VOICE SCRAMBLER

By

Captain Qazi Muhammad Shoaib Ijaz

Captain Muhammad Asif Siddiqui

Captain Syed Hassan Abbas Zaidi

Submitted to the Faculty of Electrical Engineering Department

National University of Sciences and Technology, Islamabad

in partial fulfillment for the requirements of a B.E Degree in

Telecommunication Engineering

June 2014

CERTIFICATE OF CORRECTNESS AND APPROVAL

Certified that work contained in this thesis “VOICE SCRAMBLER”, was carried out by Captain Qazi Muhammad Shoaib Ijaz, Captain Muhammad Asif Siddiqui and Captain Syed Hassan Abbas Zaidi under the supervision of RVF Brigadier Muhammad Khan Minhas (R) for partial fulfillment of Degree of Bachelor of Telecommunication Engineering, is correct and approved.

Approved by

_____

(RVF Brigadier Muhammad Khan Minhas)

EE Department

MCS

Dated:_____

ABSTRACT

Threats to communication systems which have existed since men started sending messages are eavesdropping, modification, replay, masquerading, penetration and repudiation. Implementation of speech secrecy mechanism has always been a challenge in these systems. The term scrambling has been and still is used on occasions to describe the encryption process to protect voice communication. The aim of scrambling is to corrupt as much as possible a speech signal in such a way that it cannot be recovered to an intangible speech signal, after it has been transmitted over a channel. While digital scrambling provides very high security, it is usually difficult to implement due to its fundamental requirement of much wider bandwidth as compared to analog signals.

This project is about the design and implementation of a real time voice scrambling technique on wired communication systems. We have selected Linear Feedback Shift Register technique as our scrambling algorithm which is implemented on an audio signal in digital domain and provides more security than analog domain implemented algorithms. Moreover, selected technique does not require extra bandwidth. We have designed and developed an electronic circuit for practical implementation of project. The scheme is also being realized on contemporary digital signal processor DSP TMS 320C6713 based development board.

Implemented technique has produced satisfactory results and its strength can be increased further by including more delayed output values in the algorithm.

DEDICATION

We dedicate our work to our parents, whose prayers and guidance has led us to our success.

ACKNOWLEDGEMENTS

In the name of Allah, Most Gracious, Most Merciful

All praises and glory to Almighty Allah (SubhanahuWaTaalaa) who gave us courage and patience to carry out this work. Peace and blessing of Allah be upon last Prophet Muhammad (Peace Be upon Him).

We are grateful and would like to express our sincere gratitude to our project supervisor RVF Brigadier Muhammad Khan Minhas (R) for his invaluable guidance, continuous encouragement and constant support in making this project possible. We really appreciate his guidance from the initial to the final level that enabled us to develop an understanding of this research project. Without his advice and assistance it would be a lot tougher to completion.

Lastly we would like to thanks any person which contributes to our final year project directly or indirectly. We would like to acknowledge their comments and suggestions, which were crucial for the successful completion of this study.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

Abbreviation	Word in Full
DSP	Digital Signal Processor
PCM	Pulse Code Modulation
SNR	Signal-to-Noise Ratio
R.M.S	Root Mean Square
PSTN	Public Switched Telephone Network
LFSR	Linear Feedback Shift Register
CDMA	Code Division Multiple Access
LIC	Low Intensity Conflict
ADC	Analog to Digital Convertor
DAC	Digital to Analog Convertor
PWM	Pulse Width Modulation
RC	Resistor Capacitor
IC	Integrated Circuit

CHAPTER 1

INTRODUCTION

1.1 Background

Secure communication is when two entities are communicating and do not want a third party to listen in. For that they need to communicate in a way not susceptible to eavesdropping or interception. Secure communication includes means by which people can share information with varying degrees of certainty that third parties cannot intercept what was said. Other than spoken face-to-face communication with no possible eavesdropper, it is probably safe to say that no communication is guaranteed secure in this sense, although practical obstacles such as legislation, resources, technical issues (interception and encryption), and the sheer volume of communication serve to limit surveillance.

There are following three requirements for implementation of security on landline communication for military as well as civil organization on large scale:

i. Low grade voice security

ii. Less processing time

iii. Low cost device

Keeping in view the above three considerations, 'scrambling' is a better option to be used.

In 1898, Nikola Tesla demonstrated a radio controlled boat in Madison Square Garden that allowed secure communication between transmitter and receiver.

One of the most famous systems of secure communication was the Green Hornet. During WWII, Winston Churchill had to discuss vital matters with Franklin D. Roosevelt. At first, the calls were made using a voice scrambler as this was thought to be secure. When this was found to be untrue the engineers started work on a whole new system, the Green Hornet or SIGSALY. Anyone listening in would just hear white noise but the conversation was clear to the parties. As secrecy was paramount, the location of the Green Hornet was only known by the people who built it and Winston Churchill, and if anyone did see him entering the room it was kept in, all

they would see was the Prime Minister entering a closet labeled 'Broom Cupboard.' It is said that because the Green Hornet works by a one-time pad it cannot be beaten.

The first voice scramblers were invented at Bell Labs in the period just before World War II. These sets consisted of electronics that could mix two signals, or alternately "subtract" one signal back out again. The two signals were provided by telephones for one and a record player for the other. Sets of matching pairs of records were produced containing recordings of noise, which would then be played into the telephone and the mixed signal sent over the wires. The noise would then be subtracted back out at the far end using the matching record, leaving the original voice signal intact. Eavesdroppers would hear only the noisy signal, unable to understand the voice inside.

One of those, used for telephone conversations between Winston Churchill and Franklin D. Roosevelt was intercepted and unscrambled by the Germans. At least one German engineer had worked at Bell Labs before the war and came up with a way to break them. Later versions were sufficiently different that the German team was unable to unscramble them.

1.2 Description

There are number of ways to make a speech secure with varying strength and processing. For landline communication at tactical level in military and in large civil organization, there is a need for low level voice security, less processing and a low cost device. Keeping in view the above three considerations, ‘scrambling’ is a better option to be used. A scrambler is a device that transposes or inverts signals or otherwise encodes a message at the transmitter to make the message unintelligible at a receiver not equipped with an appropriately set descrambling device. Scrambler manipulates a data stream before transmitting. The manipulations are reversed by a descrambler at the receiving side. Scrambling is widely used in satellite, radio relay communications and PSTN modems.

While digital scrambling provides very high security, it is usually difficult to implement due to its fundamental requirement of much wider bandwidth as compared to analog signals. In order to address these issues, usually analog scrambling techniques are utilized. However, such techniques provide relatively low security as compared to digital scrambling. Keeping both aspects in view, digital scrambling technique has been selected as a trade off.

A scrambler replaces sequences (referred to as whitening sequences) into other sequences without removing undesirable sequences, and as a result it changes the probability of occurrence of vexatious sequences. Clearly it is not foolproof as there are input sequences that yield all-zeros, all-ones, or other undesirable periodic output sequences. A scrambler is therefore not a good substitute for a line code, which, through a coding step, removes unwanted sequences.

The implementation of digital scrambling technique offers several challenges. Being computationally intensive, the scheme requires high speed processing for real time applications. Since it has to process the human speech and also apply the algorithms for scrambling/descrambling and for this purpose an effective program has to be written.
Keeping in view the additional bandwidth requirements for digital scrambling we have selected linear feedback shift register (LFSR) algorithm which does not require extra bandwidth because it only manipulates the data bits without adding bits into the data. It also provides the benefit of synchronisation at receiver by removing long sequences of 1 and 0's.

1.3 Application

Analogue landlines are not encrypted, and it is very easy to tap them. Such tapping requires physical access to the line, easily obtained from a number of places, e.g. the phone location, distribution points, cabinets and the exchange itself. Tapping a landline in this way can enable an attacker to make calls which appear to originate from the tapped line or provide an opportunity to eavesdropper who can listen important information conveyed from sender to receiver.

The project will provide a low grade secure and reliable real time communication system on wired medium for Armed Forces especially at platoon level to perform security tasks in current LIC environment. The project can also be used by large organizations other than military for internal landline communication.

1.4 Model Diagram

Model diagram shown in figure 1.1 has two parts:

i. Transmitter
ii. Receiver

Human speech signal is fed to microphone which converts speech signal into electrical signal which is analog in nature. This analog signal is then processed through ADC of transmitter block

which converts it into digital data. This digital data is then fed to scrambler block which performs scrambling algorithm on digital data to scramble it. After processing of data through scrambler block, it is fed to DAC which converts digital data back to analog signal for transmission over analog line. This analog signal is then transmitted over wired medium.

Towards receiver side, received analog signal is again fed to ADC for conversion of analog signal into digital. In descrambler block of receiver, reverse process of scrambler block is carried out to recover the original contents being transmitted from user to the intended listener. After performing descrambling algorithm, digital data is converted to analog using DAC. Analog signal obtained from DAC is fed to speaker for conversion of electrical signal into sound signal so that intended listener can listen it.

