5
Low Latency Reconfigurable MAC Design

While the concept of Software Defined Radio (SDR) has been around for more than 20 years, there has been a recent acceleration in the adoption of SDRs. The availability of the resulting wide variety of SDR platforms has fueled the pace of PHY layer research. On the other hand, while active research is being done in the area of MAC protocols, it has not been able to keep up with the fast pace of PHY layer research. These systems have eventually made their way into the classrooms and labs, thus giving communication engineers an experiential learning opportunity. They have provided students cost effective options to acquire real-world signals and analyze them using digital signal processing techniques. In essence, this has done for communications engineering students, what the sound card did for students learning audio signal processing. On the other hand, computer science students have been left with the option of learning about MAC protocols only through text books or by using software simulations. This is because most SDR systems do not meet the stringent latency and performance requirements required for creating real-world communication links; and the few that do are priced out of reach for classroom sizes typically found in Indian engineering colleges. In this chapter, this situation has been analyzed and the emergence of a new design space for MAC layer prototyping systems has been discussed. Key thesis contributions discussed in this chapter are

- An experimental system and methodology is proposed to measure transmit-receive (Tx-Rx) turn-around times of current SDR systems. The methodology proposed using fast sampling power sensor techniques to measure the power received at the radio front ends. Results are presented, which show that current SDR systems may not achieve the latency requirements of many MAC layer protocols
- A new architecture for SDR system is proposed which focusses on latency and cost requirements. This design uses commercial off-the-shelf transceiver chips. A board-level design is presented using EDA software.
- Design feasibility is demonstrated using simulated results with 802.11ac and QPSK signals. Simulation results are shown for both spectrum and modulation measurements.

5.1 Motivation

The concept of Software Defined Radios (SDRs) was first introduced in a research paper by [Mitola, 1995]. SDR is defined as a radio which can be reprogrammed in the field to instantly change its communication protocol to a different standard. This allows the radio to quickly adapt to changing requirements such as new channel conditions. While this concept has been around for more than 20 years, there has been a recent acceleration in the adoption of SDRs. This has been driven by advancements in software, processing technologies, and radio transceivers. It has resulted in a number of commercial SDR platforms such as the Ettus Research USRP, beeCube systems, RICE WARP, Airblue, and hobbyist platforms such as bladeRF and hackRF. These systems, and many others described in [Cass, 2006], have eventually made their way into classrooms and labs, thus giving students the opportunity to learn about PHY layer concepts with real-world signals. It has allowed students to rapidly prototype new algorithms such as carrier frequency offset estimation, synchronization, and multi-carrier aggregation, and validate these prototypes with real-world signals captured over the air. This has impacted the communication

engineering curriculum in the same way that the sound cards have impacted the audio signal processing curriculum.

The key word in Mitola's definition of SDR, which is left open for creative interpretation is "instantly". All the systems described above meet the "instant reprogramming" requirement for physical layer algorithms. However, these platforms are not suitable for meeting the low latency and real-time performance needs required to ensure that radios can establish and sustain communication links. Figure 5.1 shows how the existing SDR systems map onto the product landscape defined by latency and price requirements. USRP and other host-based systems offer good choices for lower cost of development while trading off latency. On the other hand, RICE WARP and other platforms offer choices for Field Programmable Gate Array (FPGA) implementation which enable low-latency applications, but it comes at the cost of increased coding complexity. This current situation has led to the emergence of a new design space as shown in Figure 5.1.

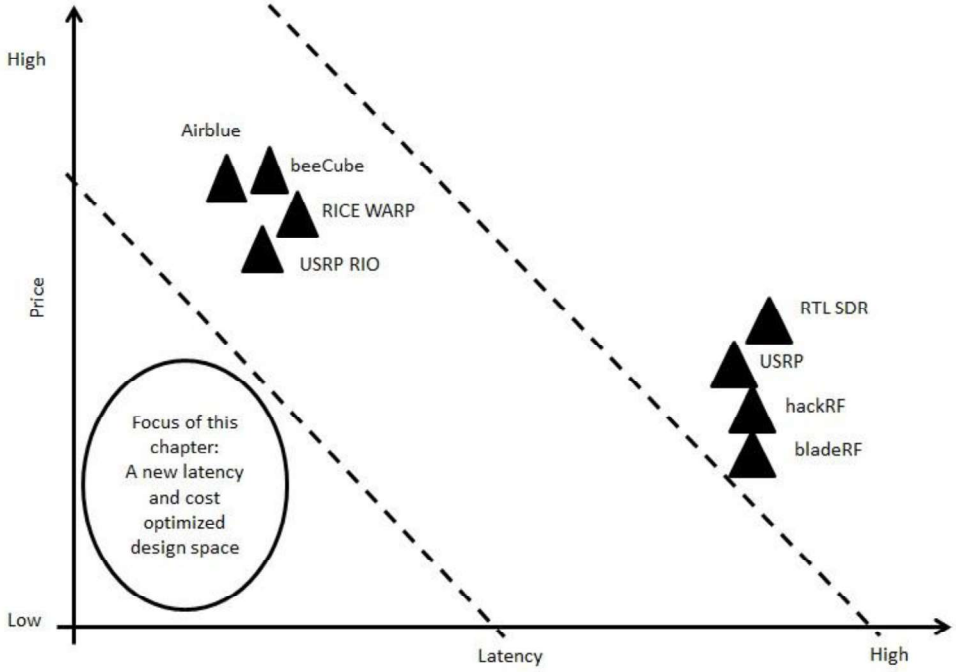

Figure 5.1: Emergence of a New Design Space Driven by Low Cost and Low Latency Requirements.

The focus of this chapter is to describe the requirements of this new space and explore the technology and business feasibility of designing systems that can play in this space. Rest of the chapter is organized as follows. Section 5.2 outlines some of the key requirements for MAC layer protocols, centered around the ideas of latency, processing speed, and cost. Section 5.3 presents experimental results which prove why traditional SDR platforms are not suitable for meeting the stringent latency requirements of MAC protocols. Section 5.4 discusses how advances in commercial off-the-shelf technology and the right design trade-offs can make the design of such low-cost latency sensitive systems a reality. Section 5.5 concludes the chapter and describes scope for future work.

