Digital Radio Baseband and Testbed for Next Generation Wireless Systems

Gilbert Lityo

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2016-101
http://www.eecs.berkeley.edu/Pubs/TechRpts/2016/EECS-2016-101.html

May 13, 2016

University of California, Berkeley College of Engineering

MASTER OF ENGINEERING - SPRING 2016

Electrical Engineering and Computer Science

Integrated Circuits

Digital Radio Baseband and Testbed for Next Generation Wireless Systems

Gilbert Lityo

This **Masters Project Paper** fulfills the Master of Engineering degree requirement.

Approved by:

1. Capstone Project Advisor:

Signature: _____ Date _____

Print Name/Department: Borivoje Nikolic/EECS

2. Faculty Committee Member #2:

Signature: _____ Date _____

Print Name/Department: Vladimir Stojanovic/EECS

Table of Contents

Problem Statement……...2

Technical Contributions

Knowledge Domain…….4

Approach…….....8

Design and Validation……10

Future Work……...13

Industry Analysis…….....14

Market Analysis…..…….…….…….……...…...16

Intellectual Property (IP)…….…….…….……18

Bibliography…….…….…….…….…….....20

Problem Statement

The goal of team digital radio baseband is to develop an open-source hardware generator for a software defined radio (SDR). SDR is a radio in which the operating physical layer of the system is implemented by software. The main purpose for this open-source project is to remove the necessity to design hardware for each block from scratch. Currently, most of the existing designs to build a SDR have been published in papers and academic journals. However, there is no basic infrastructure available for new designers to create a SDR. They have to construct the system from the very beginning. As a result, many designers spent their time trying to set up the system and many of them eventually produced similar SDR from one to another. This problem has caused the development in SDR to be not as swift as the development in software field since various open-source software are available for everyone. Many SDR designers are not adding new value for SDR improvement. Through this open-source project, our team hopes that SDR designers can take advantage of our generic hardware generators to help reduce the design time and cost significantly. Hardware generators remove the necessity to designing hardware for each communication protocol from scratch, providing instant hardware designs.

In order to build this system, our group is responsible for designing hardware generators of the digital signal processing (DSP) unit inside the SDR. DSP unit can be mapped out into a schematic diagram which consists of various functional units. Each unit is defined as a block on the diagram. By connecting these different functional blocks on the diagram correctly, we are able to create a whole radio system. To create these blocks, we are using a hardware construction language created at UC Berkeley called Chisel (Constructing Hardware in a Scala Embedded Language). We are working with

PhD students and Professors to integrate our blocks diagram and create generic hardware generators for an open-source hardware. Consequently, each generator needs to be parameterizable for anyone. Parameterization gives the users the flexibility to use our systems and modify them according to their SDR design. They will be able to input any conditions and requirements as parameters in our generic platform.

Hardware designers typically use Verilog (IEEE 1364) as their standard hardware description language (HDL). Verilog is widely used to design and verify both digital and analog circuits. However, designing parameterizable hardware generators in Verilog will require a huge amount of time as different parameters require new modules, resulting in lengthy lines of code. To avoid this, Chisel is used as it supports advanced hardware design using highly parameterized generators and layered domain-specific hardware languages. Chisel is also a hierarchical, object oriented, functional programming language, while generating low-level Verilog design. Different modules created in Chisel can be reused. Thus, the parameterizable hardware generators can be designed with Agile methodology with faster development time. On the other hand, Chisel is quite a new tool, especially for Chisel DSP environment, and still under development with regular update and bug fix. Hence, it poses a challenge to adapt and be proficient with the new language to develop these parameterizable hardware generators.

Knowledge Domain

In order to start building generic hardware generators, we must first understand the concept of the system that we are trying to create and the tools that we are using to implement the model. The system that our team is developing is the DSP and the tools that we are using is Chisel. Through the lecture slides given by Prof. Borivoje Nikolic, presentations and help from the collaborating PhD students we were able to understand the concept of our system and implement it with Chisel.

A DSP unit has two regions of operation, a transmitting region and a receiving region. Each region can function independently either as a transmitter or a receiver of a signal, but to have a complete radio system, it is required that the system has the capabilities to transmit and receive signal. A simple complete radio can be created with basic transmitter and receiver. However, to build a complete SDR, it is required to have a transmitter and a receiver with a digital signal processing capabilities such as in figure 1.

