Design and Implementation
of a
Link Level Adaptive Software Radio

Richard A. Killoy

B.S.E.E., University of Kansas, Lawrence, Kansas 1997

Submitted to the
Department of Electrical Engineering and Computer Science
and the Faculty of the Graduate School of
the University of Kansas
in partial fulfillment of the requirements for the degree of
Masters of Science.

_____
Professor in Charge

_____

_____
Committee Members

_____
Date Thesis Accepted

Acknowledgements

The success of this research relies heavily on the contributions made by my colleagues. This project is being implemented as part of the RDRN project at the University of Kansas. As such, many individuals have contributed to the design, planning, building and testing of this radio. I would be in error if I did not take a few moments to point out the significant contributions made by the myriad of people who have helped me along the way.

I would like to thank Professor Glenn Prescott for his guidance, his technical expertise and for giving me the chance to work on such a challenging project. Above all else, thanks for talking me into going to graduate school now as opposed to later.

All the classes I've taken pale in comparison to the knowledge I've learned from Ken Filardo. Your technical wisdom, your friendship, and above all else, your patience has helped to show me what real engineering and teamwork are all about.

To Shane Haas and Craig Sparks, the teamwork and knowledge we have shared the past two years has meant the world to me. I can only hope I work with other engineers as personable and knowledgeable as you. I still think the potato gun was our best work yet.

The RF expertise and helpfulness of Dan Depardo was crucial to the success of this project. Thank you for all your time and patience.

Above all else, thanks go to my fiancée, Lorie. Thank you so much for all your support and patience while I've written this. You've done all the little things that make my life so much easier while working on this. They are too numerous to name here, but thanks just for being there. Now, finally, we can start our life together.

Abstract

Over the past few years, the software radio has emerged from research as a commercially viable and flexible digital communication system. Advances in digital technology are quickly making the software radio an attractive strategy for low-cost, multi-dimensional transceivers. A needed function in tomorrow's transceivers is the ability to change, or adapt, to the rapidly changing channels (fading channels) encountered in wireless communication systems. This paper centers around the design and implementation of such a system, a flexible software controlled radio, adaptive at both the physical and data-link layers. The microprocessor driven radio described herein is a communications and data acquisition platform which supports multiple modulation formats, varying data rates and adaptive Forward Error Correction (FEC). This provides a means for developing algorithms to control radio parameters in the face of rapidly changing channels. We have also provided a means for examining and storing the digital symbols received across the channel. These stored symbols are used to verify adaptive channel estimation algorithms offline. From these channel parameters (i.e., Doppler spread, coherence bandwidth) software radios can dynamically change their characteristics to best suit the constantly changing channel.

Table of Contents

List of Figures

List of Tables

Chapter 1 Introduction

The purpose of this research is to develop a high-speed, wireless, Asynchronous Transfer Mode (ATM) transceiver. This transceiver is a digitally controlled software radio, adaptive at both the physical and data-link layers. This receiver is being designed and implemented in conjunction with the Rapidly Deployable Radio Network Project (RDRN) at the University of Kansas.

The goal of this transceiver is to provide physical layer functionality along with link layer adaptivity for a self-configurable, wireless ATM network.

The goal of the RDRN project is to design a high-speed, wireless ATM communication system that is adaptive at both the network and data-link layers to allow for rapid deployment and for use in a changing environment. Also, RDRN hopes to create architectures, protocols, and prototype hardware and software for a high-speed (1-8 Mb/s at 10 km), wireless IP/ATM network, based on Global Positioning System (GPS) location information that can be deployed rapidly in areas of military conflicts or civilian disasters where communication infrastructures are lacking and/or destroyed.

Brief Overview of the RDRN2 Project

The RDRN project has been an ongoing research program at the University of Kansas for several years. It has concentrated on the development and requirements needed for the rapid deployment of communication networks. The need for Quality of Service (QoS) support and seamless integration with commercial networks has led to an approach that uses wireless technology for the communications links, and ATM for networking.

This second phase of research, RDRN2, builds upon the earlier success of RDRN1. For this phase, our research teams have been concentrating on the following areas:

1) Channel Estimation and Link Adaptation. We are continuing our work on algorithms to estimate and adapt to rapidly changing channels, such as those encountered in mobile networks (i.e., Rayleigh fading channels). Also, we are developing algorithms to estimate the channel state and use that information to change power level, modulation, FEC, and/or frame length. This work will result in a control algorithm for adapting to link characteristics.
2) Software Radios. We have built single, full-duplex, point-to-point radios based on the RDRN2 year one radio to test, evaluate, and validate the channel estimation and link adaptation algorithms stated above. These radios incorporate a modular RF section, and we will test and validate the channel estimation and link adaptation algorithms at 5.3 GHz, 2.4 GHz, and 1.2 GHz.
3) Adaptive Networking. RDRN is developing a flowspec tailored to mobile networks. The flowspec will facilitate the mapping of IP and ATM service classes across mobile networks and coordinate switching at the frame, ATM, and IP layers. We also studied the performance of multi-layer networks (e.g., multi-hop, short distance, peer-to-peer at lower echelons, longer hop, switched at higher echelons).
4) Resource Reservation Styles. Evaluation of resource reservation styles for rapidly mobile networks. This evaluation will determine features of the current styles for mobile networks in light of the available knowledge such as accurate timing and position information as well as network-aware applications.
5) Comparative Performance Evaluation of IP versus ATM. Evaluation of the comparative performance of IP and ATM for high mobility networks

as a function of the link capacity versus the mix of traffic (delay/jitter sensitive and non-delay/jitter sensitive). This evaluation was carried out in an emulated environment using the RDRN software ATM switch on up to 24 nodes.

The focus of this paper centers around the first two items stated above. Mainly, the building of a software radio for use in channel estimation which is adaptive at the physical and data-link layers.

RDRN2 Communication Systems Overview

RDRN is made up of two types of communication nodes: 1) end user nodes, providing wireless ATM access for users at a rate of up to 8 Mbps; and 2) edge nodes which serve as Radio Access Points (RAPs) or base stations, and provide switching and connectivity between users. Both types of nodes contain GPS receivers for location determination, software-controlled radios using GPS-derived location information and network control software. [9]

The edge nodes also have integral ATM (software) switches, and they are interconnected by high-capacity (10 to 155 Mbps), directional radio links. Edge nodes can also interface to wired ATM networks.

The RDRN architecture consists of three overlaid radio networks:

(1) a low bandwidth, low power, omni-directional orderwire packet radio network for broadcasting location information and for network configuration and management;

(2) a cellular-like ATM radio network for end-user access to edge nodes with handoffs; and

(3) a high-capacity, wireless ATM backbone network providing connection between switches, using high-capacity radios with multiple directional beams.

This paper focuses on the design and implementation of the end user nodes. Their main function is to provide wireless ATM access to the edge nodes at data rates from 1- 8 Mbps. The following diagram depicts the system setup configuration for the RDRN system.

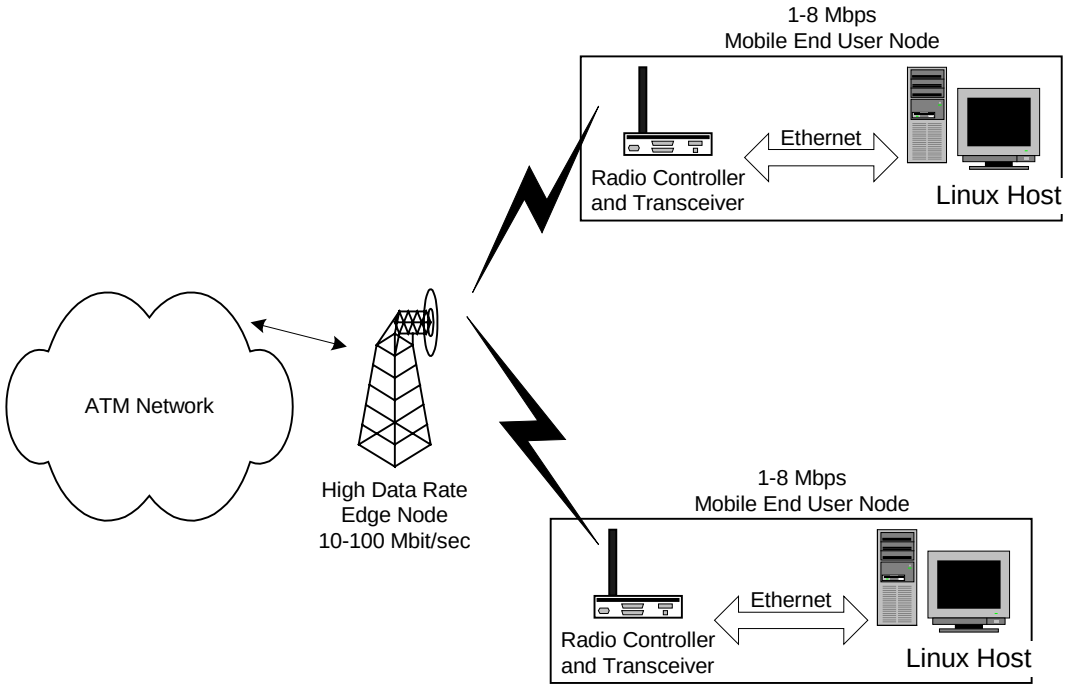

Figure 1 – RDRN System Architecture

This paper focuses on the design of the radio controller and the transceiver. These two devices perform all of the Physical Layer and Data-Link Layers functions for our communication system.

During the design phase, we considered many alternatives to implement our software radios. Our major concern was being able to accomplish the goals RDRN set out to achieve.

Table 1 – RDRN Goals and Solution Strategies Employed

Goal	Solution Strategy Employed
Channel Estimation Algorithms	The use of adaptive equalization, functioning as channel estimators. Refer to the K.U. Masters thesis "Decision Weighted Adaptive Algorithms with Applications to Wireless Channel Estimation" by Shane Haas, April 1999.
Link Level Adaptation	Software radio allows us the flexibility to change link level parameters. Table 2, next page, shows the parameters which are dynamically changeable.
Radios to Test, Evaluate, and Validate the channel estimation and link-level algorithms	Software radios implementing the use of high-speed ADC and various DSP chips. These chips can be programmed dynamically via a PowerPC processor.

RDRN is adaptable to changes in the quality of the radio communication environment. While ATM is designed to operate on high-quality (almost error-free), wired links, typical radio links suffer higher bit error rates and the link quality changes as a function of time due to mobility and changes in the environment. By estimating the channel parameters, such as multipath spread and signal-to-noise ratio, communication parameters at the link and network levels can be adapted to provide appropriate throughput and quality of service. This is the goal of our software-radio based communication system.

The software radio provides us with the ability to dynamically change many characteristics of our radio "on-the-fly." This allows us to have link-level adaptation features and be able to support a myriad of modulation formats and bit rates. The following table depicts parameters which we have the ability to change.

Table 2 – Dynamically Adjustable Radio Parameters.

Adaptive Parameter	Allowable Values
RF Section	
Transmit Power Control, TPC	5 dBm to 25 dBm in 2 dB increments
Carrier Frequency	Carrier frequency can be tuned within RF bandwidth in increments of 10 MHz
*RF Front End	Static Design, but supports 5.3 GHz, 2.4 GHz, and 1.2 GHz band.
Physical Layer	
Modulation format	BPSK, QPSK
Data Rate	1, 2, 4 MSPS
Data Link Layer	
FEC	Limited by complexity of combinational logic circuit implemented in Alteras
Multiple Access Scheme	TDMA, FDMA or hybrid combination of both
RF Preamble for TDMA	Limited by length of time slot
HDLC Frame Length	Limited by Ethernet Packet Size, 1.5 kB

* RF Front End is not dynamically adjustable. Must be manually switched in and out.

Chapter 2 Background

In any communication system, whether wireless or wired, a radio performs the functions depicted in Figure 2 below.

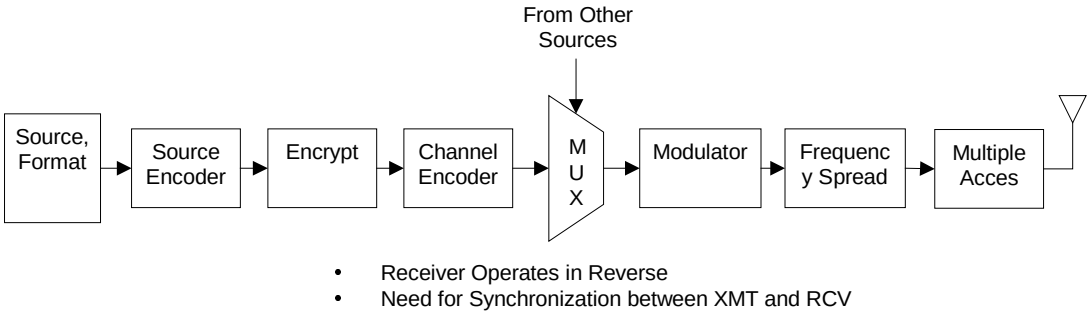

Figure 2 - Block Functionality of a Digital Communication System

While some functional blocks may not be needed in some systems, Figure 2 outlines the functionality of a generic radio system. The only blocks which need be present are the modulator/demodulator and the source. The question still remains, how best to implement these blocks to maximize information transfer.

The wireless communications environment, especially in urban areas, can be quite hostile. The interference, in and out of band, can be quite large, thereby imposing severe constraints upon the transceiver architecture. Each user in a wireless environment must share the resources the communication network has available. The most precious of these resources is the bandwidth each user is allocated. This limited bandwidth is directly related to how quickly information can be sent on the network.

Information theory, more specifically Shannon's Theorem, says the maximum amount of information that can be sent across a communication link with a given Signal to Noise Ratio (SNR) and bandwidth is:

$$c = W \log_2 [1 + SNR]$$

where c = channel capacity [bits/sec]
W = transmitted bandwidth [Hz]
SNR = received signal to noise ratio

Shannon's Theorem puts a theoretical maximum speed limit on the rate of information exchange. It is important to note, that although Shannon says we can achieve this capacity, he does not tell us how to achieve it. For this reason, most communication links are run at a bit rate below the Shannon limit.

The limited bandwidth available to each user impacts the design of the RF section. The transmitter must employ modulation, amplification and filtering such that it minimizes adjacent channel interference. The receiver, on the other hand, must be able to filter the desired band, filter the desired channel, and amplify it while minimizing the interference from strong neighboring channels.

One of the most important design parameters in any receiver is the overall system noise figure (NF). All components generate thermal noise, and this noise adds to, or corrupts, the signal we are trying to detect. Noise figure is a measure of the degradation in SNR as a signal propagates through a system. From Shannon's Theorem, above, we can see that more noise means less capacity.

As we cascade components together in a receive chain, more and more thermal noise is generated, thereby adding to the overall system NF. It can be shown that the noise contributed by each stage decreases as the gain preceding the stage increases. This implies that the first few stages in a cascade are the most critical. Conversely, if a stage exhibits attenuation (loss), the noise figure of the following circuit is "amplified." Careful attention must be paid to the noise figures of these first components in a receive chain [4]. In modern receivers, low-noise amplifiers (LNA's) are typically used. These amplifiers typically have noise figures of only a few dB.

Another important parameter to be considered in the design of any transceiver is the non-linearity of any component in the transmit and receive stage. Figure 3, below, depicts how nonlinearities can effect the transmitter performance. [15]

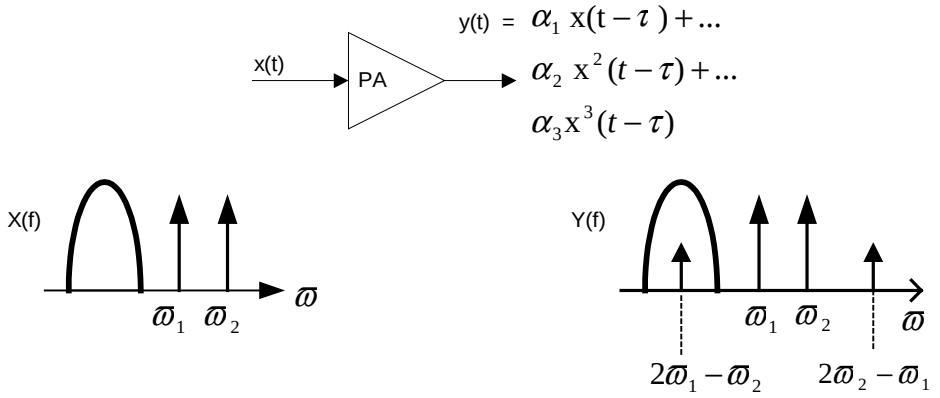

Figure 3 – Example of Nonlinearities in a Power Amplifier

It is important to note the nonlinear nature of the system. An ideal linear system will output a scaled, time-delayed version of its input. However, as shown in Figure 3, a nonlinear system will generate higher order products. These are called intermodulation products. While we may try to minimize this nonlinear behavior, it is always there to some degree.

Amplifiers and mixers in the transmit stage will often generate intermodulation products that fall out-of-band. These interferers can only be attenuated to a small degree by front end bandpass filters and therefore must be acceptably small by design. [15]

These product terms can be quite large and can be received in-band by the receiver. The nonlinear nature of mixers and amplifiers in the receiver front end becomes extremely important at this point. As shown above, odd-order nonlinearities can yield intermodulation products that fall within the desired channel.

Although distorting the amplitude, this effect is important even if the signal carries information only in its phase or frequency, because the zero-crossing points of the desired signal are corrupted by the intermodulation products.

Another important concern in the design of a transceiver is the dynamic range of the signals. A typical communications system operating in a fading environment experiences path loss such that the receiver requires a dynamic range greater than 90 dB. In systems with large dynamic range, crosstalk and input noise can become a problem. Also, the amount of isolation between the transmitter and the receiver becomes important. If the receiver is not sufficiently isolated from the transmitter, these interfering signals may have enough power to desensitize the LNA (for example, if the interferer is above the 1 dB compression point).

Super-Heterodyne Receiver

The most popular receiver architecture in use today, and the one used for the RDRN receiver, is the super-heterodyne receiver. Practically all receivers built today are of this type. A receiver's job is to perform carrier-frequency tuning (mixing), filtering and amplification, all while minimizing noise. The popularity of the super-heterodyne lies in the fact that it performs the first two of these functions in a simple and efficient manner. The following diagram depicts a system level diagram of a super-heterodyne receiver.

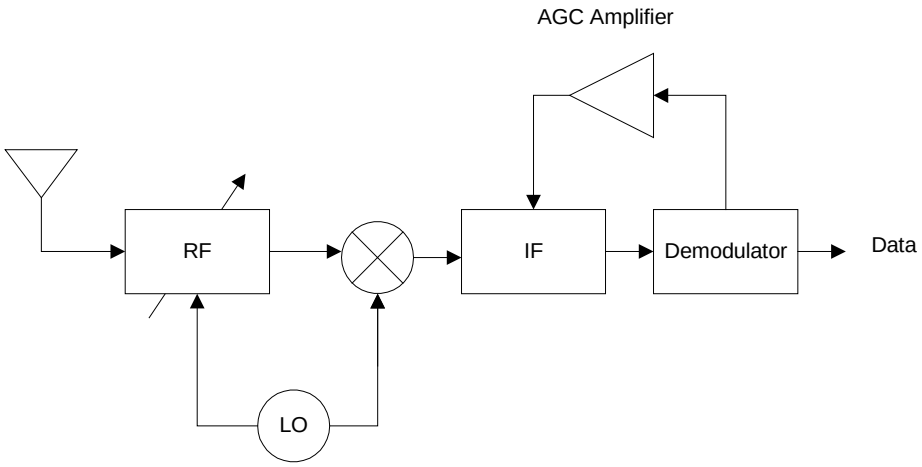

Figure 4 – Super-Heterodyne Architecture

If a receiver is to extract a narrow-band signal accompanied by large interferers from a relatively high frequency band, a bandpass filter with appreciable Q is needed. If a second-order LC filter is to provide 60 dB of attenuation at 45 kHz from the center, this is an equivalent Q on the order of 10^7, a value difficult to achieve even in such filters as surface acoustic wave (SAW) devices. [15]

It is important to note that typical filters exhibit a trade-off between loss and Q. Consider also that a lossy circuit magnifies the noise figure of the following stage and you can see why wide bandwidth, low-loss filters are desired at the RF stage.

Attempting to extract the information band directly from the RF is not very practical. The super-heterodyne receiver first translates this large bandwidth down to a more manageable frequency before attempting to extract the narrow information bandwidth. This relaxes the Q required of the channel select filter.

In choosing the LO and IF frequencies, one must concern themselves with the problem of image. While each transceiver impose constraints upon its own signal emissions, it may no have control over signals in other bands.

The mixer will develop an IF output when the input signal frequency is greater or less than the LO frequency by an amount equal to the IF frequency. That is, there are two frequencies, $\left| f_{LO} \pm f_{IF} \right|$, which will result in f_{IF} at the mixer output. In other words, the bands symmetrically located above and below the LO frequency are downconverted to the same center frequency. This image can be removed with the use of an image-reject filter. [8]

The local oscillator frequency can be either higher or lower than the center of the desired band. Called “high-side injection” and “low-side injection,” each has trade-offs. It is usually desirable to use the latter so as to minimize the LO frequency and hence facilitate the design of the oscillator. On the other hand, if the image bands below and above the desired band exhibit different amounts of noise, then f_{LO} must be chosen so as to avoid the noisy image band.

