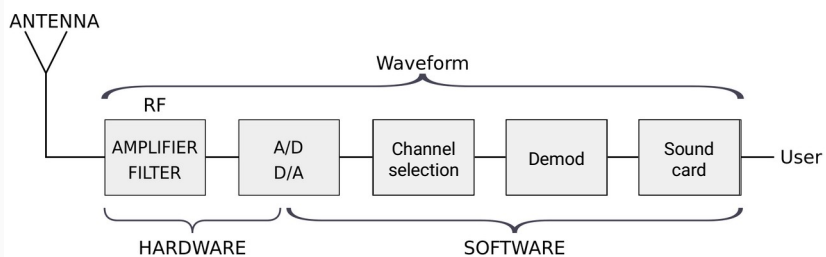
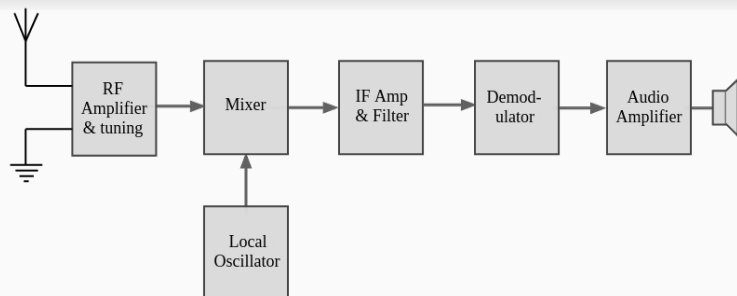


Szoftverrádiók

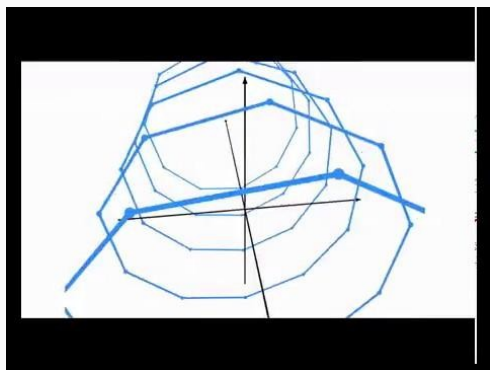


Előadó: Csókás „Csoki” Bence

Miből áll a szoftverrádió?



Minél többet szoftverből, csak a minimálisat HW-ben



- Komplex sinusoid függvény
- Ezzel fogunk lekeverni

Matek

valós input jel keverés LO

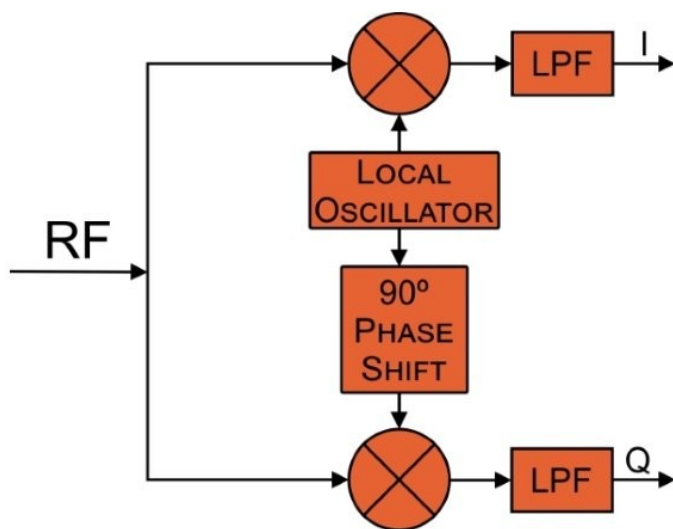
$$f(t) \cdot e^{j\omega t}$$

$$e^{j\omega t} = a(t) + jb(t)$$

$$f(t) \cdot e^{j\omega t} = f(t)a(t) + jf(t)b(t)$$

Magyarul, ha sin-al és cos-al keverünk egyszerre, megkapjuk a komplex jel Re és Im részét