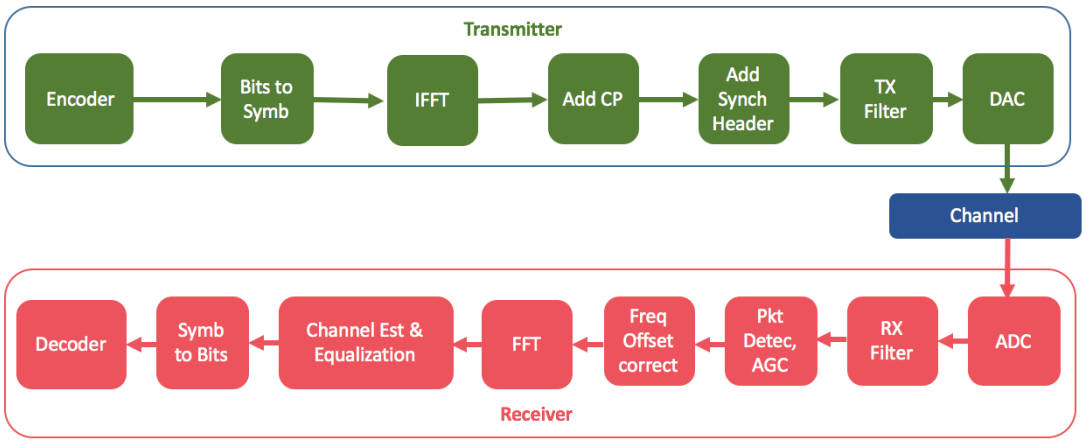

Figure 1. Generic DSP schematic

From figure 1, we can see that both the transmitter and the receiver consist of different functional blocks that have different roles in processing the signal. In general, before a data is transmitted from a transmitter in the form of analog signal, it is first represented in binary bits (0 and 1). It is then transformed into complex numbers through the modulator (Bits->Symbols block in figure 1) and inversed into the frequency domain with the IFFT block and eventually converted into an analog signal through the DAC block. The other remaining blocks are there to preserve the power of signal, prevent errors in the data and reduce noise from mixing with the signal.

In contrast, a receiver does the opposite of a transmitter. It receives an analog signal, converts it into digital through the ADC block, inverses it into complex number with FFT block, transforms it into bits with a demodulator (Symbols->Bits block in figure 1) to eventually retrieves the information contained in the signal. The other remaining blocks are there to also reduce the noise, remove errors in the data and also preserve the power of signal.

Due to the short time of our project, our capstone group is only focusing on the receiver side. The figure below (Figure 2) is a chain of blocks diagram that creates a comprehensive receiver. The red squares are showing the blocks that our team have worked on. Each team member is working individually and collaboratively on these different blocks. After each block is finished, a minimum viable product (MVP) is developed with the PhD student. The MVP is constantly being updated by integrating new completed blocks.