With either high or low injection, the image is always $2f_{IF}$ from the desired RF spectrum. In systems where adjacent channels are being used by other users, the

amount of power located at this image frequency can be quite substantial. To achieve a low loss in the channel bandwidth and high loss in the image band, one can choose $2f_{IF}$ to be sufficiently large.

As $2f_{IF}$ increases, so does the center of the down-converted band. The choice of IF depends on trade-offs between three parameters: the amount of image noise, the spacing between the desired band and the image, and the loss of the image-reject filter. A high $2f_{IF}$ leads to good rejection of the image, while a low $2f_{IF}$ allows greater suppression of nearby interferes. [15]

Digital Radios & Software Radios

The basic job of any receiver, whether it be a software radio or not, is to amplify the weak signals which it receives and demodulate these signals, all while not adding significant noise. A receiver should not allow adjacent channels or out-of-band interferers to significantly degrade the desired signal. This is known as selectivity. Finally, a receiver should be able to detect an appreciably small signal level, this is called sensitivity.

An ideal software radio is a multi-band, multimode radio with dynamic capability defined through software in all layers of the protocol stack, including the physical layer. Each band and mode is realized in a separate software-defined personality. A software radio can take on the profile of many different receiver implementations simply by reprogramming it. [10]

In the past, hardware was used to implement each of the blocks depicted in the Super-Heterodyne diagram, Figure 4. The idea behind software radios is to implement the functionality of certain blocks in software, as opposed to hardware, giving the radio the ability to perform multiple system configurations by simply reprogramming the radio.

In the past, we would have implemented this solely with analog parts, which unfortunately contained components with analog tolerances. The result was a radio that performed adequately, but only in a given role. The only way to alter the radio performance was to physically change the components inside the radio.

With the advent of high-speed digital circuitry, we can convert our analog signals to digital as soon as possible in the receive chain. Once we have done this, we can perform a myriad of functions without losing any generality or information. For example, we can subject our received signal to multiple hypothesis testing, without adding extra noise. In addition, if we choose to change the way our radio behaves, we simply reprogram the digital section.

Modern software radios utilize high-speed analog-to-digital converters (ADC), and digital signal processors (DSP) to implement many of the functions illustrated in Figure 4. The ideal software radio would place the ADC at the antenna, however, this is not currently possible. Nyquist tells us that to sample a signal without losing information, we must sample at $f_s \geq 2 \times f_h$, where f_h is the highest frequency content of our desired signal. For example, modern cell phones operate at a carrier around 1.5 GHz, and a bandwidth of 200 kHz. Nyquist tells us that we require a sampling frequency slightly larger than $f_s = 3$ GHz to sample this signal without loss of information. This is not practical with today's technology.

Therefore, the question arises as to where to place the ADC. Most modern systems place the ADC at the IF. Using a technique known as sub-sampling, the design engineer is able to sample the received signal at an intermediate frequency. This technique aliases the desired signal bandwidth down to a lower, more useable, frequency band. Refer to the following section for an example of how sub-sampling is used.

The following diagram depicts a typical software controlled radio (SWR).

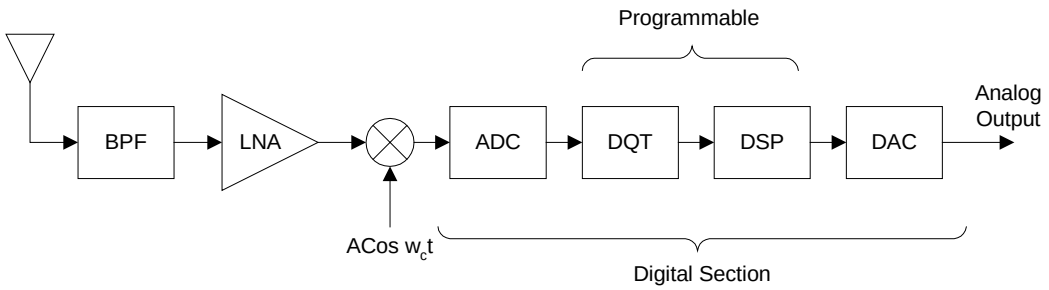

Figure 5 – Typical Software Radio

From Figure 5, we can see the essential components in any software radio system are the following:

- A/D and D/A converters. ADC's provide the link between the analog and the digital realm. DAC's provide the same, allowing digital signals to be converted back to a form which users can hear or see.
- Digital Downconverters. Downconverters provide the equivalent function of a mixer. Modulating parts of a digital spectrum to a different frequency band.
- DSPs. These processors perform the essential functions of demodulation and detection. This includes all the necessary synchronization issues for coherent detection. In addition, they can perform decoding functions. Modern DSP's perform millions of operations per second, and are easily reprogrammed.
- Microcontroller. The microcontroller is used to program different parts of the digital system. It initializes registers and tells the radio what functions to perform. It carries the software that controls the hardware.

Now that we have a better understanding of how a software radio may be implemented, let's look at some of the significant advantages software radios and DSPs provide. These include:

- No need for tuning. Digital filters require no tuning of their components. A digital filter will always act exactly the same as any other digital filter with the same coefficients.

- Linear phase filters. The design of FIR filters allows us to make filters with exact phase linearity. This constant phase delay helps to reduce intersymbol interference (ISI).
- Flexible bandwidth selection. If we need to select different channels from within our band, we simply reprogram the filter coefficients to achieve the desired response.
- Multiple modulation formats can be supported.
- Digital simulation is an "exact" simulation. In other words, the actual digital system should behave exactly like our digital simulations.
- System can be upgraded easily with software.

Digital IF and Subsampling Receivers

A digital IF architecture refers to the fact that we have replaced our analog components with digital ones from the IF section downward. This second set of mixing and filtering can be implemented more efficiently in the digital domain. Figure 5, from the previous page, shows a typical block diagram for a digital IF receiver. If we examine the digital section more carefully, we can get a better appreciation for what digital signal processing is necessary to complete the demodulation and detection.

The following diagram, Figure 6, shows a more detailed view of the required digital section.

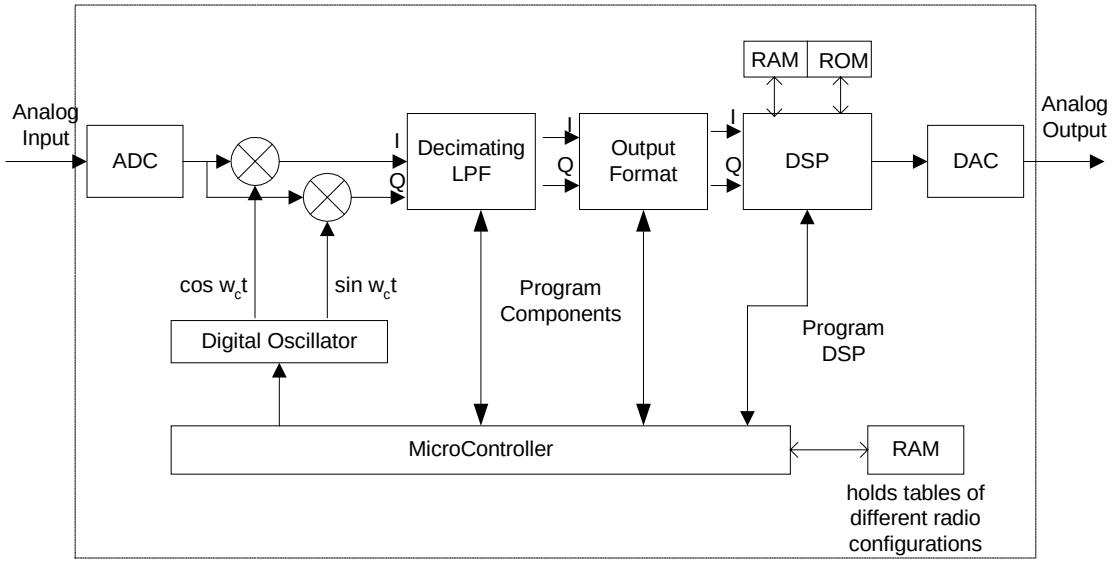

Figure 6 – Detailed View of the Digital Section of a Software Radio

This is the architecture we have used in our RDRN radios. Placing the ADC at the IF section allows us to use a technique called subsampling. The idea is that a bandlimited signal with bandwidth Δf, $(f_H - f_L)$, can be translated to a lower band if sampled at a rate equal to or greater than $2\Delta f$. This operation, if performed correctly, will produce replicas of the spectrum with no aliasing. The sample rate should be selected such that

$$\frac{2f_H}{k+1} < f_s < \frac{2f_L}{k}$$

where the maximum value of k is :

$$k < \frac{f_L}{f_H - f_L}$$

The largest value of k determines the maximum amount of sub-sampling that can be performed. [11]

The following figure depicts an example, the one used for our radio, of sub-sampling.

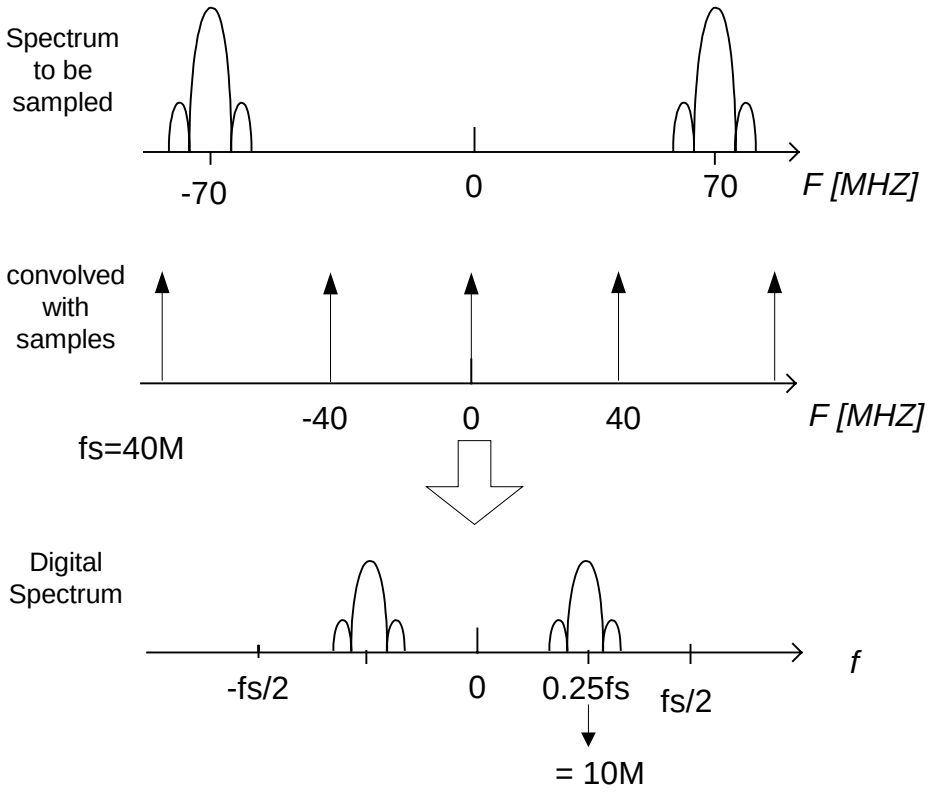

Figure 7 – Sub-Sampling at the IF

I would be remiss not to point out that there are additional architectures and various other hardware techniques available to implement software radios (i.e. FPGAs). However, the RDRN radio's which we designed follow the basic super-heterodyne structure and implement a software radio as depicted in the preceding section.

Decimation

One of the by-products of subsampling is redundant samples. For example, if we sub-sample a 70 MHz IF waveform at 40 MSPS, a copy will be placed at 10 MHz. We do not need 40 MSamples to represent this signal, some form of rate reduction is needed. Therefore, we can literally throw away the extra samples. This process of discarding unneeded samples is called "decimation."

Care must be taken with digital filters that utilize decimation techniques to understand the ramifications in the frequency domain. Of primary concern is the "noise" level increase due to signals that may be aliased inside the band of interest. The potential magnitude of these signals may render significant portions of the previously thought usable bandwidth, unusable for applications that require significant (>60 dB) attenuation of undesired signals.

Consider a digital filter with sampling frequency f_s, whose frequency response is shown in Figure 8A, the top spectrum. At first glance the usable bandwidth would appear to be the 3 dB bandwidth of the main lobe.

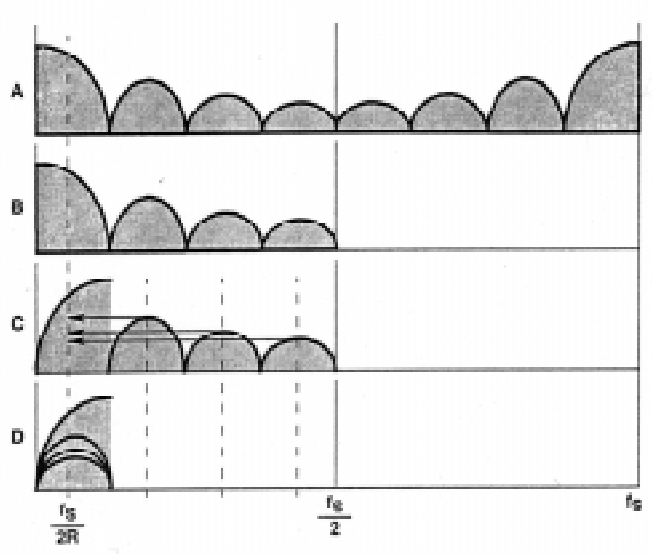

Figure 8 – Aliasing Effects of Decimation

This filter is to be decimated to rate of $\frac{1}{8} f_s$. We concern ourselves with those elements less than $\frac{f_s}{2}$, as shown in Figure 8B. The decimation process will fold the various lobes of the frequency response around the new sampling folding frequency $\frac{f_s}{2R}$. The first lobe is folded over the dotted line and a significant portion of the first

lobe appears in the passband of the filter. Any unwanted signals in this part of the spectrum will appear in the band of interest with the greatest amplitude. The second lobe is translated down to be centered on the dashed line. The third lobe is spectrally inverted and translated to be centered on the dotted line. The fourth lobe is translated too. If more lobes were present, the process would continue to spectrally invert the odd numbered lobes prior to translation to $\frac{f_s}{2R}$. This process is shown in Figure 8C.

The final diagram, Figure 8D, allows up to create an alias profile by summing the alias elements. An alias profile is used to determine what bandwidth yields the desired suppression of unwanted signals for a particular application. [6]

Due to this excessive aliasing of noise and possible interferers, the decimation process is always preceded by anti-aliasing filters. Modern DSP techniques implement the anti-aliasing filter and decimation all in one structure. In our DSP chips, these dual decimating-filter structures are the Cascade-Integrate and Comb (CIC) filters located in the Harris Digital Quadrature Tuner (DQT).

Transceiver Performance Metrics

Testing the operation of the transceiver is very important to ensure full compliance with the requirements specifications set forth. Modern wireless environments are very harsh, so the transceiver needs to be tested to be sure it will operate in such a realistic environment.

Noise figure (NF), the input 3rd-order intermodulation product (IIP3), in-band intermodulation, out-of-band intermodulation, cross modulation, and reciprocal mixing all effect receiver sensitivity and dynamic range.

The sensitivity of an RF receiver is defined as the minimum signal level that the system can detect with acceptable signal-to-noise ratio. It can be shown that the minimum detectable signal level that can be detected is (assuming conjugate matching in front end):

$$P_{in,Min} = -174 \text{ dBm} + 10 \log B + NF + SNR_{req}$$

The first three terms represent the total integrated noise power of the system and are usually referred to as the noise floor of the receiver. It is also interesting to note that since $P_{in,Min}$ is a function of the bandwidth, a receiver may appear very sensitive simply because it employs a narrowband channel, yet this is at the expense of a reduced information rate. [15]

Dynamic Range (DR) is generally defined as the ratio of the maximum input level that the circuit can tolerate to the minimum input level at which the circuit provides a reasonable signal quality. We base the definition of the upper end of the dynamic range on the intermodulation behavior and the lower end on the sensitivity. This is sometimes called the spurious-free-dynamic range (SFDR).

The upper end of the DR is defined as the maximum input level in a two-tone test for which the third-order IM products do not exceed the noise floor. It can be shown that the maximum input power is:

$$P_{in,Max} = \frac{2P_{IP3} + F}{3}$$

where F = -174 dBm + NF + 10 log B (noise floor)

From these definitions, the SFDR is the difference (in dB) between $P_{in,max}$ and $P_{in,min}$:

$$\begin{aligned} SFDR &= \frac{2P_{IIP3} + F}{3} - (F + SNR_{min}) \\ &= \frac{2(P_{IIP3} - F)}{3} - SNR_{min} \end{aligned}$$

The SFDR represents the maximum relative level of interferers that a receiver can tolerate while producing an acceptable signal quality from a small input level.

The output of the transmitter may also contain harmonics, spurs, and thermal noise, potentially interfering with the users in other channels. Harmonics are generated in the modulator and the power amplifier (PA), while spurs arise from mixers, oscillators, and parasitic resonances and nonlinearities in the PA.

In our frequency division duplexed (FDD) system, output spurs that fall in the receive band can be particularly troublesome because the front end filter does not attenuate these signals. In addition, the front end LNA amplifies these large signals, thereby de-sensitizing the LNA. This is known as receiver desensitization.

Chapter 3 System Description

The RDRN radio being implemented in this research must be a flexible, modular design, capable of adaptively controlling selectable characteristics of the radios performance. In addition, it must rely heavily on software control for reprogramming and implementation of its adaptive control algorithms. In addition, it must support multiple RF front ends.

The radio must meet a number of requirements as dictated by the goals of the RDRN research. The following tables list the requirements of the transmit and receiver modules.

Table 3 – Transmitter Requirements

Transmitter Requirements
• Transmit within RF bands at 5.3 GHz, 2.4 GHz, and 1.2 GHz • Frequency Agility – ability to select from multiple channels within each RF band • Average power output of +30 dBm (1W) from each antenna • RF Bandwidth = 100 MHz • Channel BW = 10 MHz • Selectable QPSK or BPSK modulation • Bit rates from 1 Mbps to 8 Mbps • Multiple Access: FDD with TDMA/FDMA • Transmit Power Control (2 dB steps from 5 dBm to 25 dBm) • Transmit Power ON/OFF capability (TDMA) • TxData received from PowerPC • LO Spurious level ≤ -28 dBm • LO Phase Noise ≤ -155 dBc/Hz at 50 MHZ offset

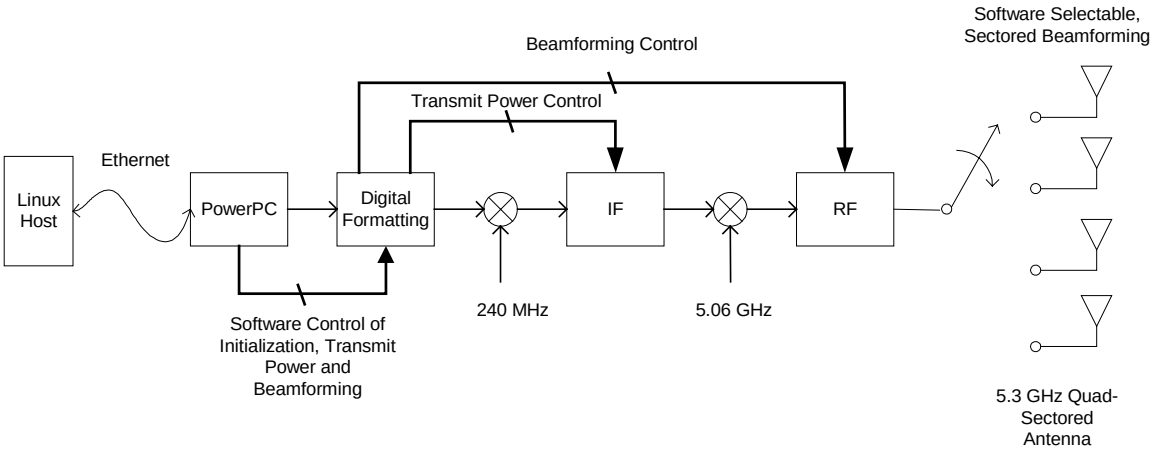

Figure 9 – Transmitter Block Diagram

The figure above shows the complete transmitter subsystem. Data to be transmitted originates from the PowerPC. The symbol rate can be selected for 1, 2 or 4 MSymbols/sec. For QPSK this translates to a maximum 8 Mbps bit rate, while for BPSK this is 4 Mbps. There are two mixing stages, one at the IF (240 MHz), and another at the RF (5.3 GHz). The omni directional antenna is implemented with four patch antennas. This patch array is controlled by the PowerPC. The PowerPC can selectively choose which patch to radiate.

Chapter Four presents a detailed discussion of each of the sections shown above.

Table 4 – Receiver Requirements

Receiver Requirements
• Receive within RF bands at 5.3 GHz, 2.4 GHz and 1.2 GHz • Frequency Agility – ability to select from four channels within RF band • Full Duplex Operation • Digital IF Architecture (subsampling) • Demodulate QPSK or BPSK modulation • Receiver band selectivity = 100 MHz • Receiver channel selectivity = 10 MHz • Bit Rates from 1 Mbps to 8 Mbps • Dynamically configurable – must be a multidimensional radio with varying software profiles (Software Radio) • RxData passed to PowerPC • NF_{max} <= 11 dB • Receiver sensitivity ≤ -85 dBm • Dynamic Range ≥ 65 dB

Worthy of mention above is the dynamic reconfiguration ability of the transceiver. This software control is what makes the radio such a versatile platform. Also, the digital subsampling IF places the ADC as close to the antenna as possible while not sacrificing signal quality. The following diagram, Figure 10, shows a system level view of the receive chain. A thorough discussion of each stage is left until Chapter Five.