5.2 MAC Layer Requirements

A generalized framework for evaluating experimentation systems for cross layer design has been described earlier in Chapter 4. This framework contains six metrics, mainly cost, latency, throughput, hardware agility, software portability, and extensibility. Attempts to build a system that scores high on all of these metrics will result in a system that ends up being insufficient for any use-case. Hence, it is very important that one clearly understands the key care-abouts of the application at hand. With this objective in mind, this thesis analyzes the requirements specifically for MAC layer prototyping systems. In this application space, the most important criteria is low latency, which in-turn is driven by processing speed, purchase price, and coding complexity, as shown in Figure 5.2.

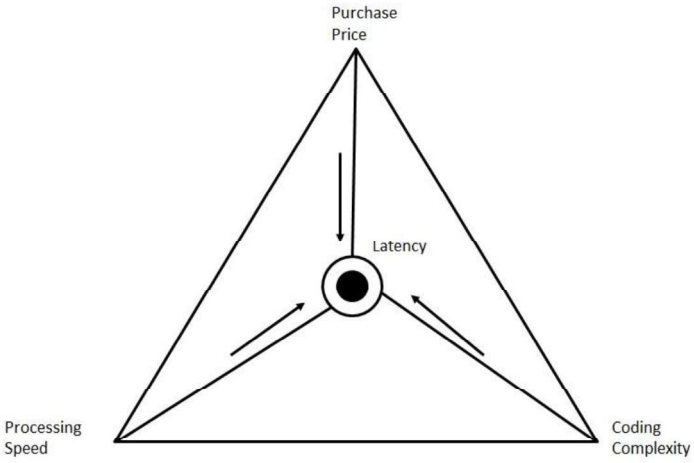

Figure 5.2: Key Design Decisions for MAC Prototyping Systems.

A MAC layer prototyping system should ideally have deterministic, also referred to as real-time, behavior. It should be able to sustain latency requirements within few tens of microseconds, and in the best effort case, to within hundreds of microseconds. To better understand these latency requirements, let us evaluate the different communication applications with human physiological real-time needs, as depicted in Figure 5.3.

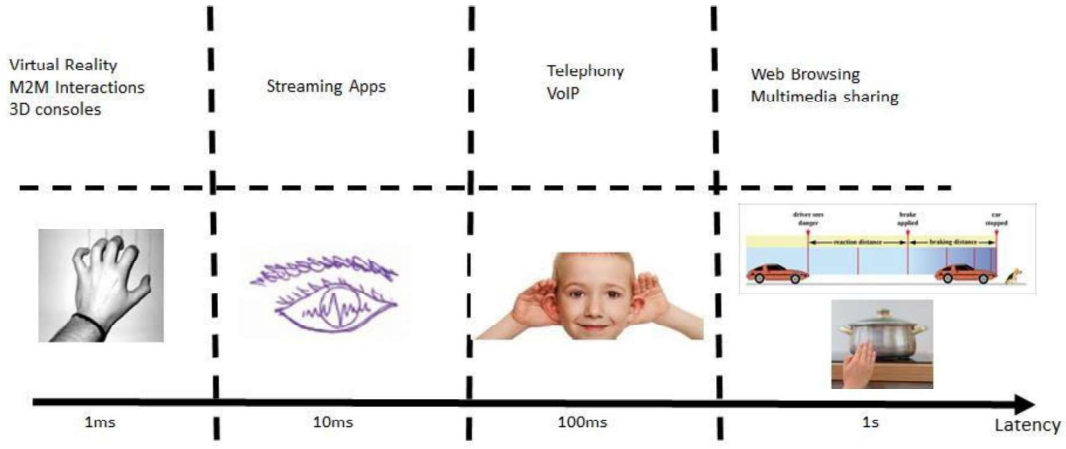

Figure 5.3: Mapping Communication Applications to Physiological Real-Time Needs of Human Beings

The evolution of wireless standards from 802.11a/b/g to 802.11n to 802.11ac clearly shows how the demand for data rates is increasing over time. This basic demand is mainly driven by applications such as Internet browsing and multimedia distribution. However, as web technology evolves, it will have to satisfy basic human physiology. For this to be enabled, it is important that the web incorporates a notion of real-time experience, from which it has stayed away till now. In nature, real-time behavior is experienced whenever the response time between two communication nodes is faster as compared to the time constants of the application system environment. To better understand this, four types of physiological real-time constants are considered: muscular, audio, visual, and tactile. Humans can react to sudden changes of situations with their muscles. For example, a driver instinctly hits the brake on the car on sensing an unforeseen accident. If technology were to replicate this scenario, as in the case of real-time browsing interaction, it's response time would have to be in the range of muscular reaction time of 0.5 to 1 seconds. For this application, any system which has a reaction time on the lower side of this range, e.g. 0.2 seconds, would be an over-designed system. This reaction time has been serviced by initial 802.11b and 3G cellular systems. The human audio system places the next stringent demand on shorter real-time latency. If two communicating nodes receive the audio signals within 70ms to 100ms, real-time interaction is experienced. Translating this to distance, using the speed of sound, implies that any real-time discussions cannot be carried out if distance is more than 100ft, which is the minimum latency requirement for telephony system set by International Telecommunications Union (ITU). Given the fact that light travels about a million times faster than sound, it has been feasible to build many applications that satisfy this requirement. As technology has evolved, modern 3rd generation systems and 4th generation systems have allowed Internet video conferencing over cellular standard. This poses its own latency requirements, as lip synchronization between the video stream and the sound track needs to be within the same time lag. If this does not happen, the sound seems disconnected to the moving image.