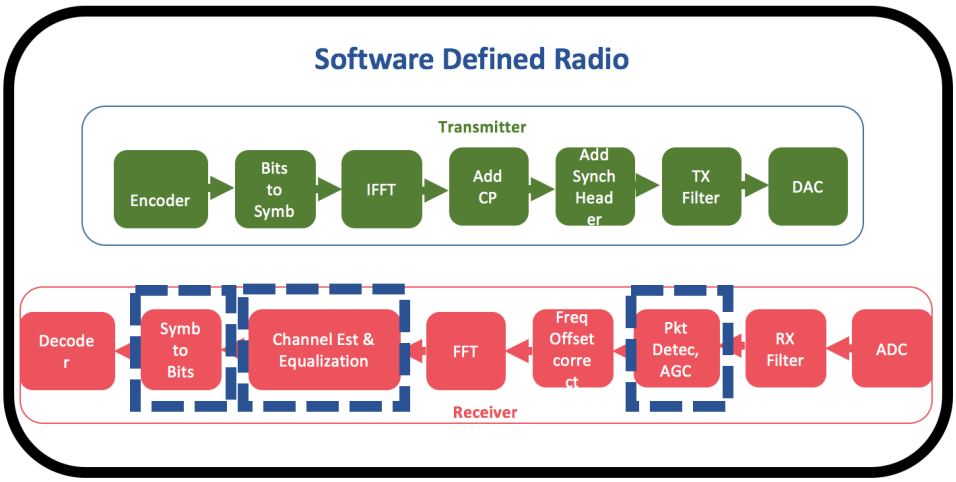

Figure 2. Completed individual blocks

In this paper, I will specifically discuss the automatic gain control (AGC) block as my technical contribution for this project. AGC is the Pkt detect, Synch, AGC block on figure 2. Before going further into the theoretical design of this block, it is best to understand the importance and reasoning behind this block. A receiver (Figure 2) works by having a source (i.e., radio station and transmitter) feeds in a signal into the receiver unit. The first stage of a receiver is an analog filter block that helps to remove the noise embedded in a signal. The signal is then processed with all of the remaining blocks in figure 2 and finally translated into a clean output (i.e., images, voices, and texts). During the process of transmission from the source to the receiver, there are some interferences caused by the environment (represented as Channel impairments and interference block on figure 2), such as reflection of a signal by a building, diffraction from different materials and scattering from leaves (Figure 3). These interferences are causing noise to be added to the signal and also fluctuating the signal power. This issue is called multipath fading.

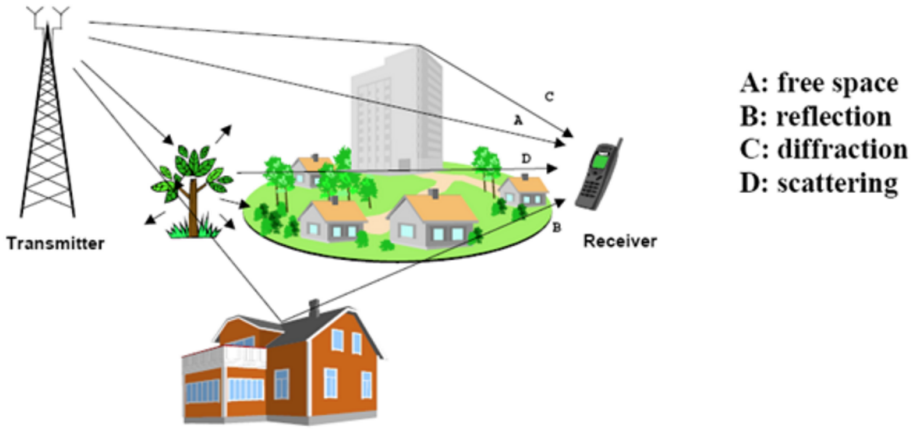

Figure 3. Environment effects on signal

In order to address this issue, the signal that comes to the receiver has to be improved through filtering to remove the noise and amplification or reduction to adjust the fluctuated strength of the signal. Amplification or reduction process can be described as increasing or decreasing gain respectively. This process can be achieved by using an AGC. An AGC has a purpose of controlling the output signal amplitude by setting a suitable gain value, depending on the strength of the signal. An increasing gain is set when there is a weakened signal. A weakened signal is normally due to fading, attenuation caused by the signal propagating over certain media in the environment. On the other hand, a decreasing gain is set when there is an overpowered signal. Depending on the power and distance of the transmitter, sometimes an overpowered signal can be received by the receiver. AGC block role is to provide a constant output signal amplitude within an acceptable range so that the circuits following the AGC block will have a stable and functioning system. AGC automatically adjust the gain of the receiver to maintain the output from fluctuating too much.

Approach

To implement an AGC, the first step is to learn the current designs of an existing AGC. There are two common closed loop AGC structures, which are Feedback AGC and Feedforward AGC (Pérez 2011). Figure 4 is showing the simplified blocks diagram of both feedback and feedforward AGCs.

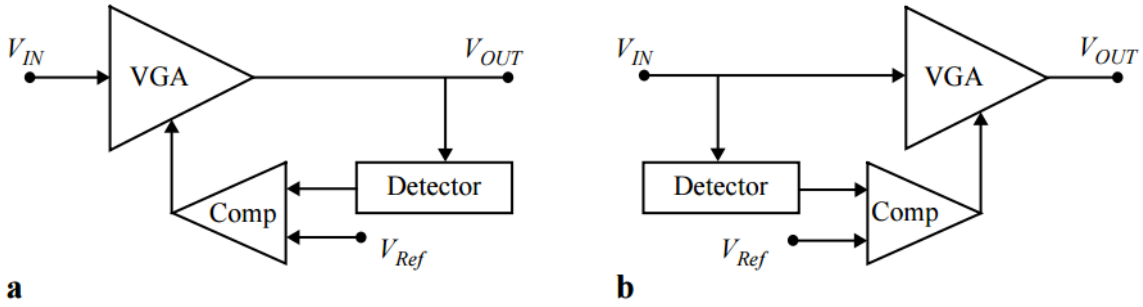

Figure 4. Simplified blocks diagram of feedback (a) and feedforward (b) AGCs (Pérez 2011)