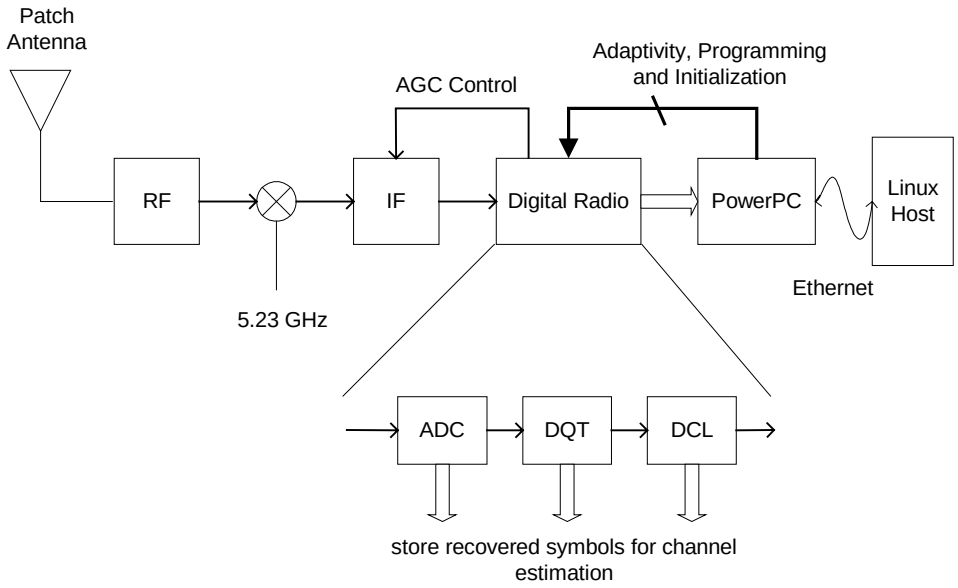

Figure 10 – Receiver Block Diagram

An omni-directional antenna collects RF energy within our desired band. There is one mixing stage which uses high-side injection to downconvert the received signal to our IF, 70 MHz. High-side injection allows us to place the image band at a higher frequency than 70 MHz, in this case 5.37 GHz.. After some amplification and filtering, digital subsampling is used to alias our spectrum to 10 MHz. At this point, two DSPs are used to finish the demodulation process and recover the transmitted symbols.

A thorough discussion of each of the stages in the receive chain is presented in Chapter Five.

Based on the requirements presented in the preceding tables, we based our prototype design heavily of modularity. To incorporate each of the RF bandwidths specified, we had to separately construct multiple RF fronts ends. These front ends connect to a static IF/digital stage that remains the same for each band.

At the heart of the radio, is the PowerPC microprocessor. This processor performs all initialization and programming tasks for the radio. It carries the link

layer adaptation algorithms, and controls the flow of digital data to and from the radio.

Multiple Access Scheme and Bandwidth Utilization

The wireless radio spectrum is a shared resource. Therefore, users must share these resources to allow effective communication. The RDRN radio utilizes frequency-division-duplexing (FDD) on the transmit and receive paths. This facilitates the use of full-duplex communications by using separate channels for transmitting and receiving information.

In addition, if multiple users are sharing the same channel, we have made provisions for TDMA possible. To get a better understanding of how the spectrum is used, Figure 13, below, gives a graphical representation of where the channels lie.

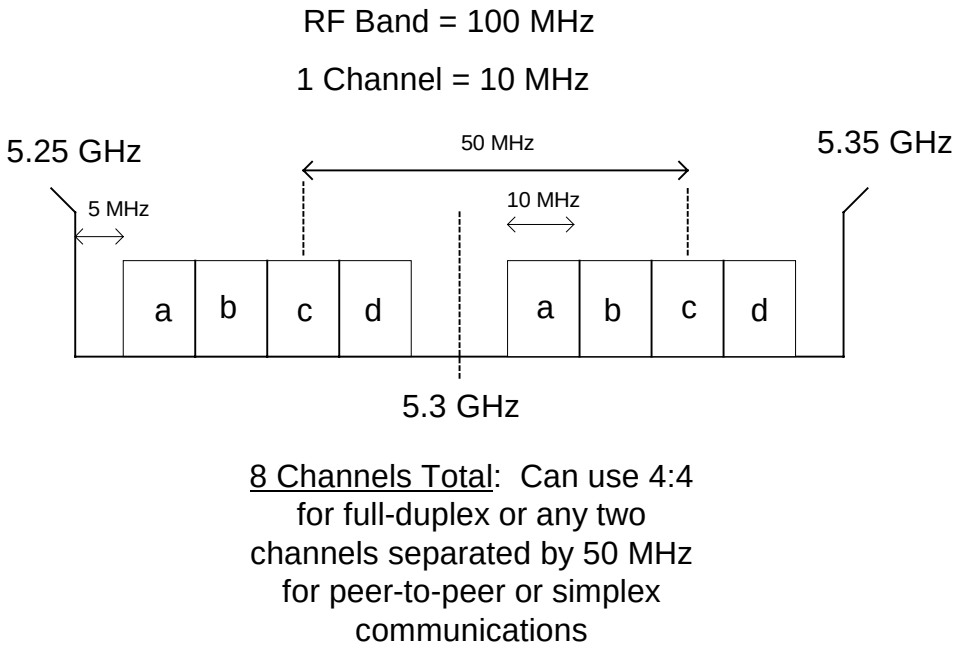

Figure 11 – Spectrum Utilization and Channel Assignment

There are four channels, each with 10 MHz of bandwidth, within each RF band. The forward and reverse channels are separated by 50 MHz. Each RF band is 100 MHz wide. We have designed the RDRN radio to incorporate the use of modular RF stages. Each RF band (i.e., 5.3 GHz, 2.4 GHz and 1.2 GHz) uses the same spectrum assignment as the one depicted above.

The next two chapters detail how the RDRN software radio was implemented. They provide block diagrams for connectivity between components, power levels, gains and/or NF for both the transmitter and the receiver sections. Each diagram will be followed by a step-by-step explanation of each section and how it operates.

The following figure shows the major sub-sections of the radio. As shown, there are three boards: the PowerPC microprocessor (lower board), the main digital/IF radio (middle board), and the modular RF front end (top board).

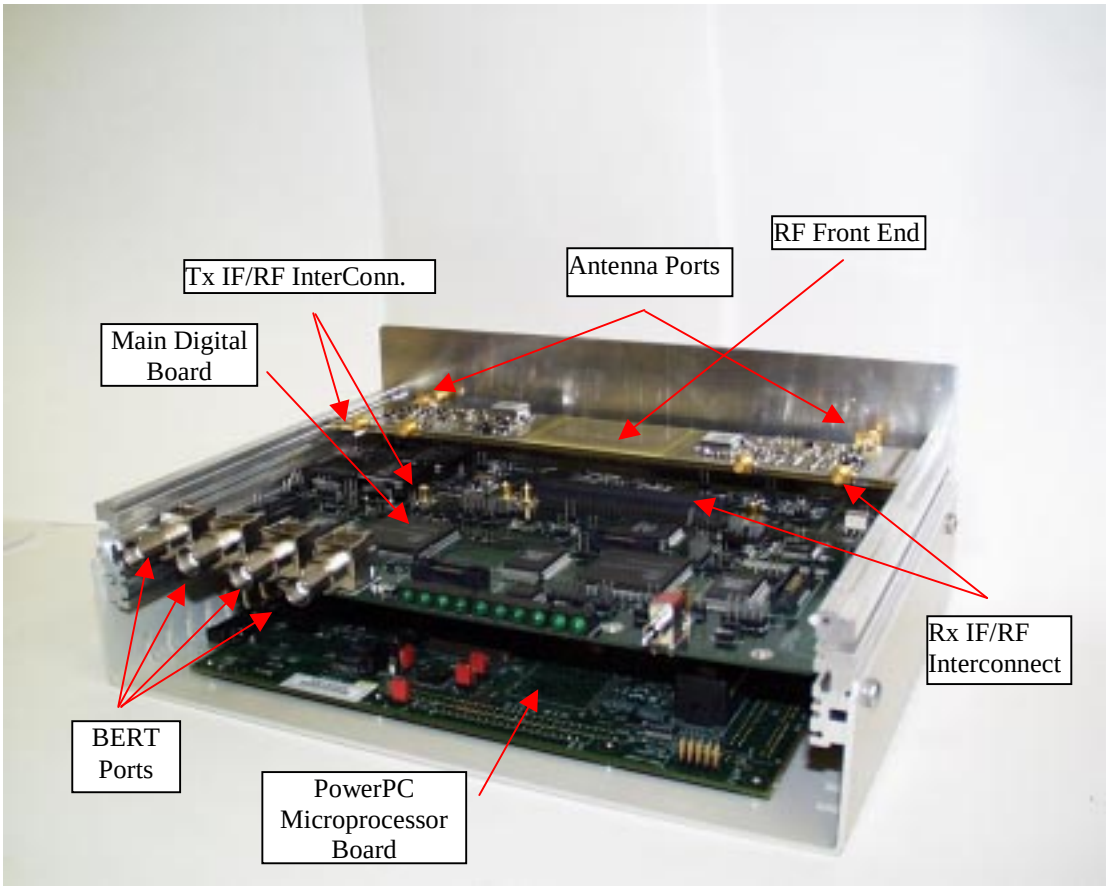

Figure 12 – Completed RDRN2 Radio, including housing

Link Budget

The table on the following page, Table 5, shows a link budget for the RDRN2 radios. This budget is calculated for the 5.3 GHz link with a data rate of 1 Mbps using BPSK modulation. The reader may notice some cells depicted in red. This file is imported as a worksheet from Excel; these red cells depict values that must be entered by the user for a correct margin calculation.

Explanation of Values Calculated for Link Budget

1. Transmit Power from power amplifier (PA).
2. Circuit Losses between the PA and the antenna.
3. TX antenna gain referenced to an isotropic radiator at peak level (i.e., no offset)
4. Summation of 1-3 yields the Effective Isotropic Radiated Power (EIRP).
5. Free space path loss assumes an Additive White Gaussian Noise (AWGN) channel.
6. 2 Ray Model varies with (1/d)^4. Note: this is not used in this budget but is provided as a reference. [16]
7. The fade allowance is the difference between the required Eb/No (the ratio of Energy-per-bit over the Noise Spectral Density) for a fading channel (Rayleigh/Rician) and for the AWGN case. Both extremes are calculated, but it is left to user to pick the fade margin.
8. Allowances are made for polarization losses, weather fades, or any number of other losses.
9. Total path loss is the amount of signal level lost due to propagation from the transmitter to the receiver. It is the sum of 5, 7 and 8.
10. Received Isotropic power refers to the power that would be received if the receiving antenna were isotropic.
11. RX antenna gain referenced to an isotropic radiator at peak level.
12. Edge of coverage loss is due to the off-axis antenna gain (compared to peak gain) and to the increased range for users at the extreme edge of communication coverage (a nominal 2 dB loss is shown here).
13. The input power to the receiver, tallied from 10, 11 and 12.

14. Computes the system equivalent temperature from the system noise figure and background antenna temperature.
15. Figure of merit for a receiver, the G/T is formed from 11 minus 14.
16. Form the noise spectral density by multiplying boltzmann's constant by 14.
17. Form the received signal to noise spectral density by subtracting 16 from 13.
18. The data rate listed in dB-bps.
19. Eb/No = (1/R)(Pr/No). Form this by subtracting 18 from 17.
20. We can achieve the same BER with a decreased Eb/No if we employ coding. None is used here.
21. Accounts for the difference between theoretically predicted detection performance, and the performance of the actual detector.
22. The required Eb/No is derived from the desired BER and the modulation format.
23. The difference between the required Eb/No and the received Eb/No gives us the final margin.

Table 5 – Link Budget for 5.3GHz Link

Link Budget for RDRN2 Radios

Date: April '99 rkilloy@ittc.ukans.edu

fc = 5.30 GHz ** items in "yellow" need
k = -228.6 dBW/Hz to be filled in by user

1.	Transmit Power	1	W	0	dBW
2.	Transmitter Ckt Loss			-1	dB
3.	Transmitter Ant. Gain (peak dBi)			5.0	dBi
4.	Transmit EIRP			4.0	dBW
5.	Free Space Path Loss			-120.6	dB
	Distance [miles]	3			
6.	2Ray Model Path Loss	-135.3	dB		
	ht [m] =	2			
	hr [m] =	2			
	Path Diff. [m] =	0.0017			
7.	Fade Allowance			-4.0	dB
	for Rayleigh Channel, Eb/No |req =	44.0	dB		
	for AWGN Channel, Eb/No |req =	9.10	dB		
8.	Other Losses			-1	dB
	Polarization Loss [dB]	-1			
	Misc. Loss [dB]	0			
9.	Total Path Loss			-125.6	dB
10.	Received Isotropic Power			-121.6	dBW
11.	Receiver Antenna Gain (peak dBi)			5.0	dBi
12.	Edge of Coverage Loss (Pointing Loss)			-1	dB
13.	Received Signal Power			-117.6	dBW
14.	System Equivalent Temperature			36.1	dB-K
	in K =	4106.4			
	Receiver NF [dB]	11.5			
	--> Te =	3806.4			
	Antenna Background Temp. [K]	300			
15.	System G/T			-31.1	$dB\text{-}K^{-1}$
16.	Noise Spectral Density, No=kTs			-192.5	dBW/Hz
17.	Received Pr/No			74.9	dB-Hz
18.	Data Rate, R =			60.0	dB-bps
	in Mbps =	1			
19.	Received Eb/No			14.9	dB
20.	Coding Gain			0.0	dB
21.	Implementation Loss			-1	dB
22.	Required Eb/No			9.10	dB
	desired Pe =	0.00001	(for TCP to work)		
	Pe = Q(Sqrt(2*Eb/No)) for BPSK				
23.	Margin			4.8	dB

Power Requirements

The following table shows measured power requirements for various sections of the radio. It is important to keep in mind that the digital section of the Altera CPLDs can be powered from the +5 V supply or the +3.3 V supply. This supply voltage determines the output levels. This is accomplished by jumping CON3 (Refer to Appendix A, "Setting the Jumpers and Headers") to the appropriate supply. The power requirements for the Power PC (PPC) are also included.

This radio has been designed to run off of a single +12 V supply. There is linear regulator for +9 V and switching regulators for +5 V and +3.3 V. Therefore, most of these measurements were made from a +12 V supply.

Note: These values do not include Power Amplifiers or LNA's. These devices are housed external to this radio. There is a separate 5V supply on board for the sole purpose of powering the PA's.

Table 6 – Power Requirements for Various Sections

	Voltage [V]	Current [A]	Power Needed [W]
Regulators (alone)	12	40m	0.480
Analog Sections	12	120m	1.44
9V sections	12	190m	2.28
Total for Analog Sections	12	350m	4.2
5V Digital. (Quiescent Only)	5V	1.6	8
3.3V Digital. Powering only Altera's (Quiescent Only)	3V	333m	1
Total Power for Digital			9 W
Total Power PPC (Quiescent), datasheet claims 8W			5 W
Total Power	**12 V**	**1.52 A**	**18.2 W**

The following diagram shows the radio interconnections between the PowerPC microprocessor, the digital radio, and the RF front ends.

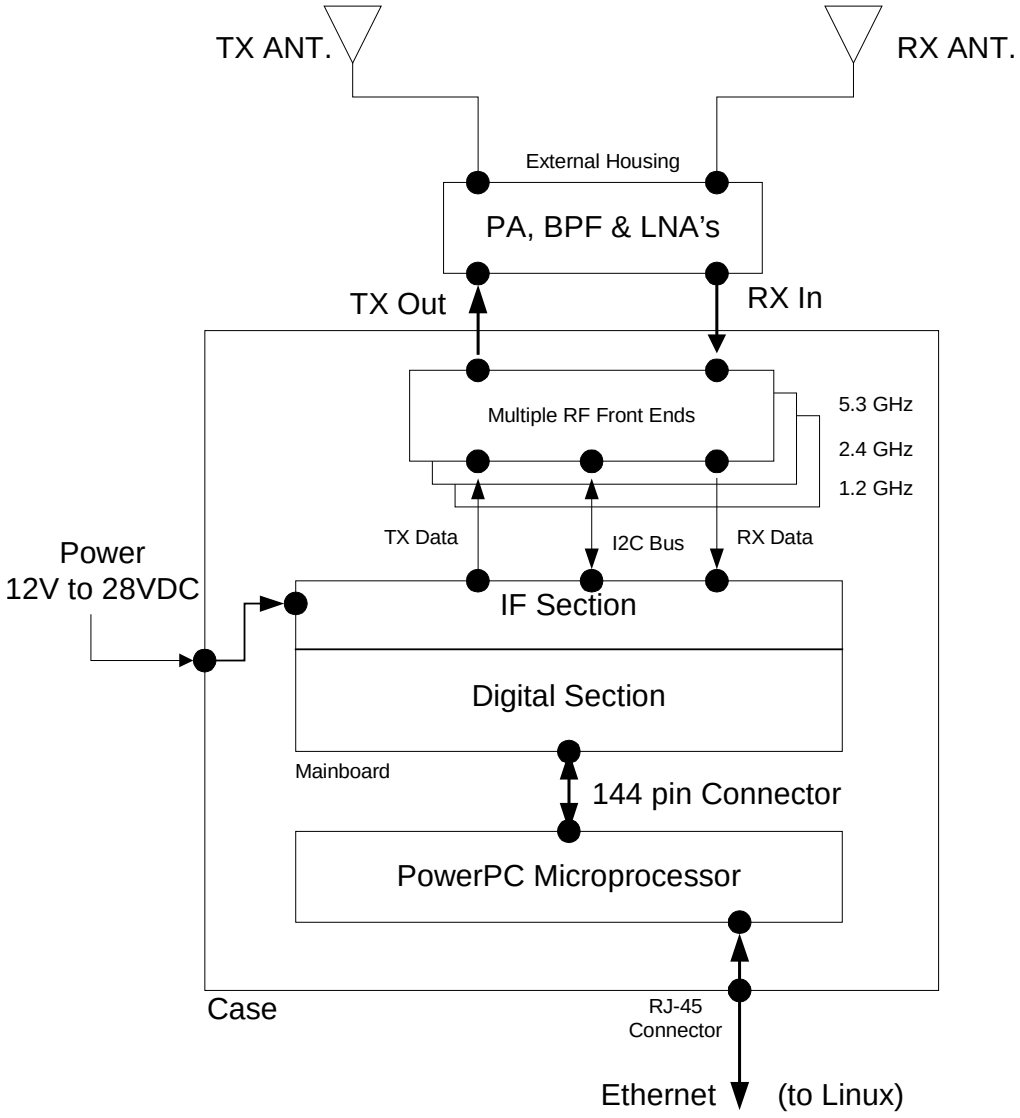

Figure 13 – Radio Interconnect Block Diagram

The following photograph shows the integrated digital and IF circuitry. Important areas are outlined in red.

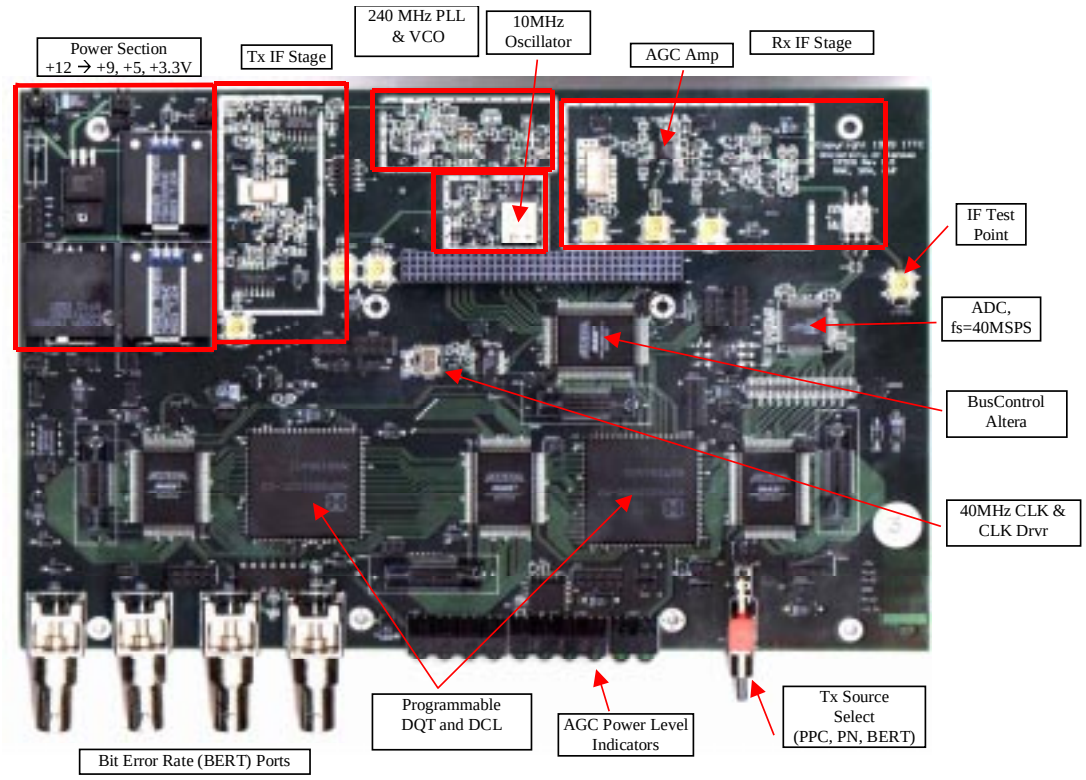

Figure 14 – Digital and IF Subsection

The main board is shown above. This part of the radio contains the power subsystem, the IF section, and the digital sections.

