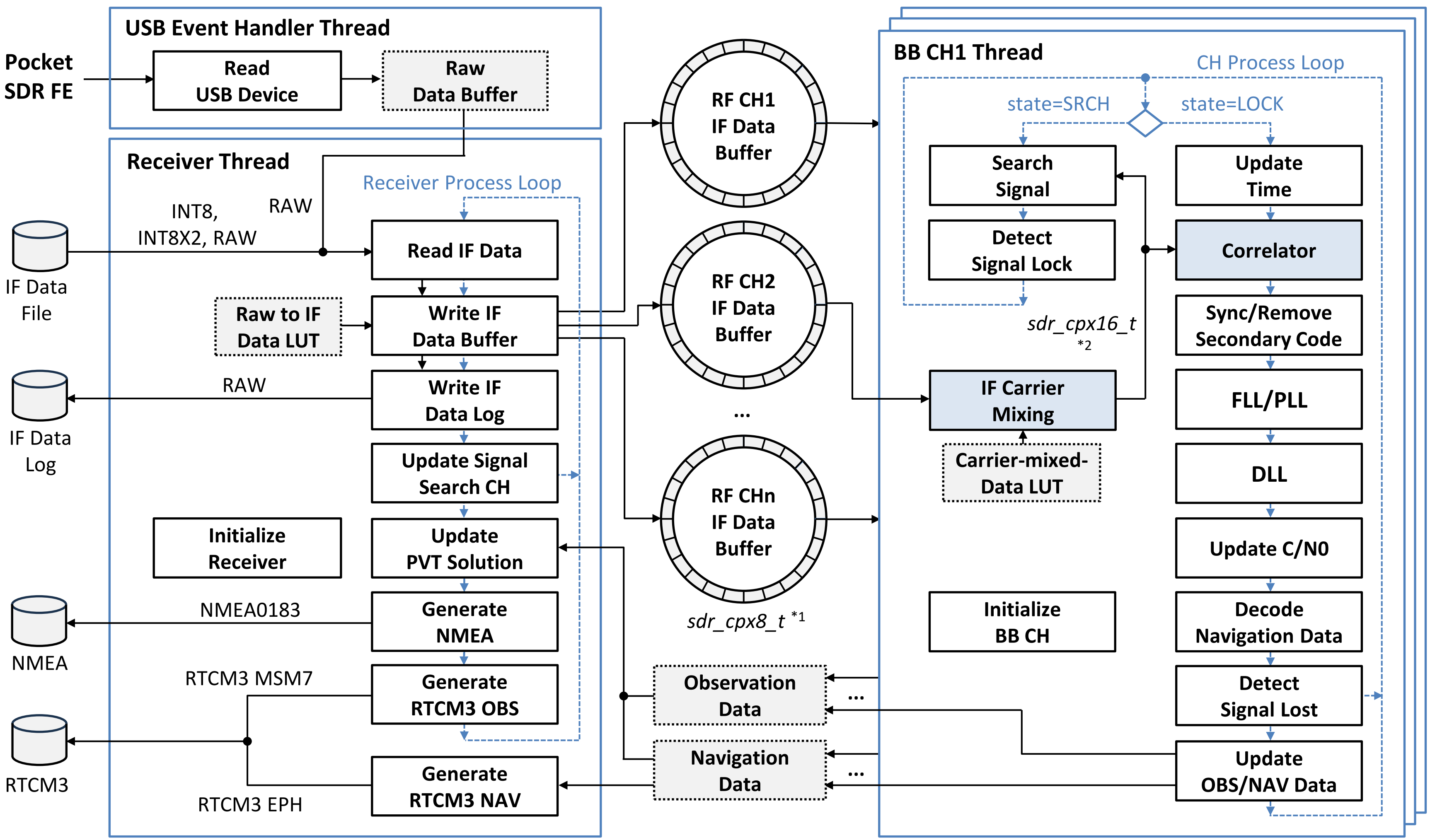
System	Signal	Navigation Data	Cycle (s)	Sync (syms)	Subframe/Message Structure (bits)	FEC
GPS, QZSS	L1C/A	LNAV	6.0	8	(24 + 6 (parity)) x 10	-
	L2C, L5Q	CNAV	12.0	8	8 (sync) + 268 + 24 (CRC)	1/2(600,300)
	L1Cd	CNAV-2	18.0	52	9 (SF1) + 600 (SF2) + 274 (SF3)	BCH(52,9), BI(38x46), LDPC(1200,600), LDPC(548,274)
GLONASS	L1C/A, L2C/A	GLO-STR	2.0	30	1 + 76 + 8 (Hamming)	Hamming
	L3OCd	GLO-L3-STR	3.0	20	20 (sync) + 256 + 24 (CRC)	1/2(600,300)
	L1OCd	GLO-L1-STR	2.0	12	12 (sync) + 222 + 16 (CRC)	1/2(500,250)
Galileo	E1C, E5b-I	I/NAV	2.0	10	114 + 6 (Tail) (even) 82 + 24 (CRC) + 8 + 6 (Tail) (odd)	BI(30x8), 1/2(250,125)
	E5a-I	F/NAV	10.0	12	214 + 24 (CRC) + 6 (Tail)	BI(61x8), 1/2(500,250)
	E6B	C/NAV	1.0	16	14 + 448 + 24 (CRC) + 6 (Tail)	BI(123x8), 1/2(1000,500)
QZSS	L1S	L1S	1.0	8 x 3	8 (sync) + 218 + 24 (CRC)	1/2(500,250)
	L6D, L6E	L6D, L6E	1.0	32	32 (sync) + 17 + 1695 + 256 (RS)	RS(255,223)
BeiDou	B1I, B2I, B3I	D1, D2	6.0	11	(26 + 4 (parity)) + (22 + 8 (parity)) x 9	BI(30x10), BCH(15,11,1)
	B1C-D	B-CNAV1	18.0	72	14 (SF1) + 600 (SF2) + 264 (SF3)	BCH(21,6), BCH(51,8), BI(36x48), 64ary-LDPC(200,100), 64ary-LDPC(88,44)
	B2a-D	B-CNAV2	3.0	24	264 + 24 (CRC)	64ary-LDPC(96,48)
	B2b-I	B-CNAV3, B2b-PPP	1.0	28	462 + 24 (CRC)	64ary-LDPC(162,81)
NavIC	L1-SPS	IRN-L1-NAV	18.0	52	9 (SF1) + 600 (SF2) + 274 (SF3)	BCH(52,9), BI(46x38), LDPC(1200,600), LDPC(548,274)
	L5-SPS, S-SPS	IRN-NAV	12.0	16	262 + 24 (CRC) + 6 (Tail)	BI(73x8), 1/2(572,286)
SBAS	L1C/A	SBAS	1.0	8 x 3	8 (sync) + 218 + 24 (CRC)	1/2(500,250)
	L5I	L5 SBAS	1.0	4 x 6	4 (sync) + 222 + 24 (CRC)	1/2(500,250)

FEC: Forward Error Correction, 1/2: Convolutional Code (R=1/2, K=7), BI: Block Interleave, LDPC: Low Density Parity Check, RS: Reed Solomon

PVT

- **Update Observation Data**
  - Decode week/TOW fields in navigation data
  - Generate pseudorange with week/TOW and code offset
  - Resolve 100 ms ambiguity
- **Update Navigation Data**
  - Decode ephemeris fields in navigation data
  - LNAV (GPS/QZSS), GLO-STR (GLONASS), I/NAV, F/NAV (Galileo), D1, D2 (BDS), IRN-NAV (NavIC)
- **Generate PVT Solution**
  - Resolve msec ambiguity in pseudorange (L5Q, L5S, G3OCP, L5Q SBAS)
  - Point positioning with L1 pseudorange by calling RTKLIB pntpos()
  - Adjust epoch cycle within 20 ms
- **Outputs**
  - NMEA 0183 (GNRMC, GNGGA, GNGSA, GxGSV)
  - RTCM 3.4 MSM (MT1077, 1087, 1097, 1117, 1127, 1137, 1107) with extensions
  - RTCM 3.4 EPH (MT1019, 1020, 1045, 1046, 1044, 1042, 1041) (decoded ephemeris)
  - Receiver Log (CSV text)
  - Raw IF Data Log (RAW8, RAW16 or RAW32)
  - Raw IF Data Tag file (ver. 0.14 ~)

S/W Structure

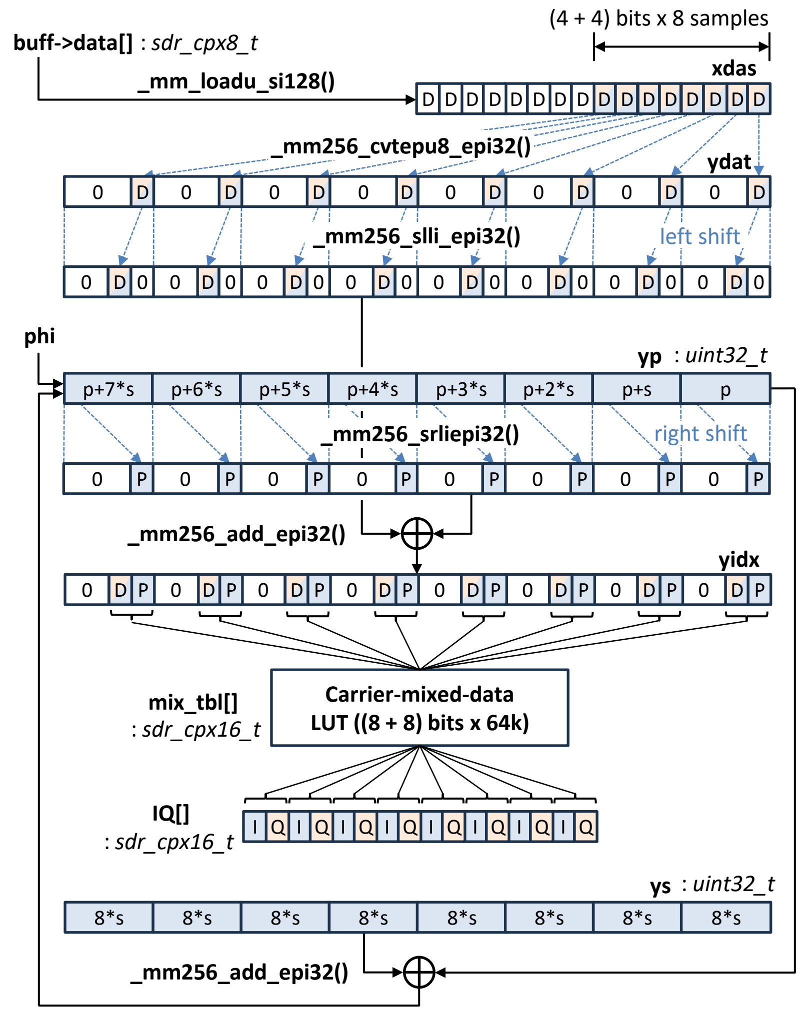


*1 (4 + 4) bits Fixed Point Complex
*2 (8 + 8) bits Fixed Point Complex

Optimization: IF Carrier Mixing

PocketSDR/src/sdr_func.c (ver.0.13)

```
void mix_carr(const sdr_buff_t *buff, int ix, int N, double phi,
              double step, sdr_cpx16_t *IQ) { // step = Δphi / sample
  const uint8_t *data = buff->data + ix;
  double scale = (double)(1 << 24) * NTBL;
  uint32_t p = (uint32_t)((phi - floor(phi)) * scale); // initial phase
  uint32_t s = (uint32_t)(int)(step * scale); // phase step / sample
  int i = 0;
  __m256i yp = _mm256_set_epi32(p+s*7, p+s*6, p+s*5, p+s*4, ...);
  __m256i ys = _mm256_set1_epi32(s*8);
  for ( ; i < N - 16; i += 8) {
    int idx[8];
    __m128i xdas = _mm_loadu_si128((__m128i *)(data + i));
    __m256i ydat = _mm256_cvtepu8_epi32(xdas);
    __m256i yidx = _mm256_add_epi32(_mm256_slli_epi32(ydat, 8),
                       _mm256_srli_epi32(yp, 24));
    _mm256_storeu_si256((__m256i *)idx, yidx);
    IQ[i  ] = mix_tbl[idx[0]]; // IQ(t)=data(t)*exp(-j*2*pi*(f*t+phi))
    IQ[i+1] = mix_tbl[idx[1]];
    IQ[i+2] = mix_tbl[idx[2]];
    ...
    IQ[i+7] = mix_tbl[idx[7]];
    yp = _mm256_add_epi32(yp, ys);
  }
  for (p += s * i; i < N; i++, p += s) {
    int idx = ((int)data[i] << 8) + (p >> 24);
    IQ[i] = mix_tbl[idx];
  }
}
```



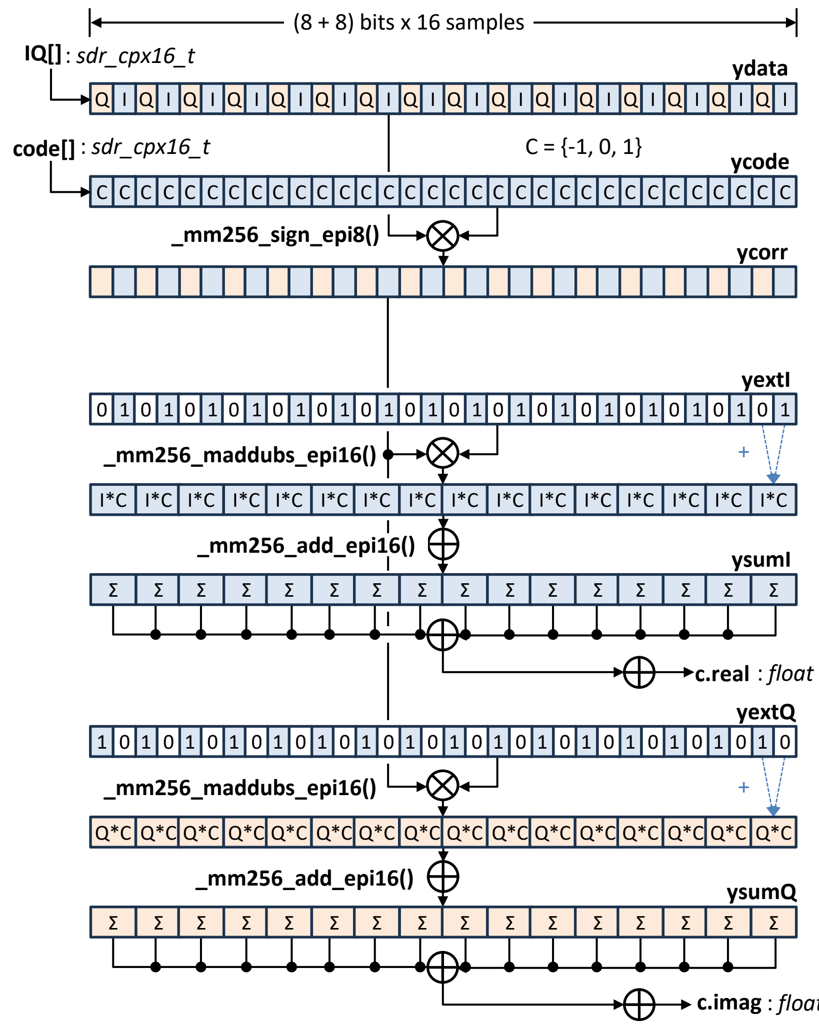
_mm_*(), _mm256_*(): SSE2, AVX2 Intrinsics [1]

[1] https://www.intel.com/content/www/us/en/docs/intrinsics-guide/index.html

Optimization: Correlator

PocketSDR/src/sdr_func.c

```
void dot_IQ_code(const sdr_cpx16_t *IQ, const sdr_cpx16_t *code, int N,
                 float s, sdr_cpx_t *c) { // c: correlation
  int i = 0;
  (*c)[0] = (*c)[1] = 0.0f;
  __m256i ysumI = _mm256_setzero_si256();
  __m256i ysumQ = _mm256_setzero_si256();
  __m256i yextI = _mm256_set_epi8(0,1,0,1,0,1,0,1,0,1,0,1,...);
  __m256i yextQ = _mm256_set_epi8(1,0,1,0,1,0,1,0,1,0,1,0,...);
  for ( ; i < N - 15; i += 16) {
    __m256i ydata = _mm256_loadu_si256((__m256i *)(IQ + i));
    __m256i ycode = _mm256_loadu_si256((__m256i *)(code + i));
    __m256i ycorr = _mm256_sign_epi8(ydata, ycode);
    ysumI = _mm256_add_epi16(ysumI, _mm256_maddubs_epi16(yextI, ycorr));
    ysumQ = _mm256_add_epi16(ysumQ, _mm256_maddubs_epi16(yextQ, ycorr));
    if (i % (16 * 256) == 0) { // to avoid overflow
      sum_s16(ysumI, (*c)[0])  // c.real += sum(ysumI), ysumI = 0
      sum_s16(ysumQ, (*c)[1])  // c.imag += sum(ysumQ), ysumQ = 0
    }
  }
  sum_s16(ysumI, (*c)[0]) // c.real += sum(ysumI)
  sum_s16(ysumQ, (*c)[1]) // c.imag += sum(ysumQ)
  for ( ; i < N; i++) {
    (*c)[0] += IQ[i].I * code[i].I;
    (*c)[1] += IQ[i].Q * code[i].Q;
  }
  (*c)[0] *= s * SDR_CSCALE; // correlation I
  (*c)[1] *= s * SDR_CSCALE; // correlation Q
}
```



