

TELECOMMUNICATIONS ENGINEERING

Engineering Software-Defined Radio: From Simple PSoC Receivers to Advanced Space Weather Stations

Published: November 11, 2026 | Tags: Software Defined Radio | SDR Design | Cypress PSoC | TangerineSDR | RF Engineering | Satellite Telemetry | Digital Signal Processing

The evolution of radio frequency (RF) communication has undergone a radical transformation over the past three decades, moving from rigid, hardware-centric architectures to the flexible, algorithm-driven domain of **Software-Defined Radio (SDR)**. In a traditional radio system, fundamental functions such as modulation, demodulation, and filtering are performed by dedicated analog components like inductors, capacitors, and crystals. In contrast, an SDR shifts these processes into the digital domain, utilizing high-speed **Analog-to-Digital Converters (ADCs)** and powerful Digital Signal Processors (DSPs) or Field Programmable Gate Arrays (FPGAs) to manipulate signals via software.

The Core Theoretical Framework of Software-Defined Radio

To understand the design of systems like the **Mercury SDR** or the **TangerineSDR**, one must first grasp the mathematical foundations of signal processing. At the heart of SDR lies the **Nyquist-Shannon Sampling Theorem**, which states that to accurately reconstruct a signal, it must be sampled at a rate at least twice the highest frequency component present in the signal (the Nyquist rate). However, in modern SDR design, we often employ **Quadrature Sampling** (I/Q sampling).

Quadrature sampling involves splitting an incoming signal into two components: the **In-phase (I)** and the **Quadrature (Q)**, which is shifted by 90 degrees. This allows for the representation of complex signals and simplifies the process of frequency shifting and demodulation without the production of unwanted image frequencies. The relationship is mathematically expressed as:

By manipulating these I and Q streams, software can emulate any modulation scheme, from simple Amplitude Modulation (AM) and Frequency Modulation (FM) to complex digital schemes like **Quadrature Amplitude Modulation (QAM)** used in modern LTE and Wi-Fi networks.

The Architecture of a Simple SDR Receiver: The PSoC Approach

One of the most elegant examples of an entry-level but highly capable SDR is the design based on the **Cypress PSoC (Programmable System-on-Chip)**, specifically the CY8C3866. This device is unique because it integrates both programmable analog and digital blocks on a single silicon die,

making it an ideal candidate for a **Simple SDR Receiver** operating in the HF (High Frequency) range of 3.5 to 18 MHz.

Key Components of the PSoC SDR Design

- **Front-End Filtering:** Before the signal reaches the PSoC, a Band Pass Filter (BPF) is required to suppress out-of-band signals that could cause aliasing or overload the converter.
- **The Mixer Stage:** In the Michael Hightower (KF6SJ) design, the PSoC's internal components are configured to act as a Tayloe Detector (a high-performance switching mixer). This stage converts the RF signal down to a Low Intermediate Frequency (Low-IF) or Zero-IF (Baseband).
- **Delta-Sigma ADC:** The CY8C3866 features a high-resolution Delta-Sigma ADC which digitizes the down-converted I and Q signals.
- **Digital Filter Block (DFB):** The internal hardware DFB performs real-time decimation and filtering, reducing the computational load on the CPU core.

The 3.5 to 18 MHz range covers several critical amateur radio bands (80m, 40m, 20m, etc.). By using a PSoC-based architecture, engineers can achieve a high degree of integration, reducing the physical footprint and power consumption compared to traditional discrete-component SDRs.

Advanced Instrumentation: TangerineSDR and the Personal Space Weather Station (PSWS)

Moving from simple hobbyist receivers to research-grade instrumentation leads us to the **TangerineSDR** and the **Mercury SDR** project. These systems are part of a broader initiative known as the **Personal Space Weather Station (PSWS)**, developed in collaboration with TAPR (Tucson Amateur Packet Radio) and academic institutions.

The PSWS is designed to measure the effects of solar activity on the Earth's ionosphere. This requires capabilities far beyond a standard receiver, including:

- **Precision Time-Stamping:** Using a GPS-Disciplined Oscillator (GPSDO) to provide sub-microsecond accuracy for RF samples. This allows researchers to compare data from multiple receivers globally to triangulate signal origins or observe ionospheric disturbances.
- **Wide Dynamic Range:** The ability to detect very weak signals in the presence of strong local interference, necessitating high-bit-depth ADCs (14 or 16 bits).
- **Data Magnetometry:** Integration with ground-level magnetometers to measure small fluctuations in the Earth's magnetic field (measured in nanoteslas).

The Mercury SDR Module

The Mercury SDR serves as the high-performance receiver engine for the TangerineSDR ecosystem. Unlike simple receivers that might rely on a host PC for all processing, the Mercury design often utilizes an **FPGA (Field Programmable Gate Array)** to handle high-speed digital down-conversion (DDC). The DDC reduces the high-speed sample stream from the ADC into a manageable bandwidth for transmission over Ethernet or USB.

Technical Comparison: Receiver Architectures

The following table compares different levels of SDR implementations based on technical specifications and use cases.