Provisions have been made on this board for testing. There are four BNC ports for use with a bit-error-rate-tester (BERT), a 3-position switch for transmit source selection, and LEDs indicating the AGC amplifier gain. This radio can select between three sources for accepting the transmit data: the PowerPC, the BERT, or an internally generated PN sequence (length = 63 bits).

Altera CPLDs have been used to allow us to monitor each symbol as it travels through the receive chain. These Alteras are located between most digital IC's in the receive path. They simultaneously pass data through and switch data out to special high-density connectors (Mictor) located throughout the board. These Mictor connectors provide a small footprint, high-density way to output up to 64 digital

signals to a logic analyzer. The logic analyzer can store these symbols to hard disk for later analysis.

The following figure shows the modular RF board. This board is for operation within the 5.3 GHz band. There are two other boards similar to the one shown below for operation within the 2.4 GHz and 1.2 GHz bands. This board performs both transmit and receive RF functions.

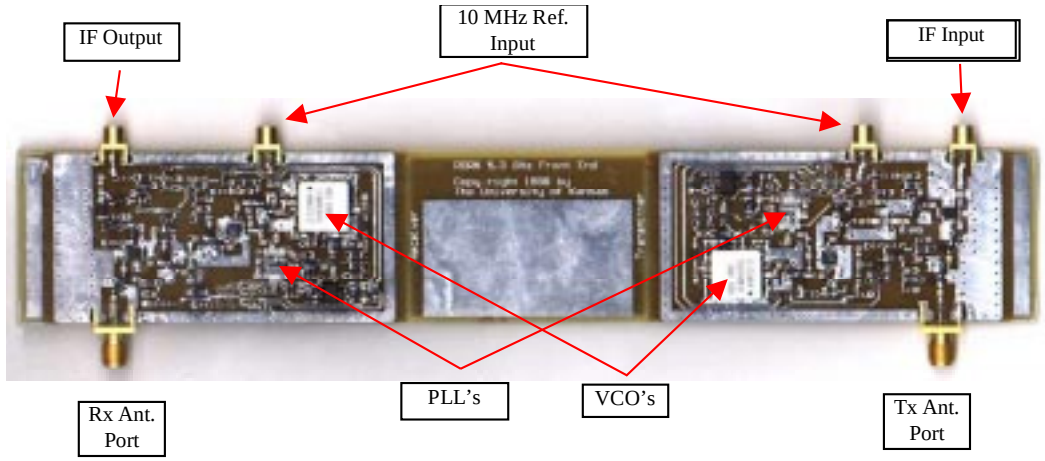

Figure 15 – 5.3 GHz RF Front End

For proper operation, the RF board requires two things. First, the on-board PLL's need to be programmed to operate at a desired frequency. This is accomplished through the I2C bus via a ribbon cable connected to the main board. This I2C bus is controlled via a Complex Programmable Logic Device (CPLD) located on the main board. This IC interfaces with the PowerPC microprocessor to receive updates or changes to the RF carrier frequency. Secondly, the RF board requires a stable, 10 MHz reference signal. This crystal reference oscillator is located on the main digital board and passed to the RF board through an MCX connector. This reference signal is used by the PLL's for carrier generation.

The Control Busses

In order to function as a true software radio, the user must have a way to change given radio parameters when desired. In addition, control algorithms can be developed that will watch certain parameters such as bit-error rate (BER), and automatically control the other characteristics to improve the link quality. These control algorithms would reside in the PowerPC microcontroller and be accessible to the user through the Ethernet interface.

The following is a block diagram, Figure 14, depicting all the control busses used throughout the radio. These busses are used by various transmit or receive sections and allow the user to dynamically adjust a myriad of radio parameters on-the-fly. The figure shows the main sections in the transmit/receive architecture and the data/control busses used by each stage. A thorough discussion of each stage will be discussed following this section.

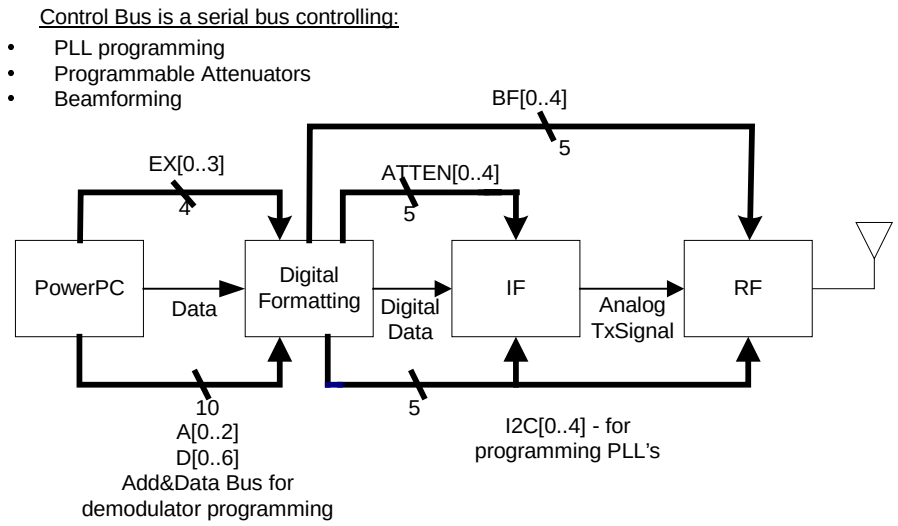

Figure 16 – Radio Sections and Bus Connectivity

At the heart of the radio is the PowerPC microprocessor. This processor controls the functionality of the entire radio. As shown in Figure 16, the PowerPC

sends all the control information and address and data bus information to the digital section for processing.

The control bus, EX[0..3], is a serial bus which controls the programming of the PLL's (i.e., I2C Bus), the programmable attenuators (i.e., Transmit Power Control), and the beamforming. This serial bus is converted into three separate parallel busses(ATTEN[0..4], BF[0..4], and I2C[0..4]) by combinational logic on the main digital board. This combinational logic controls the timing and output of these busses to the rest of the board.

The address and data bus, A[0..2] & D[0..6], output by the PowerPC are control busses used solely for programming the DSP processors (the Harris Chipset) on the main board. These two ICs function as a digital quadrature tuner (DQT) and a digital costas loop (DCL). They must be initialized for proper use. In addition, they can be reprogrammed at any time to change the functionality of the radio. This ability to reprogram "on-the-fly," is what makes the RDRN2 radio such a versatile platform for adaptive algorithm development.

The BusControl Altera serves a major role in the implementation of our high data rate radio. It not only acts as a digital controller for our many busses, but must perform digital switching for the quadrature modulator if needed. It also provides all the functionality needed to link the digital radio with the PowerPC. This Altera must perform the combinational logic necessary to provide all of the following functions:

- Control the programming of the Software Radio (SWR) by adjusting registers located in the Harris Digital Quadrature Tuner and Digital Costas Loop IC's. This is done via the address bus, A[0..2], and data bus, D[0..6].
- Control the programming of the digital PLLs. There are three PLLs located within the radio. They provide critical phase-locked loop functionality for frequencies of 240 MHz, and 5.3 GHz. This programming is done via the I2C bus, I2C[0..4].
- Along with I2C bus functions, miscellaneous control signals arrive to the radio via the EX[0..3] bus. The combinational logic must demultiplex these onto there appropriate busses for distribution throughout the radio. These include a beamforming bus, BF[0..4], and a programmable attenuation bus, ATTEN[0..4].

The following diagram shows the combinational logic implemented inside the BusControl Altera.

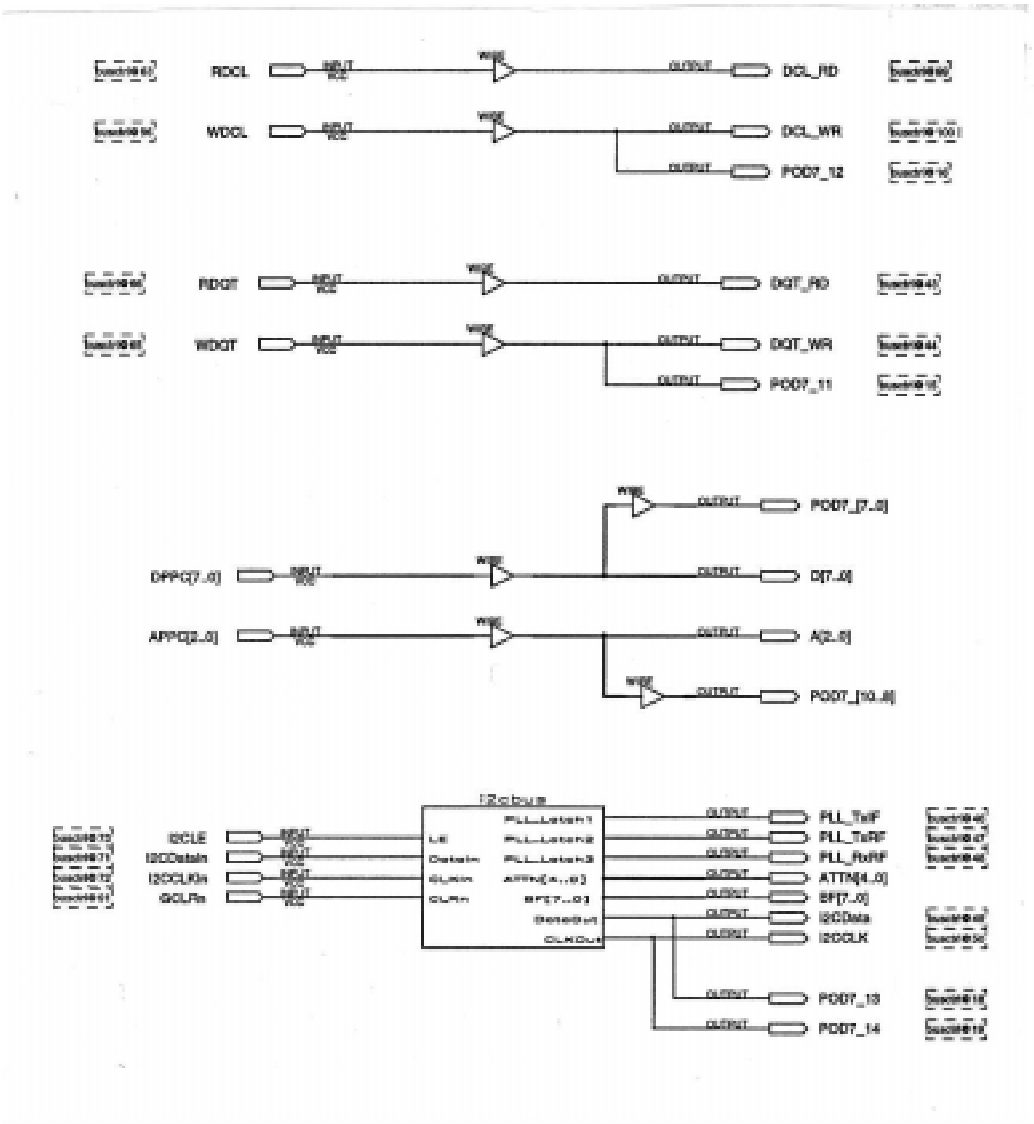

Figure 17 – BusControl Altera Internal Logic Diagram

The combinational logic which makes up the sub-assembly labeled I2Cbus above is shown below.

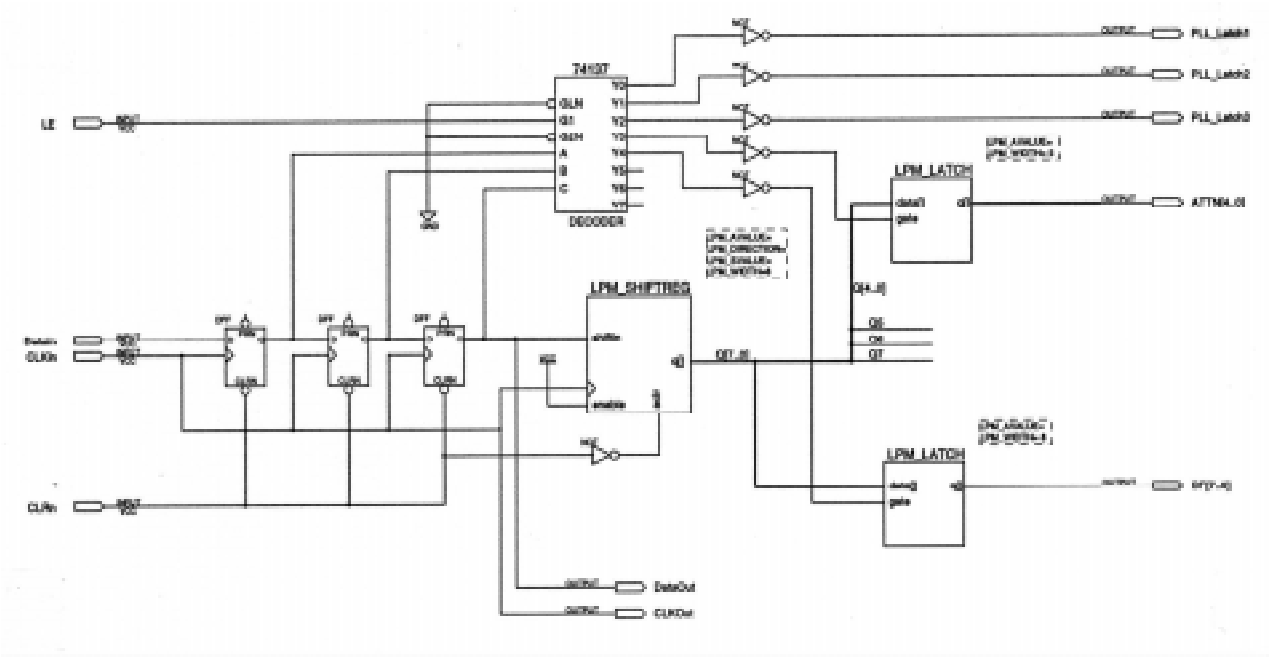

Figure 18 – Combination Logic Implemented by I2Cbus, internal to BusControl

Altera

This diagram shows how the control bus coming from the PowerPC is parsed into its respective parallel busses. The first three flip-flops are used for storing the decoding address bits. These address bits correspond to the particular chip (TxPLL, RxPLL, IFPLL, attenuators or beamforming) for which the data is intended for. The data stream shifts in serially from the DataIn input pin. The last three bits are used for decoding the 7413 decoder. The other bits gets shifted into the shift register and finally shifted out to output port DataOut (for the PLL).

The three address bits determine which device will receive the data stream. For example, if the address is 000h, then PLL number 1 will receive the data. If the address is 001h, the 5 bits right before the address bits will be latched to control the attenuators. If the address is 100h, then the 8 bits before the address bits will be latched to control beam forming.

Chapter 4 Transmitter Implementation

The previous sections in Chapter 3 talked about system level architecture and how the radio is implementing its software control features. This chapter details the transmit signal chain. It includes detailed block diagrams from baseband to the antenna. In addition, it includes signal level analysis for desired, spurious and image signal levels.

Baseband to IF

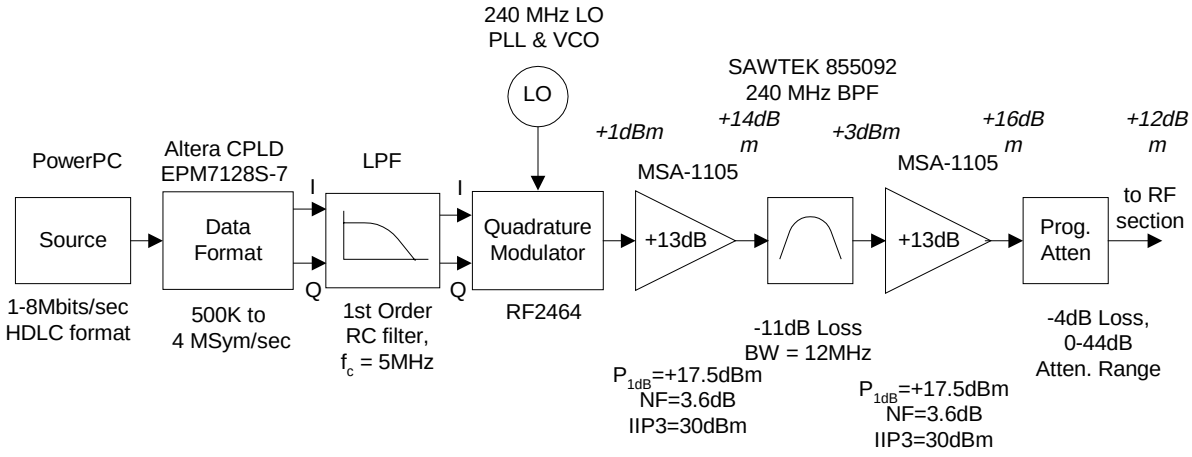

Figure 19 – Transmitter Block Diagram, baseband to IF

This diagram shows the system level functionality as a signal to be transmitted propagates from the PowerPC source through the IF section. This digital and IF section remains the same regardless of the RF stage (5.3GHz, 2.4GHz, or 1.2GHz) being used. Following the IF stage, the transmit signal is sent to the RF board via a coaxial cable. The following table lists the desired and undesired signal levels as they propagate through the IF stage.

Table 7 – Transmit Signal Level for baseband to IF

	IF Stages					
	LPFout	*Qmod*	*Amp1*	*SAWout*	*Amp2*	*Prog.Atten. (= 20dB)*
TX Power (dBm)	-3	+1	+14	+3	+16	-12
LO Spurious (dBm)		-25	-12	-68	-55	-59
TX Image, fi = 480 MHz		-80*	-67	-123	-110	-114

As shown in the table, undesired signal levels are not a problem for the IF stage. Our image and spurious signals are well controlled by the 240 MHz SAW filter. The following figure shows a board level view of the digital and IF transmit sections.

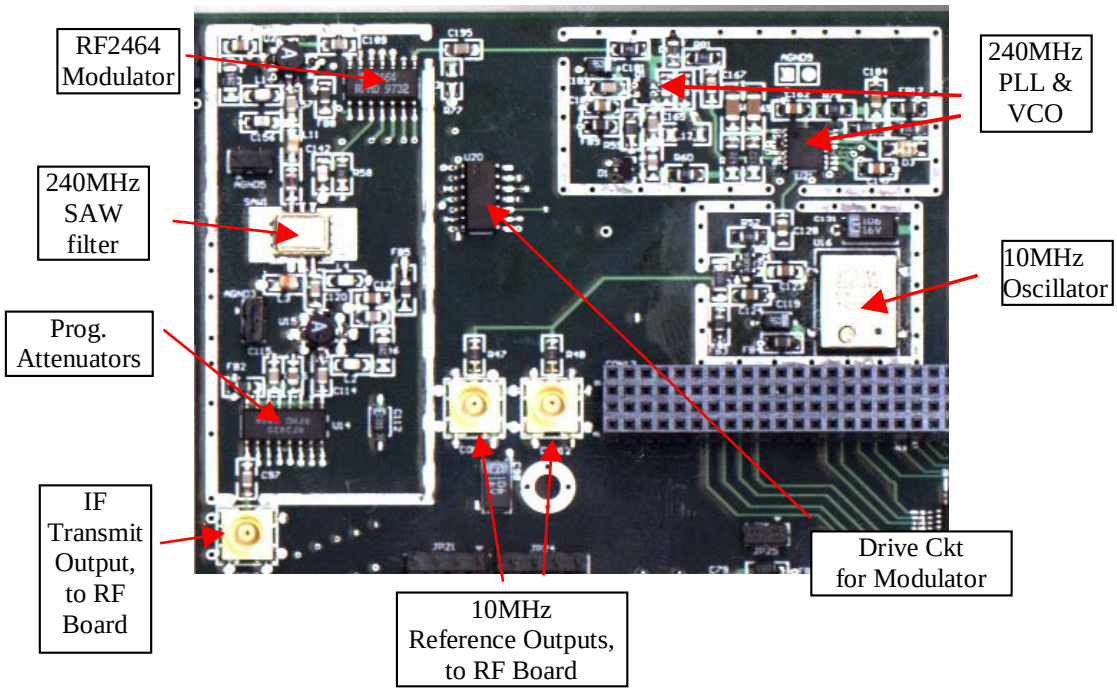

Figure 20 – Close-Up View of Transmit IF section

There are three main components which make up the transmit IF stage. These are a 10 MHz reference oscillator, 240 MHz PLL and VCO and the actual signal chain.

These three sections can be seen above. The 10 MHz reference is output to the RF section along with the transmit signal via the MCX connectors.

Input Stage & Formatting

The transceiver is physically connected to the outside world via an Ethernet link to a Linux host. All data to be transmitted is sent to the radio via this link. The PowerPC microprocessor provides the Ethernet functionality needed. Data to be transmitted over the wireless channel is first converted to an HDLC format and finally differentially encoded. This coded data is then passed to the transmit stage of the radio through the use of a serial port controlled by the PowerPC.

The data stream at this point is 1-4Mbps BPSK or 1-8Mbps QPSK. The BusControl Altera CPLD provides the combinational logic necessary to provide proper switching for the data to be transmitted.