_mm256_*(): AVX2 Intrinsics [1]

[1] https://www.intel.com/content/www/us/en/docs/intrinsics-guide/index.html

Optimization: Performance

Single Thread Execution Time of API Functions (μs)

CPU/OS = AMD Ryzen AI 9 HX370 (12C/24T), Windows 11

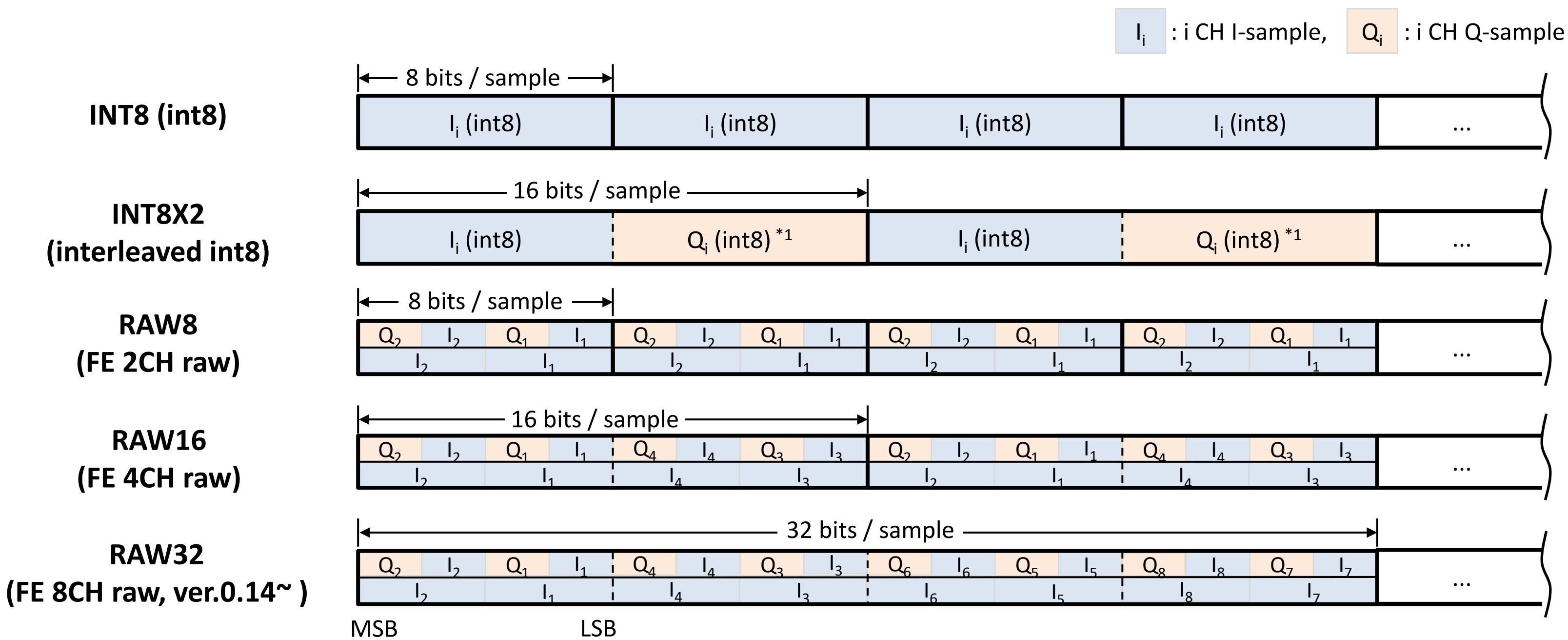
S/W	N =	API Function (sdr_*())		
		mix_carr	corr_std	corr_fft
Pocket SDR ver.0.13, C with AVX2	12,000	1.6	4.8	42.3
	24,000	4.9	6.5	82.6
	48,000	9.7	14.6	85.9
	96,000	16.2	29.4	617.1
Pocket SDR ver.0.13, C w/o AVX2	12,000	4.7	36.2	48.7
	24,000	7.4	73.9	95.9
	48,000	16.9	148.9	107.2
	96,000	33.8	301.3	661.4
Pocket SDR ver.0.11, C with AVX2	12,000	8.0	14.3	63.1
	24,000	17.5	26.7	96.3
	48,000	34.2	55.2	187.7
	96,000	69.6	110.2	641.7
Pocket SDR ver.0.13, python 3	12,000	32.2	48.6	81.1
	24,000	59.6	86.3	169.5
	48,000	116.8	168.4	342.4
	96,000	258.7	330.6	909.7

S/W = Pocket SDR ver.0.13, C with AVX2/NEON

CPU/OS	N =	API Function (sdr_*())		
		mix_carr	corr_std	corr_fft
Intel Core i5-8259U (4C/8T), Ubuntu 22.04 LTS	12,000	5.5	9.1	99.3
	24,000	10.3	17.4	178.8
	48,000	20.5	35.1	251.4
	96,000	41.3	76.5	832.5
M2 Mac mini (Apple M2, 8C/8T), macOS 14.6.1	12,000	5.5	9.3	127.3
	24,000	8.5	18.8	267.7
	48,000	17.1	37.1	461.8
	96,000	36.0	75.6	1161.7
Raspberry Pi 5 (Cortex-A76, 4C/4T), Raspberry Pi OS	12,000	16.1	27.1	319.0
	24,000	32.1	54.7	545.6
	48,000	64.2	108.8	894.6
	96,000	129.2	218.1	3582.6
Raspberry Pi 4 (Cortex-A72, 4C/4T), Raspberry Pi OS	12,000	24.7	49.5	759.3
	24,000	49.5	89.1	1435.4
	48,000	98.8	177.5	3373.6
	96,000	197.6	357.2	11343.5

mix_carr: IF carrier mixing, corr_std: Standard (N+E+P+L) correlator with IF carrier mixing, corr_fft : parallel correlator by FFT,
FFT performance optimized by FFTW wisdom at N = 48000

IF Data Formats



*1 Polarity inverted due to MAX2771 convention

2 bits IQ-sampling

bits	00	01	10	11
I	+1	+3	-1	-3
Q *1	-1	-3	+1	+3

2 bits I-sampling

bits	**00	**01	**10	**11
I	+1	+3	-1	-3

3 bits I-sampling

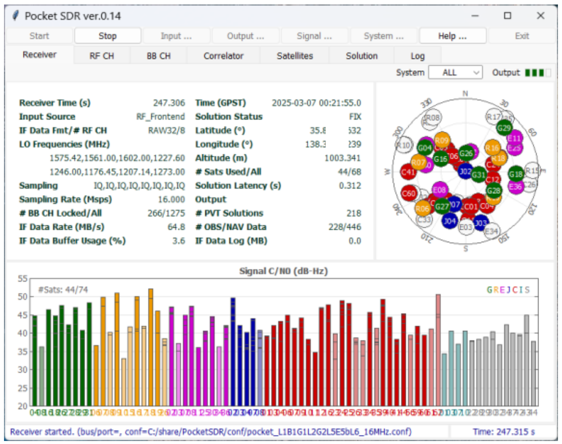
bits	0000	1000	0001	1001	0010	1010	0011	1011
I	+1	+3	+5	+7	-1	-3	-5	-7

IF Data Tag File (<path>.tag)

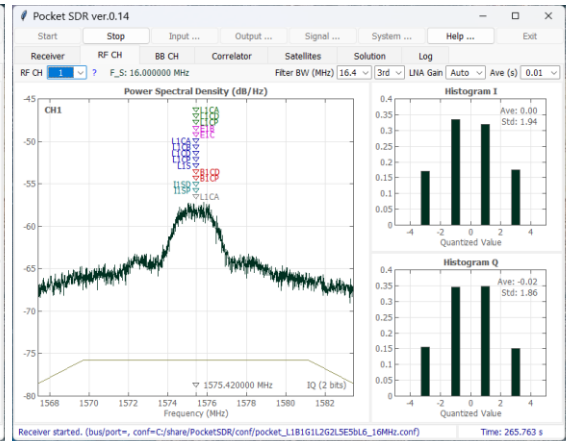
```
PROG = Pocket SDR
TIME = 2024/11/01 20:48:00.776      (Start Time GPST)
FMT  = RAW16                        (Format)
F_S  = 24                           (Sampling Freq. MHz)
F_LO = 1568,1237.8,1176.45,1278.75  (LO Freq. MHz)
IQ   = 2,2,2,2                      (Sampling, 1:I, 2:IQ)
BITS = 2,2,2,2                      (Sampling Bits)
```

GNSS Receiver (pocket_sdr.py) (1/3)

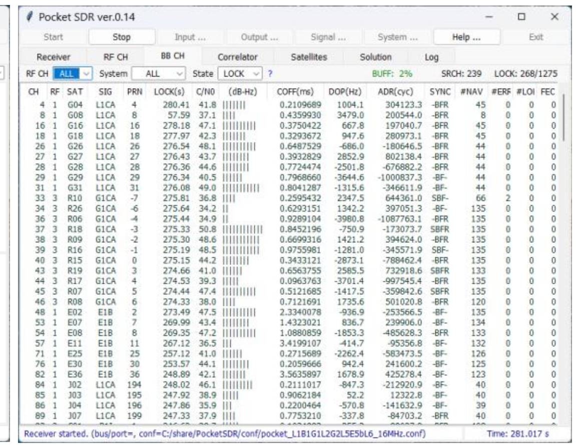
Receiver Tab



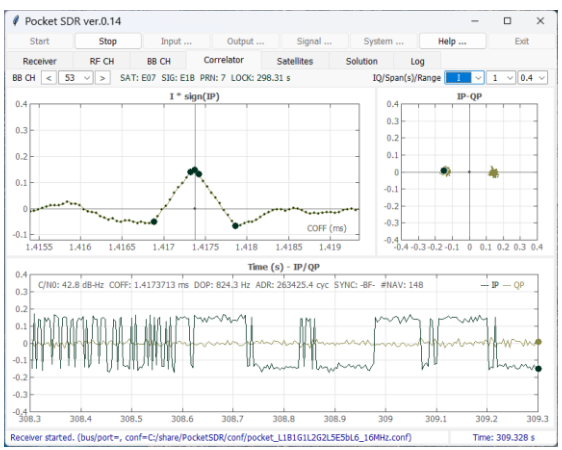
RF CH Tab



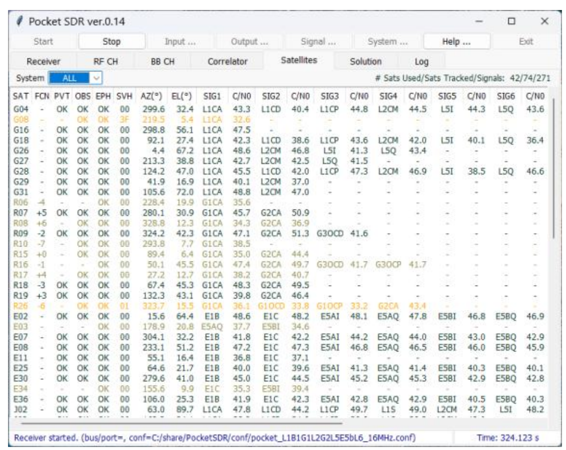
BB CH Tab



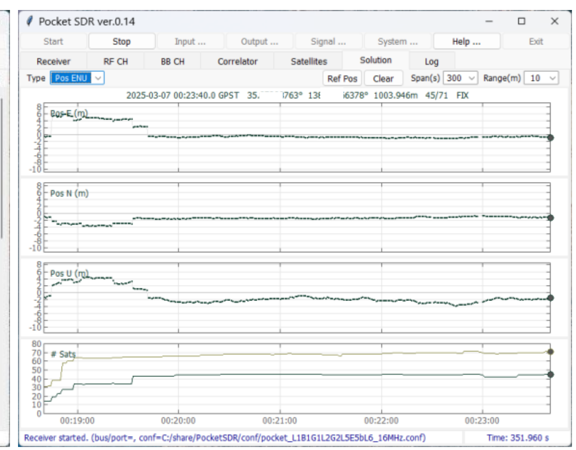
Correlator Tab



Satellites Tab

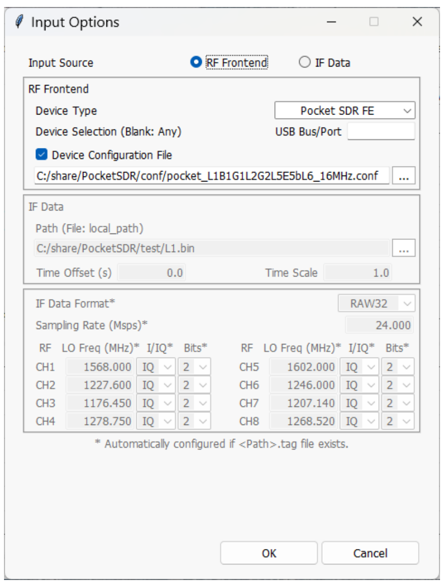


Solution Tab

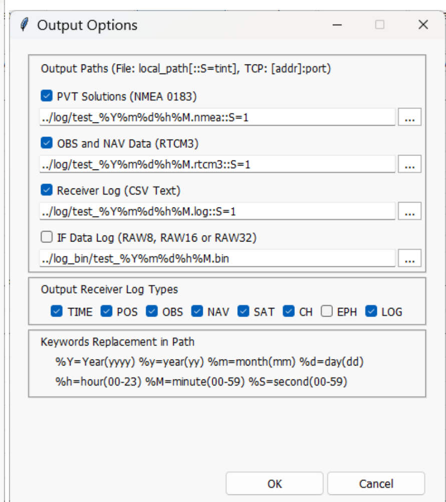


GNSS Receiver (pocket_sdr.py) (2/3)

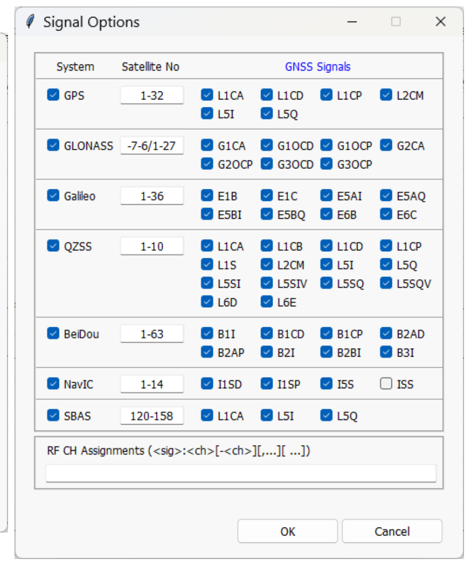
Input Options Dialog



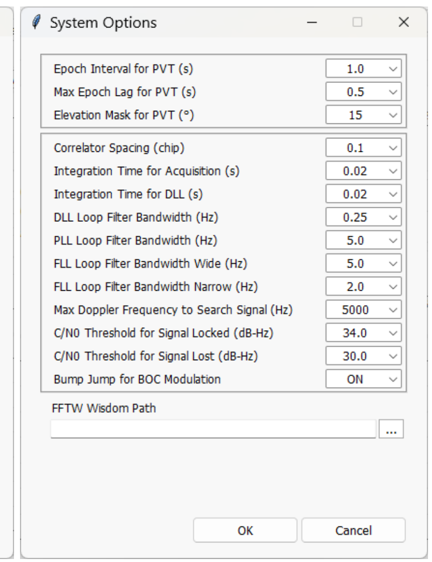
Output Options Dialog



Signal Options Dialog

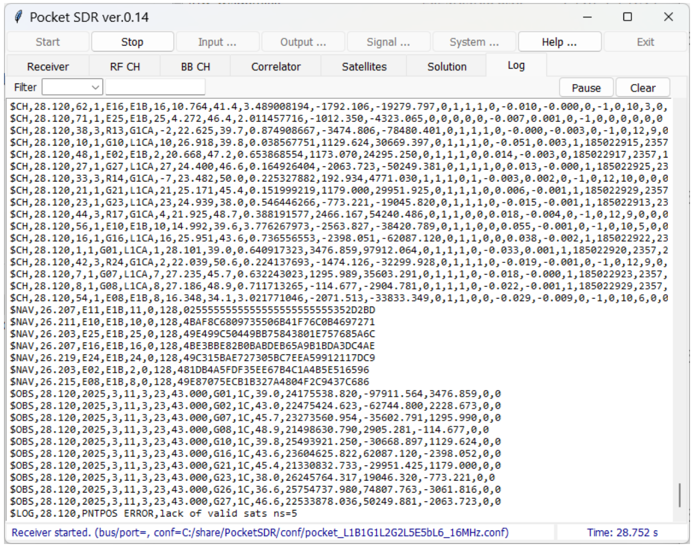


System Options Dialog



GNSS Receiver (pocket_sdr.py) (3/3)