An AGC block contains a variable gain amplifier (VGA), detector and a comparator. A detector is used to find the maximum amplitude of a signal. The maximum amplitude gives us the information of the current power of a signal. The signal power level is then inputted into a comparator (Comp) and compared with a reference voltage (Vref). This reference voltage is normally referred to the original signal power from the source. The comparator outputs either an increasing gain if the Vref has a higher power than the received signal or a decreasing gain if the Vref has a lower signal power. This gain is set to the VGA to amplify or reduce the signal power.

While both AGC structures on Figure 4 have the same components, their operations are not the same. Both structures have advantages and disadvantages. For a feedback AGC, it has an advantage in the lower input dynamic range required by peak detector. The input dynamic range required is reduced in the same way as the AGC gain range (Pérez 2011). It also has an inherently higher linearity due to the feedback loops' characteristic (Pérez 2011). On the other hand, feedback structure also has several disadvantages. First, it has instabilities with high compression or expansion as high level of feedback is required for this for this process (Pérez 2011). Second, feedback structure will always have a higher settling time compared to feedforward structure (Pérez 2011).

In contrast, feedforward AGC has no instability problems as it is configured for high compression and high expansion ratios. It also offers an ideally zero settling time as it provides a constant time that depends only on the peak detector response. However, it also has disadvantages such as it exposes high input dynamic range to the peak detector. Thus, it requires a higher linearity (Pérez 2011).

To summarize, a feedforward AGC can lead to a huge power consumption for the detector to operate, but it has a more stable system for amplifying/reducing any range of input signal power. While for a feedback AGC, it does not require a huge power for the detector to operate, but it has an unstable system when it has to amplify or reduce a very low or very high signal power.

Design and Validation

The design structure that I chose for implementing the AGC block is the feedback AGC. We believed that our DSP system will not have a very wide range of signal power, since the environment will not affect much of our receiver. Thus, the system will be stable most of the time as Input level changes can be relatively slow. Feedback AGC will then provide us with a better power consumption for our radio. The implementation design that I am following is on the figure below (Figure 5). The measure block is the detector on Figure 4 and the control block is the comparator on Figure 4.

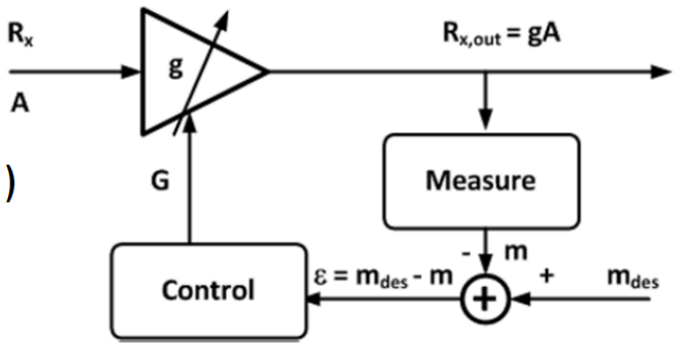

Figure 5. Feedback AGC design schematic

This concept is implemented using Chisel. The measure/peak detector will be provided by the PhD students. Thus, it can be assumed that the voltage output from the peak detector is set to be a voltage input parameter for the AGC block. The voltage reference is also set as a parameter for the AGC block. A multiplexer combined with a comparator is built to check the differences between voltage input and voltage reference. Three conditions are imposed inside the comparator. If the voltage input is bigger than the voltage reference, a negative gain signal is outputted by the multiplexer. If the voltage input is smaller than the voltage reference, a positive signal is outputted by the

multiplexer. And lastly, if there is no difference between the two of them, the multiplexer will just output a constant gain signal.

The final AGC model is shown in figure 6. An additional parameter called step size (**α** in figure 6) is required to set how fast or slow do we want to change the output of the AGC. The step size sets the gain multiplier of our AGC. The positive and negative signal outputted from the multiplexer appoints the step size to be increasing gain or decreasing gain. A register is also built to save the previous gain value in case there is a constant gain signal from the multiplexer as we want to use the old gain value.