All signals that travel between the digital radio and the PowerPC must pass through the BusControl Altera.

After passing through this section, the data rate is at 1-4Mbps BPSK or 1-4 Mbps QPSK. If BSPK is desired, the I and Q data outputs are simply tied together. From this section, data is passed through to some simple, low-pass filters.

RC Filters

The function of these filters is two-fold. First, they remove unwanted high-frequency noise from the I and Q data signals, and secondly they provide some measure of pulse shaping. In addition, these simple filters provide 20 dB reduction for the sidlobes of our transmitted signal. This is important in the receiver for adjacent channel interference.

Note: This filter is not a complex quadrature filter. It assumes the I and Q data signals are independent, which is a fair assumption. A true quadrature filter

would also have to add and subtract each I and Q component from one another before passing it to the next stage.

IF to RF

Quadrature Modulator

The RF2454 is a VHF quadrature modulator capable of performing GMSK, QPSK, DQPSK and QAM modulation. It also acts as an image reject upconverter. It operates from a single +3 or +5 V supply and provides excellent amplitude and phase balance. It operates from differential I and Q inputs to help reduce the modulation DC offset.

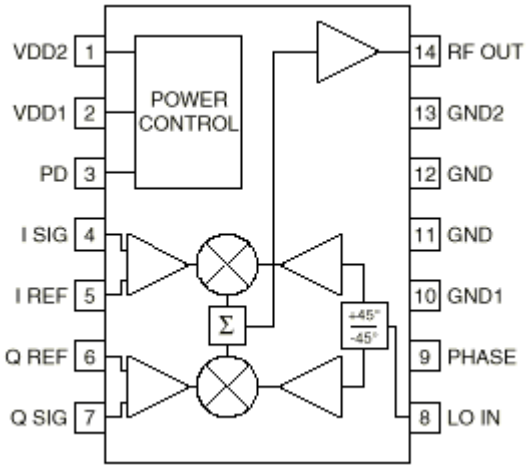

Figure 21 – RF2464 Modulator

In our application, we are using it to modulate a 240MHz carrier with a BPSK or QPSK waveform. The sideband suppression and carrier suppression are around 25 dB in addition to a low broadband noise floor. Refer to Appendix A for a complete schematic of our IF modulator implementation of the RF2464.

240 MHz PLL and VCO Carrier Generation

The modulator section above requires the use of a 240 MHz carrier. In addition, it must have highly suppressed harmonics. The following circuit shows how the 240 MHz carrier is generated.

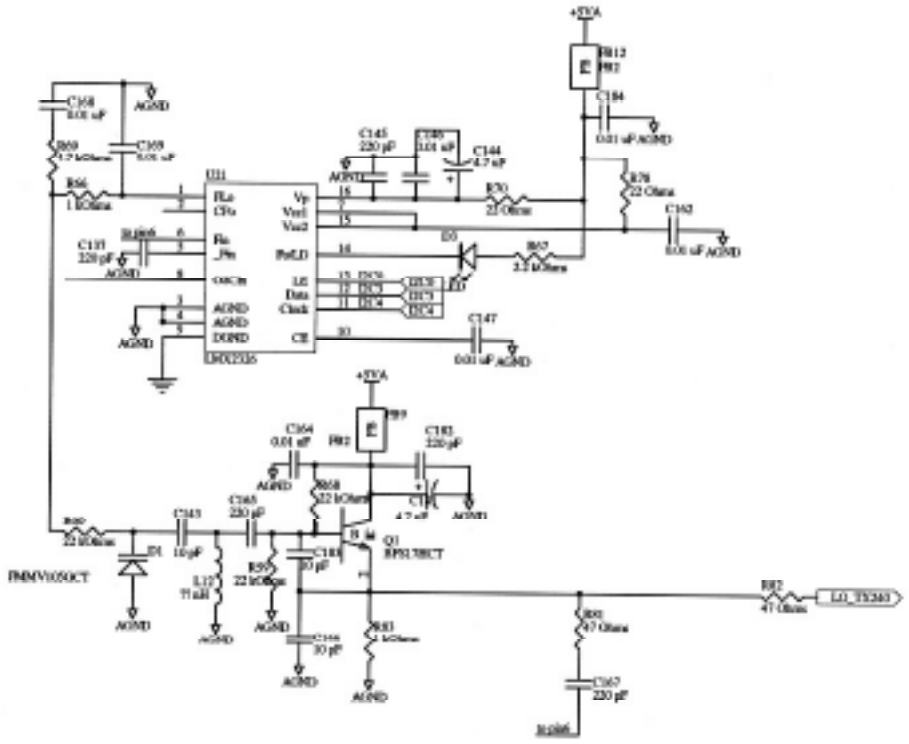

Figure 22 – 240MHz PLL and VCO

The 10MHz oscillator is used as a stable reference for the PLL. This same reference is used by all the PLLs in the radio. It is sent directly to the RF PLLs through connectors located on the main digital board.

The PLL chip uses the reference to generate an exact 240 MHz frequency carrier for the modulator. The input signal to this "phase detector" is the oscillator signal generated by the 10 MHz reference. The PLL generates multiples of the accurate 10 MHz reference frequency. The output of the phase detector is an error signal which is an average DC voltage proportional to the difference in frequency and phase of the input VCO and this multiple of the reference.

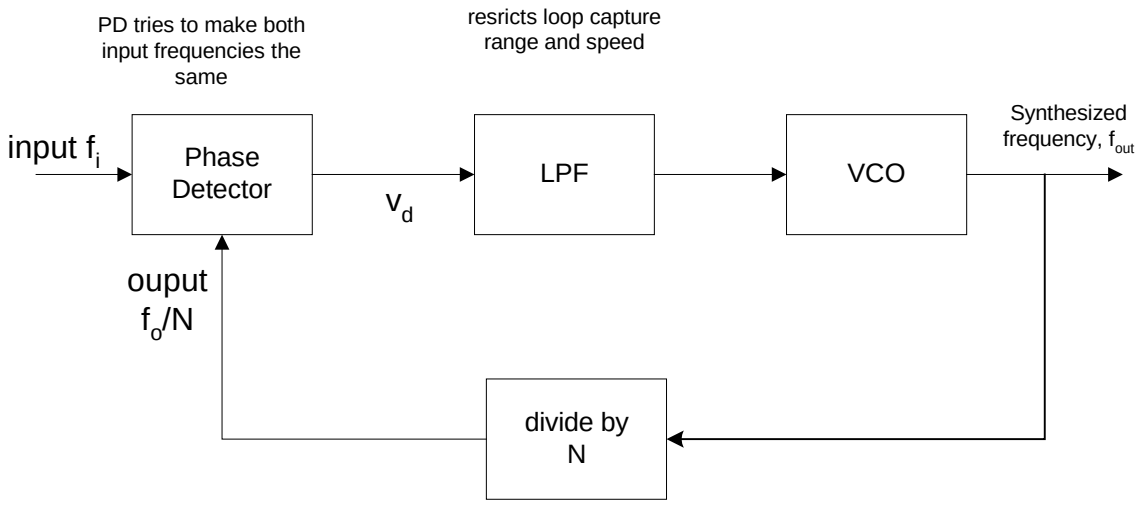

Figure 23 – PLL used as Frequency Synthesizer

The VCO is a free-running oscillator whose frequency is determined by the external resonant circuit (LC network). In our design, the varactor diode in parallel with the LC network acts as a variable capacitor. By changing the bias current on the varactor, one can change its effective capacitance, thereby changing the frequency of operation and frequency range of the oscillator.

The VCO output frequency is fed back to the phase detector, as shown in Figure 23, where it is compared with the frequency of the input signal. This frequency is the stable, carrier frequency we need.

Amplification and SAW Filter

The next three devices in the transmit IF chain prepare the signal for transport to the RF board. The signal to be transmitted needs to be amplified and filtered before being used by the next mixer.

To accomplish this, we used a SAW filter centered at 240 MHz. This filter has approximately 12 MHz of bandwidth and has an insertion loss of 11 dB. The use of a SAW filter allows extremely high Q filters in a relatively small space.

To provide amplification at this stage, we used two MSA 1105s. This amplifier acts as a 50 □ gain block, providing roughly 13 dB of gain at 240 MHz.

Programmable Attenuators

Before leaving the main digital section and proceeding to the RF section, we have incorporated a programmable attenuator within our signal chain to be used in cases where transmit power control (TPC) is desired. The PowerPC can program this device using the ATTEN[0..4] control bus.

This programmable attenuator (RF2420) uses a CMOS compatible 5-bit digital word to control the attenuation level. The attenuator is programmable over a 44 dB range in 2 dB steps with a typical response time of less than 2 nsec. The input and output of the device have a low VSWR 50 Ω match and can operate from DC to 950 MHz. The maximum input power for this device is +17 dBm.

RF Section

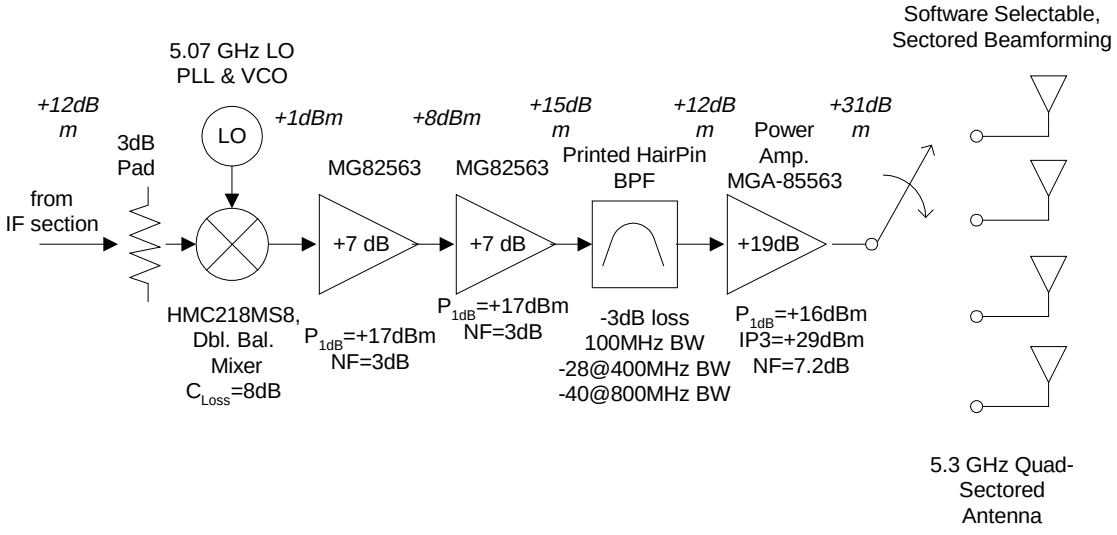

Figure 24 – Transmitter Block Diagram, IF thru RF Section

Table 8 – Transmit Output Signal Levels for IF to RF Stages

	Stage					
	3dB Pad	*Mixer*	*Amp1*	*Amp2*	*Hairpin BPF*	*Amp3*
TX Power	+9 dBm	+1	+8	+15	+12	+31
LO Spurious	+10	< -20dBm	-13	-6	-37	-18
TX Image, fi = 10.36 GHz		-8	-1	+6	-37	-18
LO Phase Noise at 50MHz offset (dBc/Hz)	-160	-168	-161	-154	-157	-138
Noise at 50MHz offset (2MHz BW)		-111	-101	-93	-96	-77

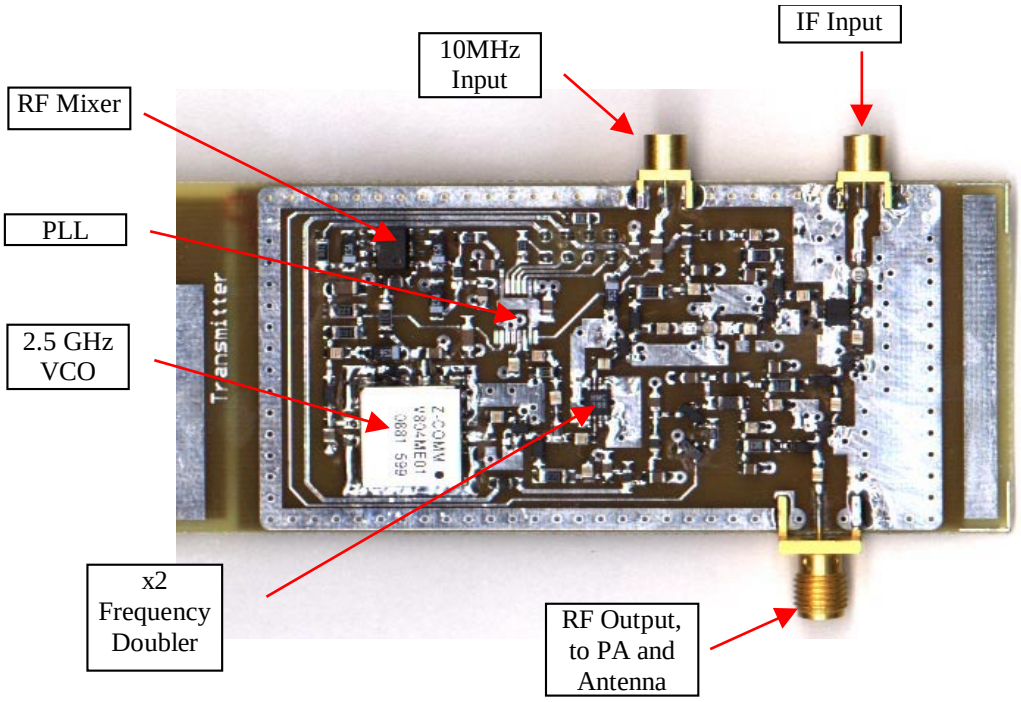

Figure 25 – Close-Up view of the RF Transmitter Stage

From the IF section, our transmit signal is sent to the transmit RF section via a coaxial cable. Once onboard, a 3 dB pad is used to attenuate the signal slightly and provide a slightly better VSWR at the input.

The RF board consists of three main sections: LO generation, RF mixing, and finally amplification/filtering.

LO Generation

To translate our signal from 240 MHz to 5.3 GHz requires the use of a 5.06 GHz local oscillator(LO). As in the IF section, a PLL and VCO are needed to provide a stable signal that will not drift (refer to Figure 23 – PLL/VCO for IF stage).

The only difference between LO generation at the RF and at IF is that we used a higher frequency PLL and purchased a commercial VCO. Although there are no VCO's operating at up to 5.06 GHz, we designed our circuit to generate a stable frequency at 2.53 GHz. With the addition of a frequency doubler (HMC188MS8), we are able to generate a stable 5.06 GHz LO signal to be used by the oscillator.

RF Mixer

The RF Mixer is used to translate our 240 MHz IF to 5.3 GHz. The HMC218MS8 is a GaAs SMT double balanced mixer. The device can be used as an up converter, down converter or bi-phase modulator. It provides a high dynamic range, excellent isolation, and a very small package size. The conversion loss for this device is 7 dB.

RF Amplification and Filtering

This stage in the RF chain is used amplify the transmit signal up to a desired level before power-amplification. The MGA-82563 is a GaAs MMIC amplifier that uses low power and provides approximately 7 dB of gain.

This is a wideband amplifier (0.1 to 6 GHz) who's input and output are matched to 50 Ω across the entire bandwidth. This simplifies our design by eliminating the need for external matching.

This amplifier can operate off of +3 V and has a +17.0 dBm P_{1dB} compression point. This makes it useful as a pre-driver for a power amplifier.

The remaining three devices in the RF signal chain are located externally. These are a printed hairpin BPF, a power-amplifier and the antennas. Refer to Figure 13 for a diagram showing the interconnections between cases.

The printed hairpin filters provide a bandpass function with only minimum signal loss. The insertion loss for these filters is approximately 3 dB while providing relatively high Q with a bandwidth of 100 MHz at 5.3 GHz..

To provide power amplification, the MGA-83563 is used. This amplifier is a GaAs RFIC that offers approximately +19 dB of gain and power output of +22 dBm while operating in saturation.

Quad-Sectored Patch Antenna Array

A quad-sectored, patch array is used as the antenna system. It provides 360° coverage in the horizontal plane. This array is used with the 5.3GHz RF boards. At present, no antenna's have been designed for either the 2.4 or 1.2 GHz systems.

The patch antennas for this system have been designed to provide maximum efficiency and low loss. The antenna patches integrate the BPF with the PA functions on a single substrate for the transmitter. In addition, the BPF and LNA are integrated in a similar fashion for the receive patches. Figure 26, below, show the integrated filter and amplifier for the transmit array patches.

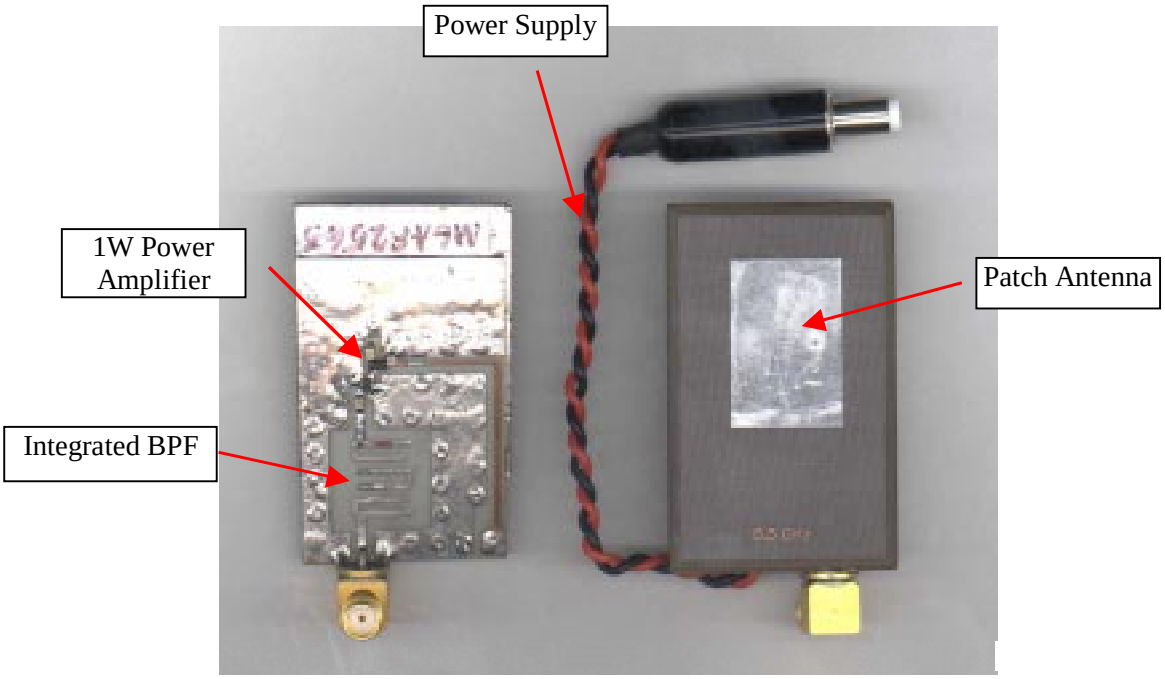

Figure 26 – Integrated Transmit Patch and Power Amplifier

This integration means we have no loss through microwave cables. Typical cables would have appreciable loss (≈3 dB) and degrade the efficiency of the PA. On the receive side, cable loss at the antenna translates directly to an increased NF. If we minimize this loss by integrating the LNA with the patch, we save appreciably on the total system NF for our radio.

The VSWR for the antenna sub-assembly is approximately 1.5:1 over the band we are using.

Chapter 5 Receiver Implementation

As shown in Chapter 2, the architecture for the receiver section is superheterodyne. The receiver can operate with one of three modular front ends: 5.3 GHz, 2.4GHz and 1.2GHz. The IF and digital sections remain the same for each band used. The receiver is a software controlled radio (SWR) with a subsampling, digital IF.

Control of the receiver is accomplished through the PowerPC microprocessor. This microprocessor talks to the receiver through the same set of busses shown for the transmit stage. Refer to Figure 14 for a diagram showing these busses. The user can control a range of parameters which govern the functionality of the radio. The parameters under the users control are listed in Table 2.

The following block set of diagrams show the receiver signal flow from the antenna to baseband. Included below each diagram are tables giving a complete signal level analysis.