To allow for a seamless video experience, modern TV sets have a picture refresh rate in the order of 100 Hz. This rate is appropriate as human eyes have a resolution slower than 100Hz. This translates into a 10 ms latency requirement, as shown in Figure 5.3. Moving further to the left in this figure, shows that the toughest real-time latency is of the order of milliseconds. A physiological requirement for this is driven by the tactile sensing of human bodies. Few examples are considered next. When we move fingers back and forth over a table with a scratch. We experience the scratch in the same position using a high sensing resolution. When moving an object on a touch screen for example, we experience tactile-visual control is in the same order of magnitude. If one moves their hand at a rate of 1 m/s, at 1 ms latency, then the image would follow the finger with a displacement of approximately 1 mm. A latency of 100 ms would create a, not so visually pleasant, 100 mm (4") displacement. Similarly, a screen update resolution finer than 10ms is necessary, so that we can experience a similar experience when moving a mouse and tracking the cursor on the display. Another example where the 1ms latency requirement can be experienced is when we move a 3D object with a joy stick in a virtual reality environment. Likewise, a real-time cyber physical experience can be experienced if the control system adheres to this extreme latency time constraint. In all of these examples, the latency requirements required are far shorter than what the current software defined radio prototyping systems allow for. In fact, they miss the target requirements by nearly two orders of magnitude.

Another example that warrants low latency is described next. Most MAC protocols use techniques such as Code Division Multiple Access (CDMA), Time Division Multiple Access (TDMA), Carrier Sense Multiple Access (CSMA), and smaller variants thereof [Nychis, 2009]. These techniques are used in many commercial wireless networks such as Bluetooth, Zigbee, and Wireless Local Area Network (WLAN). As a specific example, TMDA-based protocols specifically require precise scheduling of time to ensure that transmissions occur during the

prescribed time slots. Modern contention-based protocols also require the ability to precisely control timing requirements to implement techniques such as back-off periods, contention

windows, and inter-frame spacing. However, it is important to note that not all MAC protocols have low latency requirements. For example, the MAC protocols used in the licensed cellular bands assume that a particular time or frequency slot is available for a specific user and hence do not need extremely fast transmit-receive turnaround times. Research in this application space is mainly being driven by a handful of organizations who have the license to operate in these bands and can afford to use higher cost prototyping systems. Most of the contributions to the 3GPP standards, as an example, are being driven by these organizations. On the other hand, research in the unlicensed band is being driven by researchers in universities and entrepreneurs all over the world. Over the years, they have been making recommendations to the IEEE standards using pure theoretical and simulation-based findings. It is getting increasingly important that these recommendations are validated using real-world prototyping systems in order to increase the likelihood of being ratified by the standards bodies. This is driving the need for a low cost experimentation platform which can truly democratize the evolution of these standards.

Processing Speed: In a contention-based protocol, a back-off mechanism is used to reschedule the transmission assuming that the packet loss was due to collision. While the concept of back-off is related to precise scheduling, it also requires the ability to reschedule the transmission quickly without requiring a full packet transmission. A good experimentation system should support fast carrier sense techniques. A simple, yet elegant way, to detect the presence of a carrier is to measure the power of the received signal. More complicated techniques involve full demodulation to get to the decoded bits. The latter is more time-consuming and hence generally avoided. A related functionality requirement is the ability to do a fast recognition to detect the incoming packets. An experimentation system in the new design space should have processing capability which allows the MAC to quickly identify the incoming packet without decoding the entire packet. Such a system should also be able to generate dependent packets quickly and transmit them with precise timing relative to the previous packets. Examples of such packets are ACKs for error control and RTS/CTS for identifying channel access.

Purchase Price: In order to provide true hands-on experience, it is important that students have full access to an experimentation system. To make this financially viable for classroom sizes typically found in Indian engineering colleges, the cost of each lab station should be less than USD 100. If one were to translate all these requirements into a single metric such as the Bill of Materials (BoM), then the design goal should be to have a BoM of less than USD 75. To drive the manufacturing costs down, it is also important that the board schematic be open source so that it can be manufactured locally.

Coding Complexity: Students would ideally prefer one language to write both MAC and PHY layer algorithms. Experimentation systems should have well-documented and well-supported build tools. Ideally a single programming language should be able to target different processing elements on the system. Platform should support a reconfigurable Zigbee-like stack which students can easily modify. Programming languages should be able to support seamless distribution of algorithms across compute nodes. Ideally, platform should support C/C++ or Phython programming since most computer science students learn this language. Reconfigurability of the radio front end is not as important and can be traded off to make programming easier. One of the benefits of host-based programming is that the operating system has sufficiently matured to give elements such as multi-threading and application development, at zero cost. Some of these benefits are lost as one move to a FPGA-based system. Some of this risk can be mitigated by providing a higher layer API, which can be provided as open source code, along with various lab modules and algorithm building blocks.

To summarize, in this section, the key requirements of a MAC layer prototyping system have been described. In Section 5.3, some of the challenges of meeting these requirements with the current SDR systems will be outlined.

5.3 Challenges with SDR for Mac Layer Prototyping

New physical layer algorithms and their experimental results using SDR platforms have been widely published in the literature, as described earlier in this thesis. Availability of ample processing power on general purpose processors and lesser stringent requirements for latency are two of the key factors driving this proliferation. Likewise, the use of open source software, such as GNU radio by USRP, cannot be emphasized enough in driving this rapid adoption. As described earlier, these systems have also made their way into the classrooms and labs, where students have the opportunity to learn about PHY layer techniques with real-world signals. However, it has been a challenge to replicate this success for teaching MAC layer protocols as current SDR systems do not meet many of the key requirements described in Section 5.2. For example, in order to support the desired processing speeds, MAC protocols typically require computational partitioning of data and control blocks among parallel heterogeneous computational engines. This can be a difficult task on traditional SDR systems. Likewise, these systems support a buffered model of computation, in which a block of data is transferred from the acquisition subsystem to the processing subsystem. Each of these steps require a distinct amount of time as shown in Figure 5.4. The sum of these times is generally referred to as transmit-receive turnaround time. Note that, the receive time and transmit time are a function of the size of the packets; which are often asymmetric across transmit and receive functions.