Log Tab

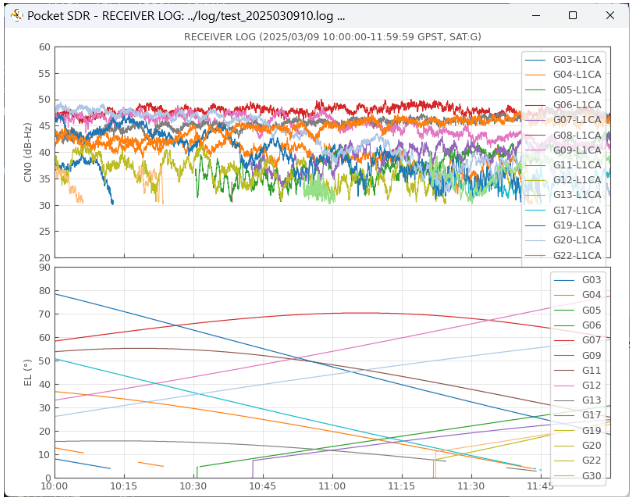


Receiver Logs

```
$TIME,time,year,mon,day,hour,min,sec,tsys (time info)
$OBS,time,year,month,day,hour,min,sec,sat,code,cn0,pr,cp,dop,lli,fcn (OBS data)
$NAV,time,sat,sig,nerr,size,data (raw NAV data HEX dump)
$POS,time,year,month,day,hour,min,sec,lat,lon,hgt,Q,ns,stdn,stde,stdu,dtr (position solution)
$SAT,time,year,month,day,hour,min,sec,sat,pvt,obs,cn0,az,el,res (satellite info)
$EPH,time,sat,sig,IODE,IODC,SVA,SVH,Toe,Toc,Ttr,A,e,i0,OMEGA0,omega,M0,delta-n,OMEGAdot,Idot,Crc,Crs,Cuc,Cus,Cic,Cis,Toes,
  Fit,Af0,Af1,Af2,TGD,code,flag (decoded ephemeris)
$CH,time,ch,rfch,sat,sig,prn,lock,cn0,coff,dop,adr,ssync,bsync,fsync,rev,srev,err_phas,err_code,tow_v,tow,week,type,nnav,
  nerr,nlol,nfec (receiver channel info)
$LOG,time,message (error, warning, general info message)
```

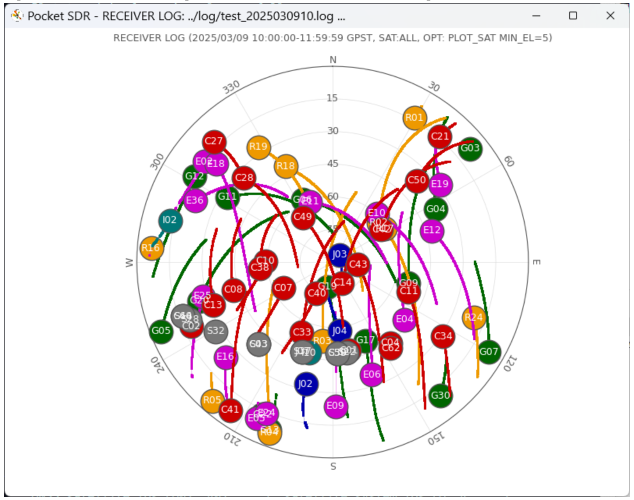
Receiver Log Viewer (pocket_plot.py)

```
pocket_plot.py test_202503091[01].log -type CN0,EL -sat G ¥
  -legend -style -
```

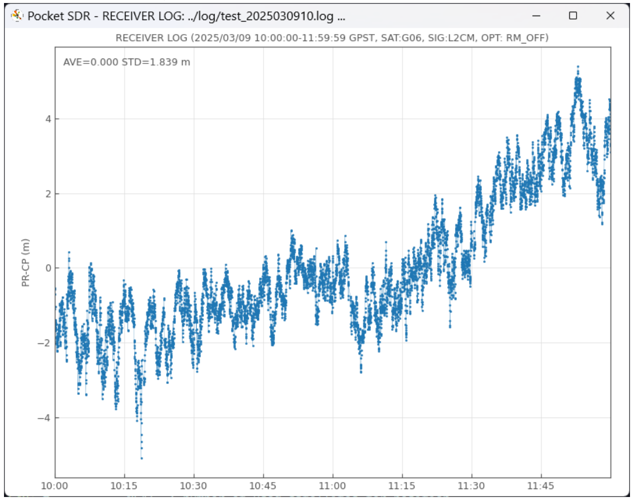



```
pocket_plot.py test_202503091[0-1].log -type POS,NSAT ¥
  -style . -stats
```


```
pocket_plot.py test_202503091[0-1].log -type SKY -sat ALL ¥
  -opt PLOT_SAT -opt MIN_EL=5 -color sys
```




```
pocket_plot.py test_202503091[0-1].log -type PR-CP ¥
  -sat G06 -sig L2CM -stats -opt RM_OFF
```



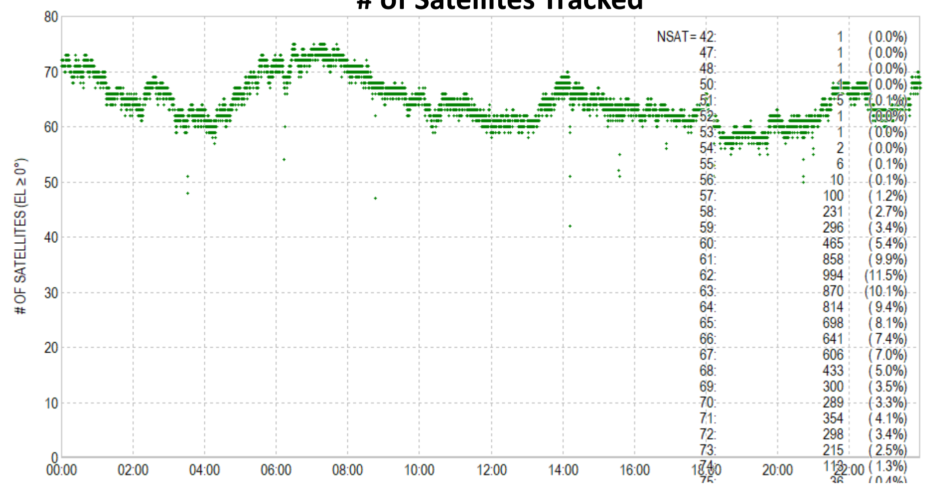
Signal Tracking Performance

Pocket SDR ver.0.14

Signals Tracked (RINEX)

```
G   20 C1C L1C D1C S1C C1L L1L D1L S1L C2S L2S D2S S2S C5I
       L5I D5I S5I C5Q L5Q D5Q S5Q
R   20 C1C L1C D1C S1C C4A L4A D4A S4A C2C L2C D2C S2C C3I
       L3I D3I S3I C3Q L3Q D3Q S3Q
E   32 C1C L1C D1C S1C C1B L1B D1B S1B C7I L7I D7I S7I C7Q
       L7Q D7Q S7Q C5I L5I D5I S5I C5Q L5Q D5Q S5Q C6B L6B
       D6B S6B C6C L6C D6C S6C
J   44 C1C L1C D1C S1C C1L L1L D1L S1L C1S L1S D1S S1S C1Z
       L1Z D1Z S1Z C2S L2S D2S S2S C5I L5I D5I S5I C5Q L5Q
       D5Q S5Q C5D L5D D5D S5D C5P L5P D5P S5P C6S L6S D6S
       S6S C6E L6E D6E S6E
S    8 C1C L1C D1C S1C C5I L5I D5I S5I
C   32 C2I L2I D2I S2I C1D L1D D1D S1D C1P L1P D1P S1P C7I
       L7I D7I S7I C7D L7D D7D S7D C5D L5D D5D S5D C5P L5P
       D5P S5P C6I L6I D6I S6I
I    4 C5A L5A D5A S5A
```

of Satellites Tracked

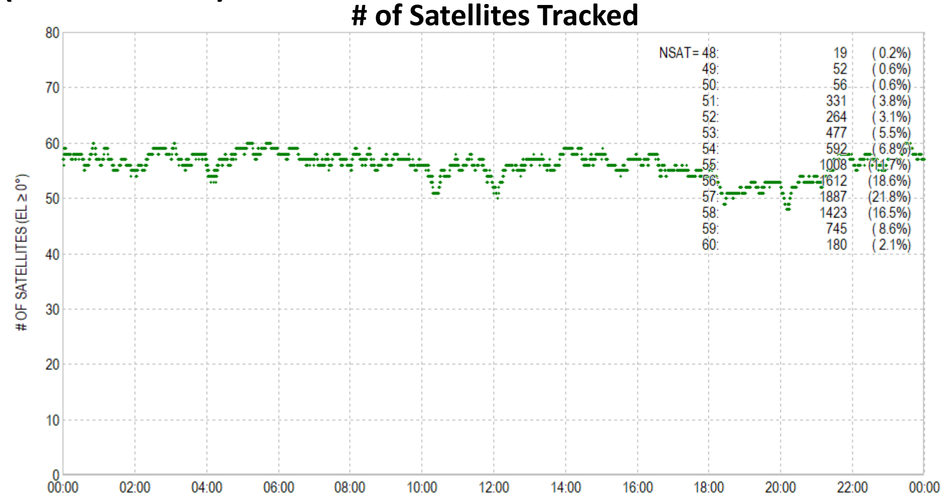


mosaic-X5 (FW 4.14.10.1) *1

Signals Tracked (RINEX)

```
G   20 C1C L1C D1C S1C C1W L1W D1W S1W C2W L2W D2W S2W C2L
       L2L D2L S2L C5Q L5Q D5Q S5Q
R   16 C1C L1C D1C S1C C2P L2P D2P S2P C2C L2C D2C S2C C3Q
       L3Q D3Q S3Q
E   20 C1C L1C D1C S1C C7Q L7Q D7Q S7Q C5Q L5Q D5Q S5Q C6C
       L6C D6C S6C C8Q L8Q D8Q S8Q
J   12 C1C L1C D1C S1C C2L L2L D2L S2L C5Q L5Q D5Q S5Q
C   24 C2I L2I D2I S2I C1P L1P D1P S1P C7I L7I D7I S7I C7D
       L7D D7D S7D C5P L5P D5P S5P C6I L6I D6I S6I
I    4 C5A L5A D5A S5A
```

of Satellites Tracked



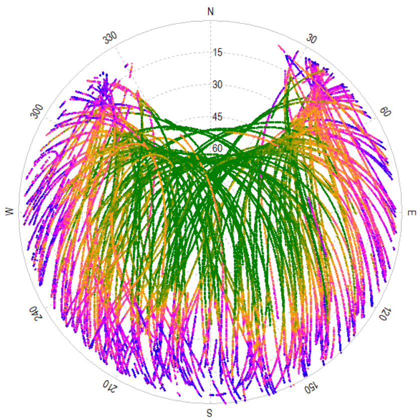
2025-03-08 0:00:00 -23:59:59 GPST, RTCM3: Pocket SDR, SBF: mosaic-X5

*1 SBAS disabled

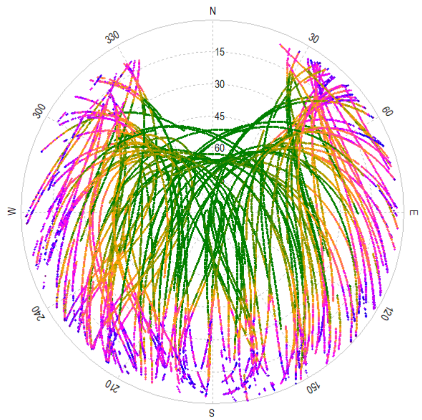
Signal C/N0

Pocket SDR ver.0.14

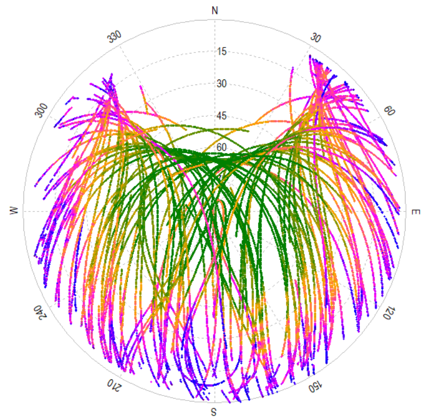
L1/G1/E1/B1



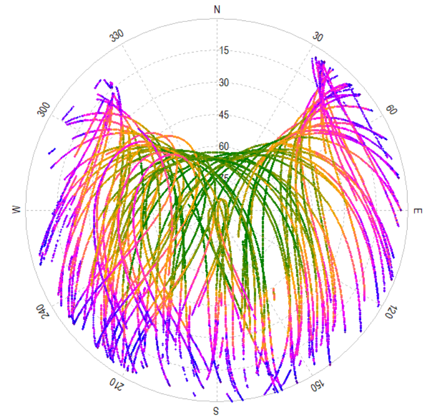
L2/G2/E5b/B2



L5/G3/E5a/B2a

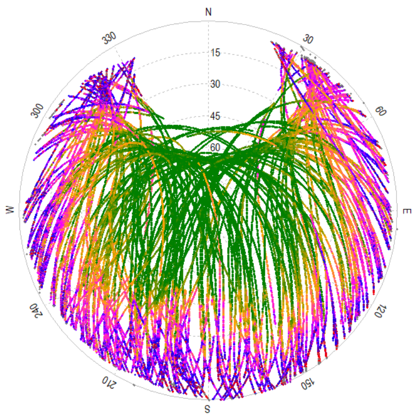


E6/L6/B3

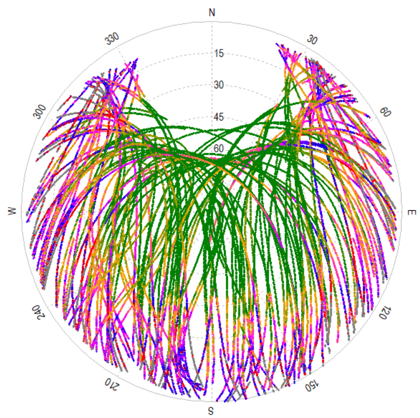


mosaic-X5 (FW 4.14.10.1)

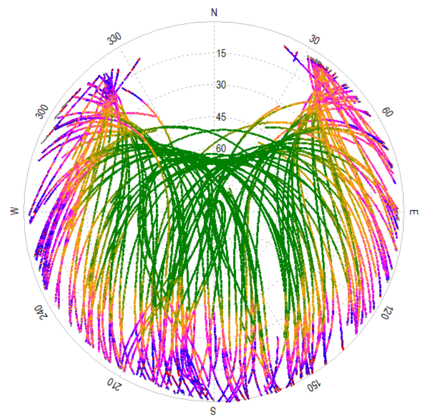
L1/G1/E1/B1



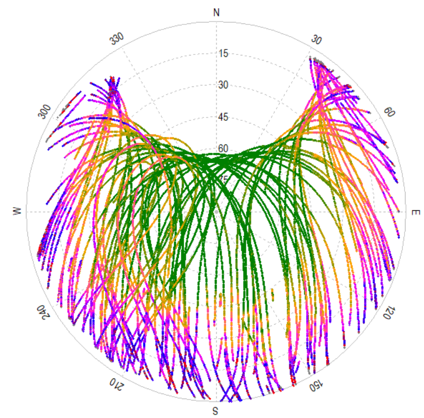
L2/G2/E5b/B2



L5/G3/E5a/B2a



E6/L6/B3



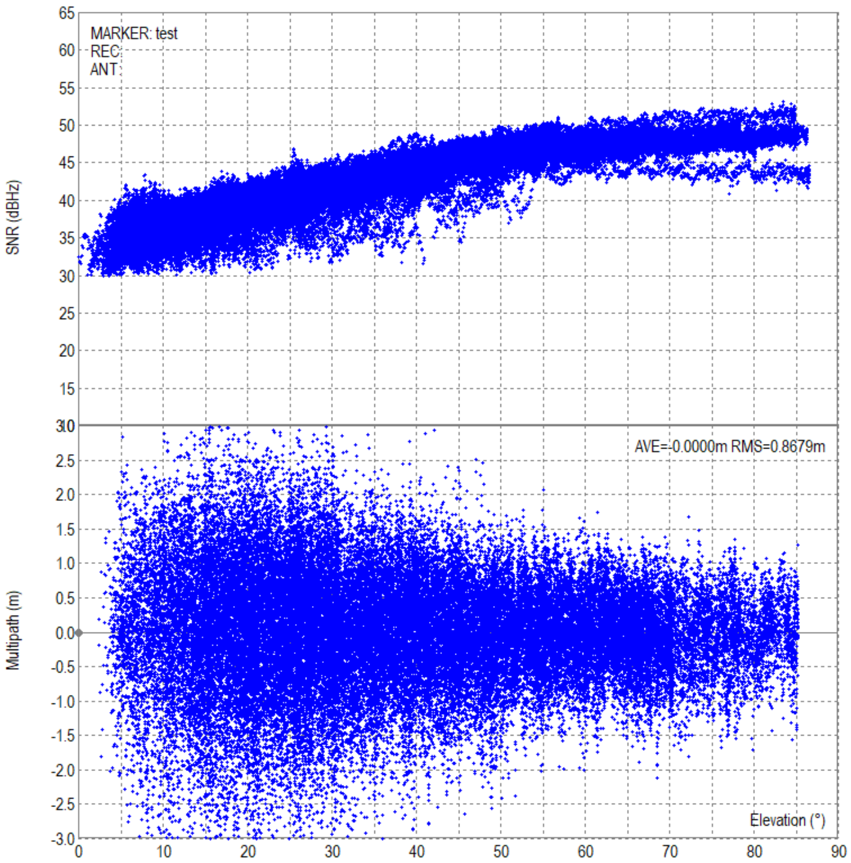
C/N0 =...45...40...35...30...25 <25 (dB-Hz)