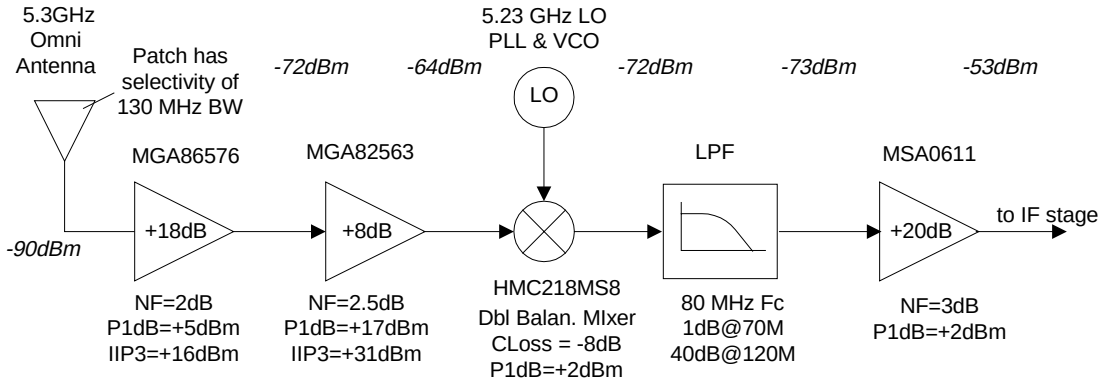

Figure 27 – Receiver Block Diagram (RF to IF)

Signal Level Table

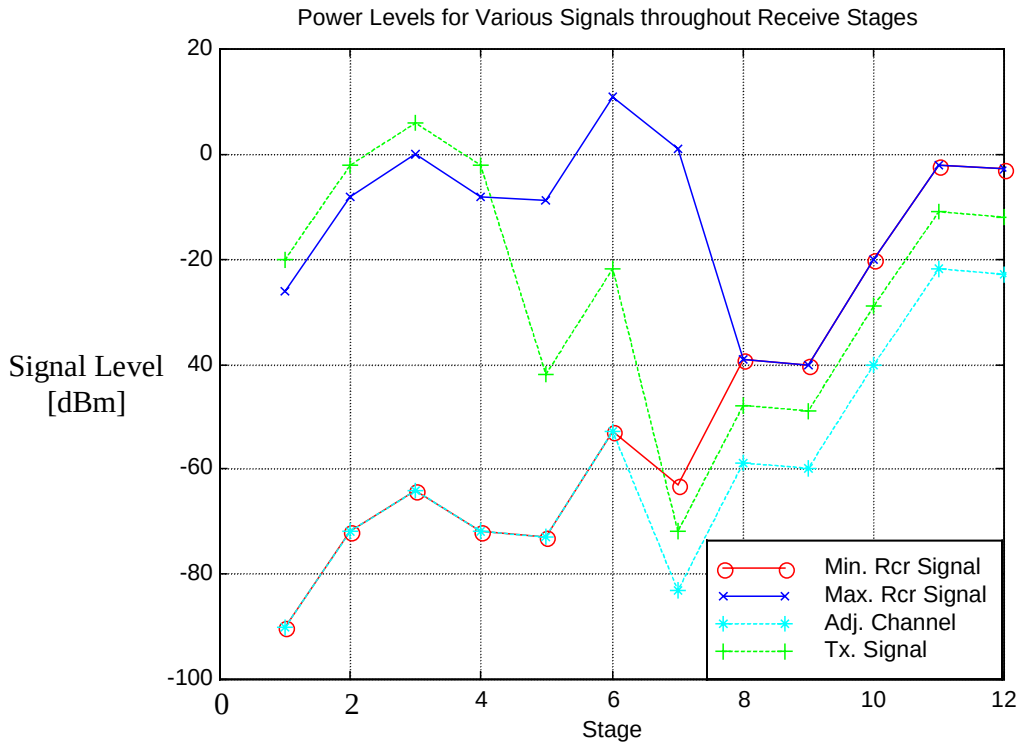

Figure 28 – Receive Signal Levels at Various Sections of Rx Chain

Table 9 – Receive Signal Levels at Various Sections of Rx Chain

P_{TX} @ 1W	Stages within Receive Chain – all Input Power Levels in dBm											
	LNA	82563	Mixer	LPF	0611	SAW	AGC	BPF	0611	2011	BPF	ADC
Desired Signal – Min	-90	-72	-64	-72	-73	-53	-63 (+24)	-39	-40	-20	-2	-3
Desired Signal -- Max.	-26	-8	0	-8	-9	+11	+1 (-40)	"	"	"	"	"
Adjacent Channel at 10MHz Offset	-90	-72	-64	-72	-73	-53	-83 (+24)	-59	-60	-40	-22	-23
	-26	-8	0	-8	-9	+11	-19 (-40)	-59	-60	-40	-22	-23
Tx Signal 50M offset, 1W, 50dB* isolation	-20	-2*	+6	-2 (-40)	-42	-22 (-50)	-72 (+24)	-48	-49	-29	-11	-12
							-72 (-40)	-112	-113	-93	-75	-76

*need 50 dB of isolation or we will desensitize the front end LNA (P_{1dB} = +5dBm)

Noise Figure Graph

The system noise figure is an important parameter which quantifies how much your SNR is degraded by thermal noise as it propagates through the various stages in the receiver. The numbers below were calculated using formulas for cascaded noise figure: [4]

$$NF_{TOT} = 1 + (NF_1 - 1) + \frac{(NF_2 - 1)}{G_1}\Lambda\ \frac{(NF_n - 1)}{G_1 G_2 \Lambda\ G_{n-1}}$$

where NF_n denotes the NF for the *nth* individual stage,
G_n denotes the gain for that same stage,

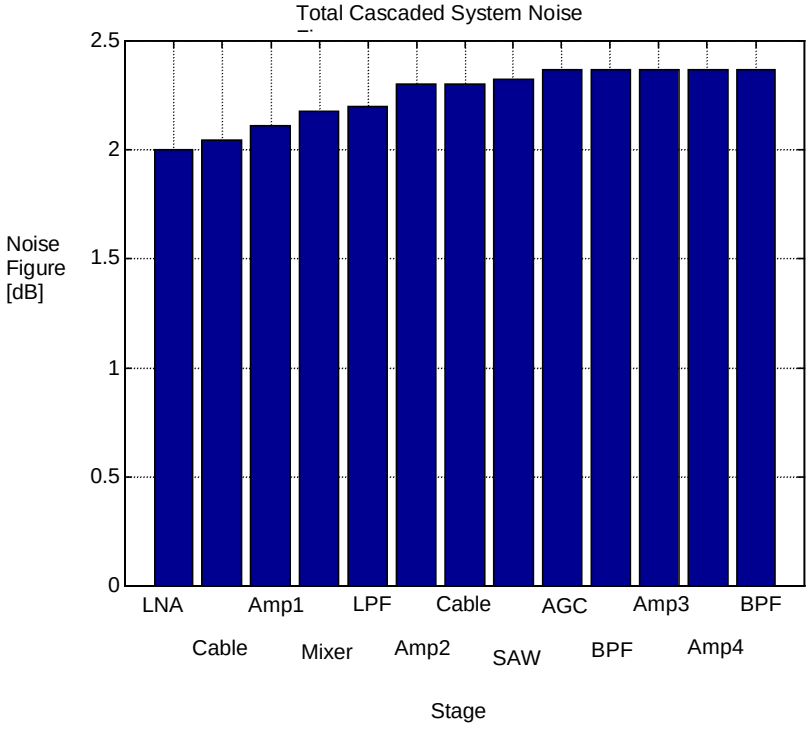

Figure 29 – Noise Figure Throughout System

RF Receiver

Patch Antenna sub-Assembly

The receive antenna for the 5.3 GHz radios consists of a patch antenna and low-noise amplifier (LNA). The patch itself provides 130 MHz of frequency selectivity, eliminating the need for a separate BPF. The LNA is embedded on the back of the patch itself. This eliminates the loss from cables and keeps the system noise figure down.

Since the overall system noise figure (NF) is dominated by the NF of the first few receive elements, it is desirable to choose an LNA which has the smallest noise figure possible. By mounting the LNA on the back of the patch, we reduce loss from filters or cables that would cause degrade the NF.

The LNA used for this receiver is the MGA-86576. This unconditionally stable amplifier allows a wide dynamic range (65 dB) by offering a 2 dB NF coupled with a 16 dBm IP_3 and P_{1dB} of 5.2 dBm. This amplifier provides approximately 18 dB of gain. The amplifier is packaged as a 50 Ω gain block, eliminating the need for conjugate matching of the output section. In addition, this amplifier has a 1.5:1 VSWR across the entire bandwidth.

These first two receive elements (antenna and LNA) are housed externally to the radio. The LNA is mounted to the back of the patch antenna. Signals are sent to the rest of the receive chain through an SMA connector.

Refer to Figure 26 for a diagram showing the integrated patch assembly.

Similar to the transmit side, the receive side RF board consists of three main sections: LO generation, RF mixing, and finally amplification/filtering.

RF Mixer

The RF Mixer is used to downconvert our 5.3 GHz carrier signal down to a more manageable 70 MHz IF. The HMC218MS8 is a GaAs SMT double balanced mixer. This is the same mixer used in for the transmit RF section. The device can be used as an up converter, down converter or bi-phase modulator. It provides a high dynamic input signal range, excellent isolation, and a very small package size. The conversion loss for this device is 8 dB.

LO Generation

To translate our signal from 5.3 GHz to our IF at 70 MHz requires the use of a 5.23 GHz local oscillator. As in the IF section, a PLL and VCO are needed to provide a stable signal that will not drift (refer to Figure 23 – PLL/VCO for IF stage).

The only difference between LO generation at the RF and at IF is that we required a higher frequency PLL, therefore we purchased a commercial VCO (V630ME13). Although there are no commercially viable VCOs operating at 5.23 GHz, we designed our circuit to generate a stable local oscillator at 2.615 GHz. With the addition of a frequency doubler (HMC188MS8), we are able to generate a stable 5.23 GHz LO signal to be used by the mixer.

The following figure shows the RF receiver section.

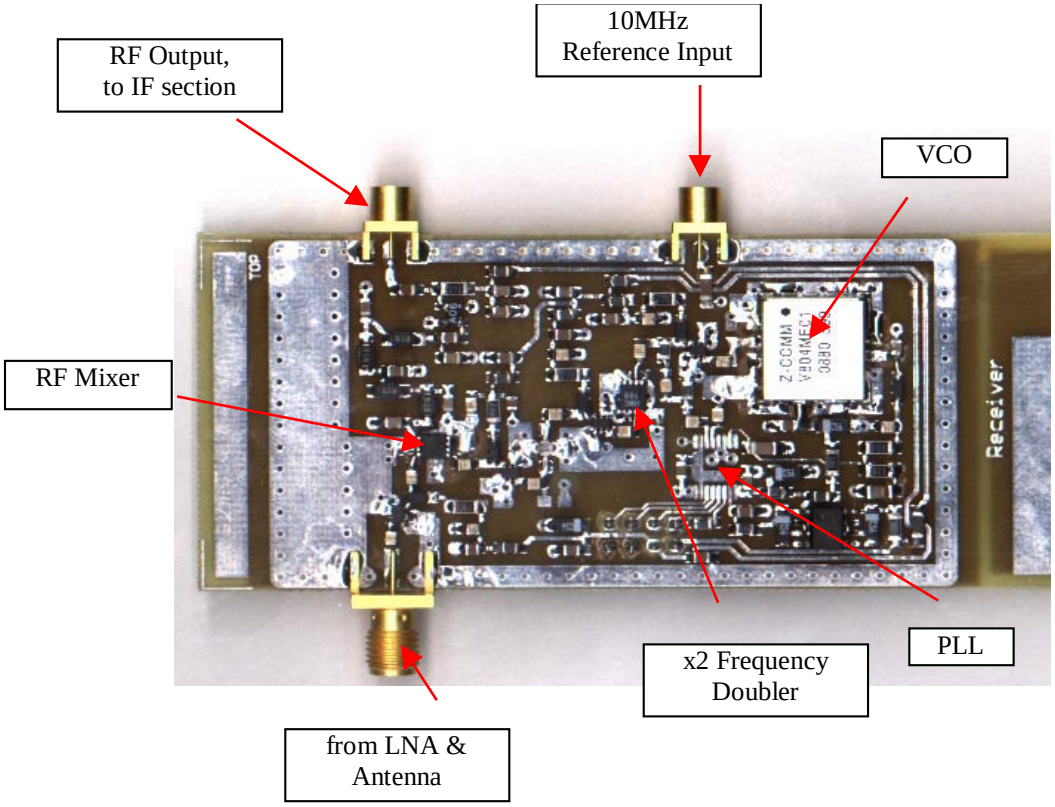

Figure 30 – Close-Up view of RF Receive Section

IF to ADC

Low Pass Filter and Amplification Section

Following the mixer, a low pass filter (LPF) is used to remove undesirable products and high-frequency interferers from our bandwidth of interest. This LPF has a cutoff at 80MHz . It provides 1dB of attenuation at 70MHz and 40dB attenuation at 120 MHz.

The MSA-0611 amplifier follows the LPF and provides +20 dB of gain. It is packaged as a cascadable 50 Ω gain block. This eliminates the need for external matching circuitry. In addition it provides a low noise figure of approximately 3 dB.

From here, our receive signal is sent to the IF section. This IF section is located on the main board. Signals are interconnected through the use of an MCX connector.

The following block diagram continues the receive signal chain. It begins at the IF and extends to the digital section.

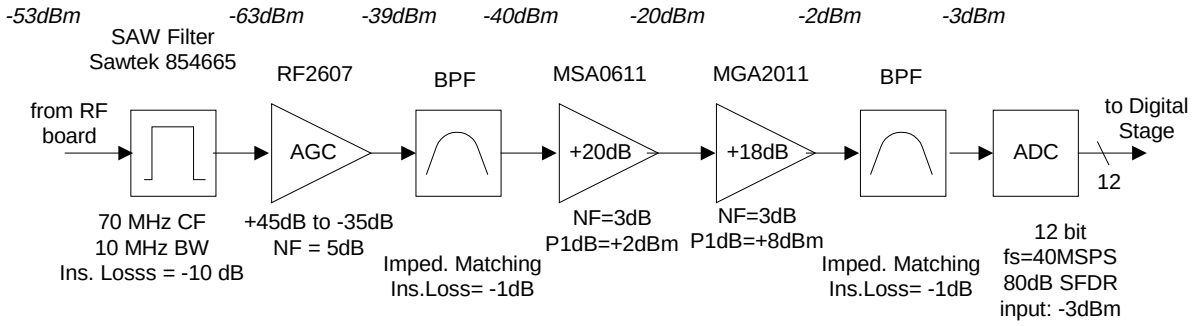

Figure 31 – Receiver Block Diagram (IF to Digital)

SAW Filter

The surface acoustic wave (SAW) filter is a bandpass filter centered at 70 MHz. It has a 10MHz bandwidth with a high insertion loss of 10 dB. The SAW filter is used to remove noise and interferes from our desired signal. It does this extremely well due to its large Q. This large Q manifests itself in a very steep rolloff; the filter is down by 50 dB within 5 MHz of the 3 dB cutoff frequencies.

The SAW device is not a 50 □ part. An input and output matching network is needed. Refer to Appendix B for detailed schematics of these matching sections. The output matching section matches the filter to the AGC section which follows.

AGC Amplifier

The RF2607 is designed to be a complete AGC amplifier for use in the receive section of CDMA or FM cellular applications. It is designed to amplify IF signals

while providing more than 90 dB of dynamic range on the gain control. This ranges from 45 dB of attenuation to 45 dB of gain.

This amplifier specifies a 5 dB noise figure and an IIP_3 of –40 dBm at maximum gain (+40 dB).

The input to this device is not 50 □. For the FM inputs (which we use) it is approximately 850 □ (single-ended). Therefore, the output section of the SAW filter preceding this device is tuned to achieve maximum gain.

The control voltage, VGC, is a DC level ranging from 0.25 V to 3.0 V. This control signal tells the IC how much gain to apply. At a VGC = 1.5 V the chip switches from attenuation to gain. This signal is controlled from the digital quadrature tuner (DQT), Harris HS50110. The input to the DQT has a level detector which outputs a "1" if the level is above a user-defined threshold, or it outputs a "0" if below. This signal is integrated and voltage divided via a simple RC network. This integrated signal provides the DC level needed to provide proper gain control.

The following diagram is a close-up view of the IF receive chain.

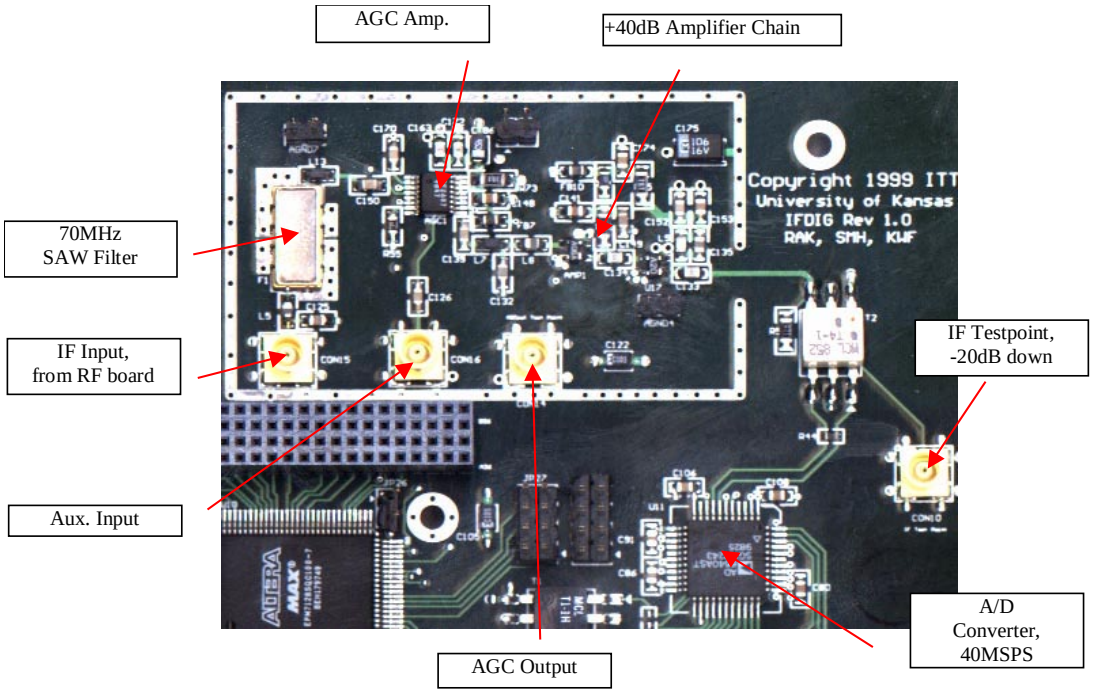

Figure 32 – Close-Up view of Receiver IF Stage

IF Amplification Chain

Following the AGC amplifier is a series of two amplifiers. These amplifiers are needed to provide the proper signal level to the analog-to-digital converter (ADC).

Together, the MSA-0611 and the MSA-2011, provide roughly +40 dB of gain. Each amplifier acts as a high-performance, 50 Ω cascadable amplifier. There is no need for matching networks.

ADC to Baseband

ADC Input Conditioning

A BPF follows the IF receive gain chain. This is a simple LC resonant tank circuit (Q □ 1.1). It has a resonant frequency around $f_o = \frac{1}{2\pi\sqrt{LC}}$ = 73 MHz and a bandwidth of approximately $BW = \frac{\omega_o}{Q} \approx$ 60 MHz.. It serves to remove any undesired harmonics, interferers and noise before the received signal is presented to the ADC.

Analog to Digital Converter (ADC)

The AD6640 is a high-speed, low power monolithic 12-bit analog to digital converter. All functions, such as track-and-hold and reference, are included on chip. In addition, the AD6640 runs from a single +5 V or +3.3 V supply and provides CMOS compatible digital outputs.

The ADC covers approximately 80 dB of spurious free dynamic range with a maximum sampling frequency of 65 MSPS. In addition, the converter offers a low 710 mW of power dissipation and outputs of either 3.3 V or 5 V. [2]

As discussed in Chapter 2, we are using a technique known as digital subsampling, or IF sampling. We are sampling our input signal (70 MHz center, 10 MHz bandwidth) at 40 MSPS. Since the bandwidth of our signal of interest is not too large, there are ranges of frequencies we can sample at to purposely introduce aliasing. This has the effect of translating our band of interest from 70 MHz down to 10 MHz. It is important to remove unwanted harmonics and high frequency terms if IF sampling is used. These higher frequency components will be aliased into our pass band and possibly cause bit errors if not removed. Final translation to DC is accomplished within the DQT.

Each analog input is centered around 2.4 V and should swing □ 0.5 V. Since these input are complementary, inputs should be no more than 2.0 V peak-to-peak. The input signal level has been designed to be no more than –3 dBm at the input. This corresponds to approximately 1 Vrms at the input.

Best performance is obtained by driving the encode and analog input pins differentially. However, the AD6640 is also designed to interface with TTL and CMOS logic families. These signals need to be clean and free from jitter.

A 1:4 impedance ratio transformer is used on the input stage. This serves three purposes: first, it isolates the ADC from the rest of the analog systems. Second, since the input impedance to the ADC is 1900 Ω, the 1:4 impedance transformer performs a 50 Ω impedance transformation in conjunction with the 280 Ω resistor , $Z_L = \frac{1}{n^2}\left(R_{260} \,//\, 1900\Omega\right) = 57\Omega$, allowing the ADC to make full use of it's dynamic range. Lastly, it serves to AC couple the signals, taking them from a single-ended mode to a differential mode.

The clock signal (ENCODE input) is also differentially driven via a 1:1 transformer.

The output of the ADC is a 12-bit parallel digital CMOS-compatible word, coded as twos complement.

Care was taken in designing the data receivers for the AD6640. We designed our outputs to drive a series of 348 Ω resistors followed by a high-speed register for latching. The AD6640 has a unique constant slew rate output stage. The output slew rate is about 1 V/ns independent of output loading. A typical CMOS gate combined with PCB trace will have a load of approximately 10 pF. Therefore, as each bit switches, 10 mA of dynamic current per bit will flow through the digital output stage. A full-scale transition can cause up to 120 mA of current to flow through the digital output stage. The series resistors will minimize the output currents that can flow. These switching currents are confined between ground and the DVcc pin. TTL gates have been avoided since they can appreciably add to the dynamic switching currents of the AD6640.

For layout, the ADC is treated as if it were an analog part. All of the grounds for the chip are tied to the analog ground plane, while the supply voltages are separated through an inductor tied to the same supply. This technique helps to reduce noise in the analog system. [3]

The following block diagram shows the complete digital section.