In general, we want to have a small step size to have a stable system as huge step size causes an abrupt change in the voltage outputted from the AGC. Excessive gain pumping causes modulation errors.

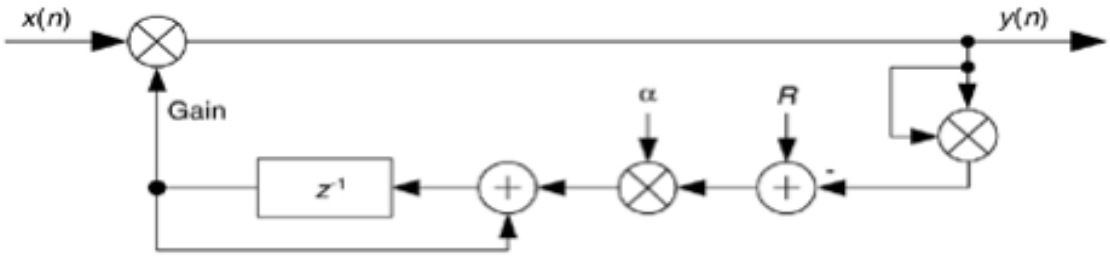

Figure 6. Complete AGC block model

Description: x(n): Input Signal, y(n): Output Signal, R: Reference Signal(Parameterized), α: Step Size(Parameterized)

The AGC block is validated using test file in Chisel. Combinations of input voltage and voltage references are used to create different scenarios that test all of the functionalities of the AGC block. The three common scenarios that might happen are the input voltage is equal, less, or greater than the reference voltage. Integer, double and fixed point values of voltage are tested to see the precision of the AGC. Combinations of

huge and small step sizes are also tested to check whether the AGC outputs a correct value and how stable is the system. Constraints are placed on the voltage as it is not allowed to have a negative value and on the step size ($0 < \alpha < 1$) to keep the system stable.

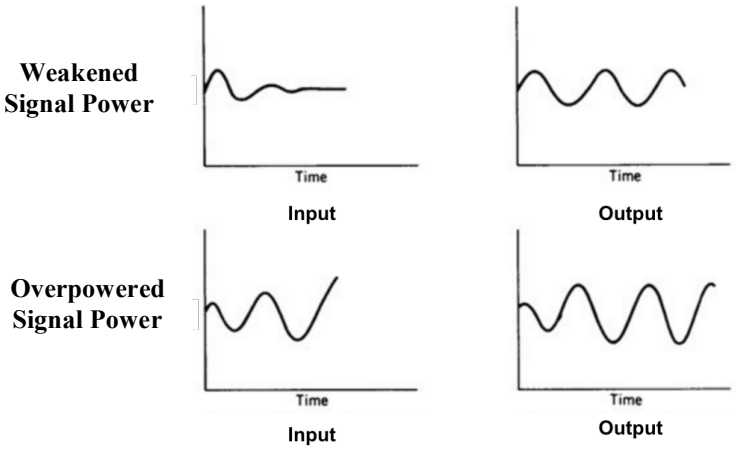

Figure 7. AGC results

The outcome of the AGC is shown in Figure 7. There are two scenarios, weakened signal and overpowered signal as the input of the AGC. In the first scenario, the AGC controls the output signal by setting an increasing gain that strengthens the input signal amplitude, resulting in a stable output signal. While in the second scenario, the AGC sets a decreasing gain that reduces the overpowered signal, outputting a stable signal similar to the first scenario. The block was validated, successfully ran in Chisel DSP environment and was generated into a Verilog design file. The final check for the block was synthesizing it using Vivado to check the utilization in FPGA.

Site Type	Used	Fixed	Available	Util%
Slice LUTs*	74	0	303600	0.02
LUT as Logic	74	0	303600	0.02
LUT as Memory	0	0	130800	0.00
Slice Registers	19	0	607200	<0.01
Register as Flip Flop	19	0	607200	<0.01
Register as Latch	0	0	607200	0.00
F7 Muxes	0	0	151800	0.00
F8 Muxes	0	0	75900	0.00

Figure 8. FPGA resource utilization by AGC block

Future Work

The next step of the process is to have an MVP product that is constantly being updated by integrating new completed blocks. The AGC block will first be integrated with the peak detector to create a complete AGC. As the PhD students are currently focused on the tape-out, the first MVP is combining the FFT and demodulator blocks to create a very minimum receiver. Afterwards, it will be connected with the channel estimation, frequency offset corrector and eventually all of the other blocks. Once a complete receiver is finished, then they will start on the transmitter. Ultimately, creating a complete SDR.