2025-03-08 0:00:00 -23:59:59 GPST , RTCM3: Pocket SDR, SBF: mosaic-X5

Code Noise and Multipath

Pocket SDR ver.0.14

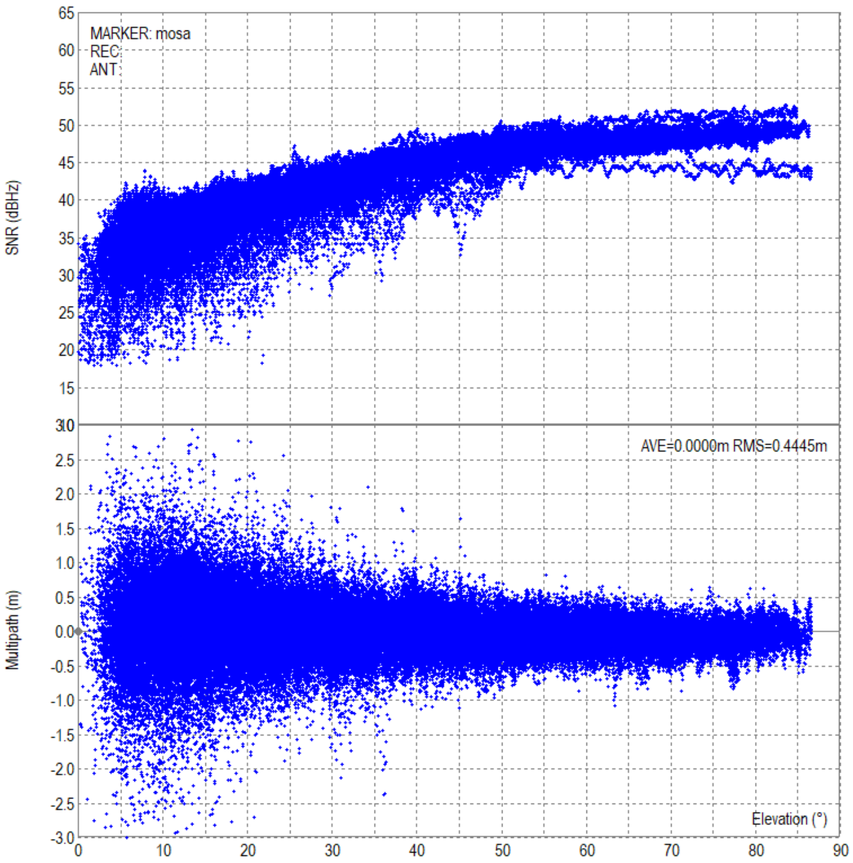
GPS L1C/A C/N0 and MP - Elevation



Correlator spacing = 0.1 chip,
DLL non-coherent integ. time = 20 ms

mosaic-X5 (FW 4.14.10.1)

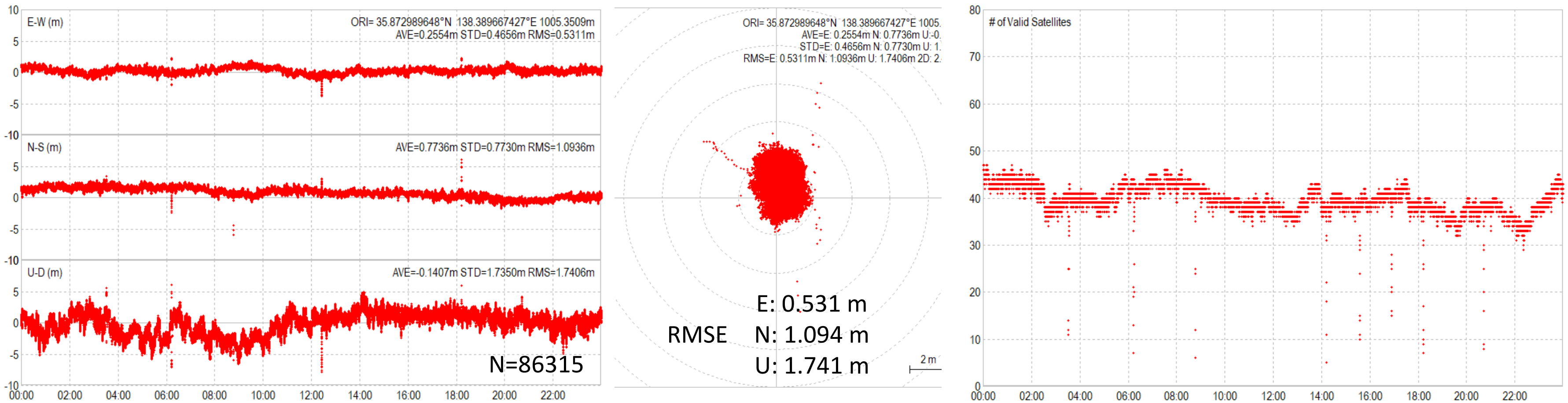
GPS L1C/A C/N0 and MP - Elevation



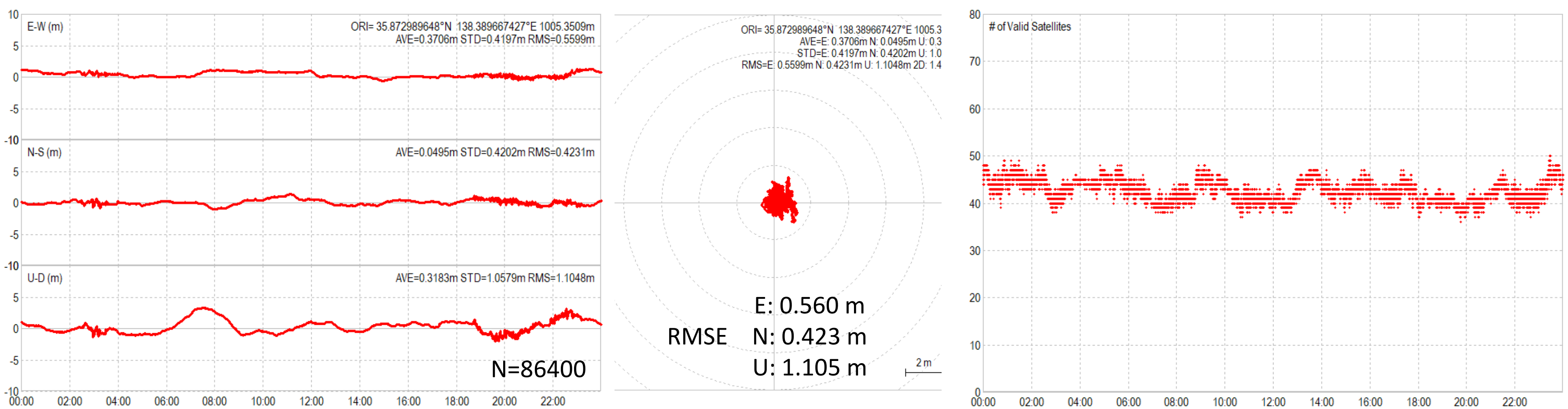
2025-03-08 0:00:00 -23:59:59 GPST, RTCM3: Pocket SDR, SBF: mosaic-X5

Positioning Solutions

Pocket SDR ver.0.14



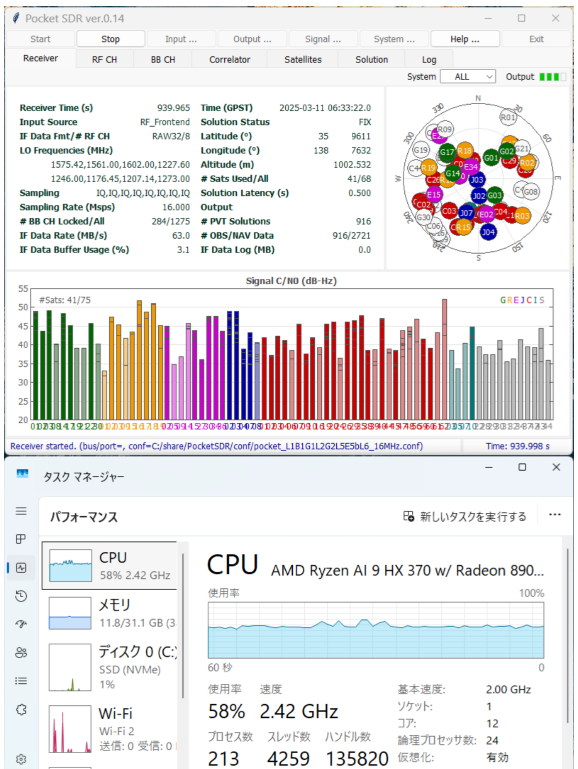
mosaic-X5 (FW 4.14.10.1)



2025-03-08 0:00:00 -23:59:59 GPST, NMEA GGA Solutions, 1 Hz

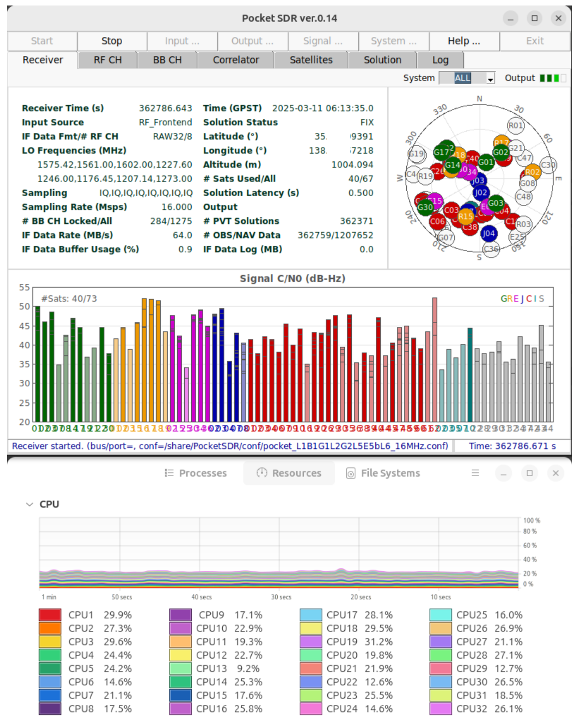
CPU Utilization

Laptop Windows PC
CPU Ryzen AI 9 HX370 (12C/24T),
RAM 32GB, SSD 1TB, Windows 11,
Fan mode: standard



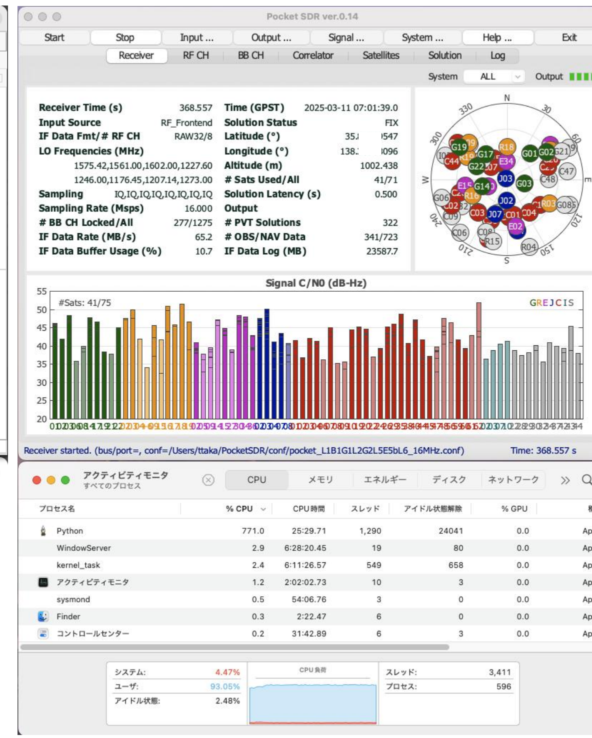
Total CPU Utilization 56-60 %
BB CH Locked/All: 284/1275 (L6D/E: 7CH)

Desktop Linux PC
CPU Ryzen 9 7945HX (16C/32T),
RAM 32GB, SSD 1TB, ubuntu 24.04 LTS,
CPB (core performance boost) disabled



Total CPU Utilization 23-25 %
BB CH Locked/All: 284/1275 (L6D/E: 8CH)

Mac mini
CPU Apple M2 (8C/8T),
RAM 16GB, SSD 512TB,
macOS Sequoia 15.3.1



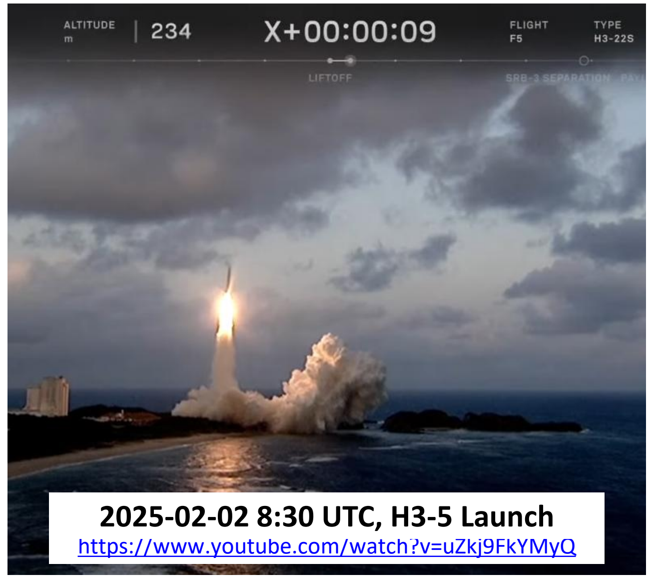
Total CPU Utilization 98-99 %
BB CH Locked/All: 277/1275 (L6D/E: 5CH)

Applications

QZS-6 First Light (1/4)

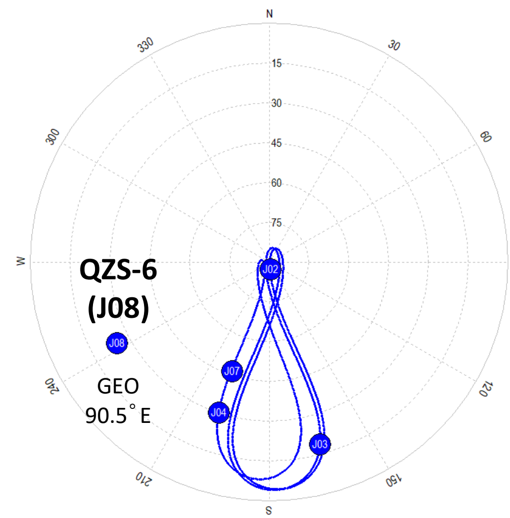
QZS-6

https://www.satnavi.jaxa.jp/ja/news/2024/11/29/10344/



2025-02-02 8:30 UTC, H3-5 Launch

https://www.youtube.com/watch?v=uZkj9FkYMyQ

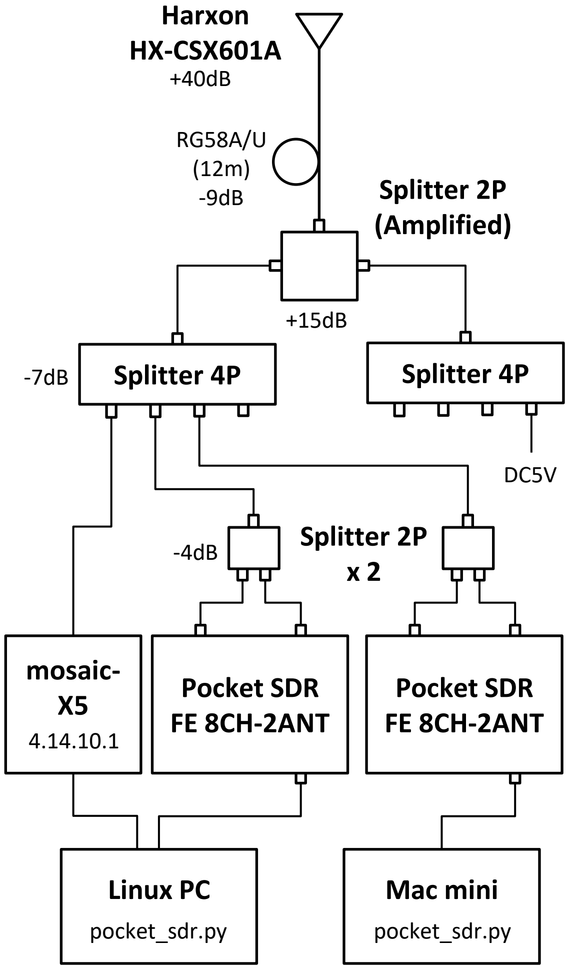


2025-03-05 00:00 UTC, Yamanashi, Japan

Satellite	Launch	Block	Orbit	SV No	SV ID	RINEX Sat ID	Signal and PRN Number										
							L1C/A	L1C	L1C/B	L1S	L1Sb	L2C	L5	L5S	L5SV	L6D	L6E
QZS-1	2010/9/11	I-Q	QZO	1	1	J01	193	193	-	183	-	193	193	-	-	193	(203)
QZS-2	2017/6/1	II-Q	QZO	2	2	J02	194	194	-	184	-	194	194	184	-	194	204
QZS-3	2017/8/19	II-G	GEO	3	7	J07	199	199	-	189	137	199	199	189	-	199	209
QZS-4	2017/10/9	II-Q	QZO	4	3	J03	195	195	-	185	-	195	195	185	-	195	205
QZS-1R	2021/10/26	IIA-Q	QZO	5	4	J04	196	196	(203)	186	-	196	196	(186)	186	196	206
QZS-5	JFY2025	III-Q	QZO	6	5	J05	(197)	197	204	-	-	-	197	-	-	197	207
QZS-6	2025/2/2	III-G	GEO	7	8	J08	(200)	200	205	-	129	-	200	-	205	200	210
QZS-7	JFY2025	III-G	QGEO	8	9	J09	(201)	201	206	-	-	-	201	-	206	201	211
Non-Standard	-	-	-	-	-	J06	(198)	(198)	(198)	-	-	-	(198)	-	-	(198)	(208)
Non-Standard	-	-	-	-	-	J10	202	202	202	-	-	-	202	-	-	202	212