Timing control refers to the ability to prescribe and measure the amount of time between events of interest. Precise timing control is important in digital communication systems because it can affect the ability to maintain communication links. Such applications often require response-time guarantees, which are measured as the time between the receipt of packet by the system and when the system responds with a message. Figure 5.4 shows these different time stamps. Let $\tau 1$ denote the time instant when the first sample is sent from the transmitter to the receiver. Let $\tau 2$ denote the time instant when the *nth* sample is transmitted, where *n* denotes the block size. Let $\tau 3$ denote the time instant when the receiver finishes processing the incoming packet, which includes decoding the received packet and creating a response packet, such as an ACK. At this time instant, the receiver is ready to transmit the first sample. Let $\tau 4$ denote the time instant when the *nth* sample is finished transmitting. Let τt denote the transmit time at the transmitter, τp denote the process time, and τr denote the transmit time at the receiver, as defined by the following simple equations.

$$\tau = t2 - t1 \quad (5.1)$$

$$\tau = t3 - t2 \quad (5.2)$$

$$\tau = t4 - t3 \quad (5.3)$$

Latency requirements and precise time scheduling refers to a tight control over these measurements. State of the art literature currently lacks a well-defined test procedure to accurately measure these times. This problem has been overcome by describing a test procedure that allows us to make precise measurements of the three quantities described in the above equations. Measurements of τp have been done in [Nychis *et al*, 2010] purely from the software perspective, using ping commands. However, these measurements do not take the overall system effects of both hardware and software into account. Additionally, this process does not allow separation of the overall time into the three components as shown in Figure 5.4. The method described in this chapter is fast and is able to make time measurements at finer granularity.

Figure 5.5 shows the proposed experimentation system which uses a power meter to measure power at the RF transmit (Tx) and receive (Rx) output ports of the SDR hardware. Power meters are capable of doing fast and accurate measurements of power by allowing users to trade

off measurement accuracy over time by carefully controlling the number of averages in the measurement.

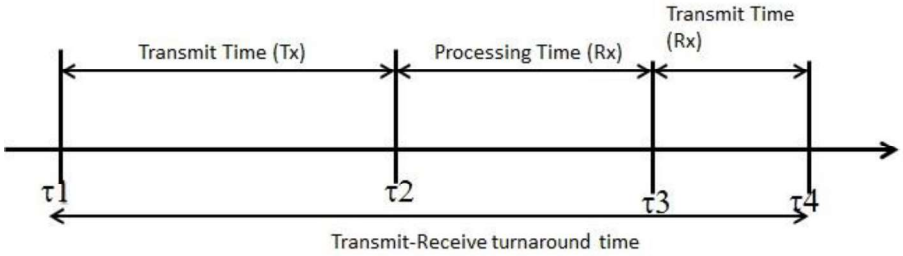

Figure 5.4: Definition of Precise Timestamps Required for MAC Protocols.

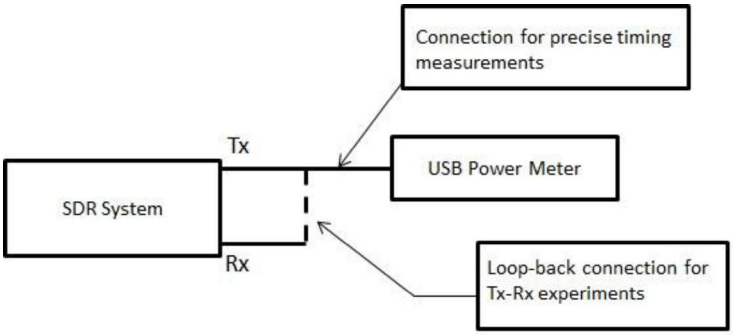

Figure 5.5: Novel Experimentation Systems using Power Meter for Fast Power Measurements

Figure 5.6 shows the measurements recorded by the power at the RF output ports. Region A indicates the time period when the transmitter is sending a message and Region B indicates the time period when the receiver is sending an acknowledgement. The proposed method introduces a GPS time-stamped marker in the first sample of the transmitted packet. Next step is to measure the time instant when the sample arrives at the RF port, indicated by a rise in the measured power level, and the time instant after the nth sample arrives at the RF port, indicated by a drop in the measured power. The experiments done in this thesis show that the power measured by the meter settled to a value within ±1dB in less than 80 microseconds. Using this test procedure, both τt and τr can be accurately measured by minimizing any uncertainty in the measurement.

To prove the validity of the test procedure, an experimentation system has been setup, as shown in Figure 5.7. It contains a laptop connected through an Ethernet cable to National Instruments Universal Software Radio Peripheral (NI- USRP 2932). NI-USRP 2932, which serves as a traditional SDR system for the experiment, contains a radio front end which can be tuned to frequencies from 400 MHz to 4,400 MHz. It offers support for 20 MHz baseband I/Q bandwidth and support for streaming data at a rate of 25 MS/s. The key parameters for this custom scheme are listed in Table 5.1.