Bibliography:

J. P. A. Pérez, S. C. Pueyo and B. C. López. (2011). *Automatic Gain Control Techniques and Architectures for RF Receivers* , 2011 :Springer

Industry Analysis

The goal of our Capstone Project is to develop open-source hardware generators for a Software Defined Radio(SDR), specifically the baseband processing components. SDR is a radio in which the operating physical layer of the system is replaced by software. Baseband processing components serve to correct any errors the environment might have created in the message received during transmission, and decode it after correction. Hardware generator is a hardware design methodology in which the design is created on demand based on the users' specified parameters.

To understand the prospect of profitability in the industry our product fits in the best, the Software Defined Radio Industry, we adopt Porter's Five Force Analysis, where the five forces are substitutes, buyers, new entrant, rivalry, and suppliers. We will analyze how these five forces influence profitability in this industry (Porter, 2008).

The force of substitutes for our product is weak. Our primary substitute is the traditional hardware based radio. SDR poses very attractive advantages that substitutes cannot provide. Its unique ability to implement multiple current communication standards on a radio is non-existent in traditional hardware based radio devices. Secondly, SDR customers will have a sustainable system with less recurring cost every time a new communication technology update emerges. As a result, SDR can create a reconfigurable radio system that has multimode and multiband to support different communications standards with lower overall cost.

Buyers have a lot of power in our industry. The two largest customer sectors in the current SDR industry are the military and the telecommunication infrastructure equipment companies. SDR is being deployed in tactical radios because it enables joint

operations between separate troops from national and international operations even though the network communications in each country are different. For the telecommunication infrastructure, they avoid creating new infrastructure for each update. According to Mobile Experts and the Wireless Innovation Forum, tactical radio manufacturers sold approximately 200,000 SDR embedded tactical radios in 2012 and the amount has been increasing annually (Pucker and Renaudeau, 2012). The buyers possess significant power because of low switching costs between different SDR vendors. Additionally, there are a limited number of buyers in a huge market, which strengthens their negotiation power. Since SDRs support a variety of standard communication standards, the buyers can just switch to another SDR vendor that support the standards if the product of current vendor, for example, becomes too expensive.

The force of new entrants is also strong because of the low barrier to entry. As there are several small SDR design/manufacturing companies, very few of which are public, it indicates that the SDR industry does not require large financial investment. From a technical perspective, the core technology dates back to a few decades ago so the information is widely available.

Although entering the market is relatively easy, there is strong rivalry based on features such as power consumption, bandwidth, and efficient architecture. Since there are already many SDR manufacturers such as Northrop Grumman Corporation, L-3 Communication Holdings Inc., and Raytheon Co. (Marketsandmarkets.com, 2014), most of the features are already covered by one or more companies' product. However, if our product, which is a design tool for SDR, can be used to build a new product that has a unique feature, the rivalry force would be weak.

However, the force of suppliers is weak. The major suppliers for SDR are similar to the suppliers for traditional radio, which includes component manufacturers, for example, antenna and other basic analog devices like analog digital converter (ADC), digital analog converter (DAC). Since these parts are standardized, the suppliers have weak bargaining power.

With weak forces in suppliers and substitute, but strong forces in new entrants, rivalry based on feature, and buyer, SDR may not seem to be an attractive industry to enter at the first glance. However, our marketing strategy explained in the following section will create a more favorable situation for our product.

Marketing

The SDR market has been constantly growing in the last few years. The SDR market size is predicted to reach $27.29 billion by 2020 with a Compound Annual Growth Rate(CAGR) of 12.5% from 2014 to 2020 (Marketsandmarkets.com, 2014). The market demand for SDR still remains strong and has yet to saturate more than 30 years after its initial inauguration (Clarke and Kreitzer, 2015). The huge growth is understandable because new communication technologies have been emerging year after year. According to IBIS industry reports, the revenues of both the Communication Equipment Manufacturing (Ulama, 2015) and Wireless Telecommunications Carriers (Blau, 2015) industries in the US currently stand at $33.8 billion and $248.7 billions respectively. This positive market trend and huge market size on the communication industry provides worthy reasoning to explore SDR development.