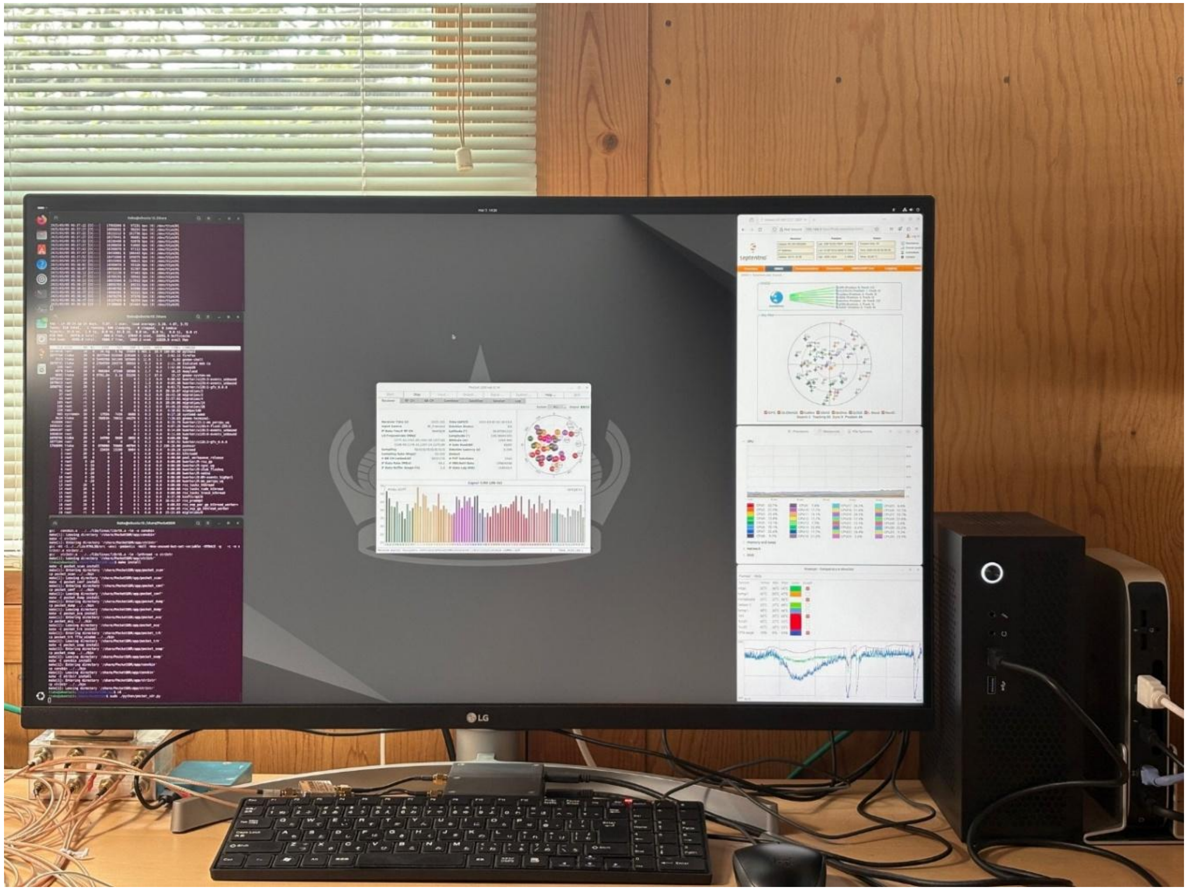
RINEX: RINEX 4.01, L5S: Normal mode, L5SV: Verification mode, (): Assigned but not used

QZS-6 First Light (2/4)



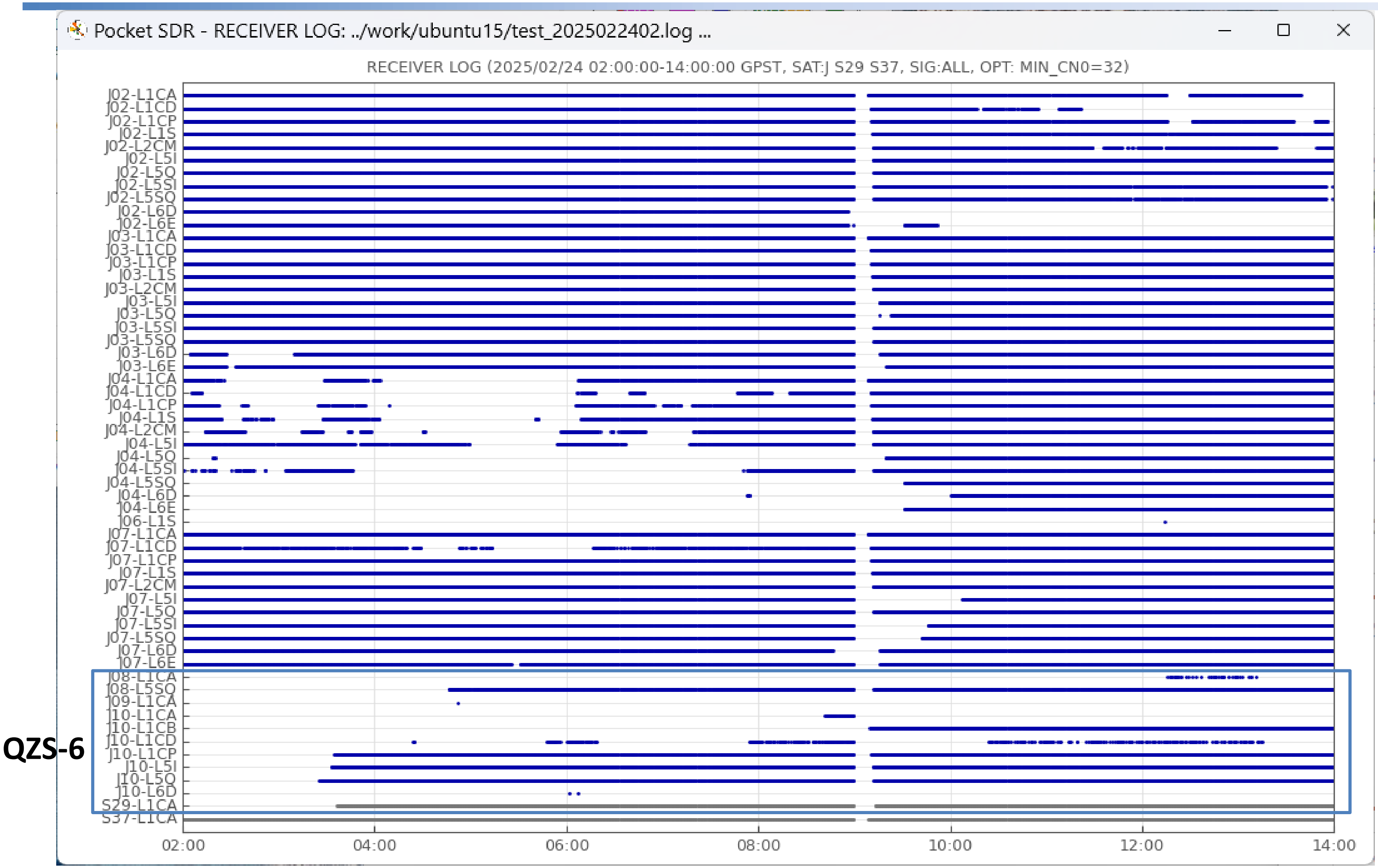
Ryzen 9 7945HX (16C/32T),
RAM 32GB, SSD 1TB,
Ubuntu 24.04 LTS

Apple M2 (8C/8T),
RAM 16GB, SSD 256GB,
macOS 15.3.1



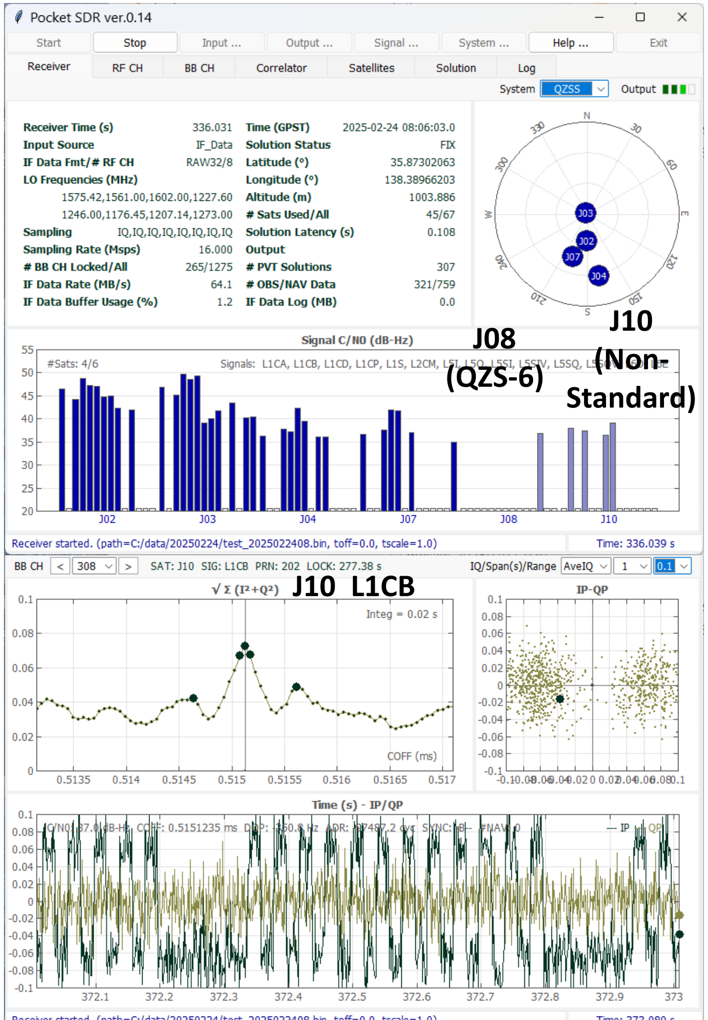
Pocket SDR FE 8CH-2ANT Configuration:
F_S = 16.000 (MHz), F_LO = 1575.42,1561,1602,1227.6,1246,1176.45,1207.14,1273 (MHz)
IQ = 2,2,2,2,2,2,2,2, BW_FILT = 16.4,16.4,8.7,16.4,8.7,16.4,16.4,16.4 (MHz)

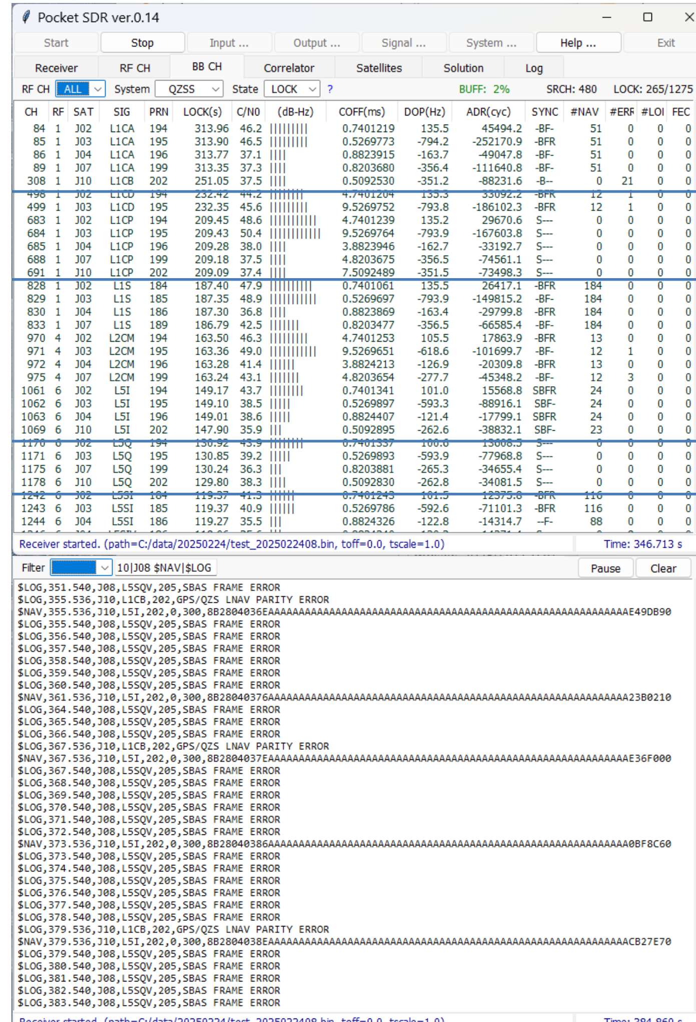
QZS-6 First Light (3/4)



QZS-6 First Light (4/4)

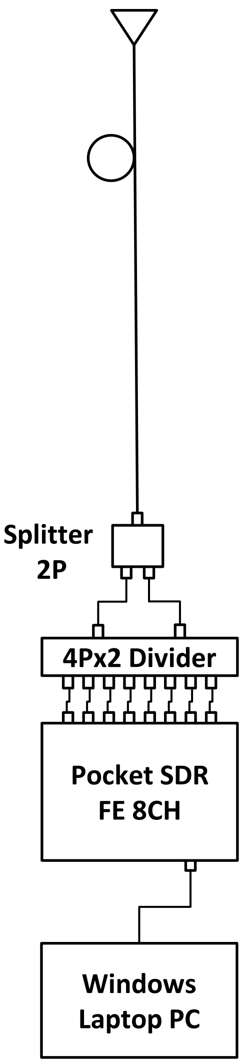
Recorded Raw IF Data (2025-02-24 08:06 - 08:19 GPST, 51.6GB)

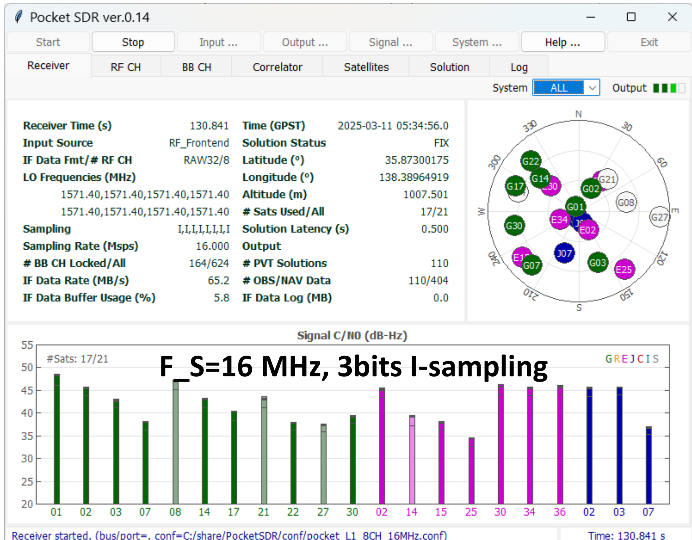


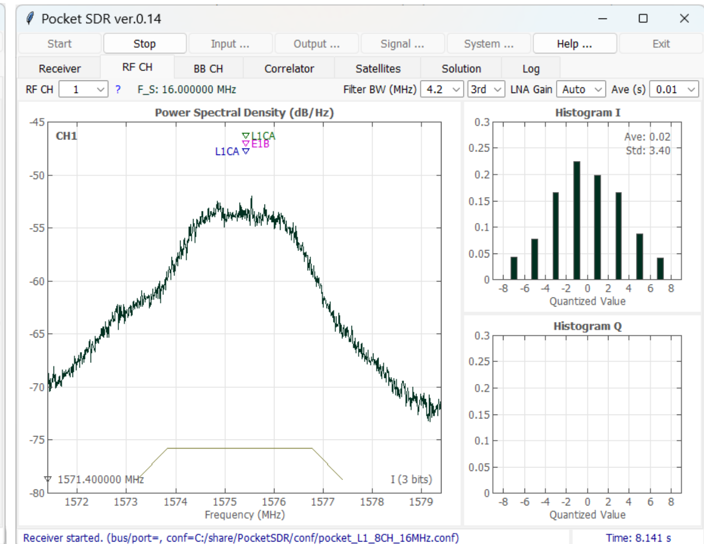


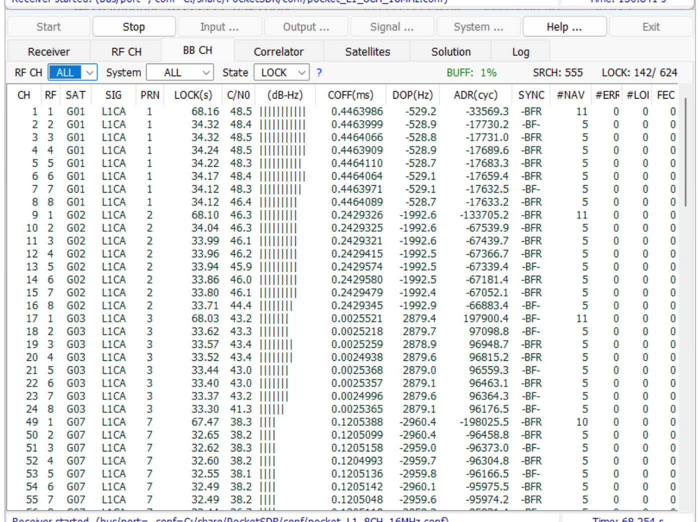
J10
PRN202
L1CB
L1CP
L5I
L5Q

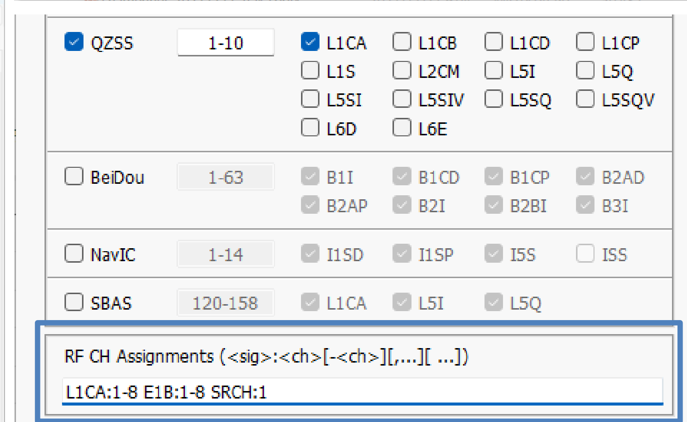
Antenna Array (1/5)