Feature	Simple PSoC SDR	Mercury / TangerineSDR	Traditional Hardware Radio
Frequency Range	3.5 MHz - 18 MHz	100 kHz - 500 MHz+	Fixed (e.g., HF or VHF)
Processing Unit	PSoC 3 (8051 + DFB)	FPGA (Intel/Altera)	Analog Circuits / MCU
Sampling Rate	~192 ksps (Effective)	122.88 Msps+	N/A (Analog)
Time Accuracy	Standard Crystal	GPSDO (High Precision)	TCXO (Moderate)
Flexibility	Software-Defined (Limited)	Fully Programmable Hardware	Hardware-Fixed
Primary Use	Educational / Amateur	Scientific / Space Weather	General Communication

The Role of SDR in Satellite Telemetry and Voluntary Observation

A significant application of SDR technology is the collection of satellite telemetry. Research by **M. Tokumitsu (2017)** highlights a prototype system designed for voluntary observers to collect satellite data. In this model, a simple SDR receiver acts as a ground station node.

Integrated Telemetry Workflow

1. **Signal Acquisition:** The SDR captures the downlink signal from a SmallSat or CubeSat, typically in the VHF/UHF bands.
2. **Local Demodulation:** The software (often GNU Radio or a custom web app) demodulates the data frames.
3. **Cloud Integration:** The data is sent to a central repository server.
4. **Data Compensation:** Because voluntary observers may have intermittent coverage, the central server uses algorithms to merge incomplete data streams from multiple observers into a single, complete telemetry set.

This distributed network approach, enabled by the low cost of SDR hardware, allows for global satellite monitoring without the need for expensive, dedicated ground station infrastructure.

Engineering Challenges: Overcoming Noise and Interference

In any SDR design, especially those covering the 3.5 - 18 MHz range, managing the **Signal-to-Noise Ratio (SNR)** is paramount. There are three primary sources of degradation in an SDR system:

1. Quantization Noise

This is inherent to the ADC. Every bit of resolution in an ADC adds approximately 6 dB of dynamic range. A 12-bit ADC provides roughly 72 dB, while a 16-bit ADC provides 96 dB. In environments with strong interfering signals, a higher bit-depth is necessary to prevent the ADC from saturating (clipping).

2. Clock Jitter

In high-speed sampling, the timing of the sample is as important as the amplitude. Phase noise in the sampling clock can manifest as reciprocal mixing, where strong signals on adjacent frequencies leak into the desired signal's bandwidth, raising the noise floor.

3. Electromagnetic Interference (EMI)

Since SDRs are often connected to computers via USB or Ethernet, they are susceptible to high-frequency noise generated by switching power supplies and high-speed digital buses.

Effective shielding and the use of **common-mode chokes** on data cables are essential design practices.

Practical Implementation: Building a Prototype SDR Receiver

For engineers looking to implement a system similar to the KF6SJ Simple SDR, the following workflow provides a technical roadmap:

Phase 1: Analog Front-End Design

Design a switchable Band Pass Filter (BPF) bank. For the 3.5 MHz to 18 MHz range, a series of 5th-order Chebyshev filters provides a good balance between roll-off and complexity. Ensure the input impedance is matched to 50 ohms to maximize power transfer from the antenna.

Phase 2: Signal Conversion

Utilize a high-speed bus or an integrated SoC like the Cypress PSoC. Configure the internal Universal Digital Blocks (UDBs) to generate the precise clock phases required for I/Q sampling. If using the Tayloe Mixer approach, ensure the sampling switch (e.g., FST3253) has low ON-resistance to minimize insertion loss.

Phase 3: Digital Signal Processing (DSP)

Implement the following chain in software (C++ or Python):

- DC Removal: Remove any offset introduced by the ADC hardware.
- Frequency Shift: Use a Numerically Controlled Oscillator (NCO) to fine-tune the signal.
- Decimation: Reduce the sample rate to the desired audio bandwidth (e.g., 48 kHz).
- Demodulation: Apply SSB (Single Sideband), CW (Continuous Wave), or Digital mode logic.

The Future of Software-Defined Radio and Global Connectivity

The convergence of SDR and web technologies is creating a new era of **Collaborative Radio**. Projects like the TangerineSDR are not merely standalone devices but nodes in a global sensor network. By sharing data across repositories, as seen in the PSWS project, we can observe the Earth's environment with unprecedented resolution.

Furthermore, the move toward **Direct Sampling**-where the ADC is placed as close to the antenna as possible-is becoming more feasible as converter technology improves. This eliminates the need for analog mixers entirely, further reducing hardware distortion and increasing the "softwarization" of the radio. As we look toward the 6G era and advanced satellite constellations, the principles found in a "Simple SDR" remain the foundation for the most complex communication infrastructures on the planet.

By abstracting the physical layer into code, we have unlocked a level of adaptability that was previously impossible. Whether it is a voluntary observer capturing satellite telemetry or a researcher monitoring solar flares, the Software-Defined Radio is the bridge between the chaotic electromagnetic spectrum and the precision of digital information. The engineering journey from the basic PSoC receiver to the sophisticated Mercury SDR illustrates a broader truth in technology: when hardware becomes flexible, innovation becomes limitless.