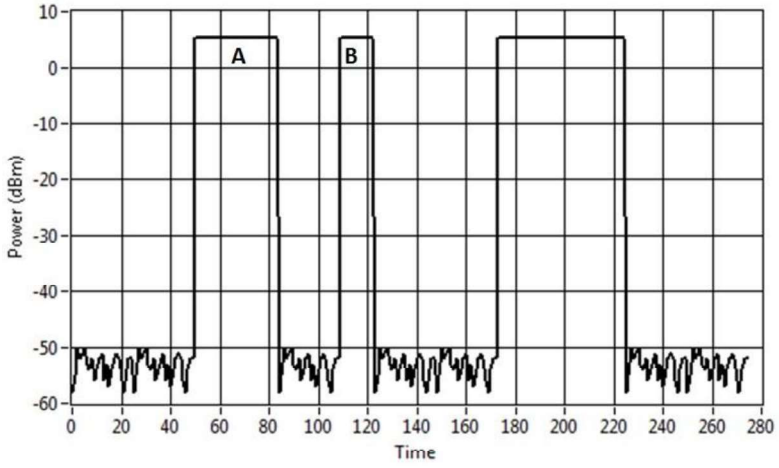

Figure 5.6: Precise Timing Measurements Achieved using Power Meters

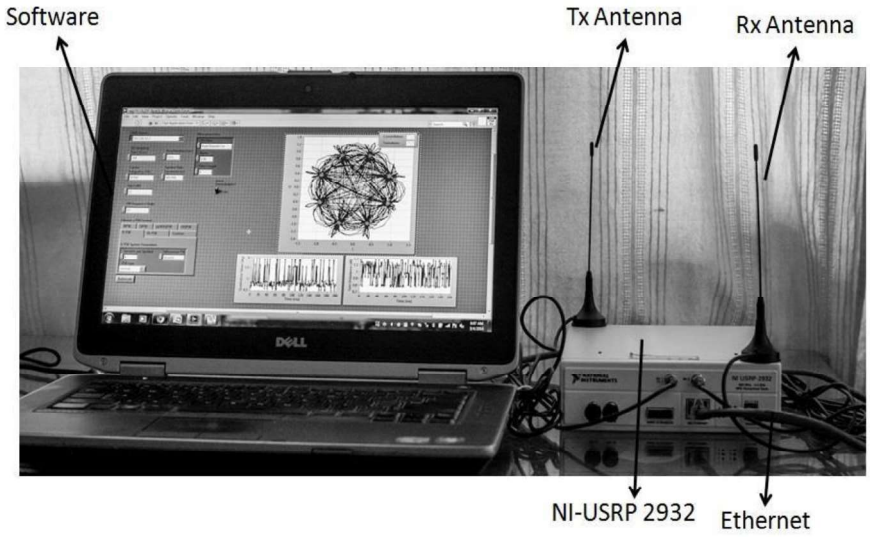

Figure 5.7: Experimentation System used to Validate New Test Procedure using Power Meters

Table 5.1: Key Parameters used in the Experimentation System

Parameter	*Value*
Center Frequency	2400MHz
Modulation Scheme	M-ary Phase Shift Keying
MAC Protocol	TDMA
Symbol Rate	1MS/s

Table 5.2 shows the results of measuring τt as a function of packet size, averaged over 1000 iterations. τt is the time taken by the transmitter to send the packet. This data shows two characteristics. First of all, the latency is pretty high with traditional SDR architecture. Secondly,

it is also interesting to note the high standard deviation and how this value scales as a function of the packet size.

Table 5.2: Transmitter Latency vs Packet Size

Packet Size	*Average Latency (ms)*	*Standard Deviation*
100	10.3	0.23
200	13.0	0.34
500	17.4	0.45
1000	19.8	0.65
2000	20.1	0.8
5000	23.5	0.9

The second set of experimentation shows measurement results of τ_r as a function of packet size, averaged over 1000 iterations, as shown in Table 5.3. This is the time taken by the receiver to transmit the created packet. Once again, a wide statistical distribution in the time measurements is noticed. It is also important to note that, for same packet sizes, the transmit time and receive times are not equal, which shows that the principle of reciprocity should not be assumed for transmit and receive channels.

Table 5.3: Receiver Latency vs Packet Size

Packet Size	*Average Latency (ms)*	*Standard Deviation*
100	11.3	0.21
200	12.3	0.31
500	16.8	0.425
1000	17.1	0.65
2000	18.7	0.78
5000	20.1	0.8

5.4 Design Feasibility

In this section, the feasibility aspects of designing a system that meets the key requirements discussed in Section 5.2 are discussed. These key requirements are latency and cost. In order to design a system that focuses on one of these key requirements, it is important that one has the flexibility to trade-off on other dimensions. Since the focus of this thesis is on a teaching platform for MAC protocols, a lower priority has been given to requirements such as throughput and hardware agility. The market research shows that commercial technology has and will continue to evolve so that FPGA and real-time processing units can be put on a single silicon fabric. This makes it feasible to build systems which students can use to implement and experiment with latency optimized algorithms. Likewise, state of the art allows ADC/DAC techniques to be merged with RF front ends on a single chip, making it feasible to lower the cost of the system. As part of the on-going research work, the plan is to provide open source MAC/PHY algorithms which can be run on the FPGA and real-time processors. This will significantly reduce the cost of ownership, allowing students to leverage the community

ecosystem, similar to what the GNU Radio has done for USRP. Thus, by using commercially available technology, trading throughput, and by building open source PHY/MAC FPGA IP, it

is possible to build latency and cost optimized systems which can be used for teaching. In the rest of this section, a very high level view and feasibility analysis of the proposed new design has been provided.

An architectural diagram of a traditional, PC-based SDR platform is shown in Figure 5.8. The block diagram of the proposed design is shown in Figure 5.9. Table 5.4 compares the relative normalized costs of these two systems along various vectors. SDR systems typically feature a general purpose processor, typically a desktop or a laptop computer, as the processing block. Most students will have access to a laptop or desktop and hence the cost of this component has been set to zero. Traditional SDR systems generally use a higher bandwidth bus such as, GigE and PCIExpress, between the processor and the analog to digital converters, as all the data processing is done on the general purpose processor. The normalized cost of this block is set to 1 unit (or a single $ sign). On the other hand, the proposed design uses a USB bus technology, which has become a commodity using current technology. Hence, the normalized cost of this block is set to zero.

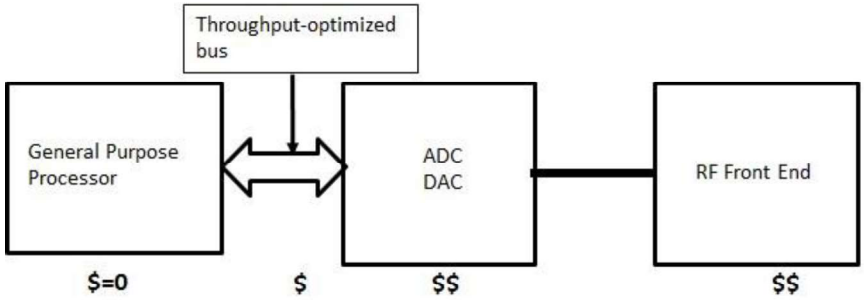

Figure 5.8: Generalized Architecture of Current PC-based SDR Systems

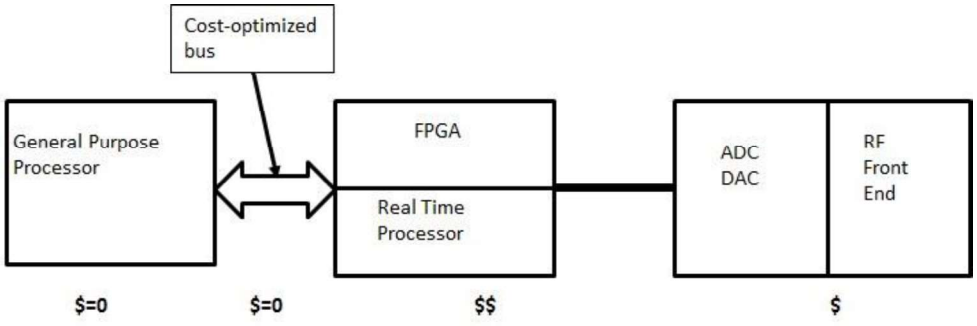

Figure 5.9: Proposed Architectural Design Capable of Meeting Latency and Cost Requirements of MAC Protocols