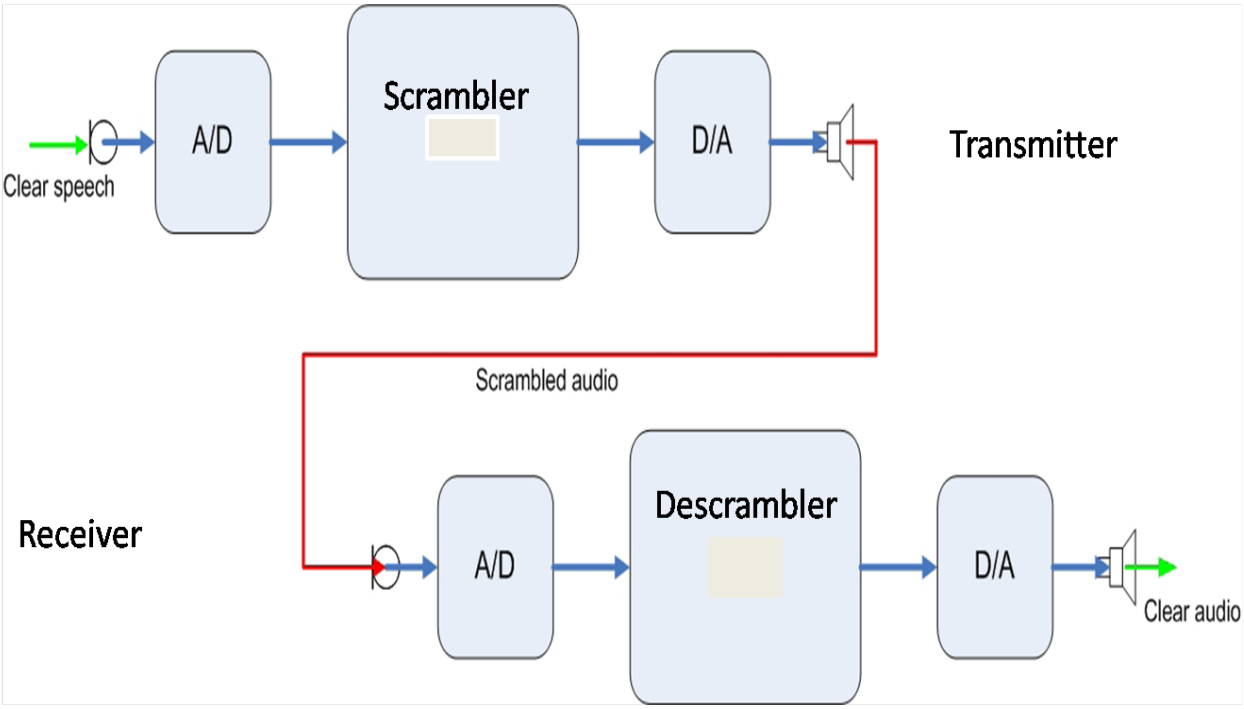


Figure 1.1: Model Diagram

1.5 Scope and Objectives

For landline communication at tactical level in military there is a need for low grade voice security, less processing and a low cost device. So the objectives of this project include designing and developing a secure communication system over wired communication channel and to achieve an optimized combination of security and robustness of performance, digital

scrambling is implemented. Electronic circuit designing and development of DSP based digital speech scrambling system is the main aim of the project.

i. Selection of Voice Scrambling Technique.
ii. Implementation of Voice Scrambling Technique through TMS 320C6713.
iii. Design of Appropriate Hardware.

CHAPTER 2

LITERATURE REVIEW

2.1 Human Voice

For the development and understanding of a voice scrambling system it is existentially to know some basic parameters of the voice itself. In fact the theory is quite more complex than illustrated in the following chapter.

The human voice is simply a sound or audio signal which is generated by a human being to communicate with one and another. Therefore, men use their vocal folds to modulate the air stream, coming from the lungs, into vibrations which make, with the rest of the human sound forming system (like: mouth hole, tong etc), a sound. This sound is treated by some different effects to make a variety of sounds like vowels or consonants.

Hence to the difference of the anatomic of every human the voice of two people never sound the same. The biggest difference is obviously recognized between the voices of men, women and children. This is due to the different fundamental frequencies (often also referred as pitch of the voice) these three groups usually have. These fundamental frequencies are mainly generated by the vocal cords which create through their opening and closing a periodic signal with the fundamental frequency f0.

The male vocal cords are between 17 and 25 mm in length and their fundamental frequency are between 85 and 155 Hz, the ones of a woman are between 12.5 and 17.5 mm in length which obtains a fundamental frequency between 165 and 250 Hz. Kids have even smaller cords, their fundamental frequency is often about 440 Hz (Babies up to 500Hz).

The human speech consists out of vowels and consonants which both are important for understanding. Vowels are voiced sounds where the vocal cords vibrate while they are produced. Those ones have a very clear, narrow and harmonic spectrum that is at lower frequencies. Some consonants are unvoiced or voiceless sounds which have a wider and higher spectrum as Figure

2.1 shows the frequency distribution for one-octave bands.

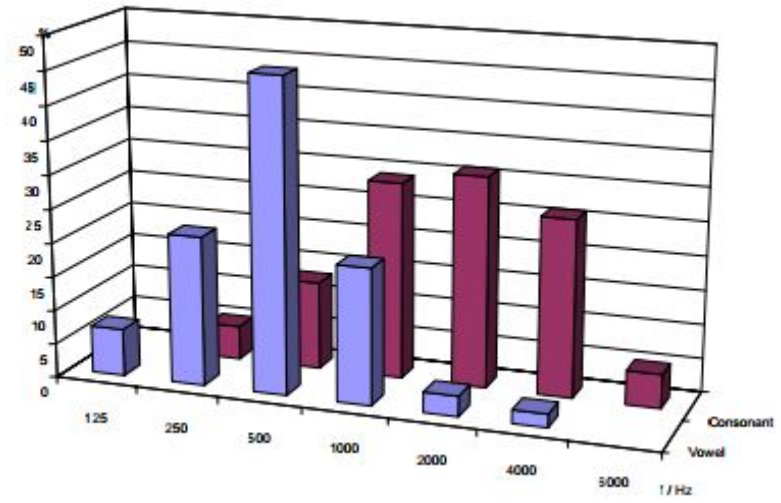


Figure 2.1: Frequency Distribution of the Human Voice

However there is a kind of diagram that is often used for speech signals which is the spectrogram. In this one, the frequencies are plotted over the time and the amplitude is marked with the intensity of the colour like in Figure 2.2.

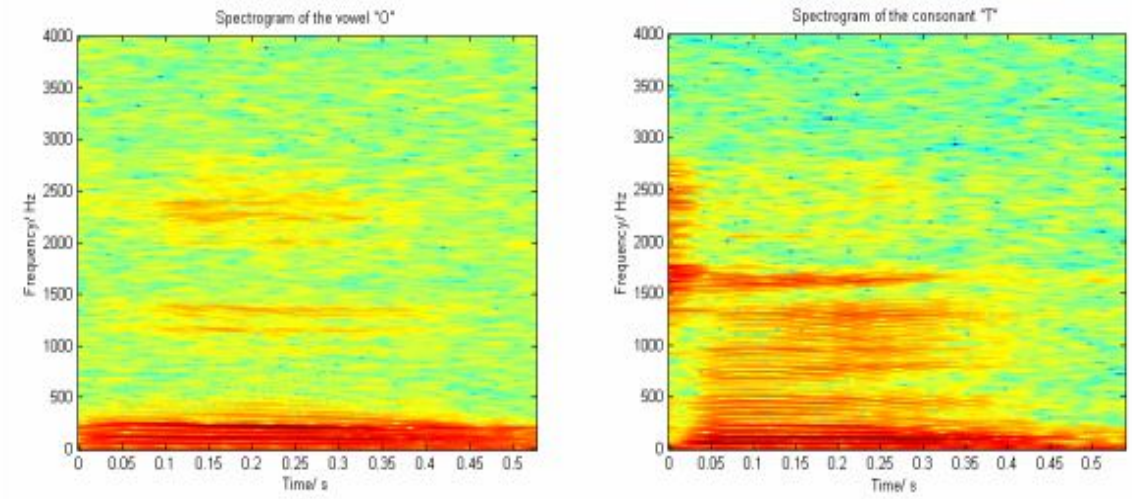


Figure 2.2: Spectrogram of Vowels and Consonants

As observed in the left spectrogram of Figure 2.2 the vowel has a predominant low spectrum. The right spectrogram shown in Figure 2.2 is the one of a consonant followed by a vowel sound. The sound of the human voice is of course a combination of the vowels and of the consonants frequencies which leads into a spectrogram like Figure 2.3 for a normal spoken sentence.

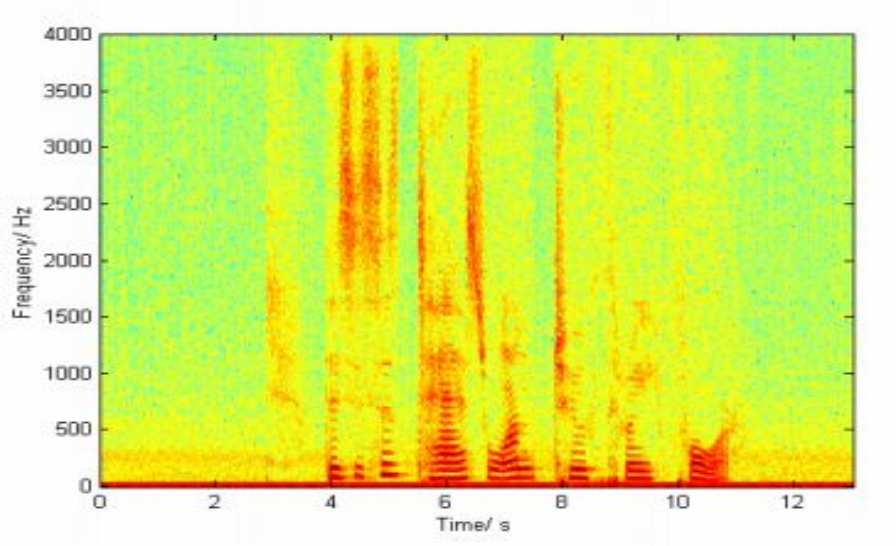


Figure 2.3: Combined Spectrogram of Vowels and Consonants

2.2 Spectrogram of a Human Voice

The complete spectrum of the human voice is individually about 1,3 – 2,5 octaves (where the ear has around 10 octaves). That makes the average range of voice about 80 Hz to 12 kHz. Where in this bandwidth are some sub bands more important than others because those are necessary for the ability of understanding and more significant for the difference of the voice.