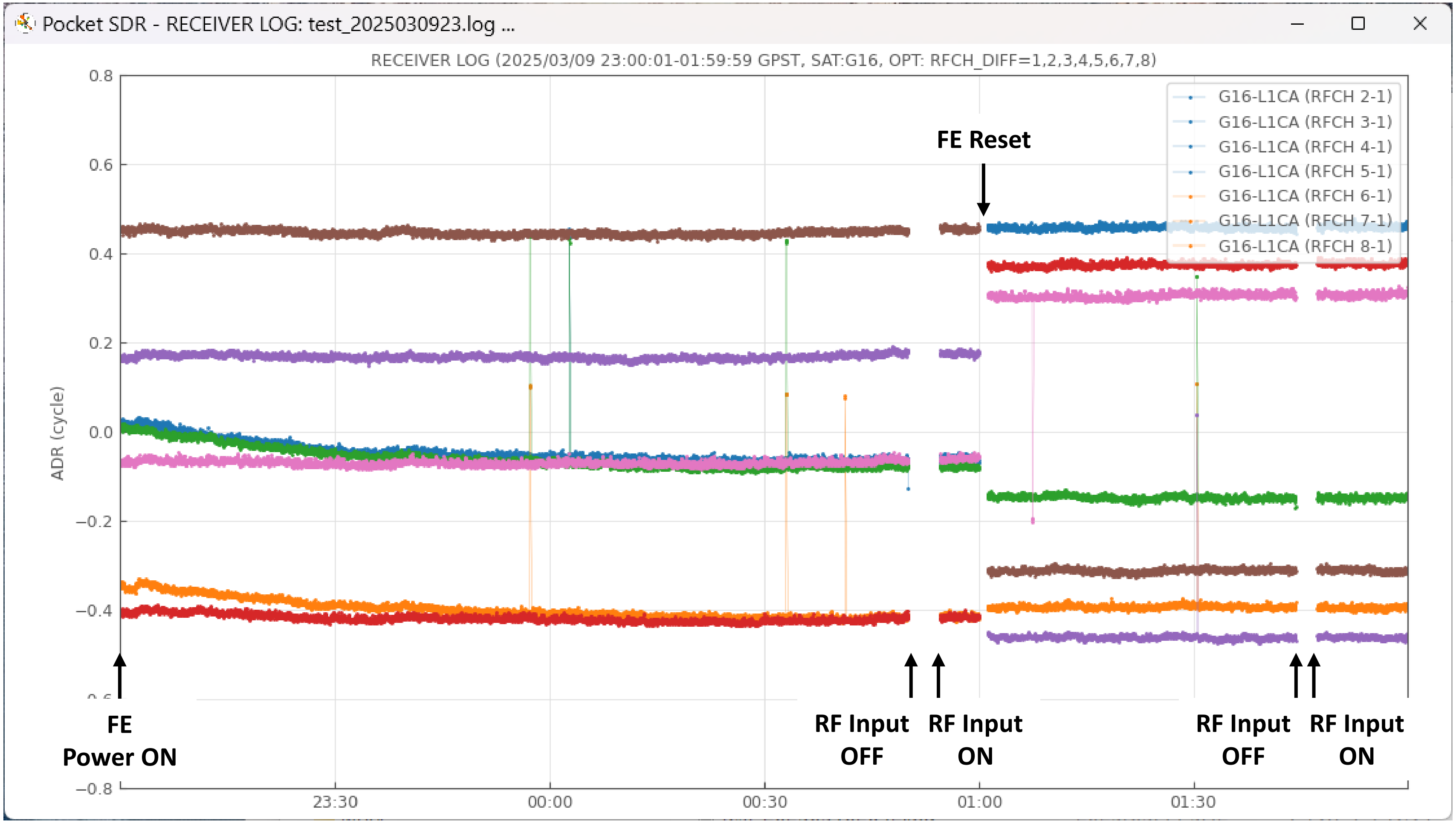




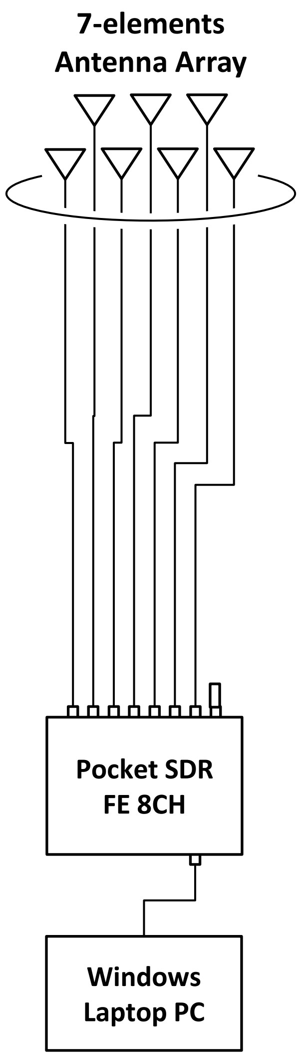
Signal RF CH Assignment Options

Antenna Array (2/5)

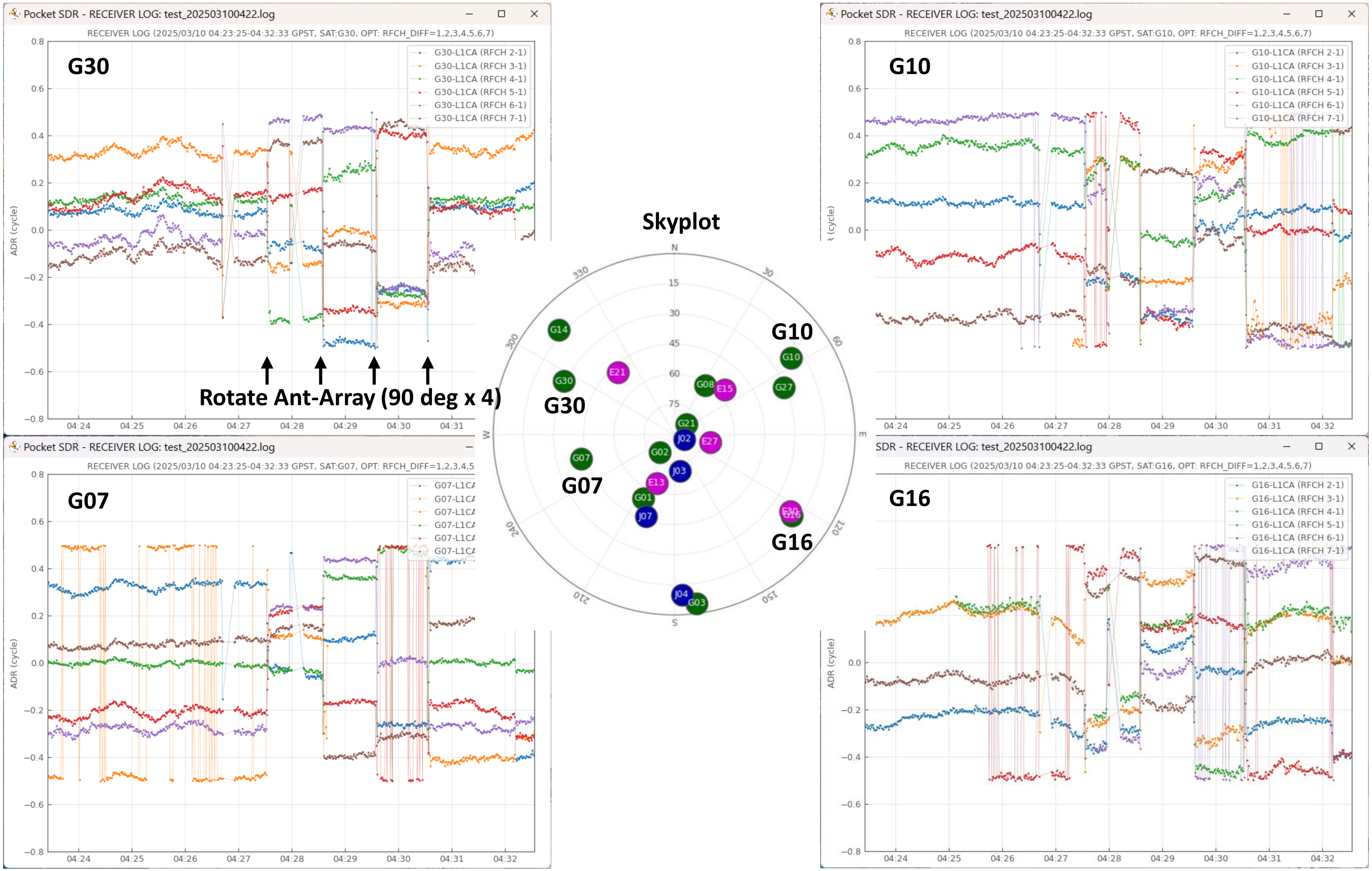
Inter-Channel Phase-Bias between RF CHs (SAT=G16, SIG=L1CA)



Antenna Array (3/5)

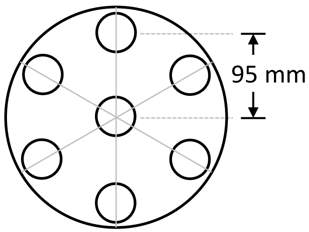


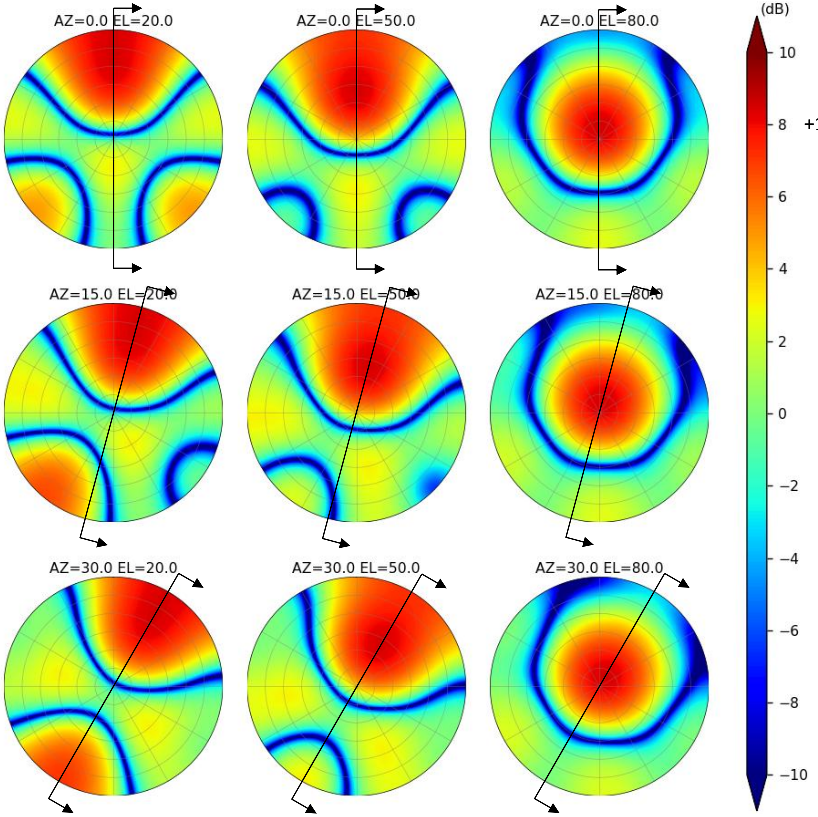
Antenna Array (4/5)

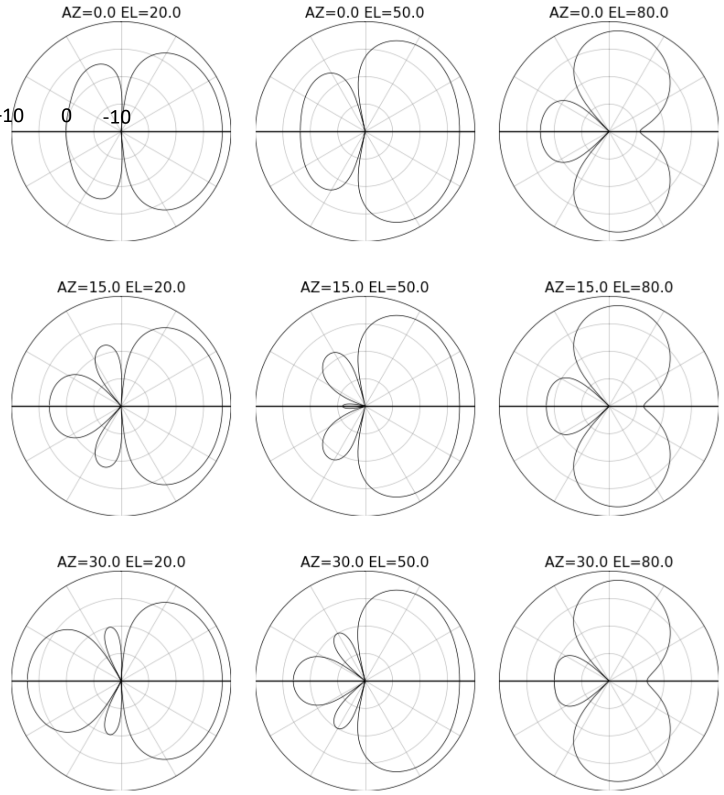


Antenna Array (5/5)

Array Gain Simulation: 7-elements Antenna Array (dB)







Appendix

Directory Structure of Package

PocketSDR (ver.0.14)

```
─┬─ bin          # Pocket SDR APs binary programs
 ├── app         # Pocket SDR APs source programs
 │   ├── pocket_conf  # Pocket SDR FE device configurator
 │   ├── pocket_dump  # Dump digital IF data of Pocket SDR FE device
 │   ├── pocket_scan  # Scan and list USB devices
 │   ├── pocket_acq   # GNSS signal acquisition
 │   ├── pocket_trk   # GNSS signal tracking and PVT generation
 │   ├── pocket_snap  # Snapshot Positioning
 │   └── convbin      # RINEX converter supporting Pocket SDR
 ├── src         # Pocket SDR library source programs
 ├── python      # Pocket SDR Python scripts
 ├── lib         # Libraries for APs and Python scripts
 │   ├── win32        # Libraries for Windows
 │   ├── macos        # Libraries for mac OS
 │   ├── linux        # Libraries for Linux or Raspberry Pi OS
 │   ├── build        # Makefiles to build libraries
 │   ├── cyusb        # Cypress EZ-USB API (CyAPI.a) and includes
 │   └── RTKLIB       # RTKLIB source programs based on 2.4.3 b34
 ├── conf        # Configuration files for Pocket SDR FE
 ├── FE_2,4,8CH  # Pocket SDR FE 2CH, 4CH, 8CH H/W and F/W
 ├── driver      # Driver installation instruction for Pocket SDR FE
 ├── doc         # Documents
 ├── image       # Image files for documents
 ├── sample      # Sample digital IF data captured by Pocket SDR FE
 └── test        # Test codes and data
```

Driver Installation for Pocket SDR FE

- **Windows**
  1. Get "EZ-USB FX3 Software Development Kit (SDK)" for Windows [1]. You need registration in the Infineon developer site to download the file.
  2. Execute the SDK installer (ezusbfx3sdk_1.3.5_Windows_x32-x64.exe). The SDK is installed to "C:¥Program Files (x86)¥Cypress¥EZ-USB FX3 SDK¥1.3" as default.
  3. If the error ".NET Framework 3.5 or later required" shown in the SDK installation, download .NET Framework 3.5 from Microsoft site [2] and install it.
  4. Attach a Pocket SDR FE 2CH or 4CH to the PC, open Device Manager, and look for the device "EZ-USB" (FE 2CH) or "FX3" (FE 4CH, FE 8CH).
  5. Select the device, right-click and execute "Update Driver", select "Browse my computer for drivers", and input the directory "driver" in SDK with "Include subfolders" checked.
  6. If the driver properly installed, the device could be recognized as "Cypress FX2LP Sample Device" (FE 2CH) or "Cypress FX3 USB StreamerExample Device" (FE 4CH, FE 8CH).
- **Linux, Raspberry Pi OS or macOS**
  1. No need to install the Cypress driver for EZ-USB FX2LP or FX3.
  2. Pocket SDR utilizes instead a cross-platform USB host driver libusb-1.0 [3] as a shared library for these OS.
  3. Refer "Installation for Linux or Raspberry Pi OS" or "Installation for macOS" [4]. The installation procedure also includes how to install the libusb-1.0 package.