Table 5.4: Cost Analysis of Proposed Design

Block	*Traditional SDR*	*Proposed Design*
General Purpose Processor	0	0
Bus	$	0
FPGA	$	$$
ADC	0	0

RF	$$	$

Next block in a traditional SDR is the analog to digital converter (ADC). Traditional SDRs have generally used higher resolution on their ADCs. In industrial data-acquisition projects, designers need to digitize an input signal that extends over a very wide dynamic range. Other examples where a wide dynamic range is needed is when one needs to accommodate signals from different sources that exhibit quite different signal ranges or to resolve small changes around a certain value. Usage of higher resolution ADC results in a higher dynamic range in the resulting signal, but it comes at the expense of higher system cost, as shown in Table 5.4. The normalized cost of ADC is set to zero for the proposed design as it uses integrated ADC and RF subsystem, as discussed next.

The RF portions used in traditional SDR systems are largely comprised of direct up/direct down topology also known as homodyne. They are generally designed to cover a wide frequency range from DC to 6GHz or higher, support a wide instantaneous bandwidth and have good spurious free dynamic range. While traditional SDRs can boast of an RF front end which is best in its class, it has driven the cost of the hardware higher and thus out of reach of a wide population of teachers and researchers. Hence, the normalized cost of this component is set to 2 units. The proposal is to combine the ADC and RF capabilities into a single transceiver chip which are commercially available in the market place today. One example of such a chip is the AD9361 which is a high performance and highly integrated RF agile transceiver from Analog Devices. This device combines an RF front end with a flexible mixed-signal baseband section and integrated frequency synthesizers. Since this device incorporates both the RF front end and the ADC/DACs, its cost relative to the cost in traditional systems is low.

Another example is the Texas Instruments TI CC2500, shown in Figure 5.10, which is a low-cost 2.4 GHz transceiver designed for very low-power wireless applications. It is available in volume for prices as low as $10 [Texas Instruments, 2014]. Due to the availability of such commercial RF transceivers, a normalized cost of 1 unit is assigned to this component while performing cost analysis of the proposed design.

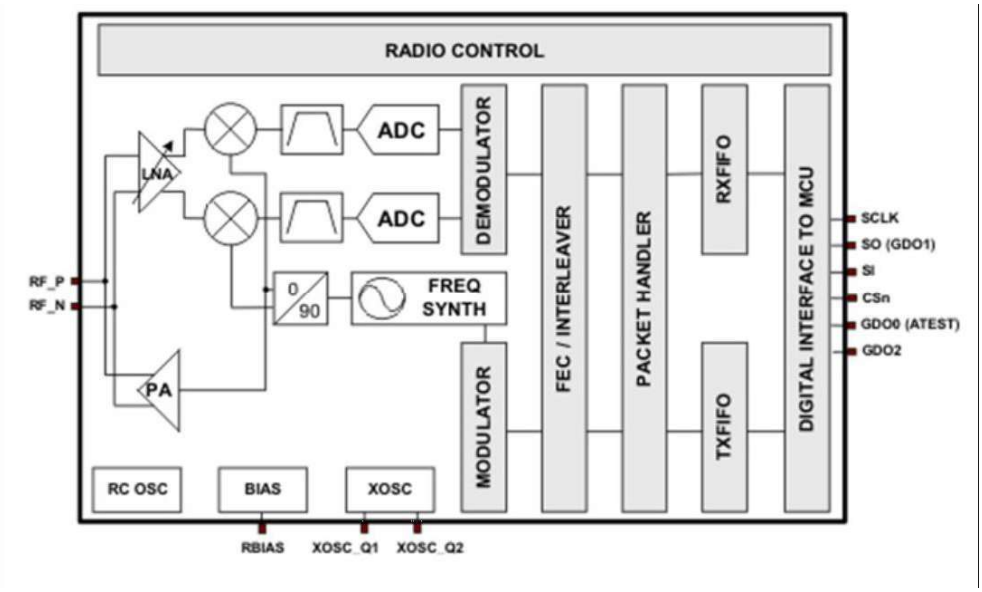

Figure 5.10: Texas Instruments RF Transceiver Chip

One of the key requirements listed in this thesis is latency. FPGAs are generally suited to meet the latency requirements of MAC protocols as they contain specialized circuitry that can perform multiple, sequential, and parallel operations within a single clock cycle. FPGAs are also made up of different types of resources, such as logic, signal processing, and memory blocks, which can be used programmed to extract optimal performance. Hence, the usage of FPGAs is recommended for the proposed design. One of the challenges with FPGA is its price and coding complexity. Hence, this component has been given a normalized cost of 2 units in Table 5.4. GNU radio provided an ecosystem of open source library for traditional SDR systems which has significantly driven the costs associated with coding complexity to zero. As part of future scope of work, a GNU-radio like environment will be created for the proposed system to help overcome the coding challenges. PHY/MAC layer algorithms will be developed and published that students can use as a starting point to conduct their experiments. Once implemented, this initiative would significantly offset the increased cost of ownership introduced by the use of FPGAs on the proposed design, leading to increased adoption by labs all over the country.

The architecture proposed in Figure 5.9 above has been simulated using Applied Wave Research (AWR) Design Environment [AWR, 2015]. The Virtual System Simulator (VSS) portion of this design environment has enabled validation of the proposed design architecture for implementation feasibility and performance. This type of analysis proves that the design proposed in the thesis is practical and can be implemented. There are various metrics that can be used to measure the performance of the proposed design. In this thesis, spectrum performance and Error Vector Magnitude (EVM) calculated from the constellation diagram has been chosen. Single carrier signals, such as QPSK and 16-PSK, and multi-carrier signals, such as OFDM signals, have been used to prove that the proposed design provides ample RF performance for a wide array of signals.