For less bandwidth consumption the frequency used in telephony ranges from approximately 300 Hz to 3400 Hz, the fundamental frequency of most speech falls in this range, so that there are always harmonic series in the voice signal to give the listener the feeling of hearing the fundamental tone. With some guard bands added you can use a sample rate of 8 kHz.[5]

2.3 Basic Scrambling Action

Data transmission through any communication system will be errorless if the timing of each device attached to the system is accurate enough. But it is difficult to ensure the accuracy of the

timing in a larger system having many devices or having very large data packets. The solution to this problem of highly accurate clocks can be found by using synchronous communication techniques. In synchronous systems, it is necessary to extract timing information from the received data, which in turn reduces the burden of internal clock circuitry. There are many methods using which timing information from the received data can be retrieved. Line coding techniques like Return to Zero coding and Manchester coding are the few familiar techniques that may be used for this purpose [1]. The problem with the usage of these line-coding techniques is that the timing benefits come at the expense of bandwidth, so these techniques are not used in Public Switched Telephone Network (PSTN), which is severely band-limited network, so one has to choose other options. Scrambling is a technique that can be used in conjunction with simpler line coding algorithms such as simple binary and RS232C protocol to achieve above-mentioned goal without sacrificing the bandwidth requirement.

In digital system, it is common practice to encounter long strings of 1's and 0's within the transmitted data that results in constant output levels. Timing information cannot be retrieved from such outputs because there will be no state-transition during these sequences which may result in transmission errors at the receiver. Scrambler can eliminate this problem by detecting undesirable sequences of bits and inserting state transitions in a pseudo random manner. If there is a long sequence of 1's, 0's will be pseudo randomly inserted in to the stream. It ensures that the probability of receiving a '1' is equal to the probability of receiving '0' and minimizes the probability of periodic or repetitive data transmission, which in turn make clock recovery easier. Simple scrambling action of a scrambler is shown in Figure 2.4. While it may not be possible to prevent the occurrence of all undesirable sequences with absolute certainty, at least most of the common replications in the input data stream can be removed by the use of a scrambler. It has been found that the transmission of short repetitive patterns could play havoc with both the equalizer and timing recovery systems.[1]

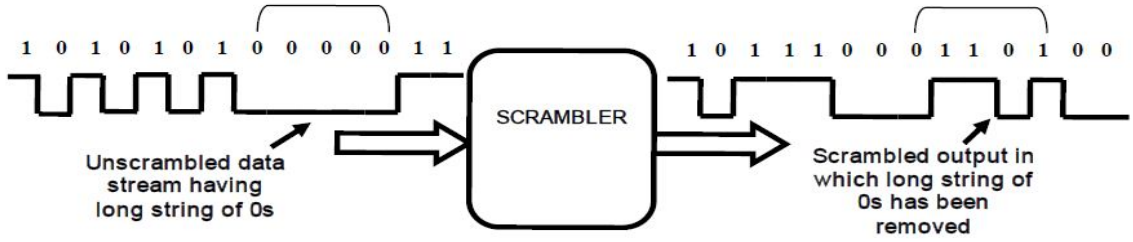


Figure 2.4: Simple Scrambling Action

2.4 Utilization of Scrambler

In the beginning, scrambler was used to reduce the effect of jitter in the PCM systems. Jitter is a very serious problem as it reduces the Signal-to-Noise Ratio (SNR) and causes more errors to be introduced at the receiver. In addition, jitter also effects the functioning of the repeaters, which has a commutative effect. The proceeding timing circuits generates systematic jitter, which degrades the transmission quality because the r.m.s. value of the jitter increases in proportion to the square root of the number of repeaters. A self-retiming circuit is used in conventional base-band PCM systems to minimize the jitter in which a timing waveform is extracted from an equalized pulse train.

There are certain impairments that vary with the statistics of the digital source in digital transmission systems and these statistics of the data source is related to the problem of timing recovery, equalization and cross talk. One of the methods to isolate the system performance from the source statistics is to use redundant transmission codes. These codes could not provide complete isolation however they generate additional problems by increasing the symbol rate. Alternative method to cope up with this problem is scrambling, which whitens the statistics of digital source. Any source is said to white if it generates statistically independent and equiprobable symbols using which system impairment can be easily analyzed. All the first order and second order statistics of any binary source can be whiten to any degree at the cost of an arbitrarily small controllable rate.[1]

2.5 Linear Feedback Shift Register

Linear Feedback Shift Register (LFSR) is used to generate pseudo random numbers. LFSR has two main parts. They are shift register and feedback function.

The shift register can move its content in both directions, either in left or in right direction. It shifts its content to adjacent positions and also shifts the content if the end position is vacant. During the shift the content of the end position bit is moved out and with the help of the feedback function the vacant position is filled. The result of the feedback function is inserted into the shift register during the shift, filling the position that is emptied as a result of the shift.

The initial value of the LFSR is called the seed, and because the operation of the register is deterministic, the stream of values produced by the register is completely determined by its

current state. Likewise, because the register has a finite number of possible states, it must eventually enter a repeating cycle. However, an LFSR with a well-chosen feedback function can produce a sequence of bits which appears random with a very long cycle.[2]

2.6 Linear Feedback Shift Register Sequences

Feedback shift register sequences have been widely used as synchronization codes, masking or scrambling codes, and for white noise signals in communication systems, signal sets in CDMA (code division multiple access) communications, key stream generators in stream cipher crypto systems, random number generators in many cryptographic primitive algorithms, and for testing vectors in hardware design. S. Golomb's popular book "Shift Register Sequences", first published in 1967 and revised in 1982 is a pioneering book which discusses this type of sequences.[4]

2.7 Types of Scrambler

1. Additive (synchronous) Scrambler
2. Multiplicative (self-synchronizing) Scrambler

2.7.1 Additive Scrambler

Additive scramblers (they are also referred to as synchronous) transform the input data stream by applying a pseudo-random binary sequence (PRBS) (by modulo-two addition). Sometimes a pre-calculated PRBS stored in the Read-only memory is used, but more often it is generated by a linear feedback shift register (LFSR). In order to assure a synchronous operation of the transmitting and receiving LFSR (that is, scrambler and descrambler), a sync-word must be used. A sync-word is a pattern that is placed in the data stream through equal intervals (that is, in each frame). A receiver searches for a few sync-words in adjacent frames and hence determines the place when its LFSR must be reloaded with a pre-defined initial state. The additive descrambler is just the same device as the additive scrambler. Additive scrambler/descrambler is defined by the polynomial of its LFSR (for the scrambler in the figure 2.5, it is $1 + x^{-14} + x^{-15}$) and its initial state.

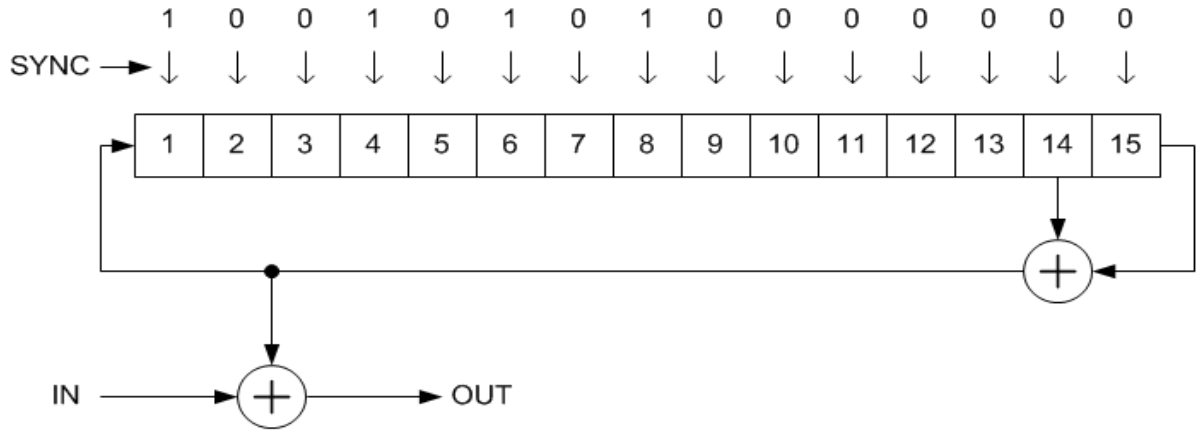


Figure 2.5: Additive Scrambler

2.7.2 Multiplicative Scrambler

Multiplicative scramblers (also known as feed-through) perform a multiplication of the input signal by the scrambler's transfer function in Z-space. They are discrete linear time-invariant systems. A multiplicative scrambler is recursive and a multiplicative descrambler is non-recursive. Unlike additive scramblers, multiplicative scramblers do not need the frame synchronization that is why they are also called self-synchronizing. Multiplicative scrambler/descrambler is defined similarly by a polynomial (for the scrambler in the figure 2.6 it is $1 + x^{-18} + x^{-23}$), which is also a transfer function of the descrambler.