There is another great future potential market for SDR. Part of current SDR users are those who use small consumer electronic devices. Tremendous growth in the Internet

of Things(IoT) market has led to various applications, for instance Smart Grid, home automation systems, and intelligent industrial system (Bushehri, 2013). IoT acts as a smart gateway to connect between multiple low-power, low-cost devices and with each other and the internet (Bushehri, 2013). All of these devices have different ranges of communications protocols and interfaces, such as Bluetooth, Wi-Fi, and ZigBee. These multiple wireless standards are implemented more efficiently using SDR technology as SDR can re-configure those devices and create a gateway that can connect everything into a whole system.

Apart from the market size, we will also discuss the "4P's" of marketing: product, price, place, and promotion. Though our project is currently set for research purposes, for this analysis, we will be assuming the scenario where we apply our technology in a business setting.

Our product is an open-source Software Defined Radio(SDR) platform which can be customized to user's needs. It can provide SDR designers with instant hardware designs to expedite their product development by only focusing on developing the value-added components.

As an open-source project, our product is available for anyone who is trying to develop SDR from scratch. Users can integrate our product on their systems without paying any licensing fee. On the other hand, as the main developer of the product, we can adapt a service business model similar to that of Linux/Red Hat. We can provide services to help potential customers transition to our framework, implement, and integrate customized features not yet implemented. The major platform for support will be available through online forum, chat, or video-conferencing. Thus, we will be able to

create a direct channel without face-to-face encounter (Wenkart, 2014). Field Engineers will also be available if requested.

We recognize the lack of trust potential users may have in adopting our platform. Therefore, we aim to promote our reputation, which will be achieved through journal publications and conferences. Sponsorship deals will also be provided for renowned corporates as their subscription for our support service will accelerate the level of credibility.

Intellectual Property (IP)

The two options considered for open-source licensing are Berkeley Software Distribution(BSD) and General Public License(GPL). The main difference between BSD and GPL is that any developer who modifies source code licensed under GPL is also required to license his/her work under GPL if he/she wants to distribute the product. In contrast, developers who enhance or modify a product licensed using BSD are not under such restriction, opening up business opportunities. As a result, BSD will encourage more people to improve and use this tool.

Open-source is chosen due to several reasons. We are currently using publicly available design architectures to create these generators. Thus, we do not have any new algorithm/design for the SDR blocks, making them unsuitable for patents. Instead, our generator aims to help others who want to build new products to use the existing design. As we are using/implementing the existing designs, making this project open-source will protect us from lawsuit. It is also beneficial to the design community as donating our idea to the public will encourage innovation by saving time/effort to design the fundamental blocks.

By building a support model around this, we can create win-win situation; by creating a lower barrier to adopt this framework, the number of SDR designers will increase which not only encourages innovation, but also generates more revenue for our business.

Bibliography

Blau, G. (2015). *IBISWorld Industry* Report 51721: Wireless Telecommunications Carriers in the US. Retrieved October 18, 2015 from IBISWorld database.

Bushehri, E. (2013). *Future Wireless Networks Will Rely On Programmable SDRs.* Electronic Design.

Clarke, B. & Kreitzer, K. (2015). How to Maximize Your Software-Defined Radio's Dynamic Range. Microwave Product Digest.

Marketsandmarkets.com. (2014). Marketsandmarkets: Software Defined Radio (SDR) Market for Communication by Component (FPGA, DSP, GPP, PSOC, Amplifier, and Software), Application (Military, Telecommunication, Short Range, Positioning, Transportation, and Public Safety), & Geography - Analysis & Forecast to 2014 – 2020. Retrieved from http://www.marketsandmarkets.com/Market-Reports/software-defined-radio-communication-market-11265833.html.

Porter, M (2008). *The Five Competitive Forces That Shape Strategy.* Harvard Business Review.

Pucker, L & Renaudeau, D. (2012). *Conquering SDR tactical radio market challenges.* Military Embedded Systems.

Ulama, D. (2015). *IBISWorld Industry Report 33422:* Communication Equipment Manufacturing in the US. Retrieved October 18, 2015 from IBISWorld database.

Wenkart, M (2014). The Marketing Bible. Norderstedt: on Demand.