IQ demodulátor



cos $\rightarrow$ 90°-os fázistolás $\rightarrow$ sin

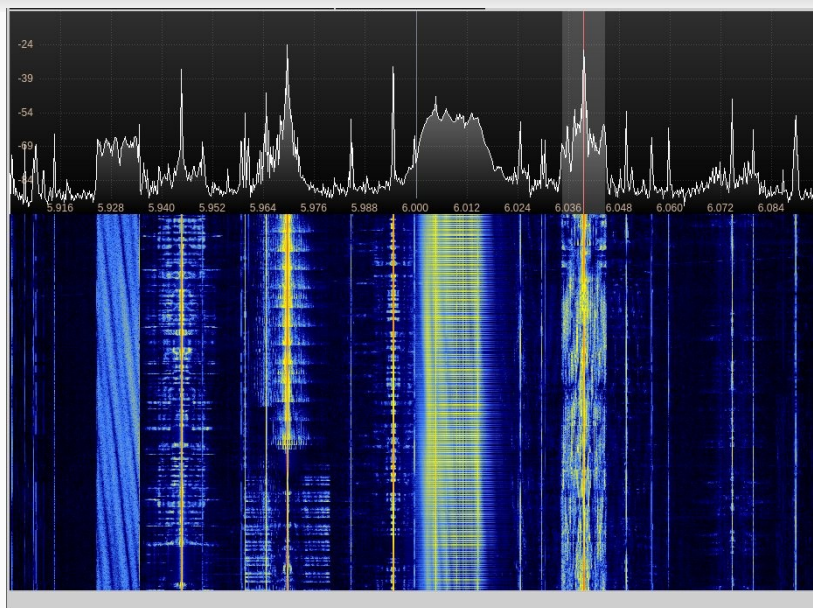
I = In-phase; Re

Q = Quadrature; Im

Komplex jelek

- Pozitív és negatív frekvencia komponensek különböznek
- Pozitív tartományban az LO feletti jelek
- Negatív tartományban az LO alatti jelek
- **Frekvenciában eltoltspektrum**

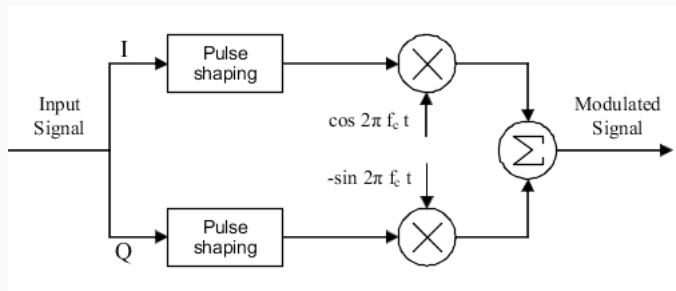
Komplex jelek



Vízszintes tengelyen a freki
Függőleges tengelyen az idő, legfelül a jelen pillanat
Amplitúdó színekkel