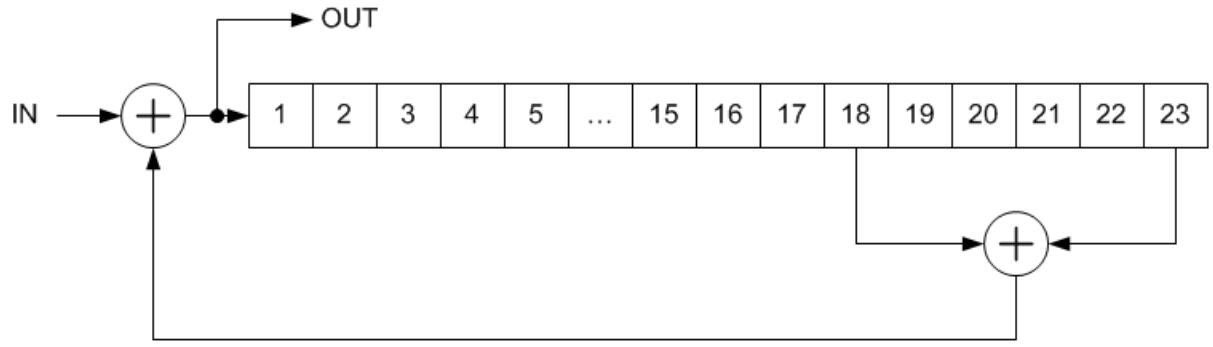


Figure 2.6: Multiplicative Scrambler

CHAPTER 3

SYSTEM DESIGN

3.1 Implemented Algorithm

There are many ways to implement scrambler but all rely on the same basic building blocks of linear feedback shift registers and modulo-2-addition functions. Many researchers have discussed the general theory of implementation of scramblers. In general, the serial data enter in linear feedback shift register, where each stage in the register delays the signal by one time unit as shown in Figure 3.1. The delayed version of the output signal is then fed back and modulo-2-addition is performed with the input signal.

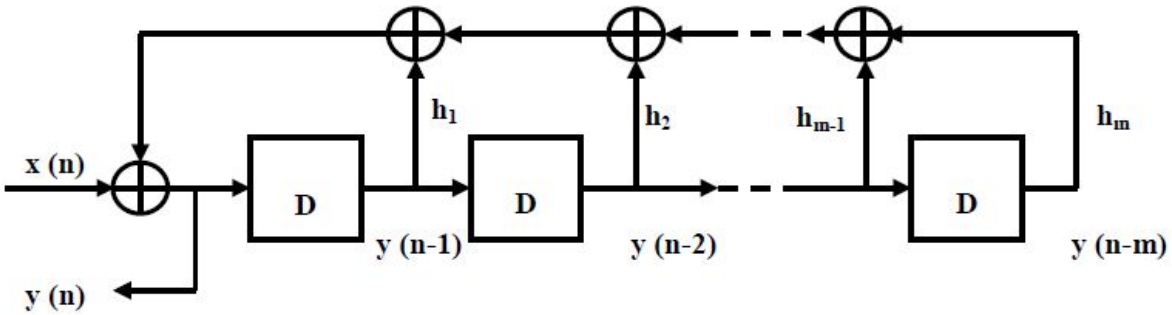


Figure 3.1: Basic Structure of a LFSR

The scrambler's input and output relation, in general, is given by

$$y(n) = x(n) + \sum_{k=1}^{m} h_k y(n-k) \qquad \longrightarrow (1)$$

Where y(n) is current output, y (n-k) is the output delayed by k times, x(n) is current input and hk is system transform function. All the constants and variable in equation (1) can only have the values '0' or '1' and all additions performed are modulo-2-additions (exclusive-OR operations).[1]

3.2 Project Components Description

3.2.1 Digital Signal Processor

A digital signal processor (DSP) is an integrated circuit designed for high-speed data manipulations, and is used in audio, communications, image manipulation and other data-acquisition and data-control applications.

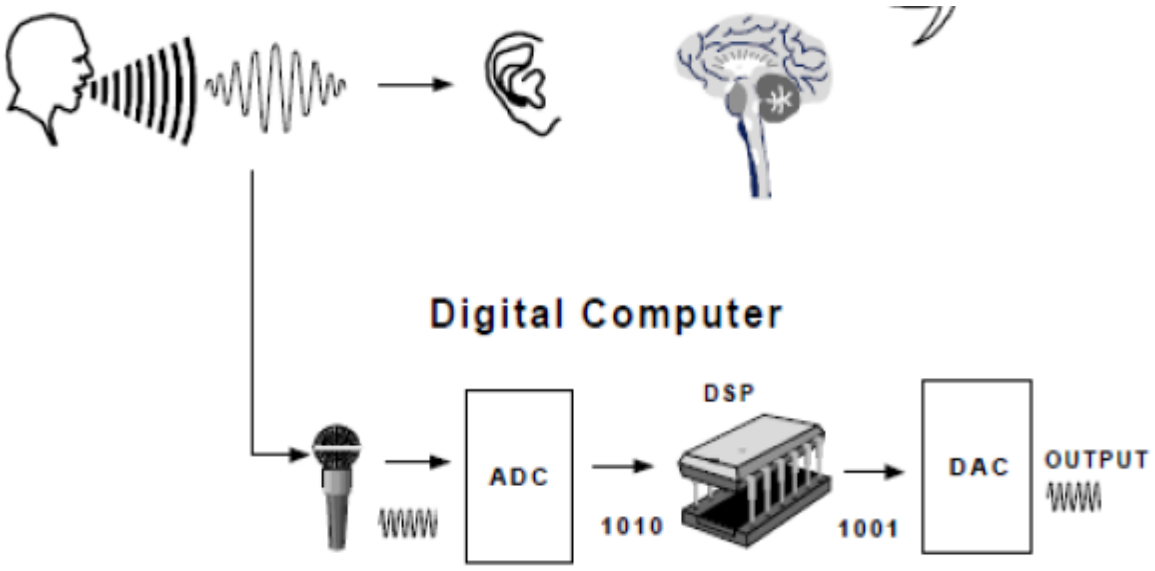


Figure 3.2: Human Voice Processing Through DSP

3.2.1.1 How Analog and Digital Signals Work Together

Digital technology such as personal computers (PCs), assist us in many ways: writing documents, spell checking, and drawing. Unfortunately, the world is analog, and electronic analog computers are not as versatile as digital computers. Therefore, in order to make use of the tremendous processing power that digital technology offers us, we must do the following: Convert the analog signals into electrical signals, using a transducer (such as a microphone, as shown in the diagram). Digitize these signals (i.e., convert them from analog to digital using an analog-to-digital converter (ADC)), as shown in the diagram.

Once the signal is in digital form, our computer can easily process it through a digital signal processor. The DSP specializes in processing these signals, which makes it slightly different from microcomputers, microcontrollers, or general-purpose microprocessors. After the DSP has

processed the signal, the output signal must be converted back to analog form so that we can sense it. This is the digital-to-analog (DAC) conversion stage in the diagram. A loudspeaker, for example, would reproduce analog signals coming from the DAC into sound.

3.2.1.2 Which Architecture is Best Suited for DSP?

Common general-purpose personal computers use processors designed with the Von Neuman architecture while the Harvard architecture is more commonly used in specialized microprocessors for real-time and embedded applications. DSPs typically use Harvard architecture, although Von Neuman architecture also exists.

3.2.1.3 Components of a Typical DSP System

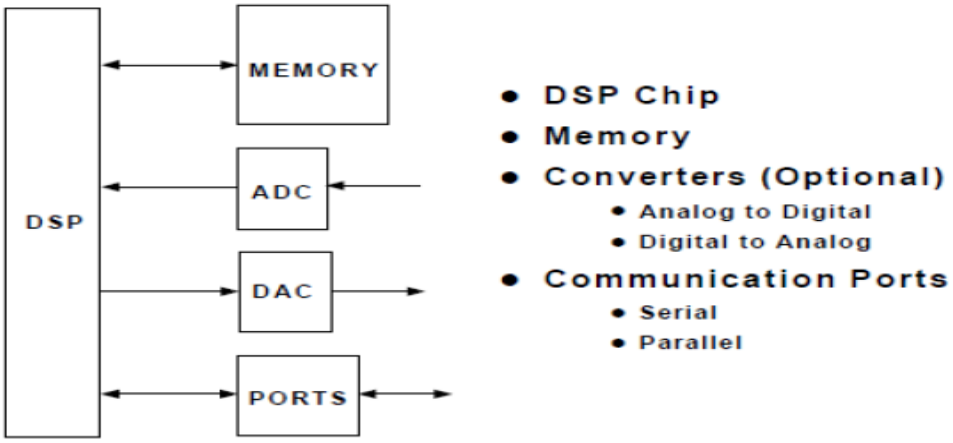


Figure 3.3: Components of DSP System

3.2.1.4 Practical Applications for DSP Systems

DSPs have found a wide variety of applications. They are used in everyday hi-fi systems as well as high-end virtual-reality applications. Generally, DSP is not an expensive technology. Some practical DSP systems are:

- Hi-Fi Equipment
- Toys
- Videophones
- Modems
- Phone Systems
- 3D Graphics Systems

- Image Processing Systems

3.2.1.5 Advantages to Digital Processing

- Programmability
- Stability
- Repeatability
- Special Applications

3.2.1.6 Programmability

- One Hardware = Many Tasks

Figure 3.4: Multitasking through DSP

3.2.1.7 Upgradability and Flexibility

- Develop New Code Upgrade
- Solder New Component

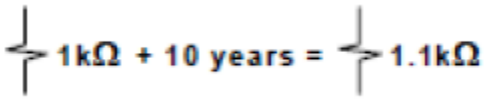
3.2.1.8 Analog Variability

Analog Circuits are affected by

- Temperature
- Aging

3.2.1.9 Tolerance of Components

- Two Analog Systems using the same design and components may differ in performance



3.2.1.10 Perfect Reproducibility

- Nearly identical performance from unit to unit
- Performance not affected by tolerance
- No drift in performance due to temperature or aging
- Guaranteed accuracy

Example is A CD player always plays the same music quality.