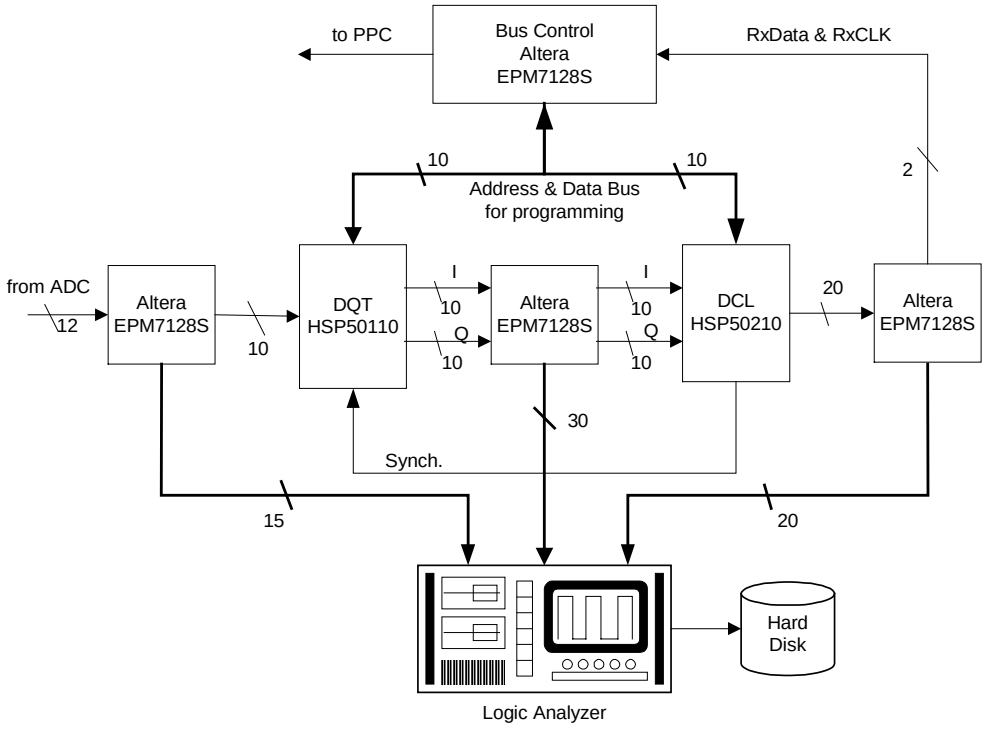

Figure 33 – Receiver Block Diagram (Baseband)

At all stages within the digital section, data can be output to a logic analyzer and data stored to disk. This allows the user to capture the received symbols anywhere within the digital receive chain. It is worth mentioning that coherence is not established until after the DQT. Therefore, any symbols stored before that are not phase or carrier coherent and must be dealt with accordingly.

Signal Processing1 Altera

This purpose of this Altera is to provide a buffer from the ADC. All of the digital words output from the ADC are simply passed straight through this CPLD. In addition, they are routed to one of the high-density Mictor connectors. This allows these digital symbols to be recorded to disk for later analysis if desired.

Digital Quadrature Tuner, the HSP50110

The Digital Quadrature Tuner (DQT) provides many of the functions required for digital demodulation. These functions include carrier LO generation and mixing, baseband sampling, programmable bandwidth filtering, baseband AGC, and IF AGC error detection. Serial control inputs are provided which can be used to interface with external symbol and carrier tracking loops. These elements make the DQT ideal demodulator applications with multiple operational modes or data rates. [6]

In our design, we are using the DQT in conjunction with the HSP50210 Digital Costas Loop to function as a demodulator for BPSK/QPSK signals. The following block diagram shows the internal functions of the DQT.

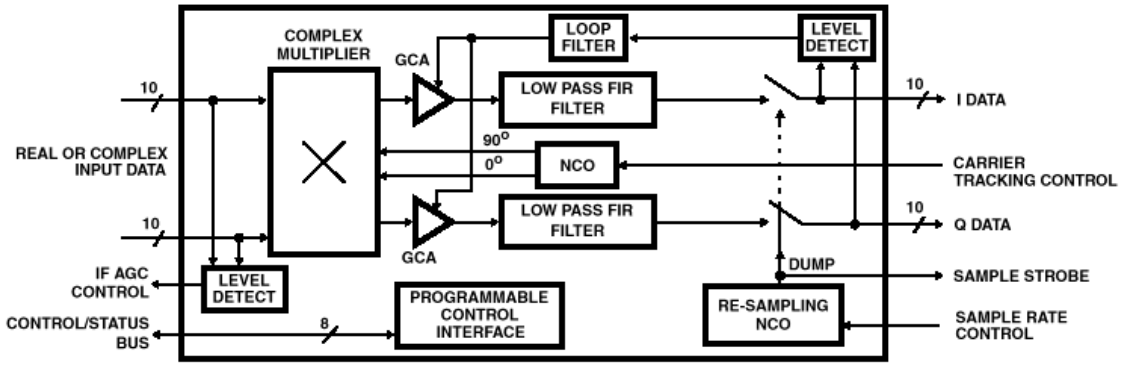

Figure 34 - HSP50110 Digital Quadrature Tuner

The DQT processes a real or complex input digitized at rates up to 52 MSPS. The channel of interest is shifted to DC by a complex multiplication with the internal LO. The quadrature LO is generated by a numerically controlled oscillator with a tuning resolution of 0.012 Hz. The output of the complex multiplier is gain corrected and fed into identical low pass FIR filters. Each filter is comprised of a decimating low pass filter followed by an optional compensations filter. The decimating low pass filter is a 3-stage cascaded-integrator-comb (CIC) filter. The CIC filter can be

configured as an integrate and dump filter or a third order CIC filter with a (sin(X)/X)^3 response. Compensation filters are provided to flatten the response of the CIC. If none of the filtering options are desired, they may be bypassed. The filter bandwidth is set by the decimation rate of the CIC filter. The decimation rate may be fixed or adjusted dynamically by a symbol tracking loop to synchronize the output samples to symbol boundaries. The decimation rate may range from 1-4096. An internal AGC loop is provided to maintain the output magnitude at a desired level. Also, an input level detector can be used to supply error signal for an external IF AGC loop closed around the A/D.

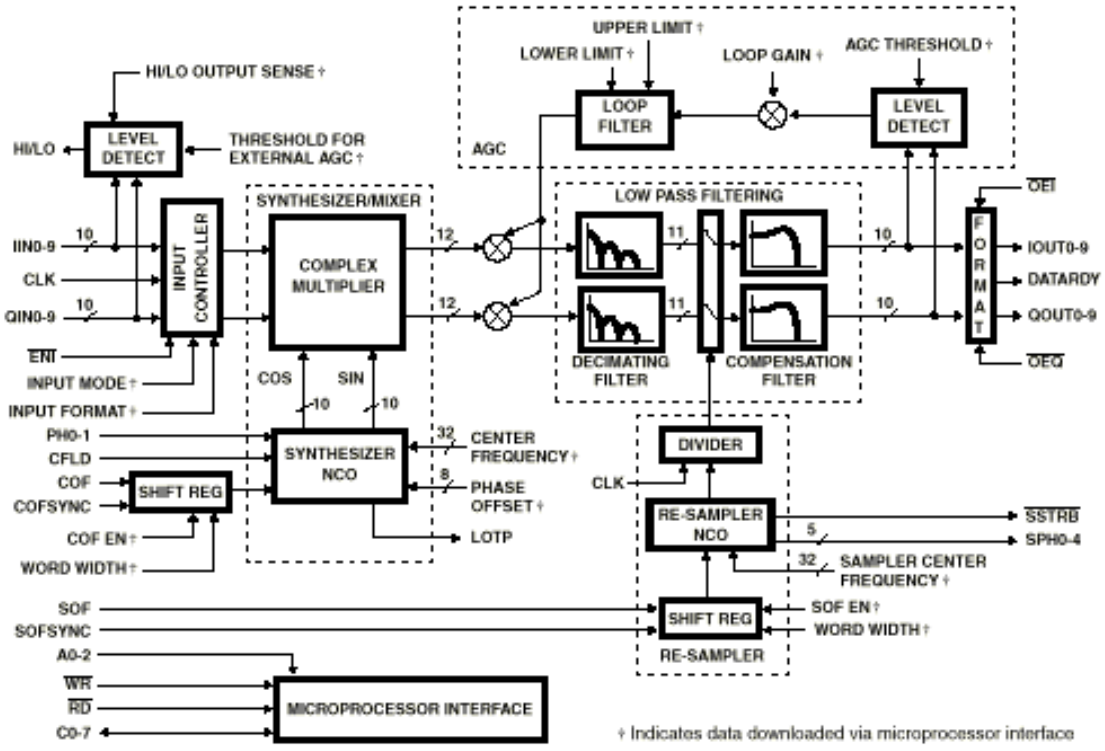

Figure 35 – Functional Block Diagram of HSP50110

The DQT output is provided in serial or parallel formats. In this design, however, I am using the parallel format. In addition, this device is configurable over a general purpose 8-bit parallel bi-directional microprocessor control bus.

Microprocessor Interface

The Microprocessor interface is used to write the HSP50210s control registers and monitor various read points within the demodulator. Data written to the interface is loaded into a set of four 8-bit holding registers, one write address register, or one read address register. These registers are accessed via the 3-bit address bus and an 8-bit data bus.

Data is read from an internal status register and a series of output holding registers. The output holding registers range in size from 8 to 32 bits, and their contents are multiplexed out a bit at a time on the data bus by controlling the address bus and asserting the /RD signal. Refer to the Appendix A for a more thorough discussion of how to program these parts.

Signal Processing2 Altera

This purpose of this Altera is to provide a means for outputting the digital samples to a connector. The digital output words are passed straight through to the DCL and also to the connector. This allows these digital symbols to be recorded to disk for later analysis if desired.

Digital Costas Loop, the HSP50210

As stated in the previous section, the HSP50210 Digital Costas Loop (DCL) is used in conjunction with the DQT to provide coherent detection of a BSPK/QPSK

signal. The DCL performs many of the baseband processing tasks required for the demodulation of BPSK or QPSK waveforms. These tasks include matched filtering, carrier tracking, symbol synchronization, AGC, and soft decision slicing. The DCL is designed for use with the DQT to provide a two chip solution for digital down conversion and demodulation. [7]

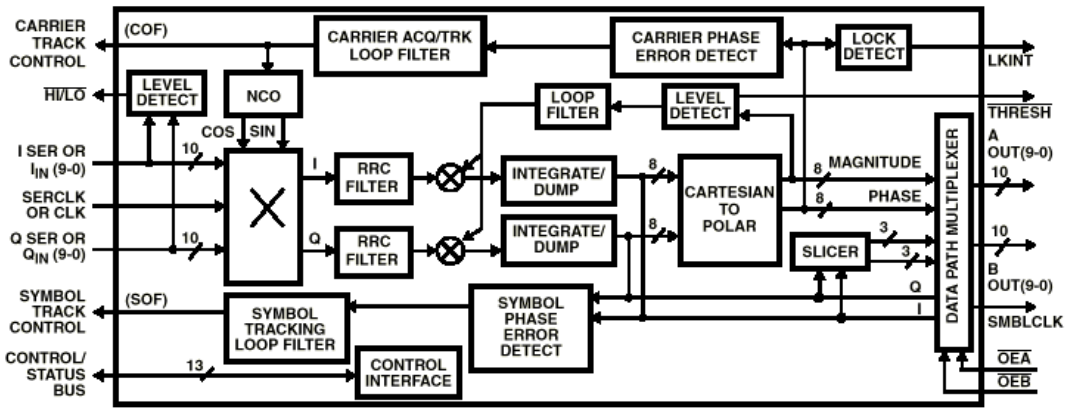

Figure 36 - HSP50210 Digital Costas Loop

The DCL processes the in-phase and quadrature components of a baseband signal which have been digitized to 10 bits. As shown in the block diagram, the main signal path consists of a complex multiplier, selectable matched filters, gain multipliers, Cartesian-to-polar converter, and soft decision slicer.

The complex multiplier mixes the I and Q inputs with the output of a quadrature NCO. Following the mix function, selectable matched filters are provided which perform integrate and dump or root raised cosine filtering (~ 0.40). The matched filter output is routed to the slicer, which generates 3-bit soft decision, and to the Cartesian-to-polar converter, which generates the magnitude and phase terms required by the AGC and Carrier Tracking Loops.

The phase-locked-loop (PLL) system solution is completed by the HSP50210 error detectors and second order Loop Filters that provide carrier tracking and symbol synchronization signals. Since we are using the DCL in conjunction with the DQT, these control loops are closed through a serial interface between the two parts. To maintain the demodulator performance with varying signal power and SNR, an internal AGC loop is provided to establish an optimal signal level at the input to the slicer and to the cartesian-to-polar converter.

The following block diagram shows a more detailed view of the internal functions performed by the Digital Costas Loop.

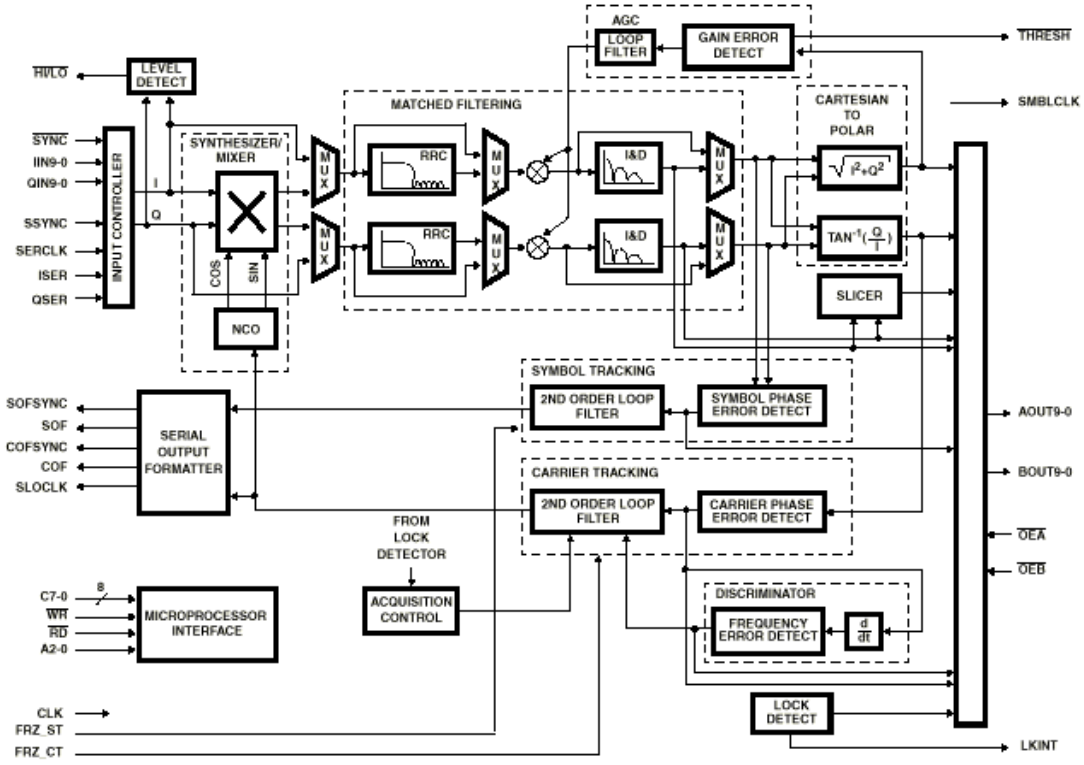

Figure 37 – Functional Block Diagram of HSP50210

Microprocessor Interface

The Microprocessor interface is used to write the HSP50210s control registers and monitor various read points within the demodulator. Data written to the interface is loaded into a set of four 8-bit holding registers, one write address register, or one read address register. These registers are accessed via the 3-bit address bus and an 8-bit data bus.

Data is read from an internal status register and a series of output holding registers. The output holding registers range in size from 8 to 32 bits, and their contents are multiplexed out a bit at a time on the data bus by controlling the address bus and asserting the /RD signal. Refer to Appendix A for a more thorough discussion of how to program this component.

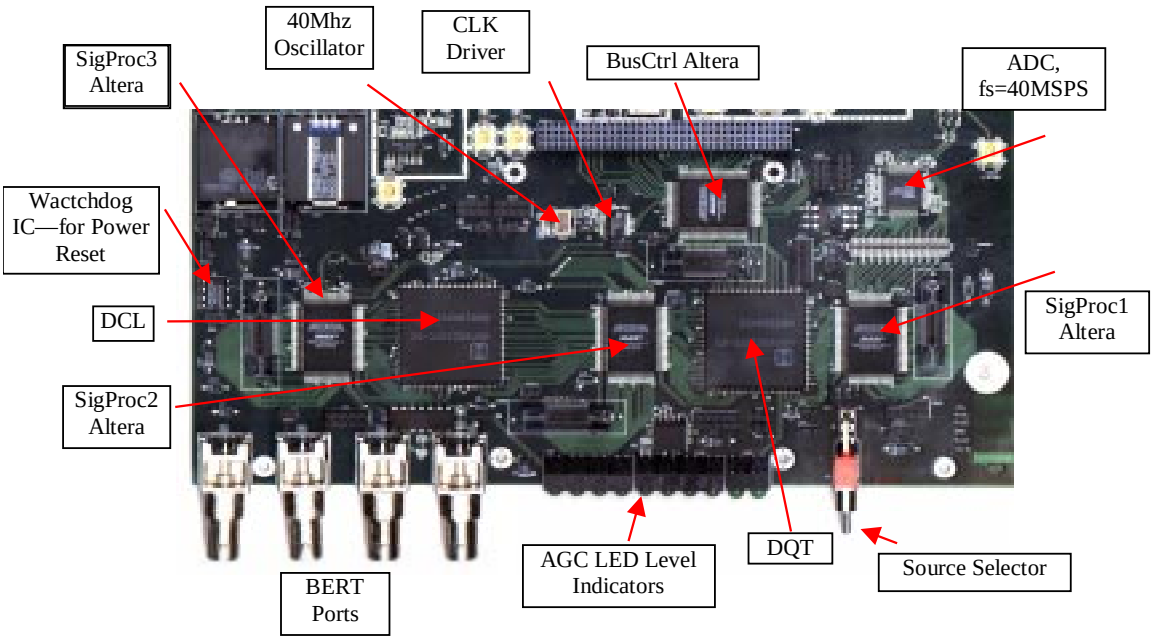

Figure 38 – Close-Up View of Receiver Digital Section

Chapter 6 Summary and Conclusions

This paper presents research centered around the design and implementation of a link-level adaptive software radio. This radio is a platform through which adaptive control algorithms can be developed to adapt the system to constantly changing wireless channels.

This radio is controlled by a PowerPC microprocessor which can monitor any given set of radio performance metrics. This processor has direct control of how the radio operates. It controls the radios initialization registers through a series of busses and can adaptively change the configuration of the radio and how it behaves.

At the present time, the radio is still in the testing and debugging phase. All initial tests show the radio performing satisfactorily. Final tests and field trials should be completed by December 1999.

Future Work

There is still considerable work to be done on this project. As stated above, this radio is merely the platform that allows the link-level adaptation to be done. The main area still needing attention is the design of the control algorithms. These algorithms will reside on the PowerPC and should monitor the radio and adaptively change its configuration to best suit the channel.

One of the areas of research RDRN is concentrating on is the area of channel estimation. This radio allows the user to receive and store digital samples of the symbols transmitted over the channel. These received symbols can be used to estimate the current state of the channel.

This radio and its controller represent the first two layers (physical and data-link levels) in the protocol stacks for the networking reference model. As a final

product, this communication system should provide seamless, wireless data links over TCP/IP networks. Other areas of the RDRN project are concentrating on the development of these wireless protocols. In the future, this radio will need to be integrated with these upper layers to produce a complete, adaptive wireless networking solution.

References

1. Altera Corporation, *MAX+PLUS□ II: Getting Started*, San Jose, CA 95134, September 1997.

2. Analog Devices, AD6640 IF Sampling A/D Converter Datasheet, 1998.

3. Analog Devices, *High Speed Design Techniques*, Norwood, MA 02062, 1996.

4. Friis, H.T., "Noise Figure of Radio Receivers," *Proc. IRE*, Vol. 32, pp. 419-422 July 1944.

5. Johnson, Howard W. & Martin Graham, *High Speed Digital Design*, Prentice Hall, Englewood Cliffs, NJ 07632, 1993

6. Harris Semiconductor, HSP50110 Datasheet, 1999.

7. Harris Semiconductor, HSP50210 Datasheet, 1999.

8. Haykin, Simon, *Communication Systems*, John Wiley and Sons, New York, 1994.

9. K. Sam Shanmugan and Joseph Evans, *Rapidly Deployable Radio Network (RDRN): Phase II, 1997 Project Summary*. University of Kansas and DARPA Information Technology Office, September 1997.

10. Mitola III, Joseph, "Technical Challenges in the Globalization of the Software Radio", *IEEE Communications Magazine*, pg. 87, Vol. 37, No. 2, February 1999.

