PulseJS

Quick development of sensors and radio applications

SDR, simplified version

❶ A SDR system is made up of a transmission/reception equipment (a « SDR »)(1), a computer(2) and an application operating the hardware and performing real-time signal processing(3). **SDR-Technologies** revolutionizes the development of SDR (Software Defined Radio) applications with **PulseJS**, a powerful and easy-to-use software engine that allows you to create signal processing applications in just **a few hours**, compared to the several weeks usually required.

```
// read the queue name from the parameters
var queue_name = argv(0);
// read the channel offset
pmr_channel_offset = parseFloat( argv(1)) * 1e6 ; // convert to Hz for DDC

// access the queue
var fifo_from_rx = Queues.create(queue_name);

// configure the DDC
var ddc = new DDC();
ddc.setCenter( pmr_channel_offset );
ddc.setOutBandwidth( 12.5e3 );

// configure a FM demodulator and plug it in the DDC
var FMdemod = new NBFM('fm');
FMdemod.configure( {'modulation_index': 0.2} );
ddc.setDemodulator( FMdemod ); // tells the DDC to call the FMDemod object before output
ddc.setAGC(true);

var IQBlock = new IQData('iq');
// now loop : read IQ block from radio, do something
while( fifo_from_rx.isFromRx() ) {
    if( IQBlock.readFromQueue( fifo_from_rx ) ) {    // load samples from input queue into IQBlock obje
        ddc.write( IQBlock);
        var fm_audio = ddc.read(); // output is IQData !!
        // do something with the audio
        // here we just dump data to see if it works
        fm_audio.dump();
    }
}
```
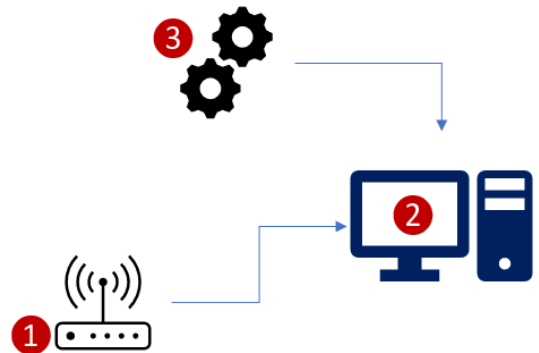

Customize your SDR

❷ PulseJS is delivered with its **starter kit** to ensure accelerated development of your application. This kit includes an **embedded PulseNode board** (ARM processor with pre-installed PulseJS, integrated GNSS, and native SDR support), **documentation** and **tutorials**. Additionally, **technical support** is included, with our team ready to put its expertise at your disposal.

https://sdrvm.sdrtechnologies.fr/

SDR Technologies
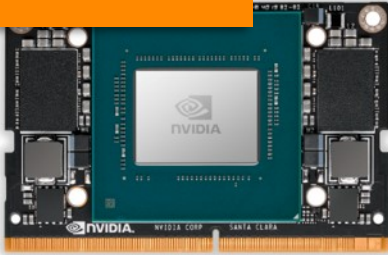
Target platform examples :

ARM , RISCV
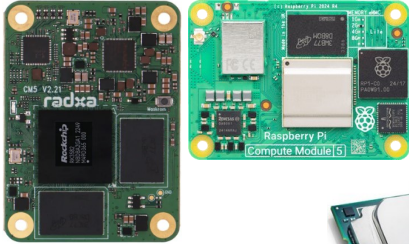
x86

NVIDIA Jetson

Options

- Formations
- Assistance and advice
- Technical support
- Specific radio drivers
- LLM APIs (Ollama, ChatGPT)

Accessibility

PulseJS simplifies engineering and development of radio applications.

- Intended for non-experts with basic knowledge of programming and signal processing
- Facilitate radio data analysis (real-time or delayed/post-processing)

Functionalities

- Executable file, platform-dependent, (x86, GPU…), licensed
- Runs on any Linux-supported processor
- Hardware supports: RTLSDR, LimeSDR, BladeRF and SoapySDR
- Multitasking: can handle several distinct tasks simultaneously
- Enables communication between multiple PulseJS systems, mesh network
- Possible reuse of existing code thanks to the add of software "plugins"

Ease of use

- Programs stored as text files, JavaScript-type language
- Executes local or downloaded (http, ftp,…) programs
- Wide range of functions available (GPS, AM, FM, SSB, FSK, GMSK, …)
- Integrated Web server
- Notifications (SMTP, MQTT, API)
- Complete online documentation

Engineered & manufactured in France

www.sdrtechnologies.fr - ✉ contact@sdr-technologies.fr

7 rue Ernest Gouin - 78290 Croissy-sur-Seine - **FRANCE**