3.2.1.11 TMS320 Family

The Texas Instruments TMS320 family of DSP devices covers a wide range, from a 16-bit fixed-point device to a single-chip parallel-processor device. In the past, DSPs were used only in specialized applications. Now they are in many mass-market consumer products that are continuously entering new market segments.

3.2.1.12 DSK – DSP STARTER KIT TMS320C6713

It includes:

- DSK – DSP STARTER KIT TMS320C6713 made *by* Texas Instruments
- Code Composer v3.1
- MATLAB- Matrix Laboratory

3.2.1.13 Basic Features of DSK TMS320C6713

The basic features of a DSP Processor are:

Feature	Use
Fast-Multiply accumulate	Most DSP algorithms including filtering, transforms etc. are multiplication- intensive
Multiple - access memory architecture	Many data-intensive DSP operations require reading a program instruction and multiple data items during each instruction cycle for best performance
Specialized addressing modes	Efficient handling of data arrays and first-in, first-out buffers in memory

Specialized program control	Efficient control of loops for many iterative DSP algorithms. Fast interrupt handling for frequent I/O operations.
On-chip peripherals and I/O interfaces	On-chip peripherals like A/D converters allow for small low cost system designs. Similarly I/O interfaces tailored for common peripherals allow clean interfaces to off-chip I/O devices.

3.2.1.14 Architecture of 6713 DSP Processor

The C67xx DSPs use an advanced modified Harvard architecture that maximizes processing power with eight buses. Separate program and data spaces allow simultaneous access to program instructions and data, providing a high degree of parallelism. For example, three reads and one write can be performed in a single cycle. Instructions with parallel store and application-specific instructions fully utilize this architecture. In addition, data can be transferred between data and program spaces. Such Parallelism supports a powerful set of arithmetic, logic, and bit-manipulation operations that can all be performed in a single machine cycle. Also, the C67xx DSP includes the control mechanisms to manage interrupts, repeated operations, and function calling.

3.2.1.15 TMS320C6713

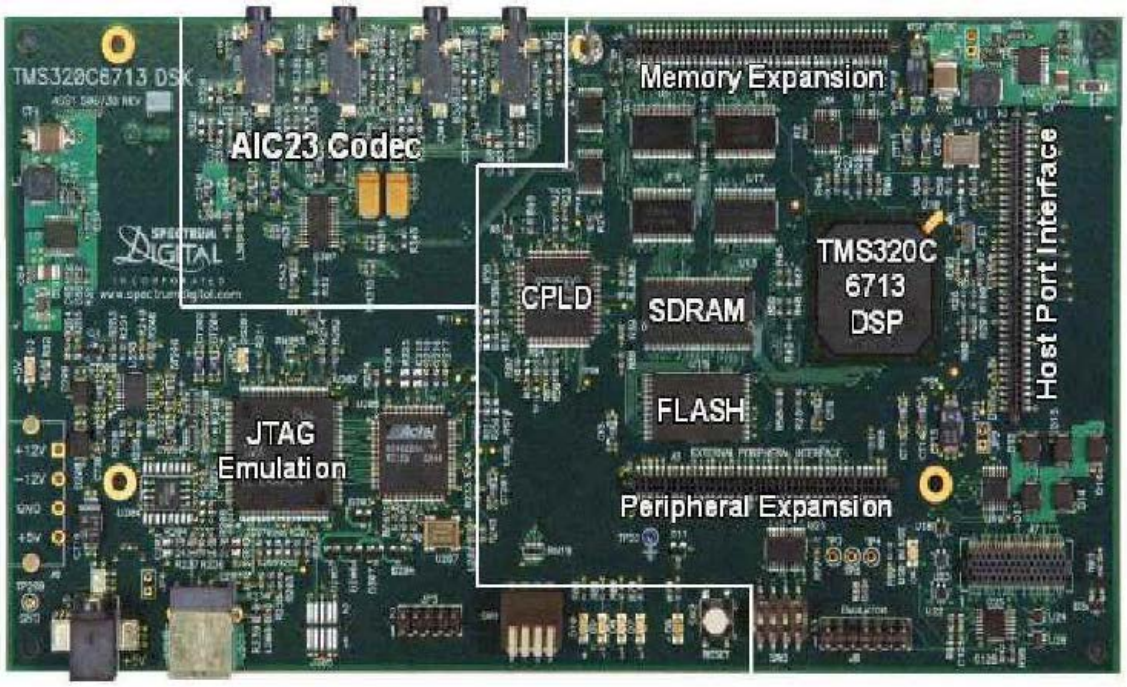


Figure 3.5: TMS 320C6713

Basic block diagram of C6713 DSK is shown in figure 3.6.

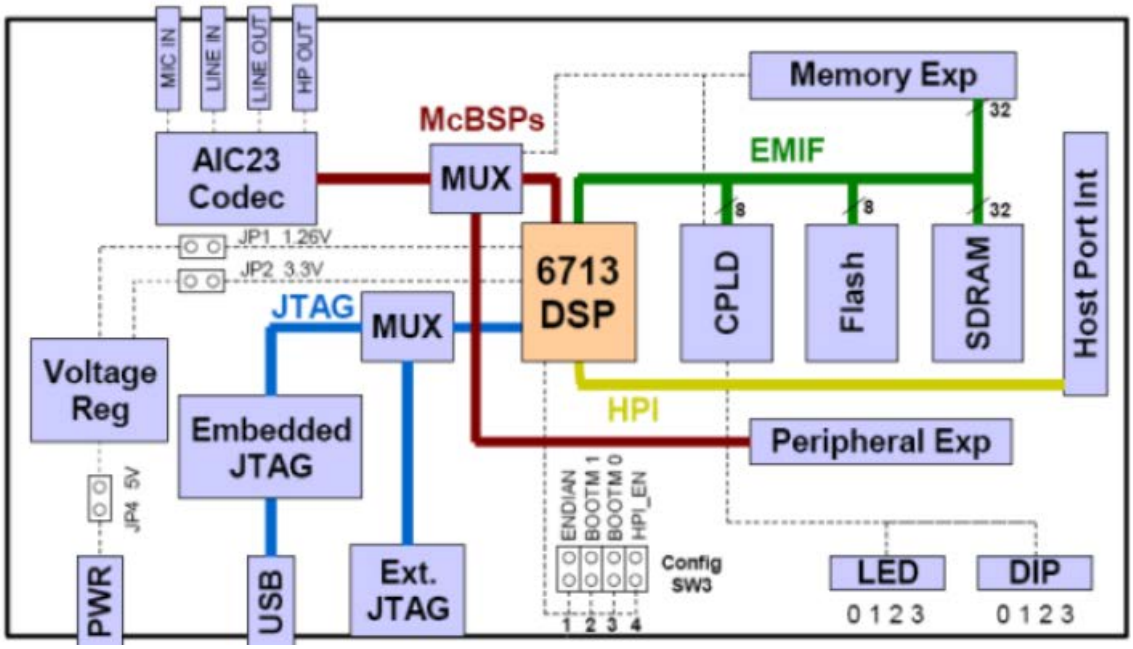


Figure 3.6: Basic Block Diagram

3.2.1.16 Specification of TMS 320C6713

- 225 MHz TMS 320C6713 floating point DSP
- AIC 23 Stereo Codec (ADC and DAC)
  - Ideal for audio applications
  - 8 – 96 KHz sample rates
- Memory
  - 16 MB Dynamic RAM
  - 512 KB nonvolatile flash memory
- General purpose I/O
  - 4 LEDs
  - 4 DIP switches
- USB interface to PC

3.2.2 LM 7805 Voltage Regulator Integrated Circuit

The **78xx** (sometimes **L78xx**, **LM78xx**, **MC78xx**...) is a family of self-contained fixed linear voltage regulator integrated circuits. The 78xx family is commonly used in electronic circuits

requiring a regulated power supply due to their ease-of-use and low cost. The 78xx lines are positive voltage regulators: they produce a voltage that is positive relative to a common ground. There is a related line of **79xx** devices which are complementary negative voltage regulators. 78xx and 79xx ICs can be used in combination to provide positive and negative supply voltages in the same circuit.

78xx ICs have three terminals and are commonly found in the TO220 form factor, although smaller surface-mount and larger TO3 packages are available. These devices support an input voltage anywhere from a few volts over the intended output voltage, up to a maximum of 35 to 40 volts depending on the make, and typically provide 1 or 1.5 amperes of current (though smaller or larger packages may have a lower or higher current rating).

7805 is a member of 78xx series of fixed linear voltage regulator ICs. The voltage source in a circuit may have fluctuations and would not give the fixed voltage output. The **voltage regulator IC** maintains the output voltage at a constant value. The xx in 78xx indicates the fixed output voltage it is designed to provide. LM7805 provides +5V regulated power supply. Capacitors of suitable values can be connected at input and output pins depending upon the respective voltage levels.