11. Prescott, Glenn, *EECS 800: DSP for Communications and Radar*, class notes and lecture, Fall 1999.

12. Pozar, David M., *Microwave Engineering*, Second Edition, John Wiley and Sons, New York, 1998.

13. Proakis, John G , *Digital Communications*, Third Edition, McGraw Hill, New York, 1995.

14. Protel International Pty Ltd., *Protel 98 Designer's Handbook*, 1997.

15. Razavi, Behzad. *RF Microelectronics*, Prentice Hall Inc., Upper Saddle River, NJ 07458, 1998.

16. Rappaport, Theodore S., *Wireless Communications – Principles and Practice*, Prentice Hall Inc., Upper Saddle River, New Jersey 07458, 1996.

Appendix A

Appendix A contains information important for programming the radio. If initialization is not performed correctly, or if wrong values are loaded into the various registers, the radio will not function correctly.

Appendix A contains the following information:

- Jumper Settings
- Connector Description with Signal Levels
- Programming the registers for the RF PLL's
- Programming the registers for the IF PLL (240MHz)
- Programming the Harris Chips
- Programming the Altera CPLD
- Initialization tables located within the PowerPC source code.

Jumper Settings & Headers

The section of the Appendix centers around two things. First, what are the various jumper settings, how do we jumper them for proper operation. Second, what functions do the various headers serve which are located throughout the RF and IF/digital parts of the radio.

Table 10 – Jumper Settings

Jumper Designator	Purpose	How to use	Location
CON3	a typo on PCB. should have been labeled a JP#. This jumper sets the output level of the Alteras. All Alteras will output the same level.	Jumper VCCIO to +5V for +5V levels. Jumper VCCIO to +3.3V for +3.3V levels	located right below power section
JP1	used for programming SigProc3 Altera	Attach programming header. Refer to section on how to program Alteras	Digital Section
JP2	for soldering panel mount switch for unforeseen use.	NA	Digital Section
JP3	Test Header	used in conjunction with JP9	Digital Section
JP4	used for programming SigProc2 Altera	Attach programming header. Refer to section on how to program Alteras	Digital Section
JP5	for soldering panel mount switch for unforeseen use.	NA	Digital Section
JP6	for soldering panel mount LED for unforeseen use.	NA	Digital Section
JP7	for soldering panel mount LED for unforeseen use.	NA	Digital Section
JP8	for soldering panel mount LED for unforeseen use.	NA	Digital Section
JP9	Test Header	used in conjunction with JP3	Digital Section
JP10	used for programming SigProc1 Altera	Attach programming header. Refer to section on how to program Alteras	Digital Section
JP11	for soldering a panel mount MASTER RESET switch	Refer to Delta List. Changes made to PCB render this inoperable as is.	Digital Section
JP12	for soldering panel mount switch for unforeseen use.		Digital Section
JP13	used for programming BusCtrl Altera.	Attach programming header. Refer to section on how to program Alteras	Digital Section
JP14	for soldering panel mount switch for unforeseen use.	NA	Digital Section

JP15	for soldering panel mount switch for unforeseen use.	NA	Digital Section
JP16	for jumpering AGND to DGND. May need to place ferrite across this.	short both pins together to short AGND to DGND	Digital Section
JP17	for jumpering +5VA to +5V. May need to place ferrite across this.	short both pins together to short +5VA to +5V	Digital Section
JP18	for soldering panel mount switch for unforeseen use.	NA	Digital Section
JP19	for soldering panel mounted LED to this. Indicates TxPLL has lock.	NA	Digital Section
JP20	for soldering panel mounted LED to this. Indicates RxPLL has lock.	NA	Digital Section
JP21	Contains I2C bus for programming RF Transmit PLL. Also contain +9V Power for RF Transmitter	Connect ribbon cable from here to RF Transmit section.	Digital Section, its mate is JP1 on RF board.
JP22	Used to provide +3.3V to digital section.	Jumper across either set of pins. Will provide +3.3V to digital section only.	Power Section
JP23	Used to provide +5VA power to Analog Section	Jumper across either set of pins. Will supply +5VA to analog section. Digital Section will not see +5V unless JP17 used also.	Power Section
JP24	Contains I2C bus for programming RF Receive PLL. Also contain +9V Power for RF Receiver	Connect ribbon cable from here to RF Receive section.	Digital Section, its mate is JP2 on the RF board.
JP25	for soldering panel mount switch for unforeseen use.	NA	Digital Section
JP26	for solder panel mount LED indicating /LOCK	NA	Digital Section
JP27	Bits 4-7 of 8 bit word intended to control beamforming.	Connect to RF switch. Talk to Dan	Digital Section – used with JP28
JP28	Bits 0-3 of 8 bit word intended to control beamforming	Connect to RF switch. Talk to Dan.	Digital Section – used with JP27
JP29	for soldering panel mounted LEDs indicating power for +9V, +5V, and +3.3V	NA	Power Section
JP30	Used to supply +5V for Power Amplifiers.	Solder wires to panel mounted connector to supply +5V to Power Amplifiers. The PA's have their own dedicated switching regulator.	Power Section

JP31	Used to supply +9V power to board.	Jumper across either set of pins.	Power Section
JP32	+12V input power should be applied here. This is the main power input for the board.	Jumper vertically. There is an AGND jumper next to this for the grounding external supply.	Power Section

Connector Description and Input/Output Levels

Table 11 – Connector Description

IF and Digital Board				
Connector Designator	Input, Output	Use	Signal Level	Location & Type
CON1	Output	Recovered CLK for use with BERT	TTL/CMOS Compatible. Can drive 50Ω	Digital Section BNC
CON2	Ouput	Recovered DATA for use with BERT	TTL/CMOS Compatible. Can drive 50Ω	Digital Section BNC
CON3	Input	Tx CLK for use with BERT	TTL/CMOS	Digital Section BNC
CON4	Input	Tx DATA for use with BERT	TTL/CMOS	Digital Section BNC
CON5	Output	Outputs decimated sampled symbols from DQT. Coherent. To be used for channel estimation.	TTL/CMOS	Digital Section BNC
CON6	Output	Outputs A & B output busses from DCL	TTL/CMOS	Digital Section Mictor
CON7	Output	Outputs sampled symbols from ADC. These are noncoherent.	TTL/CMOS	Digital Section Mictor
CON8	Output	Outputs Bus Control bits sent from PPC. Can be used to look at programming info.	TTL/CMOS	Digital Section Mictor
CON9	Output	Transmit Data IF output. This goes to RF Transmitter	240MHz fc, modulated BPSK/QPSK level: +12dBm	Analog, IF Transmitter. MCX
CON10	Ouput	IF Testpoint. This – 20dB down TP shows the input signal presented to the ADC	-3dBm nominal +4dBm FullScale	Analog, IF Receiver. MCX
CON11	Output	10 MHz Reference Osicllator for use by RF Transmit PLL	TBD	Analog, 10 MHz Oscillator
CON12	Ouput	10 MHz Reference Osillator for use by RF Receiver PLL	TBD	Analog, 10MHz Oscillator
CON13	Input & Output	144 pin Header. This is the interface to PPC. Conatins all the programming bits and control information	TTL/CMOS	Digital Section 144 pin header
CON14	Output	AGC Output Signal. One of the unused	This output will not give identical power levels as	Analog, IF Receiver.

		complementary outputs. Used for testing	the one we are using. It does not have the matching sections on output pin.	MCX
CON15	Input	IF input signal from RF Receiver Board	Min: -60 dBm Max: +10 dBm	Analog, IF Receiver. MCX
CON16	Input	Auxillary IF Input to AGC amplifier. Used for testing	TBD	Analog, IF Receiver. MCX
RF Board				
J1	Input	Tx Signal from IF Board	+12dBm	MCX, RF Transmitter
J2	Input	10 MHz Reference for PLL	TBD	MCX, RF Transmitter
J3	Input	Receive Signal from Antenna	Min: -90 dBm Max: -15 dBm	SMA, RF Receiver
J4	Output	RF output to Receive IF section	Min: -55 dBm Max: +10dBm	MCX, RF Receiver
J5	Input	10 MHz Reference for PLl	TBD	MCX, RF Receiver
J6	Output	Transmit Output to PA and Antenna	+15 dBm	SMA, RF Tranmitter

Programming the PLLs

There are three PLL chips located onboard this radio. There are two RF PLLs, and one for the 240 MHz transmit IF section. This third PLL is located within the IF section, while the other two are on the RF board. All three of the PLLs are from National Semiconductor and operate the same.

Programming each of these PLLs is identical. Each PLL is programmed from an I2C serial bus. This bus has three lines: a clock, a data signal, and a latch for strobing data into the chip. The data line and the clock signal are the same net for all three chips. The latch enable signal (LE) is separate for each chip.

All that is needed o program the PLLs for proper operation is to program three registers. There is a control register, C, and two counter registers, R and N. The PLL operates through the implementation a "swallow counter." A thorough discussion of this technique is left to the datasheet.

Refer to the Tables on the following page for frequency programming information and a comprehensive channel plan. For RF programming, find the frequency band and forward/reverse channel you wish to use. Next, read over right and find the corresponding N register value. Note: the R & C registers are the same for all RF bands and are listed at the bottom of Table 12.

Table 13 is similar to Table 12 and lists R, C and N register values for the 240 MHz PLL.

Inside the PowerPC, these register values are stored as shown in tables. Before being output to the I2C bus, the PPC shifts these numbers three bits to the left and a three bit address is added to each number. These address bits are needed by the BusCtrl Altera CPLD to determine what chip needs to be programmed.

Table 12 – Channel Plan and Frequency Programming Information for RF

Channel	Tx	Rx	TxLO	RxLO	Tx VCO	Rx VCO	N Register Values (see note below for R and C values)			
							Tx N	Rx N	Program	
5.3 GHz Frequencies (Forward & Reverse Channels)							N =		Tx N(h)	Rx N(h)
1F	5260	5310	5020	5380	2510	2690	1004	1076	100FB1	1010D1
1R	5310	5260	5070	5190	2535	2595	1014	1038	100FD9	101039
2F	5270	5320	5030	5390	2515	2695	1006	1078	100FB9	1010D9
2R	5320	5270	5080	5200	2540	2600	1016	1040	100FE1	101041
3F	5280	5330	5040	5400	2520	2700	1008	1080	100FC1	1010E1
3R	5330	5280	5090	5210	2545	2605	1018	1042	100FE9	101049
4F	5290	5340	5050	5410	2525	2705	1010	1082	100FC9	1010E9
4R	5340	5290	5100	5220	2550	2610	1020	1044	100FF1	101051
2.4 GHz Frequencies (Forward & Reverse Channels)										
F	2395	2445	2635	2515			1054	1006	101079	100FB9
R	2445	2395	2685	2325			1074	930	1010C9	100E89
1.2 GHz Frequencies (Forward & Reverse Channels)										
F	1245	1295	1005	1365			1005	1365	100FB5	101555
R	1295	1245	1025	1175			1025	1175	101005	10125D

To program the R and C registers, use the following values for the RF PLLs		
	R	*C*
5.3 GHz & 2.4 GHz Tx & Rx PLL	= 4 = 000010h	0000D2h
1.2 GHz PLL	= 10 = 000028h	0000D2h

Table 13 – 240 MHz PLL Programming

To program the 240 MHz IF PLL, use this table		
R	*N*	*C*
= 10 = 000028h	= 240 = 100F01h	0000D2h
= 4 = 000010h	= 96 = 100601h	0000D2h

Programming the Harris Chipset

At present there are two ways to program the Harris chipset for proper operation.

The PowerPC interface allows the user to set up multiple configuration tables inside the PPC code. A use can then simply select the radio "personality" he desires by choosing the table he wants. The PowerPC loads every register, every time you download it to the PPC board. At present, the only way to change a given register is to update the table, recompile and download the new data to the PowerPC board.

This can be slightly cumbersome, therefore, a basic programming interface has been designed that allows a user to quickly change individual registers. This is ideal for debugging and trying to find out the "exact" configuration that will make the radio work optimally. This interface is a dual-parallel-port laptop which has a custom 144-pin header which plugs directly onto the digital board. This program requires two parallel ports and is called "h99_rev0.bas." It is currently running on "radiator" in the DSP and Wireless Lab in room 215.

Both of the above methods provide the user with an interface for programming so they do not have to manually control the programming busses. If desired, interfacing with the Harris chips at the bit level is fairly straightforward. The microprocessor interface onboard the Harris chips is used to write to initialize this chipset. Both chips are initialized in the same way. There is a 3-bit address bus, A[0..2], and a 8-bit data bus, D[0..7], that are used to access the internal registers of the devices.

Accessing the internal registers is accomplished through the use of 4 holding registers, sometimes referred to as "shadow" registers. Table 13, below, shows the write address map for this interface. Programming is best explained through the use of an example.

Lets assume we want to program the center frequency for the NCO on the DQT. Once we examine the tables located in the back of the datasheet, we find that

this register is address 00h. To program it to operate at 10 MHz, we need to load into that register a value of 40000000h.

We want to load address 00h with 40000000h. We must first break up the data into one byte chunks and load each byte into one of the holding registers. For example, shadow register 0 receives 00h, shadow register 1 receives 00h, shadow register 2 gets 00h, shadow register 3 receives 40h (the upper bits), and finally we load into shadow register 4 the internal destination address, 0h.

Table 14 – Write Address Map for Microprocessor Interface

Address[2..0]	Description
000	Holding Register 0. Transfers to bits 7-0 of the 32-bit destination register. LSB = bit 0.
001	Holding Register 1. Transfers to bits 15-8 of a 32-bit destination register.
010	Holding Register 2. Transfers to bits 23-16 of a 32-bit destination register.
011	Holding Register 3. Transfers to bits 32-24 of a 32-bit destination register. MSB = bit 31.
100	This is the internal destination address register. On the fourth CLK following a write to this register, the contents of the holding register are transferred to the destination register. The lower 4 bits written to this register are decoded in to the destination register address. There are 9 registers for the DQT and 32 registers to program for the DCL.

The following diagram shows the control register loading sequence that uses this table.

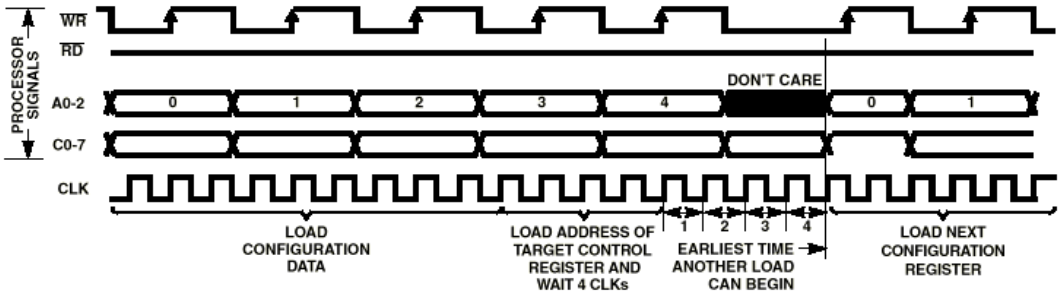

Figure 39 – Control Register Loading Sequence

The above diagram shows the loading sequence needed to properly program the Harris chipset. The address bus shown above will vary from address 0 to address 4. This is the address of the shadow registers. The data bus contains the desired data to be loaded into a specific internal register. The information on D[0..7] when the address bus is 4 is the internal destination address the data will be loaded to.

Table 15 - Programming Data for Harris Chipset

Register#	BPSK			QPSK		
	1MSym/s	2 MSym/s	4MSym/s	1 MSym/s	2MSym/s	4MSym/s
DQT0	40000000	TBD	TBD	TBD	TBD	TBD
1	19999999					
2	000000A8					
3	010BBF20					
4	00000382					
5	00200017					
6	00003FA8					
7	00000000					
8	00000000					
DCL0	006B81A2					
1	00000034					
2	0F68513E					
3	00000001					
4	00000000					
5	00000000					
6	00000020					

7	00000002					
8	00A3D70A					
9	FF5C28F6					
10	00006A00					
11	0001A86E					
12	058946A3					
13	00000000					
14	00000100					
15	00068DB8					
16	FFF97248					
17	0002AF70					
18	0002AF70					
19	00000000					
20	080179FE					
21	FD55F111					
22	FFC0FFC0					
23	000031F0					
24	00000000					
25	00000000					
26	00000020					
27	00000022					
28	00000007					
29	00000000					
30	00000000					
DCL31	00000000					

Note: Register values for the BPSK/QPSK high symbol rates have not yet been determined. We are in the debugging phase currently, and are using the 1MSym/sec BPSK for this purpose. This table will be updated when the QPSK and higher symbol rate register values are decided.

Programming the CPLDs

There are four Altera CPLDs located throughout the digital section of this radio. There is a BusCtrl Altera used for controlling the programming and initialization busses from the PowerPC. In addition, there are three signal processing Alteras located in the receive chain. These are names SigProc1,2, and 3 respectively. The SigProc1 and 2 Alteras are simply used as a buffer to pass information straight through to the next chip. In addition, they route the sampled symbols to a special Mictor connector which can be used to store these recovered symbols to a hard drive for offline analysis. The SigProc3 Altera is slightly more complex. It is used to multiplex the recovered I and Q data for BPSK and QPSK in addition to routing bits to the Mictor connectors.

All four Altera CPLDs are programmed in the same fashion. Each of the Alteras is programmed via its own 5x2 IDC header. Refer to the section on "Jumpers and Headers" to find the header needed to program a specific Altera. These headers are called JTAG ports.

After a combinational logic design is completed with the MAX +II software, it must be compiled and simulated to make sure it will work properly. A special cable, called a ByteBlaster cable, is plugged into the parallel port and the other end plugged into the JTAG port on the digital board. This is the programming interface for the Alteras.

There are two important parameters to check before you program your device. One, make sure you have the device set correctly. In our case we are programming the EPM7128SQC-100-7. In addition, make sure your design is the current project the MAX +II has selected for downloading to the chip. You can determine if your project is the current project by looking at the project field in the programming pop-up box.. It is important to note, that just because you have a design open, does not mean it is the current project. The software will automatically determine if the proper connection is there, program the Altera and verify that everything went satisfactorily.

Appendix B

Appendix B contains the following:

- Schematics for the RF sections.
- Schematics for the IF/digital sections.
- Altera combinational logic diagrams for all four CPLDs.
- Bill of Materials (BOM)
- A "delta" list showing all values that have deviated from the schematics or needed to be changed. This table points to values which have been changed or altered during the testing process.

Note: These schematics were created using Protel'98. All files are in the corresponding Protel format and can be located in the following directories.

The schematic files can be located in (all schematics files have the *.sch extension, while the "main" master project file has the *.prj extension)

/devnull/projects/rdrn2/design/rev2omni/schematics/…

The library files can be located in (all library files have the *.lib extension):

/devnull/projects/rdrn2/design/rev2omni/library files/…

The layout files can be located in (all PCB layout files have the *.pcb extension):

/devnull/projects/rdrn2/design/rev2omni/pcb/…

Altera CPLD Logic files and Harris Chipset revision files are located in:

/devnull/projects/rdrn2/firmware/ifdig/…

RF Schematics

IF and Digital Section Schematics

Altera CPLD Combination Logic Diagrams

Bill of Materials

Delta List for IFDig Rev1.0

So far the following hardware changes have been made to the IFDig Radio. These changes are not shown in the "IFDig.prj" files for Protel. If a new board is fabricated, or a new one tested these changes need to be incorporated onto the board for proper operation.

Note: So far, there have only been three(3) boards fabricated and stuffed.

Date: April 21, 1999
May 13, 1999

Schematic Location	Modifications Needed	□ Δ's already incorporated on board #:
Ifdig.prj		
RFTx.sch		
TxLO_240.sch	• U21 (PLL): Output should be taken on pin2, not pin1. Pin 1 is unused. Chip Enable (CE), pin 10, should be tied high.	1
	• C143 should be 15pF	1
	• L12 should be 22nH	1
	• R68 should be 10K Ohm	
	• R83 should be 500 Ohm	1
		1
TxIF.sch		
RFRx.sch		
RxIF.sch	• AGC1: Pin 11 should be tied to AGND	1
	• L7 changed to 0.039uH	
	• C132 changed to 36pf	1
	• L8 omitted	
	• LED1,2,3 : The pins numbers for the SSF-xx LEDs are backwards. This resulted in the anode/cathode being reversed. The fix for this is to simply mount the LEDs on the bottom of the board with the LEDs facing outward. The schematic reference for this part has been changed.	1
		1
		1
RxIFGain.sch		
ADC.sch		
DigRxTx.sch		

HSP50110.sch		
HSP50210.sch		
BusCtrl.sch		
SigProc1.sch		
SigProc2.sch		
SigProc3.sch		
BERT.sch	• U8 (WatchDog): R19 replaced with 0 Ohms. NOTE: This means we can not use the manual Reset as it stands. The ckt need to be changed to enable the manual reset to work on JP11.	1,2,3
Power.sch	• L1 and L10 need to be Power Inductors L10 is currently shorted for operation. L1 still needs to be modified.	1,2,3
Clock.sch		
PPC_IO.sch		
Delta's for Altera PRG's	**Modifications Needed**	**□ Δ's on board #'s:**
BusCtrl	• Added two DFF's to delay the PLL_Data signal. We were violating the setup time for the PLL chips. • The WDQT signal was coming in on the RDQT signal pin. This has been rerouted to the appropriate output pin. • The WDCL strobe was coming in on the WDQT signal pin. This has been rerouted to the appropriate output pin. NOTE: In the future, if we need to read from Harris Chips, this will have to be changed and a bi-directional bus made.	1
SigProc1		
SigProc2		
SigProc3		
General Notes	**Problem**	**Modified on boards num:**
	All MCX Connectors center pins are not soldered.	1
	JP21 and JP24 are too close and cannot both support a connector at the same time. Custom cables must be used.	