[1] https://www.infineon.com/cms/en/design-support/tools/sdk/usb-controllers-sdk/ez-usb-fx3-software-development-kit/
[2] https://dotnet.microsoft.com/en-us/download/dotnet-framework/thank-you/net35-sp1-web-installer,
[3] https://libusb.info/, [4] https://github.com/tomojitakasu/PocketSDR

pocket_conf (1/2)

Synopsis

```
pocket_conf [-s] [-a] [-h] [-p bus[,port]] [conf_file]
```

Description

Configure or show settings for a Pocket SDR FE device. If conf_file specified, the settings in the configuration file are set to the Pocket SDR FE device registers. The configuration is a text file containing records of MAX2771 register field settings as like follows. The register field settings are written as keyword = value format or hexadecimal format. In the case of keyword = value format, a keyword is a field name shown in MAX2771 manual [1]. Strings after # in a line is treated as comments. If conf_file omitted, the command shows the settings of the Pocket SDR FE device in the same format of the configuration file.

Options

```
-s            Save the settings to EEPROM of the SDR device. These settings are also loaded at reset
              of the Pocket SDR FE device.
-a            Show all of the register fields.
-h            Configure or show registers in a hexadecimal format.
-p [bus[,port]] USB bus and port number of the Pocket SDR FE device. Without the option, the
              command selects the device firstly found.
conf_file     Path of the configuration file. Without the option, the command shows current register
              field settings of the Pocket SDR FE device.
```

pocket_conf (2/2)

Configuration Example

```
#
#  Pocket SDR device settings (MAX2771)
#
#  [CH1] F_LO = 1575.420 MHz, F_ADC = 24.000 MHz (IQ), F_FILT =  0.0 MHz, BW_FILT =  4.2 MHz
#  [CH2] F_LO = 1176.450 MHz, F_ADC =  0.000 MHz (IQ), F_FILT =  0.0 MHz, BW_FILT = 16.4 MHz

[CH1]
CHIPEN          =      1  # Chip enable (0:disable,1:enable)
IDLE            =      0  # Idle enable (0:operating-mode,1:idle-mode)
MIXPOLE         =      0  # Mixer pole selection (0:13MHz,1:36MHz)
LNAMODE         =      0  # LNA mode selection (0:high-band,1:low-band,2:disable)
MIXERMODE       =      0  # Mixer mode selection (0:high-band,1:low-band,2:disable)
FCEN            =      0  # IF filter center frequency: (128-FCEN)/2*{0.195|0.66|0.355} MHz
FBW             =      2  # IF filter BW (0:2.5MHz,1:8.7MHz,2:4.2MHz,3:23.4MHz,4:36MHz,
                          # 7:16.4MHz)
F3OR5           =      1  # Filter order selection (0:5th,1:3rd)
FCENX           =      0  # Polyphase filter selection (0:lowpass,1:bandpass)
FGAIN           =      1  # IF filter gain setting (0:-6dB,1:normal)
ANAIMON         =      0  # Enable continuous spectrum monitoring (0:disable,1:enable)
IQEN            =      1  # I and Q channel enable (0:I-CH-only,1:I/Q-CH)
GAINREF         =    170  # AGC gain reference value (0-4095)
SPI_SDIO_CONFIG =      0  # SPI SDIO pin config (0:none,1:pull-down,2:pull-up,3:bus-hold)
AGCMODE         =      0  # AGC mode control (0:independent-I/Q,2:gain-set-by-GAININ)
FORMAT          =      1  # Output data format (0:unsigned,1:sign-magnitude,
                          # 2:2's-complement)
BITS            =      2  # Number of bits in ADC (0:1bit,2:2bit,4:3bit)
DRVCFG          =      0  # Output driver config (0:CMOS-logic,2:analog)
DIEID           =      0  # Identifiers version of IC
GAININ          =     58  # PGA gain value programming in steps of approx 1dB per LSB (0-63)
HILODEN         =      0  # Enable output driver to drive high loads (0:disable,1:enable)
FHIPEN          =      0  # Enable highpass coupling between filter and PGA
                          # (0:disable,1:enable)
PGAIEN          =      1  # I-CH PGA enable (0:disable,1:enable)
PGAQEN          =      1  # Q-CH PGA enable (0:disable,1:enable)
STRMEN          =      0  # Enable DSP interface (0:disable,1:enable)
STRMSTART       =      0  # Enable data streaming (rising edge)
STRMSTOP        =      0  # Disable data streaming (rising edge)
STRMBITS        =      1  # Number of bits streamed (1:IMSB/ILSB,3:IMSB/ILSB/QMSB/QLSB)
STAMPEN         =      0  # Enable insersion of frame numbers (0:disable,1:enable)
TIMESYNCEN      =      0  # Enable output of time sync pulse when streaming enabled by
                          # STRMEN
DATASYNCEN      =      0  # Enable sync pulse at DATASYNC
STRMRST         =      0  # Reset all counters
LOBAND          =      0  # Local oscillator band selection (0:L1,1:L2/L5)
REFOUTEN        =      1  # Output clock buffer enable (0:disable,1:enable)
IXTAL           =      1  # Current programing for XTAL (1:normal,3:high-current)
ICP             =      0  # Charge pump current selection (0:0.5mA,1:1mA)
INT_PLL         =      1  # PLL mode control (0:fractional-N,1:integer-N)
PWRSAV          =      0  # Enable PLL power-save mode (0:disable,1:enable)
NDIV            =  26257  # PLL integer division ratio (36-32767):
                          # F_LO=F_XTAL/RDIV*(NDIV+FDIV/2^20)
RDIV            =    400  # PLL reference division ratio (1-1023)
FDIV            =      0  # PLL fractional division ratio (0-1048575)
EXTADCCLK       =      0  # External ADC clock selection (0:internal,1:ADC_CLKIN)
PREFRACDIV_SEL  =      0  # Clock pre-divider selection (0:bypass,1:enable)
REFCLK_L_CNT    =      0  # Clock pre-divider L counter value (0-4095):
                          # L_CNT/(4096-M_CNT+L_CNT)
REFCLK_M_CNT    =      0  # Clock pre-divider M counter value (0-4095)
ADCCLK          =      0  # Integer clock div/mul selection (0:enable,1:bypass)
REFDIV          =      3  # Integer clock div/mul ratio (0:x2,1:1/4,2:1/2,3:x1,4:x4)
FCLKIN          =      0  # ADC clock divider selection (0:bypass,1:enable)
ADCCLK_L_CNT    =      0  # ADC clock divider L counter value (0-4095):
                          # L_CNT/(4096-M_CNT+L_CNT)
ADCCLK_M_CNT    =      0  # ADC clock divider M counter value (0-4095)
CLKOUT_SEL      =      1  # CLKOUT selection (0:integer-clock-div/mul,1:ADC-clock)
MODE            =      0  # DSP interface mode selection

[CH2]
...
```

The keywords are based on bitfield names of MAX2771 registers [1]
GRAY parameters are only shown by pocket_conf with -a option
Blue parameters are effective only for CH1 as the clock master

[1] ADI/maxim integrated, MAX2771 Multiband Universal GNSS Receiver Data Sheet, 2018 p.31 ~ p.45

pocket_dump (1/1)

```
Synopsis
  pocket_dump [-t tsec] [-r] [-p bus[,port]] [-c conf_file] [-q] [file [file ...]]

Description
  Capture and dump digital IF (DIF) data of a Pocket SDR FE device to output files. To stop
  capturing, press Ctr-C.

Options

  -t tsec     Data capturing time in seconds.
  -r          Dump raw data of the Pocket SDR FE device without channel separation and quantization.
  -p bus[,port] USB bus and port number of the Pocket SDR FE device. Without the option, the command
              selects the device firstly found.
  -c conf_file Configure the Pocket SDR FE device with a device configuration file before capturing.
  -q          Suppress showing data dump status.
  [file [file ...]] Output digital IF data file paths. The first path is for CH1, the second one is for
              CH2 and so on. The second one or the later can be omitted. With option -r, only the
              first path is used. If the file path is "", data are not output to anywhere. If the
              file path is "-", data are output to stdout. If all of the file paths omitted, the
              following default file paths are used.
              CH1: ch1_YYYYMMDD_hhmmss.bin
              CH2: ch1_YYYYMMDD_hhmmss.bin
              ...
              (YYYYMMDD: dump start date in UTC, hhmmss: dump start time in UTC)
```

pocket_acq (1/1)

Synopsis

```
  pocket_acq [-sig sig] [-prn prn[,...]] [-tint tint] [-toff toff] [-f freq] [-fi freq] [-d freq]
             [-nz] file
```

Description

```
  Search GNSS signals in digital IF data and plot signal search results.If single PRN number by -prn
  option, it plots correlation power and correlation shape of the specified GNSS signal. If multiple
  PRN numbers specified by -prn option, it plots C/N0 for each PRN.
```

Options ([]: default)

```
  -sig sig    GNSS signal type ID (L1CA, L2CM, ...). See below for details. [L1CA]
  -prn prn[,...] PRN numbers of the GNSS signal separated by ','. A PRN number can be a PRN number
              rangelike 1-32 with start and end PRN numbers. For GLONASS FDMA signals (G1CA, G2CA),
              the PRN number is treated as FCN (frequency channel number). [1]
  -tint tint  Integration time in ms to search GNSS signals. [code cycle]
  -toff toff  Time offset from the start of digital IF data in ms. [0.0]
  -f freq     Sampling frequency of digital IF data in MHz. [12.0]
  -fi freq    IF frequency of digital IF data in MHz. The IF frequency equals 0, the IF data is
              treated as IQ-sampling (zero-IF). [0.0]
  -d freq[,freq] Reference and max Doppler frequency to search the signal in Hz. [0.0,5000.0]
  -nz         Disalbe zero-padding for circular colleration to search the signal. [enabled]
  -h          Show usage and signal type IDs
  file        File path of the input digital IF data. The format should be a series of int8_t (signed
              byte) for real-sampling (I-sampling) or interleaved int8_t for complex-sampling (IQ-
              sampling). PocketSDR and AP pocket_dump can be used to capture such digital IF data.
```

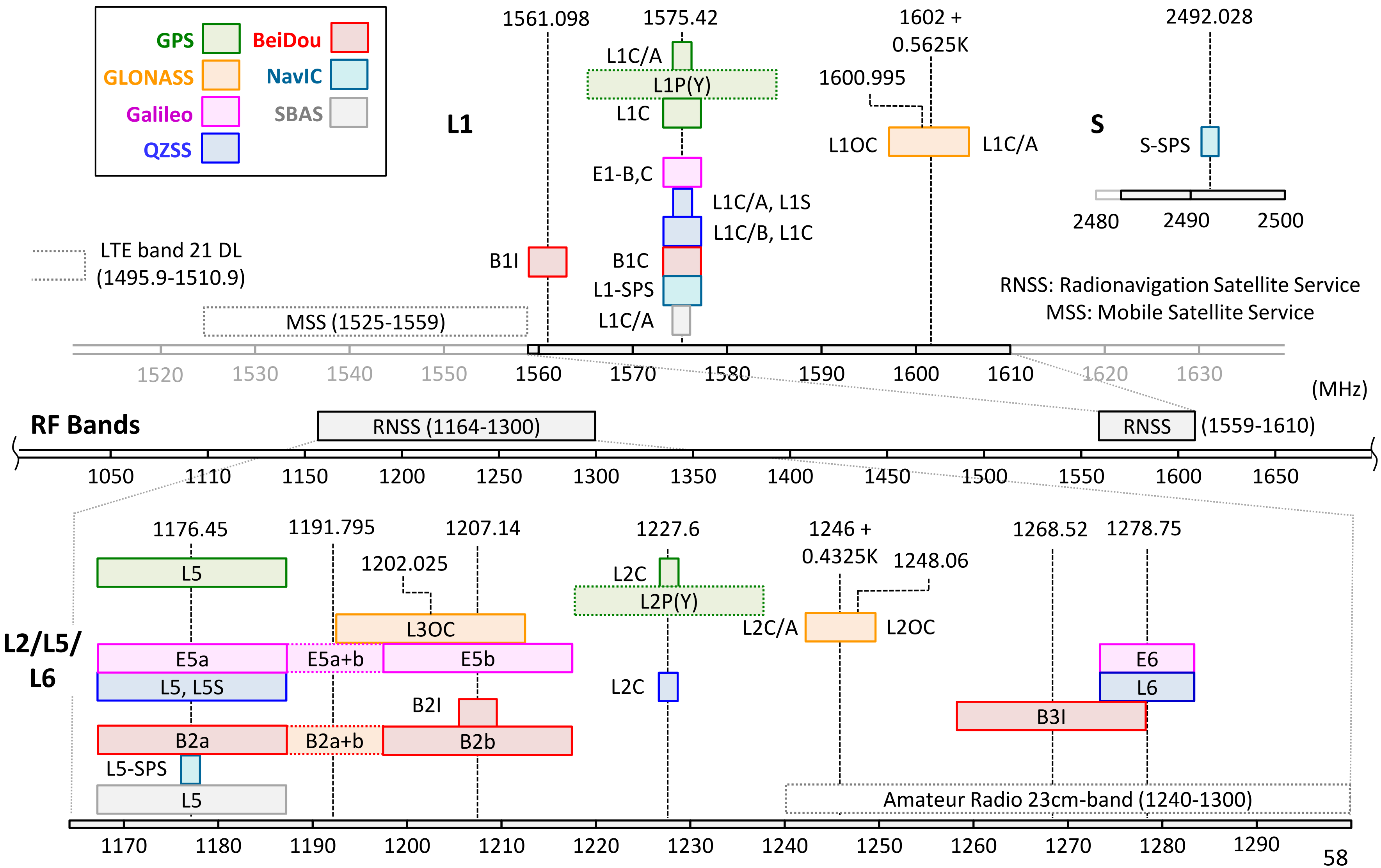
pocket_trk (1/2)

```
Synopsis
  pocket_trk [-sig sig -prn prn[,...] ...] [-fmt {INT8|INT8X2|RAW8|RAW16}] [-f freq] [-fo freq[,...]]
         [-IQ {1|2}[,...]] [-toff toff] [-ti tint] [-p bus,[,port] [-c conf_file] [-log path]
         [-nmea path] [-rtcm path] [-raw path] [-w file] [file]

Description
  It searches and tracks GNSS signals in the input digital IF data, extract observation data, decode
  navigation data and generate PVT solutions. The observation and navigation data can be output as a
  RTCM3 stream. The PVT solutions can be output as a NMEA stream. The observation data and raw
  navigation data and some event logs can be output as a log stream.

Options ([]: default)
  -sig sig [-fi freq] -prn prn[,...] ...  A GNSS signal type ID (L1CA, L2CM, ...) and a PRN number list
          of the signal. For signal type IDs, refer pocket_acq.py manual. The PRN number list shall
          be PRN numbers or PRN number ranges like 1-32 with the start and the end numbers. They are
          separated by ",". For GLONASS FDMA signals (G1CA, G2CA), the PRN number is treated as the
          FCN (frequency channel number). The pair of a signal type ID and a PRN number list can be
          repeated for multiple GNSS signals to be tracked.
  -fmt {INT8|INT8X2|RAW8|RAW16}  Specify IF data format as follows:
          INT8 = int8 (I-sampling), INT8X2 = interleaved int8 (IQ-sampling), RAW8 = Pocket SDR FE 2CH
          raw (packed 8 bits), RAW16 = Pocket SDR FE 4CH raw (packed 16 bits) [INT8X2]
  -f freq   Specify the sampling frequency of the IF data in MHz. [12.0]
  -fo freq[,...] Specify LO frequency for each RF channel in MHz. In case of the IF data format as RAW8
          or RAW16, multiple (2 or 4) frequencies have to be specified separated by ",". If the LO
          frequency is specified as 0, the IF frequency is assumed as 0 for IQ-sampling and 1/2 of the
          sampling frequency for I-sampling. [0,0,0,0]
  -IQ {1|2}[,...] Specify the sampling type (1 = I-sampline, 2 = IQ-sampling) for each RF channel
          separated by "," in case of the IF data foramt as RAW8 or RAW16. [2,2,2,2]
```

pocket_trk (2/2)

```
Options (cont.)
  -toff       toff Time offset from the start of the IF data in s. [0.0]
  -tscale     scale Time scale to replay the IF data file. [1.0]
  -ti tint    Update interval of the signal tracking status in seconds. If 0 specified, the signal
              tracking status is suppressed. [0.1]
  -p bus[,port] USB bus and port number of the Pocket SDR FE device in case of IF data input from
              the device.
  -c conf_file Configure the Pocket SDR FE device with a device configuration file before signal
              acquisition and tracking.
  -log path   A stream path to write the signal tracking log. The log includes observation data,
              navigation data, PVT solutions and some event logs. The stream path should be one of the
              followings.
              (1) local file file path without ':'. The file path can be contain time keywords (%Y,
                  %m, %d, %h, %M) as same as the RTKLIB stream.
              (2) TCP server  :port
              (3) TCP client  address:port
  -nmea path  A stream path to write PVT solutions as NMEA GNRMC, GNGGA and GNGSV sentences.
  -rtcm path  A stream path to write raw observation and navigation data as RTCM3.3 messages.
  -raw path   A stream path to write raw IF data.
  -w file     Specify the FFTW wisdowm file. [../python/fftw_wisdom.txt]
  [file]      A file path of the input IF data. The Pocket SDR FE deveice and pocket_dump can be used
              to capture such digitized IF data. If the tag file <file>.tag of the input IF data exists,
              the format, the sampling frequency, the LO frequencies and the sampling types are
              automatically recognized by the tag file and the options -fmt, -f, -fo, and -IQ are
              ignored. If the file path omitted, the input is taken from a Pocket SDR FE device directly.
              In this case, the sampling frequency, the sampling types of IF data, the RF channel
              assignments and the IF frequencies for each signals are automatically configured according
              to the device information.
```