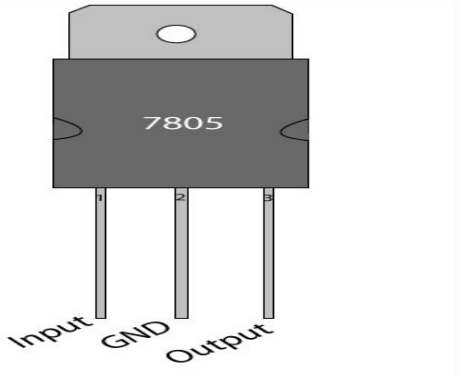


Figure 3.7: LM 7805 Voltage Regulator

3.2.2.1 Advantages

- It does not require additional components to provide a constant, regulated source of power, making them easy to use, as well as economical and efficient uses of space. Other voltage regulators may require additional components to set the output voltage level, or to assist in

the regulation process. Some other designs (such as a switched-mode power supply) may need substantial engineering expertise to implement.

- It has a built-in protection against a circuit drawing too much power. It has protection against overheating and short-circuits, making them quite robust in most applications. In some cases, the current-limiting features of the 78xx devices can provide protection not only for the 78xx itself, but also for other parts of the circuit.

3.2.3 MAX-515 Digital to Analog Converter Integrated Circuit

The MAX-515 is a low-power, voltage-output, 10-bit digital-to-analog converter (DAC) specified for single +5V power-supply operation. The MAX-515 draws only 140µA current. It comes in 8-pin DIP and SO packages. It has been trimmed for offset voltage, gain, and linearity. So no further adjustment is necessary. The MAX-515's buffer is fixed at a gain of 2.

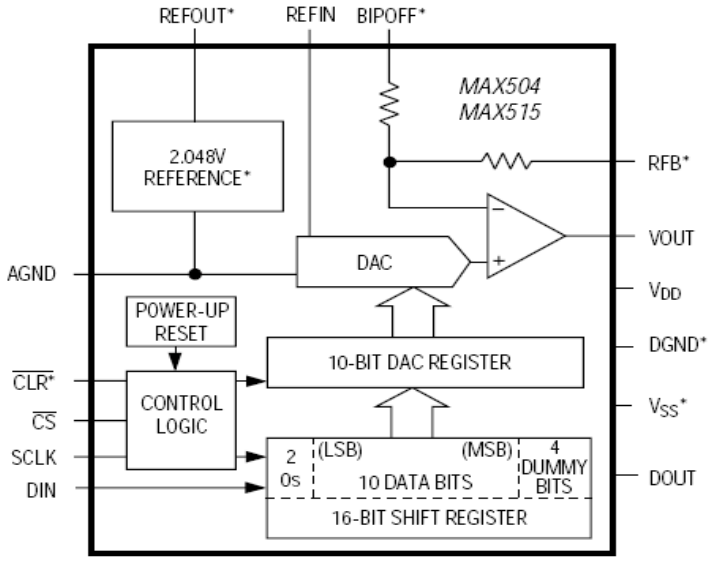


Figure 3.8: MAX – 515 Internal Model

3.2.3.1 Key Features

- Operate from Single +5V Supply
- Buffered Voltage Output
- 140µA Supply Current
- INL = ±1/2LSB (max)
- Guaranteed monotonic over temperature

- Flexible output range - 0V to VDD
- 8-Pin SO/DIP
- Power-On Reset
- Serial Data Output for Daisy-Chaining

3.2.3.2 Applications

- Audio Systems
- Battery-Operated/Remote Industrial Controls
- Battery-Powered Test Instruments
- Digital Gain and Offset Control
- Machine- and Motion-Control Devices

3.2.4 LM - 386 Integrated Circuit

The LM-386 is a power amplifier designed for use in low voltage consumer applications. Gain is internally set to 20 to keep external part count low, but the addition of an external resistor and capacitor between pins 1 and 8 will increase the gain to any value up to 200. The inputs are ground referenced while the output is automatically biased to one half the supply voltages. The quiescent power drain is only 24 milli watts when operating from a 6 volt supply, making the LM-386 ideal for battery operation.

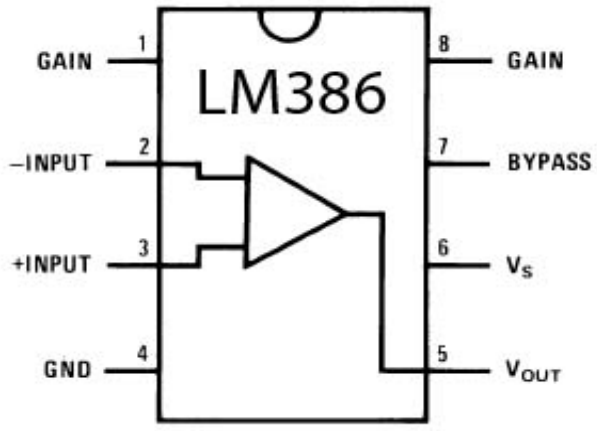


Figure 3.9: LM – 386

3.2.4.1 Features

- Battery operation
- Minimum external parts
- Wide supply voltage range 4V±12V or 5V±18V
- Low quiescent current drain 4 mA
- Voltage gains from 20 to 200
- Ground referenced input
- Self-centering output quiescent voltage
- Low distortion
- Eight pin dual-in-line package

3.2.4.2 Applications

- AM-FM radio amplifiers
- Portable tape player amplifiers
- Intercoms
- TV sound systems
- Line drivers
- Ultrasonic drivers
- Small servo drivers
- Power converters

3.2.5 Condenser Microphone

Condenser means capacitor, an electronic component which stores energy in the form of an electrostatic field. The term condenser is actually obsolete but has stuck as the name for this type of microphone, which uses a capacitor to convert acoustical energy into electrical energy.

Condenser microphone requires power from a battery or external source. The resulting audio signal is signal stronger than that from a dynamic. Condensers also tend to be more sensitive and responsive than dynamics, making them well-suited to capturing subtle nuances in a sound. They are not ideal for high-volume work, as their sensitivity makes them prone to distort.

3.2.5.1 How Condenser Microphone Works

A capacitor has two plates with a voltage between them. In the condenser mic, one of these plates is made of very light material and acts as the diaphragm. The diaphragm vibrates when struck by sound waves, changing the distance between the two plates and therefore changing the capacitance. Specifically, when the plates are closer together, capacitance increases and a charge current occurs. When the plates are further apart, capacitance decreases and a discharge current occurs. A voltage is required across the capacitor for this to work. This voltage is supplied by a battery in the mic.

3.2.6 Voltage Divider

When two resistors of specific values are connected in series and output is taken from the common point of the two resistors, such a configuration is known as voltage divider. In our project the voltage divider is used to change the input voltage from 5 V to 2.5V. This specific voltage is then fed to the microcontroller. We have used four voltage divider circuits in our project each one for solar, wind, hydro and battery. It is used as a protection for microcontroller to avoid any damage due to high voltage.

3.2.7 PIC Controller

PIC microcontrollers are electronic circuits that are used to perform various controlling tasks. Various programming techniques including C language, assembly language are used to program a controller. They are used in many electronic devices such as alarm systems, computer control system, security systems, devices which include timers etc. Many times of PIC controllers exist and vary in the range of memory. These are programmed and simulated by Circuit Wizard software. Memory units, I/O interfaces and processor all exist on a single piece of chip. It can be used to read and write data on electronic devices. In our project we have used PIC18F4550. The PIC 18F4550, is a low-power, high-performance microcontroller with 32KB of in-system programmable Flash memory. The device is manufactured using PIC's high-density non-volatile memory technology and is compatible with the industry-standard and its silent features are as follow.

- Operating frequency of 20MHz (200 ns instruction cycle)
- 32KB Flash Program Memory

- 2KB of Data Memory
- 256 bytes of EEPROM Data Memory
- 40 Pins

3.2.7.1 PIC Architecture

- Peripheral interface controllers are a family of microcontrollers by Microchip technology.
- To programmed a PIC controller is much easier with many attractive features and are suitable for many applications.
- Use of separated program and data memories.
- Separate busses for data and address allow increased data flow to and from the CPU.
- Width of data and address busses vary from each other.
- Harvard architecture is used in PIC18F4550 which makes it much efficient then von-Neumann architecture.

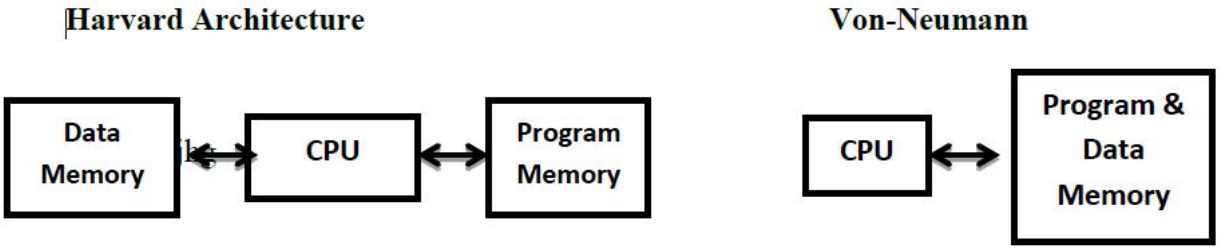


Figure 3.10: Comparisons between Architecture

3.3 Design

Detailed study on DSK TMS 320C6713 has been carried out. It has been found that compatible version of MATLAB with above kit is 2006b. Using MATLAB 2006b, we have designed a model for scrambler in SIMULINK.