Figure 5.11 shows the spectrum of the QPSK signal. It shows good performance in the in-band measurements. The dynamic range is around 40dB, which is sufficient for most radio prototyping applications. Figure 5.12 shows the IQ constellation plot of the received QPSK signal. The measured average Error Vector Magnitude (EVM) is 1%. To prove the validity of the design for higher order modulation schemes, the performance of the proposed hardware design has been tested with higher order modulation signals, such as 16-PSK. Since the IQ points are closer to each other in the constellation, as compared to QPSK, this signal is much more susceptible to noise and other impairments. Figure 5.13 shows the constellation plot of a 16-PSK signal, with the measured average EVM around 1.5%.

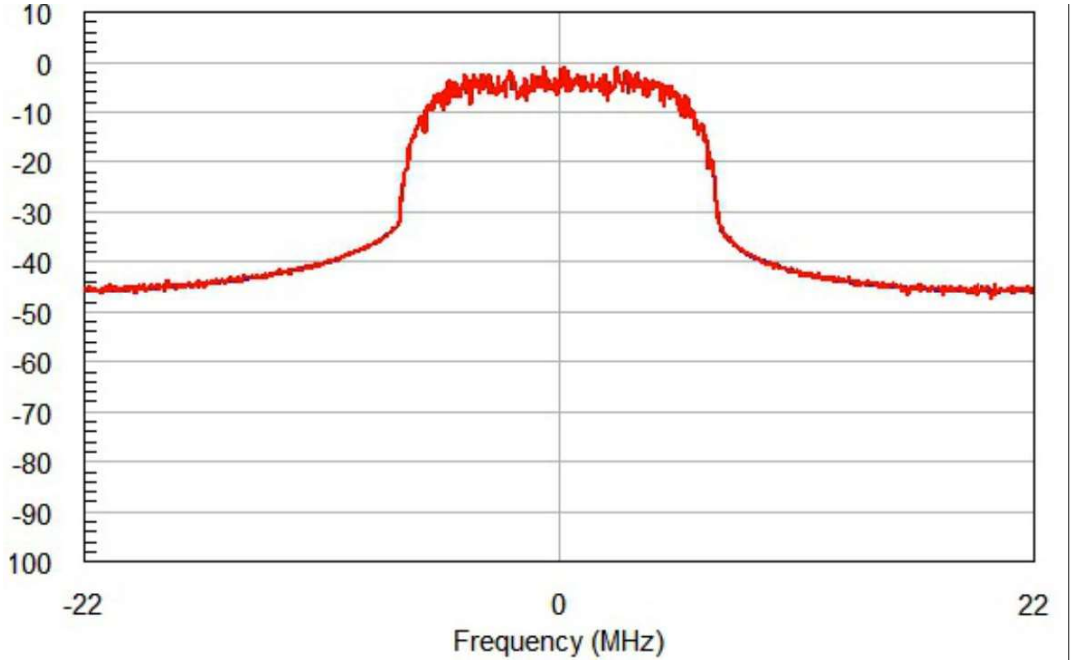

Figure 5.11: Spectrum of a QPSK signal received by the Rx port of the proposed design

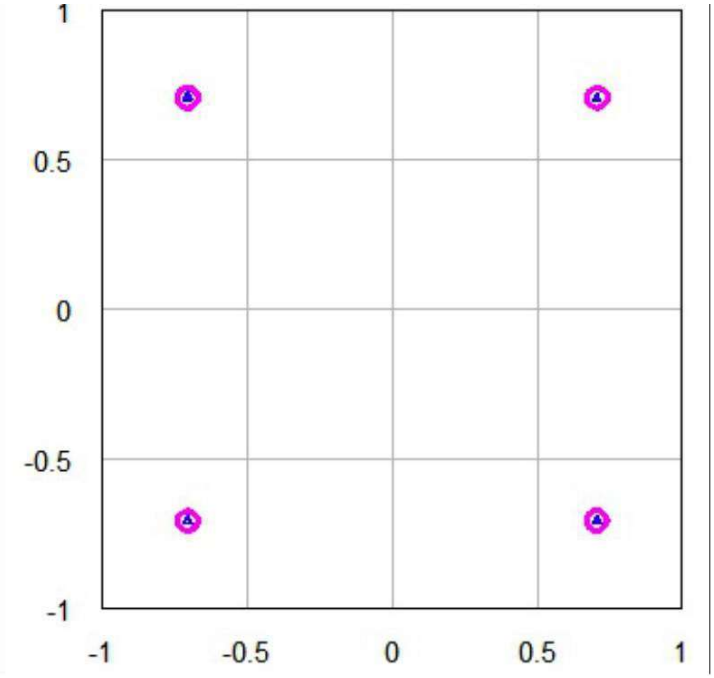

Figure 5.12: Constellation plot of a QPSK signal received by the Rx port of the proposed design

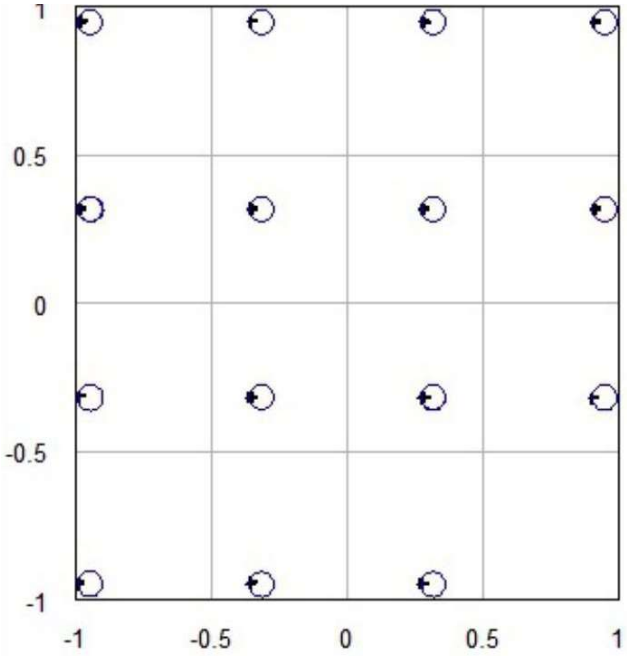

Figure 5.13: Constellation plot of a 16-PSK signal received by the Rx port of the proposed design

IEEE 802.11ac is a wireless networking standard in the 802.11 family developed in the IEEE Standards Association process, providing high-throughput wireless local area networks (WLANs) on the 5 GHz band. 802.11ac is an improvement to its predecessor, 802.11n. One of the goals of 802.11ac is to deliver higher levels of performance that are commensurate with Gigabit Ethernet networking. It provides a seemingly "instantaneous" data transfer experience and a pipe fat enough that delivering a high quality of experience. The hardware design proposed in this thesis has been tested with 802.11ac signals.