GNSS Signal Bands



GNSS Signals (1/3)

Military or Restricted Signal

System	Carrier Freq. (MHz)	Signal	I/Q	Min Rec. Power (dBW)	Modulation	Primary Code			Secondary Code		Navigation Data				Notes	Pocket SDR Signal ID
						Length (chip)	Chip Rate (Mcps)	Cycle (ms)	Length (chip)	Cycle (ms)	Data	Symbol Rate (sps)	Data Rate (bps)	FEC		
GPS [1][2][3]	1575.42	L1C/A	Q	-158.5	BPSK(1)	1023	1.023	1	-	-	LNAV	50	50	-		L1CA
		L1P(Y)*1	I	-161.5	BPSK(10)	1week	10.23	1week	-	-	LNAV	50	50	-		-
		L1M	I	?	BOC(10,5)	?	5.115	?	?	?	?	?	?	?	Block IIR-M~	-
		L1C-D	I	-163.0	BOC(1,1)	10230	1.023	10	-	-	CNAV-2	100	50	BCH,LDPC	GPS III~	L1CD
		L1C-P	I	-158.25	TMBOC(6,1,4/33)	10230	1.023	10	1800	18000	-	-	-	-		L1CP
	1227.6	L2C/A	Q	-164.5	BPSK(1)	1023	1.023	1	-	-	LNAV	50	50	-	Block IIR-M~	-
		L2P(Y)*1	I	-164.5/-161.5	BPSK(10)	1week	10.23	1week	-	1week	LNAV	50	50	-		-
		L2M	I	?	BOC(10,5)	?	5.115	?	?	?	?	?	?	?	Block IIR-M~	-
		L2C-M	Q/I	-160.0/ -158.5	BPSK(1)+TDM	10230	0.5115	20	-	-	LNAV	50	50	-	Block IIR-M~	-
											CNAV	50	25	1/2 *6		L2CM
		L2C-L			BPSK(1)+TDM	767250	0.5115	1500	-	-	-	-	-	-		(L2CL)
	1176.45	L5-I	I	-157.9/-157.0	BPSK(10)	10230	10.23	1	10 (NH)	10	CNAV	100	50	1/2 *6	Block IIF~	L5I
		L5-Q	Q	-157.9/-157.0	BPSK(10)	10230	10.23	1	20 (NH)	20	-	-	-	-		L5Q
GLONASS [4][5][6][7]	1602.0 + 0.5625K*2	L1C/A*4	I	-161.0	BPSK(0.5)	511	0.511	1	- / 2*3	- / 2*3	GLO-STR	100	50	-		G1CA
		L1P	Q	?	BPSK(5)	5110000	5.11	1000	-	-	?	?	?	-		-
	1600.995	L1OCd	Q	?	BPSK(1)+TDM	1023	0.5115	2	2 (MC)	4	GLO-STR	250	125	1/2 *6	GLO-K2~	G1OCD
		L1OCp	Q	?	BOC(1,1)+TDM	4092	0.5115	8	-	-	-	-	-	-		G1OCP
		L1SC	I	?	?	?	?	?	?	?	?	?	?	?		-
	1246.0 + 0.4375K*2	L2C/A*5	I	-167.0	BPSK(0.5)	511	0.511	1	- / 2*3	- / 2*3	GLO-STR	100	50	-		G2CA
		L2P	Q	?	BPSK(5)	5110000	5.11	1000	-	-	?	?	?	-		-
	1248.06	L2CSI	Q	?	BPSK(1)+TDM	?	0.5115	?	?	?	?	?	?	?	GLO-K2~	-
		L2OCp	Q	?	BOC(1,1)+TDM	10230	0.5115	20	50	1000	-	-	-	-		G2OCP
		L2SC	I	?	?	?	?	?	?	?	?	?	?	?		-
	1202.025	L3OCd	I	?	BPSK(10)	10230	10.23	1	5 (BC)	5	GLO-STR	200	100	1/2 *6	GLO-K1~	G3OCD
		L3OCp	Q	?	BPSK(10)	10230	10.23	1	10 (NH)	10	-	-	-	-		G3OCP
Galileo[8]	1575.42	E1-A	Q	?	BOC(15,2.5)	?	2.5575	?	?	?	G/NAV	?	?	?	PRS	-
		E1-B	I	-157.0	CBOC(6,1,1/11)	4092	1.023	4	-	-	I/NAV	250	125	1/2 *6	OS, SoL, CS	E1B
		E1-C	I		CBOC(6,1,1/11)	4092	1.023	4	25	100	-	-	-	-		E1C
	1176.45	E5a-I	I	-155.0	BPSK(10)	10230	10.23	1	20	20	F/NAV	50	25	1/2 *6	OS, CS	E5AI
		E5a-Q	Q		BPSK(10)	10230	10.23	1	100	100	-	-	-	-		E5AQ
	1207.14	E5b-I	I	-155.0	BPSK(10)	10230	10.23	1	4	4	I/NAV	250	125	1/2 *6	OS, SoL, CS	E5BI
		E5b-Q	Q		BPSK(10)	10230	10.23	1	100	100	-	-	-	-		E5BQ

*1 AS ON, *2 K = {-7, ..., +6}, *3 Odd FCN, *4 L1OF, *5 L2OF, *6 Convolutional Code (R=1/2, K=7), NH: Neuman Hoffman Code, MC: Manchester Code, BC: Barker Code

GNSS Signals (2/3)

Military or Restricted Signal

System	Carrier Freq. (MHz)	Signal	I/Q	Min Rec. Power (dBW)	Modulation	Primary Code Length (chip)	Primary Code Chip Rate (Mcps)	Primary Code Cycle (ms)	Secondary Code Length (chip)	Secondary Code Cycle (ms)	Navigation Data Data	Navigation Data Symbol Rate (sps)	Navigation Data Data Rate (bps)	Navigation Data FEC	Notes	Pocket SDR Signal ID
Galileo (Cont.)	1191.795	E5a+b*7	-	(-152.0)	8-PSK(10)	10230	10.23	1	100	100	-	-	-	-		-
	1278.75	E6-A	Q	?	BOC(10,5)	?	5.115	?	?	?	G/NAV	?	?	?	PRS	-
		E6-B	I	-155.0	BPSK(5)	5115	5.115	1	-	-	C/NAV	1000	500	1/2 *6	CAS, HAS	E6B
		E6-C	I		BPSK(5)	5115	5.115	1	100	100	-	-	-	-		E6C
QZSS [9][10][11][12]	1575.42	L1C/A	I/Q	-158.5*8	BPSK(1)	1023	1.023	1	-	-	LNAV	50	50	-		L1CA
		L1C/B	I	-158.5	BOC(1,1)	1023	1.023	1	-	-	LNAV	50	50	-		L1CB
		L1C-D	I	-163.0*9	BOC(1,1)	10230	1.023	10	-	-	CNAV2	100	50	BCH,LDPC		L1CD
		L1C-P	Q	-158.25	BOC(1,1)	10230	1.023	10	1800	18000	-	-	-	-	Block I	L1CP
			I	-158.25*10	TMBOC(6,1,4/33)	10230	1.023	10	1800	18000	-	-	-	-	Block II ~	L1CP
		L1S	I	-161.0/-158.5	BPSK(1)	1023	1.023	1	-	-	L1S	500	250	1/2 *6	SLAS	L1S
	1227.6	L2C-M	I	-160.0/ -158.5	BPSK(1)+TDM	10230	0.5115	20	-	-	CNAV	50	25	1/2 *6		L2CM
		L2C-L			BPSK(1)+TDM	767250	0.5115	1500	-	-	-	-	-	-		(L2CL)
	1176.45	L5-I	I	-157.9/-157.0	BPSK(10)	10230	10.23	1	10 (NH)	10	CNAV	100	50	1/2 *6		L5I
		L5-Q	Q	-157.9/-157.0	BPSK(10)	10230	10.23	1	20 (NH)	20	-	-	-	-		L5Q
		L5S-I	I	-157.0*11	BPSK(10)	10230	10.23	1	-	-	L5S	500	250	1/2 *6	Normal mode	L5SI
									2 (MC)	2	L5S	500	250	1/2 *6	Verif. mode	L5SIV
		L5S-Q	Q		BPSK(10)	10230	10.23	1	20 (NH)	20	-	-	-	-	Normal mode	L5SQ
									2 (MC)	2	-	-	-	-	Verif. mode	L5SQV
	1278.75	L6D	I	-155.7	BPSK(5)+TDM*12	10230	2.5575	4	-	-	L6D	2000	2000	RS	CLAS	L6D
		L6L			BPSK(5)+TDM	1048575	2.5575	410	2 (MC)	820	-	-	-	-	Block I	-
		L6E			BPSK(5)+TDM*12	10230	2.5575	4	-	-	L6E	2000	2000	RS	MADOCA-PPP	L6E
BeiDou [13][14][15][16][17]	1561.098	B1I	I	-163.0	BPSK(2)	2046	2.046	1	20 (NH)	20	D1	50	50	BCH		B1I
								1	-	-	D2	500	500	BCH	GEO	B1I
		B1Q	Q	?	BPSK(2)	?	2.046	?	?	?	?	?	?	?		-
	1575.42	B1C-D	I	-159.0/ -161.0	BOC(1,1)	10230	1.023	10	-	-	B-CNAV1	100	50	NB-LDPC	BDS-3	B1CD
		B1C-P	I		QMBOC(6,1,4/33)	10230	1.023	10	1800	18000	-	-	-	-		B1CP
		B1A-D	Q	?	BOC(14,2)	?	2.046	?	?	?	?	?	?	?		-
		B1A-P	Q			?	2.046	?	?	?	-	-	-	-		-
	1176.45	B2a-D	I	-156.0/ -158.0	BPSK(10)	10230	10.23	1	5	5	B-CNAV2	50	25	NB-LDPC	BDS-3	B2AD
		B2a-P	Q		BPSK(10)	10230	10.23	1	100	100	-	-	-	-		B2AP
	1207.14	B2I	I	?	BPSK(2)	2046	2.046	1	20 (NH)	20	D1	50	50	BCH		B2I
								1	-	-	D2	500	500	BCH	GEO	B2I

*7 AltBOC, *8 -164.0 dBW (SVID=7), *9 -167.2 dBW (SVID=7), *10 -162.4 dBW (SVID=7), *11 -162.6 dBW (SVID=3), *12 +CSK by Nav Data

GNSS Signals (3/3)

Military or Restricted Signal

System	Carrier Freq. (MHz)	Signal	I/Q	Min Rec. Power (dBW)	Modulation	Primary Code Length (chip)	Primary Code Chip Rate (Mcps)	Primary Code Cycle (ms)	Secondary Code Length (chip)	Secondary Code Cycle (ms)	Navigation Data Data	Navigation Data Symbol Rate (sps)	Navigation Data Data Rate (bps)	Navigation Data FEC	Notes	Pocket SDR Signal ID
BeiDou (Cont.)	1207.14	B2Q	Q	?	BPSK(10)	10230	10.23	1	?	?	?	?	?	?		-
		B2b-I	I	-160.0/ -162.0	BPSK(10)	10230	10.23	1	-	-	B-CNAV3	1000	500	NB-LDPC	BDS-3	B2BI
											B2b-PPP	1000	500	NB-LDPC	BDS-3, GEO	B2BI
		B2b-Q	Q	?	BPSK(10)	10230	10.23	1	?	?	?	?	?	?	BDS-3	-
	1191.795	B2a+b*13	-	?	8-PSK(10)	10230	10.23	1	?	?	-	-	-	-		-
	1268.52	B3I	I	-163.0	BPSK(10)	10230	10.23	1	20 (NH)	20	D1	50	50	BCH		B3I
								1	-	-	D2	500	500	BCH	GEO	B3I
		B3Q	Q	?	BPSK(10)	?	10.23	?	?	?	?	?	?	?		-
		B3A-D	I	?	BPSK(10)	?	10.23	?	?	?	?	?	?	?	BDS-3	-
		B3A-P	I		BPSK(10)	?	10.23	?	?	?	-	-	-	-		-
NavIC [18][19]	1575.42	L1-SPS-D	Q	-159.6	BOC(1,1)	10230	1.023	10	-	-	IRN-NAV	100	50	BCH,LDPC	NVS-01~	I1SD
		L1-SPS-P	I	-158.2	CSBOC(6,1,4/33)	10230	1.023	10	1800	18000	-	-	-	-		I1SP
	1176.45	L5-SPS	*14	-159.0	BPSK(1)	1023	1.023	1	-	-	IRN-NAV	50	25	1/2 *6		I5S
		L5-RS-D	*14	?	BOC(5,2)	?	2.046	?	?	?	?	50	25	1/2 *6		-
		L5-RS-P	*14	?	BOC(5,2)	?	2.046	?	?	?	-	-	-	-		-
	2492.028	S-SPS	*14	-162.3	BPSK(1)	1023	1.023	1	-	-	IRN-NAV	50	25	1/2 *6		(ISS)
		S-RS-D	*14	?	BOC(5,2)	?	2.046	?	?	?	?	50	25	1/2 *6		-
		S-RS-P	*14	?	BOC(5,2)	?	2.046	?	?	?	-	-	-	-		-
SBAS *15 [20]	1575.42	L1C/A	I	-	BPSK(1)	1023	1.023	1	-	-	SBAS	500	250	1/2 *6		L1CA
	1176.45	L5-I	I	-	BPSK(10)	10230	10.23	1	2 (MC)	2	L5 SBAS	500	250	1/2 *6	PRN120-158	L5I
		L5-Q	Q	-	BPSK(10)	10230	10.23	1	2 (MC)	2	-	-	-	-		L5Q

*13 ACE-BOC, *14 Interplex Modulation, *15 including QZSS L1Sb, BeiDou BDSBAS-B1C/B2a

[1] IS-GPS-200K, Navstar GPS space segment/navigation user interfaces - interface specification, 2019, [2] IS-GPS-800F, Navstar GPS space segment/user segment L1C interface - interface specification, 2019, [3] IS-GPS-705A, Navstar GPS space segment/user segment L5 interface - interface specification, 2010, [4] GLONASS interface control document - navigation radiosignal in bands L1, L2, version 5.1, 2008, [5] GLONASS interface control document - code division multiple access open service navigation signal in L1 frequency band, edition 1.0, 2016, [6] GLONASS interface control document - code division multiple access open service navigation signal in L2 frequency band, edition 1.0, 2016, [7] GLONASS interface control document - code division multiple access open service navigation signal in L3 frequency band, edition 1.0, 2016, [8] European GNSS (Galileo) open service signal-in-space interface control document (OS SIS ICD), Issue 1, Revision 3, 2016, [9] Quasi-Zenith satellite system interface specification - satellite positioning, navigation and timing service (IS-QZSS-PNT-003), 2018, [10] Quasi-zenith satellite system interface specification - sub-meter level augmentation service (IS-QZSS-L1S-003), 2018, [11] Quasi-zenith satellite system interface specification - centimeter level augmentation service (IS-QZSS-L6-003), 2018, [12] Quasi-zenith satellite system interface specification - positioning technology verification service (IS-QZSS-TV-004), 2023, [13] BeiDou navigation satellite system signal in space interface control document - open service signal B1I, version 3, 2019, [14] BeiDou navigation satellite system signal in space interface control document - open service signal B1C, version 1.0, 2017, [15] BeiDou navigation satellite system signal in space interface control document - open service signal B2a, version 1.0, 2017, [16] BeiDou navigation satellite system signal in space interface control document - open service signal B3I, version 1.0, 2018, [17] BeiDou navigation satellite system signal in space interface control document - Precise Point Positioning service signal PPP-B2b, version 1.0, 2020, [18] Indian Regional Navigation Satellite System, Signal in space ICD for standard positioning service version 1.1, 2017, [19] NAVIC signal in space ICD for standard positioning service in L1 frequency version 1.0, 2023, [20] RTCA, Minimum Operational Performance Standards for Airborne Equipment Using Global. Positioning System/Wide Area Augmentation System, DO-229, 1996