3.3.1 Scrambler Block

On the top of model shown in figure 3.11, C6713DSK processor is added to inform SIMULINK about the processor for which model is designed. Audio input is fed to C6713 DSK ADC block which is converting analog audio input to a digital according to set parameters. These parameters

include sampling frequency, type of input data and quantization levels. This digital output is then fed to Linear Feedback Shift Register algorithm as shown in figure 3.11:

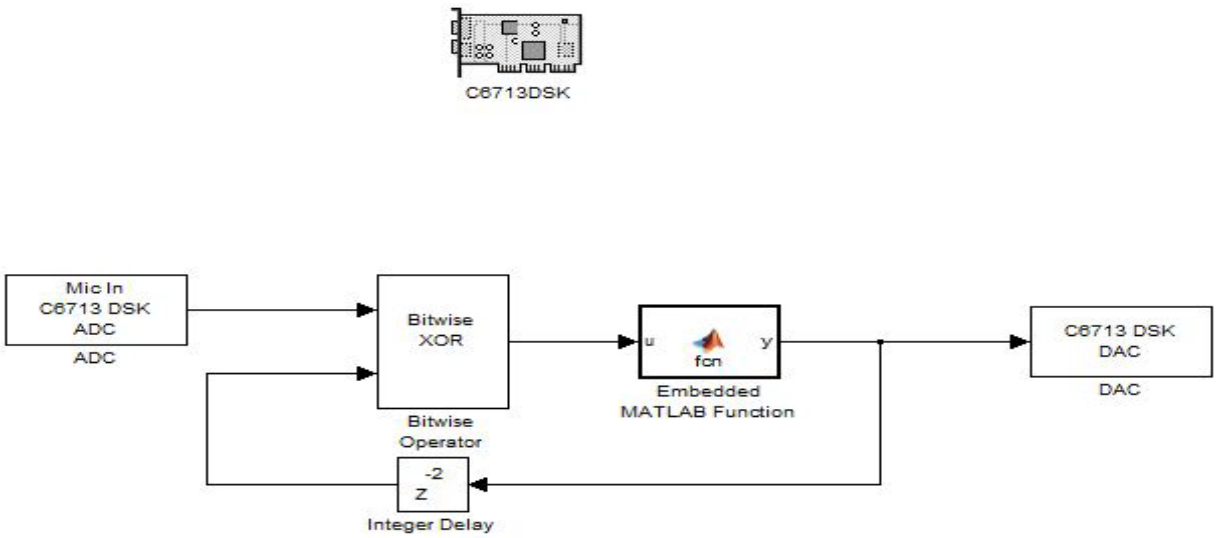


Figure 3.11: Scrambler Block

Input digital data is XOR with second previous output value and then result is passed through an embedded matlab function which acts as a buffer. Buffer is added because matlab does not support direct feedback of output to XOR function. Output obtained after passing through embedded matlab function is scrambled audio in digital form which is converted to analog using C6713 DSK DAC block.

3.3.2 Descrambler Block

In the model shown in figure 3.12, again C6713DSK processor is added at the top to inform SIMULINK about the processor for which model is designed. Scrambled audio input received is fed to C6713 DSK ADC block which is converting analog audio input to a digital according to set parameters. These parameters include sampling frequency, type of input data and quantization levels. This digital output is then fed to a descrambler algorithm which does inverse of scrambler as shown in figure 3.12:

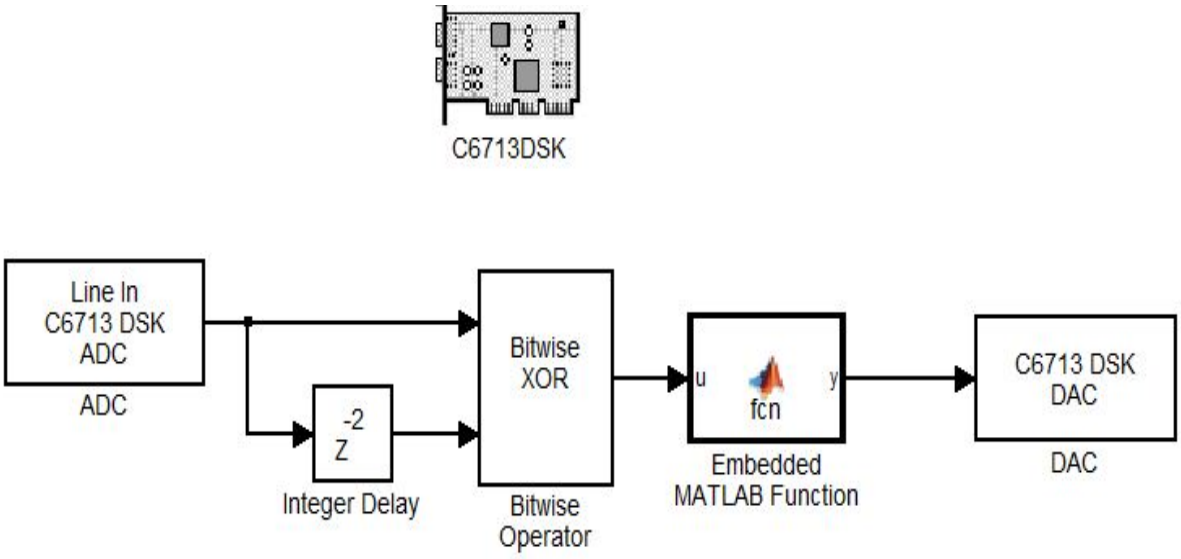


Figure 3.12 Descrambler Block

Input digital data is fed to design algorithm which does inverse process of Scrambler. Second previous input value is XOR with current input value to get descrambled data. Output obtained after processing through algorithm is descrambled audio in digital form which is converted to analog using C6713 DSK DAC block.

This model is then imported to CC Studio 3.1 which converts it to C program. This program is then burnt and executed on DSK TMS 320C6713, result will be demonstrated in the demonstration part.

3.3.3 Practical Circuit Designing and Implementation

Simulation of practical circuit is carried out in Proteus 8.0 first and then practical implementation has been done.

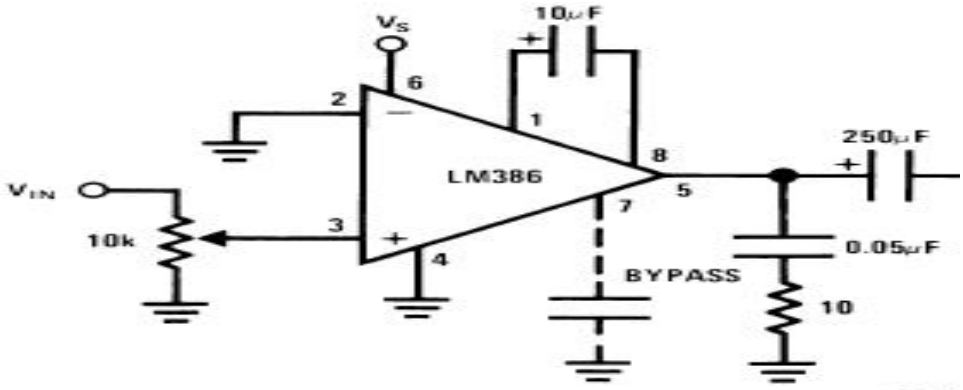


Figure 3.13 Input Signal Amplifier

Voice Signal is input to condenser microphone which is further amplified using LM-386 low voltage audio power amplifier.

This amplified analog audio signal is fed to pin 2 of PIC 18F 4550 which is operating at 20 MHz frequency set by a combination of 2 Capacitors and Crystal Oscillator as shown in figure 3.14.

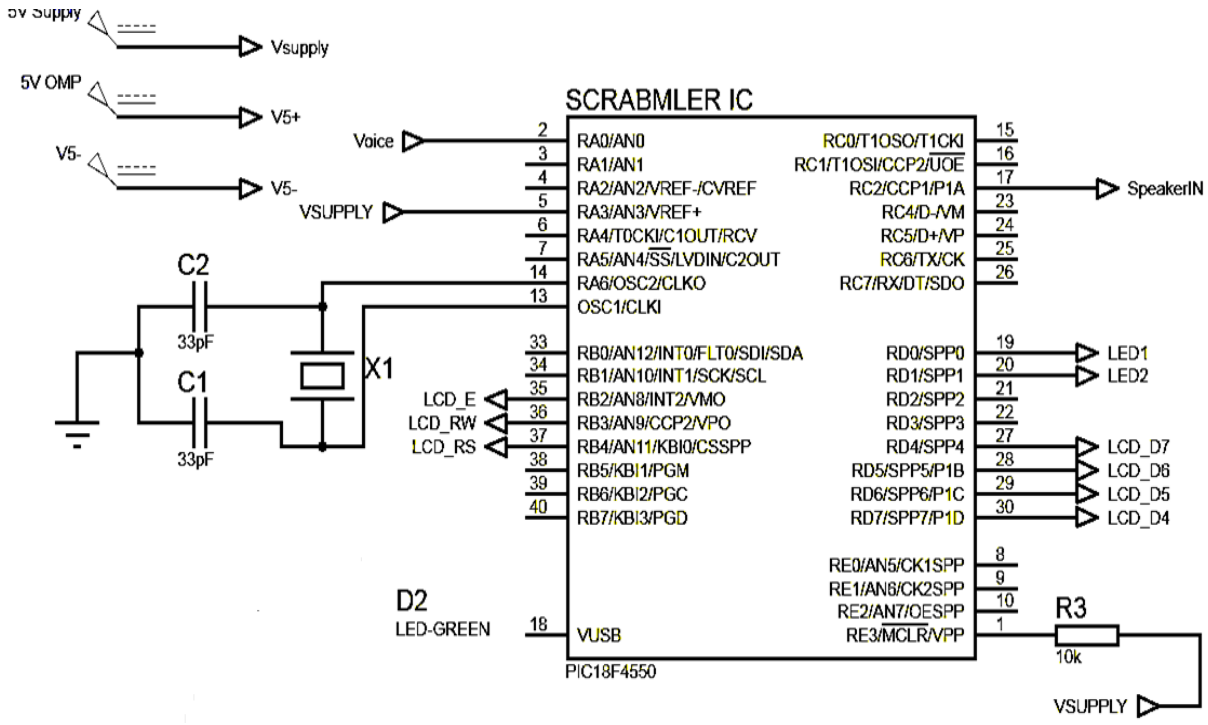


Figure 3.14 PIC 18F4550

Code for scrambling is attached as Appendix - B.