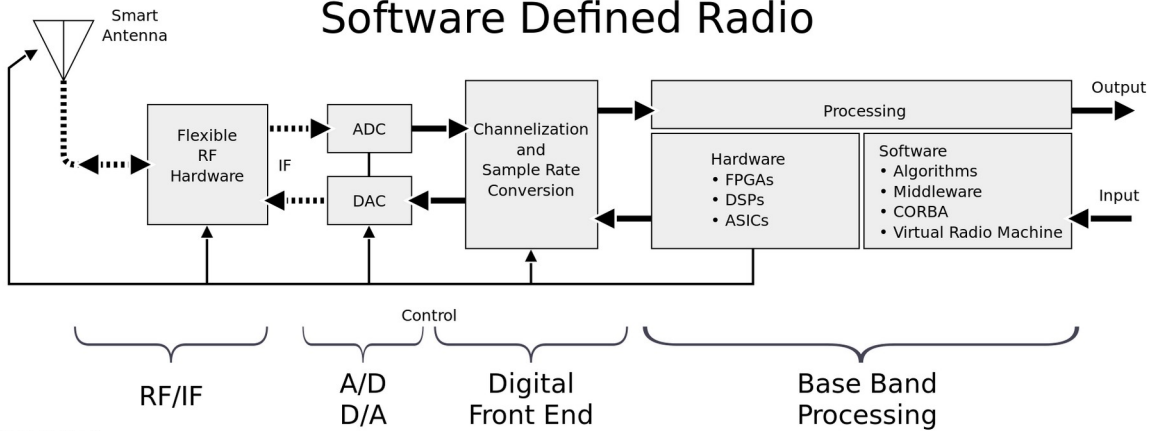
IQ modulátor

- Ugyanaz visszafelé

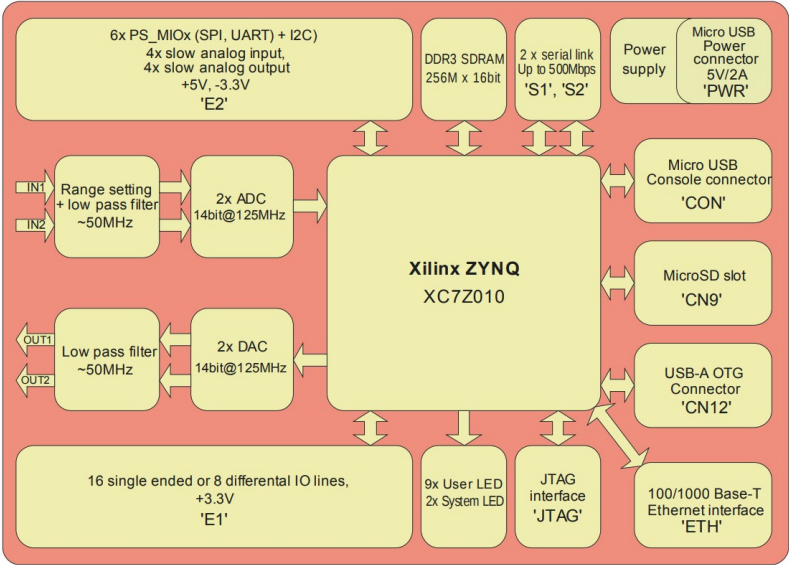


Hardver felépítése

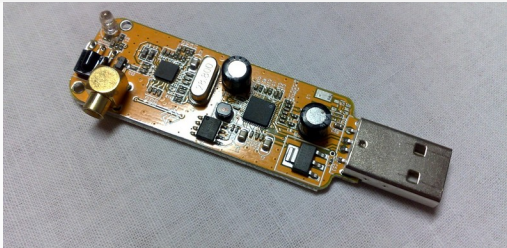
Software Defined Radio



Red Pitaya STEmlab Block Diagram



Piacon lévő eszközök



- DVB-T vevők (Realtek RTL2832U)
- HackRF
- LimeSDR
- NI USRP
- Red Pitaya



RTL: FM vétel funkció, a driver szoftverből dolgozta fel!
RTL-SDR projekt: szoftverrádióvá alakítás