Figure 5.14 shows the constellation plot of an 802.11ac signal received by the Rx port of the proposed design. Figure 5.15 shows the Error Vector Magnitude (EVM) per OFDM symbol of an 802.11ac signal. The quality of the constellation plot and EVM values validates that the proposed hardware design can be practically used for MAC layer prototyping with 802.11ac and next generation of WiFi signals.

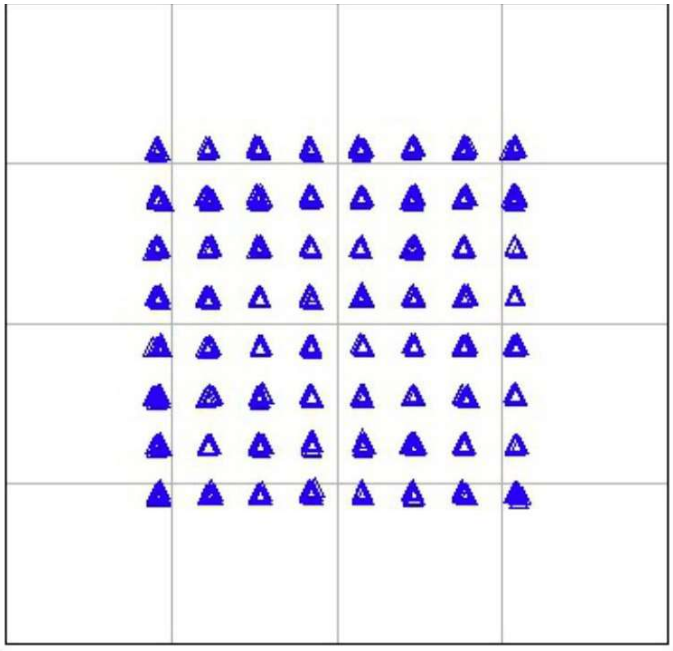

Figure 5.14: Constellation plot of a 802.11ac signal received by the Rx port of the proposed design

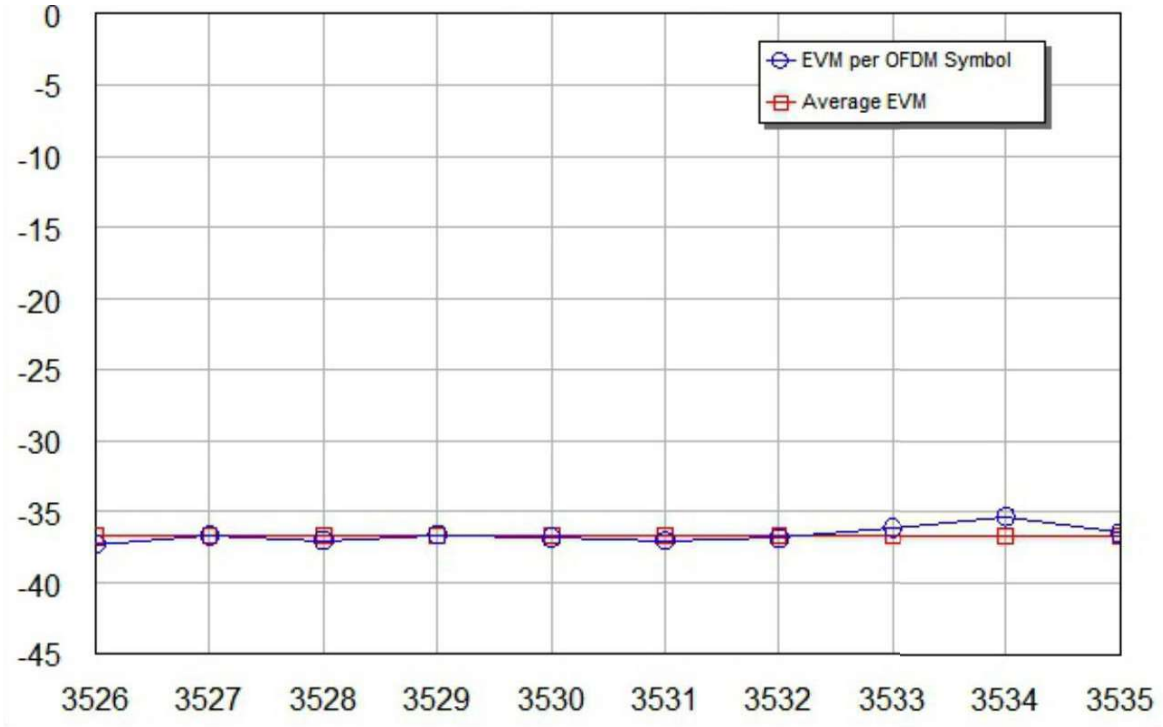

Figure 5.15: EVM per OFDM symbol of a 802.11ac signal received by the Rx port of the proposed design

5.5 Summary

PC-based SDR systems have made their way into the classrooms and labs, thus providing

PHY layer engineers with an experiential learning environment. On the other hand, students are

still learning about MAC layer protocols in theory or with the aid of software simulations. In this chapter, this current situation is analyzed and the emergence of a new design space for MAC layer prototyping systems is discussed. The key requirements of such systems, namely latency, processing speed, and cost are discussed in this chapter. This chapter also explains how availability of commercial technology and careful trade-off with other requirements, such as throughput and frequency agility, is making it feasible to build a system that meets design objectives of low cost and low latency.