Paul Adams and Lincoln Young

EE542

Design Specification

This project will undertake to design, develop and implement a software defined radar using off the shelf RF and analog circuit components coupled with a Raspberry Pi for the microcontroller and system interface. The MIT Open CourseWare Coffee Can Radar course [1] will be used as a design template for the hardware portion. We intend to demonstrate the capabilities of a simple radar architecture, coupled with small system-on-a-chip, and distributed sensing and processing.

GENERAL DESCRIPTION

Radio Frequency (RF). The radar will emit in the unlicensed ISM band at 2.4 GHz with a signal bandwidth below 48 kHz, so as to be in the range of audio analog-to-digital converters (ADC). The radar waveform will be a Frequency Modulated Continuous Waveform (FMCW) with period of approximately 40 milliseconds. The transmit and receive antennas will be physically separate and implemented with empty coffee cans as a cavity for a simple monopole emitter. The RF signal conditioning chain will consist of off the shelf Mini-Circuits parts.

Power Supply. The system will implement 5 and 12 VDC rails to a solderless breadboard using AA batteries connected in series. A low dropout voltage regulator will maintain 5 volts to the active RF components.

Analog Circuit. Keeping to the design called out in, the breadboard will contain a function generator, tunable in DC bias and signal period. A video amplifier with low-pass-filter will be implement to condition the receive signal. The circuits will be mapped to a PCB for performance improvements.

Audio Interface. The received signal will be mixed with the transmitted signal such that the baseband signal of interest will be in the audio band. A simple USB plug and play audio card will convert the analog audio signal to a digital signal sampled at 48 kHz with 16 bits of resolution.

Microcontroller. A Raspberry Pi (R-Pi) 2 B+ will be used to process raw data and prepare results for display. The R-Pi uses a Broadcom ARMv7 chipset and will be loaded with a Linux kernel distributed by Arch Linux for Arm [3] (alarm).

Middleware. Python will be used to program the R-Pi and implement the real-time application. It will interface with the Jack [4] audio connection kit as well as Advanced Linux Sound Architecture (ALSA) [5] serving as the audio interface driver.

Application Software. Matlab will be used to prototype algorithms and data processing. Testing and debugging routines will be written using a combination of Matlab and Python. Python applications will run on the R-Pi to do initial data routing and reduction and then make results available on a network port. A second R-Pi will ingest these reports and publish to a web interface.

ENVIRONMENT

Environmental considerations include signal radiation and system hardware components. The radar will conform to ISM broadcast by staying within the allocated bandwidth and radiating less than 10 mW of power.

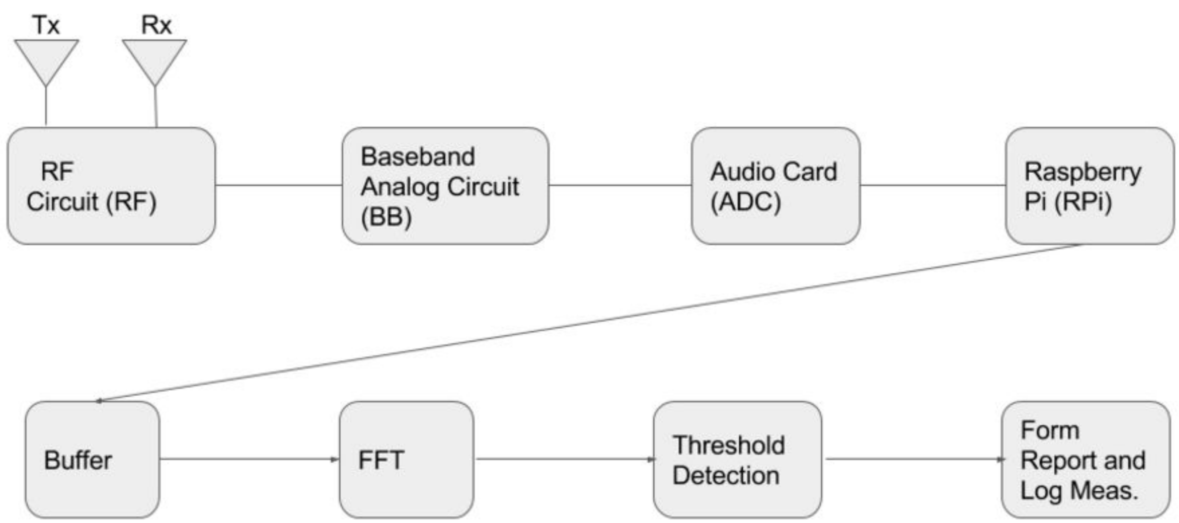
INPUT/OUTPUT

Analog to RF. Soldered wires will connect the Modulator to the Oscillator. RF signals will migrate through SMA connectors and cables. The RF signal will radiate into the environment and reflect energy from various scatterers. The received signal will terminate at a ¼ inch audio cable.

Analog to Digital. The audio cable will connect to a USB A/D converter and streamed to the host's PCM device.

Application Software. The main program will receive audio data and reduce that to detection reports. These reports will be published to a network socket and subscribed by a remote R-Pi which will collect results over time and publish to a web interface.

FUNCTIONAL DESCRIPTION



Works Cited

Charvat, G.L. *MIT IAP 2011 Laptop Based Radar: Block Diagram, Schematics, Bill of Material, and Fabrication Instructions*, 1998. Print.