Output of the scrambler is PWM which is converted to analog using MAX 515 integrated circuit. At receiver side, reverse process of scrambling is done using same PIC 18F4550 shown in figure 3.14. Code for descrambling is attached as Appendix - C. At transmitting end, synchronization has been carried out using pin 35 and 36 for indication of start of scrambling and acknowledgement respectively. At receiving end pin 36 is used for indication of start of scrambling and pin 35 for acknowledgment.

CHAPTER 4

ANALYSIS AND EVALUATION

4.1 Results and Analysis

The working of the project is scrutinized and compared with the goals set at the beginning. It is determined whether or not the project was able to fulfill all the requirements specified. The system is analyzed in two dimensions depending upon the performance while testing and the level of security achieved by the process.

4.2 Performance Analysis

The performance of the project is adhered to the success criteria of a number of crucial factors. Some of the important ones are described as under:

4.3 Residual Intelligibility of Scrambling Algorithm

Residual intelligibility is very difficult to determine quantitatively. Residual intelligibility tests were performed using subjective tests. Fifty random digits were spoken and voice data was recorded. This voice was then processed using the implemented scrambling algorithm which in this case is LFSR. Fifteen different students were asked to listen to the scrambled voice data and write down the numbers they heard. In case the student could not understand a number being spoken, he or she was told to write down their best guess.

This method gave a general idea of the intelligibility of the scrambled voice. The results of the test were favorable. On average, a student was able to guess three out of fifty digits correctly. This implies a very low residual intelligibility of the scrambled output.

4.4 Bandwidth Expansion

Bandwidth is a limiting factor of voice communication channels. It was required that the scrambling algorithm used should not increase the bandwidth of the original signal because of implementation of algorithm on the existing systems. As explained in Article 2.3, implemented algorithm does not increase the bandwidth requirement of original signal after scrambling because it only changes the sequence of bits of input digital data and does not add redundant bits into it. So this implies no bandwidth expansion. The result is therefore favorable.

4.5 Cryptographic Strength of Scrambling Algorithm

Cryptanalysis of the system is an important process in secure speech communication systems. In LFSR, there are several advantages to be gained by using the digital domain approach as compared with the analog domain technique. Scrambled speech signal cannot be descrambled by an inverse permutation in the analog domain and this property provides extra cryptanalytic strength to the scheme.

4.6 Final Testing Phase

The voice data was sent to practical implemented circuit. Here the data was processed and the scrambled output was sent over the wire pair to the receiver. The received signal was descrambled and the recovered original signal was heard in the handset of the receiver side. The voice quality obtained was adequate. The volume of the speech was slightly lower than the volume received without scrambling and descrambling data.

Wire pair used between transmitter and receiver was tapped but the sound heard in the headset was unintelligible. Hence, an eavesdropper on the line could not be able to understand the conversation.

4.7 Conclusion – Project Success or Failure

LFSR based speech scrambler implemented and evaluated in terms of security, residual intelligibility and descrambled speech quality. The system is realized on taxes instrument DSK TMS 320C6713 first and then practically through hardware implementation. The results which are based on subjective tests provide very low residual intelligibility and good quality of scrambled and descrambled signals correspondingly. Cryptographic strength of the scrambler is good because of implementation of algorithm in digital domain.

Testing phase showed that all the requirements of the project are met. Bandwidth of the signal is not expanded. With the correct descrambling algorithm, the scrambled speech could be recovered correctly. An eavesdropper could not understand the scrambled speech. Hence the project can be deemed as a success.

4.8 Recommendations for Future Work

The project can be improved upon to provide better results in terms of voice quality, residual intelligibility and increased cryptographic security. Current speech processing research activities

for communication applications are aimed at improving performance under given channel conditions. Research into analog and digital speech scrambling methods show promise of significant advances in the next few years, and could be combined in some instances to achieve double benefits.

Following are recommended for improvement of the project in future:

i. Different algorithms to be designed for a single device which can be switched by user at any time in order to reduce the eavesdropping probability.

ii. Since this scrambling technique is cost effective, therefore, it can be used for wireless communication systems also.

iii. Moreover, for the enhancement of security the scrambling process can be carried out by exploiting the AES on DSP.

BIBLOGRAPHY

Guidance for this project has been obtained from web sources and research papers.

[1] DSP Based Implementation of Scrambler for 56Kbps Modem by Davinder Pal Sharma davinder.sharma@sta.uwi.edu (Department of Physics University of the West Indies St. Augustine, Tinidad & Tobago) and Jasvir Singh j_singh00@rediffmail.com (Department of Electronics Technology Guru Nanak Dev University Amritsar -143105, India

[2] Secure Speech with LFSR by Arathi k Chandran, Sreela Sreedhar
PG Student, Associate Professor, HOD, Department of Computer Science & Engineering, Toc H Institute of Science and Technology, Kerala, India
International Journal of Research in Engineering and Technology

[3] Speech Encryption and Decryption Using Linear Feedback Shift Register (LFSR) by Tin Lai Win, and Nant Christina Kyaw
World Academy of Science, Engineering and Technology Vol:2 2008-12-28

[4] Communication Systems Security, Appendix A, Draft, L. Chen and G. Gong, 2008

[5] Implementation of a real-time voice encryption system by
Markus Albert Brandau, Feb 2008

[6] Linear Feedback Shift Registers in SAGE by Timothy Brian Brock
May 16, 2006

[7] Digital Signal Processing and Applications with the C6713 and C 6416 DSK by Rulph Chassaing

[8] Code Composer Studio, Getting Started Guide, SPRU 509, May 2001, Texas Instruments Inc.

APPENDICES

APPENDIX-A

CODE FOR SCRAMBLING

```
unsigned int v = 0;
unsigned int z1,z2,z3,z4,z5=0;
unsigned int sinput=0;
unsigned int soutput=0;

void scrambler(){
sinput=v;
soutput=sinput^z2^z5^z3;
z5=z4;
z4=z3;
z3=z2;
z2=z1;
z1=soutput;
v=soutput;
}

void interrupt(){
  v=adc_read(0);
if(PORTC.F0==0){
scrambler();
v=v>>2;
}
else
v=v>>2;
 //v1=v<<2;
 //v2=v>>8;
PWM1_Set_Duty(v);
UART1_Write(v);
//UART1_Write(v2);
  if (v>100)
  {
   PORTD.F1=0;
  }
  else
  PORTD.F1=1;

  INTCON.TMR0IF=0;
   TMR0IF_bit = 0;
   TMR0H        = 0xF3;
   TMR0L        = 0xCB;
```

```
}

void main() {
TRISC=0;
TRISD=0;
TRISB=0;
PORTB=255;
TRISC.F0=1;
TRISA.F0=1;
TRISD.F0=0;
TRISD.F1=0;s
PORTD.F0=1;
PORTD.F1=1;
TRISC.F0=1;
TRISC.F6=0;
TRISA |= 0x01;
  LATA = 0;
  ADCON1=0x0E;
  ADCON0=0x00;
  ADCON2=0x80;
  ADCON0.ADON = 1;
TMR2 = 0x00;
PR2 = 22;
CCPR1L = 0x00;
CCP1CON = 0x0C;
TRISC &= 0xFB;
T2CON = 0x04;
INTCON.T0IF = 0;
TMR0L = 0xFE;
T0CON = 0xC0;
INTCON.T0IE = 1;
INTCON.GIE = 1;
PWM1_Init(100000);
T0CON        = 0x88;
 TMR0H        = 0xF3;
 TMR0L        = 0xCB;
 GIE_bit      = 1;
 TMR0IE_bit       = 1;
UART1_Init(57600);
while(1){
if(PORTC.F0==0)
PORTD.F0=0;
else
PORTD.F0=1;
}
}
```

APPENDIX-B

CODE FOR DESCRAMBLING

```
unsigned char tmp;
unsigned int v = 0;
unsigned int v1 = 0;
unsigned int v2 = 0;
unsigned int i = 0;
unsigned int z1,z2,z3,z4,z5=0;
unsigned int sinput=0;
unsigned int soutput=0;
unsigned char x=0;
unsigned char f1=0;
unsigned char f2=0;

void descrambler(){
sinput=v;
soutput=sinput^z2^z5^z3;
z5=z4;
z4=z3;
z3=z2;
z2=z1;
z1=sinput;
v=soutput;
}
void interrupt()
{
if (PIR1.RCIF)
{
v=UART1_Read();
PWM1_Set_Duty(v);
if(v<100)
PORTD.F1=0;
else
PORTD.F1=1;
}
PIR1.RCIF = 0;
}

void main() {
TRISC=0;
TRISD=0;
TRISB=0;
```

```
PORTB=255;
TRISC.F0=1;
TRISA.F0=1;
TRISD.F0=0;
TRISD.F1=0;
PORTD.F0=1;
PORTD.F1=1;
TRISC.F0=1;
TRISC.F7=1;
PWM1_Init(100000);
PWM1_Start();
INTCON.GIE = 1;
INTCON.PEIE = 1;
PIE1.RCIE=1;
UART1_Init(57600);
while(1){
if(PORTC.F0==0)
PORTD.F0=0;
else
PORTD.F0=1;
}
}
```