Piacon lévő eszközök

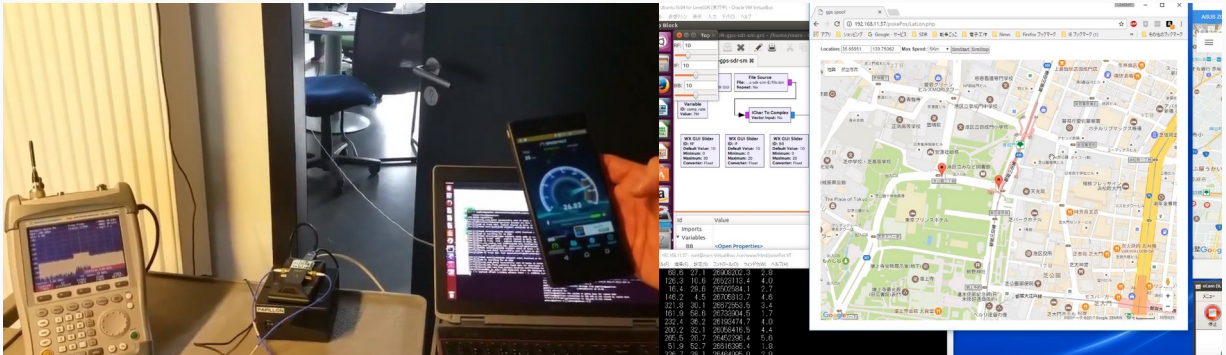
	HackRF One	Ettus B200	Ettus B210	BladeRF x40	RTL-SDR	LimeSDR
Frequency Range	1MHz-6GHz	70MHz-6GHz	70MHz-6GHz	300MHz-3.8GHz	22MHz-2.2GHz	100kHz-3.8GHz
RF Bandwidth	20MHz	61.44MHz	61.44MHz	40MHz	3.2MHz	61.44MHz
Sample Depth	8 bits	12 bits	12 bits	12 bits	8 bits	12 bits
Sample Rate	20MSPS	61.44MSPS	61.44MSPS	40MSPS	3.2MSPS	61.44MSPS (Limited by USB 3.0 data rate)
Transmit Power	-10dBm+ (15dBm @ 2.4GHz)	10dBm+	10dBm+	6dBm	N/A	0 to 10dBm (depending on frequency)
Price	\$299	\$686	\$1,119	\$420 (\$650)	~\$10	\$299 (\$289 pre-order)

Mire használható?

GSM / 4G bázisállomás

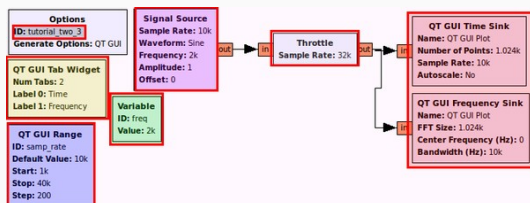
GPS adó

..és még sok más



Mire nem?

GNU Radio



- Blokk alapú jelfeldolgozó library
- Széleskörűen konfigurálható blokkok
- Köztük a jeltovábbítás megoldott
- Pythonban, C++-ban programozható
- GNU Radio Companion: egyszerű GUI

Játszásra jó még: Gqrx, SDR#

Példa - AM demod szoftverrádióval

```
1. #include <math.h>
2. #include <stdio.h>

3. int main()
4. {
5.     double i, q, s;
6.     for(;;) //végtelen ciklus
7.     {
8.         i=((unsigned char)getchar()-127); //beolvasás
9.         q=((unsigned char)getchar()-127); //interlaced I, Q
10.        s=sqrt(i*i+q*q); //pillanatnyi amplitudó
11.        putchar((unsigned char)(s+127)); //kiírás
12.    }
13. }
```

Gyakorlaton hasonló feladat vár

AM demod hiányosságok

- Csatornaválasztás
 - Szűrés az elején (LPF)
 - vagy lekeverés+szűrés
- Decimálás & szűrés a végén
 - a hanghullám hangkártya mintavételi frekvenciájára illesztése

```
rtl_sdr -s 240000 -f 118200000 -g 20 - | tcc -lm -run minidemod.c \  
| sox -t raw -r 240000 -e unsigned -b 8 -c 1 - -t raw - rate 48000 \  
| mplayer -quiet -rawaudio samplesize=1:channels=1:rate=48000 -demuxer rawaudio
```

Read like this:

```
rtl_sdr (acquires samples) | minidemod (demodulates) \  
| sox (low pass filter + decimation) | mplayer (audio output) `
```

Unix pipe

`tcc -run`: lefordítja és futtatja

Köszönjük a figyelmet!

Kérdések?