PERFORMANCE ANALYSIS IN IOT BASED HOME AUTOMATION USING HYBRID RS-LDPC CODES

NG LAY WEI

UNIVERSITI SAINS MALAYSIA

2018

PERFORMANCE ANALYSIS IN IOT BASED HOME AUTOMATION USING HYBRID RS-LDPC CODES

by

NG LAY WEI

Thesis submitted in partial fulfilment of the requirements for the degree of Bachelor of Engineering (Electronic Engineering)

JUNE 2018

ACKNOWLEDGEMENT

I would like to express my deepest gratitude to my supervisor, Dr. Nor Muzlifah Mahyuddin. Her support and guidance throughout the time of conducting the final year project are appreciated. The constructive advice given helped me in completing the thesis.

Secondly, I would like to thank Associate Professor Dr. Mohd Fadzli Bin Mohd Salleh for being my examiner of the thesis. The comments given for pre-viva helped me in improving the program.

I would like to express my appreciation to my family and friends for their moral support and encouragement throughout the thesis. Special thanks to Mr. Anthony Uwaechia who has shared his knowledge and patience to help me in completing the coding part of the program.

Lastly, I would like to thank School of Electrical and Electronic Engineering, Universiti Sains Malaysia for providing me all the facilities needed for my studies. I would like to convey me appreciation to all the lecturers who are working hard to deliver their knowledge to me throughout my studies.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

BER	Bit Error Rate
BCH	Bose–Chaudhuri–Hocquenghem
CRC	Cyclic Redundancy Check
FEC	Forward Error Correction
HTML	Hyper Text Markup Language
IoT	Internet of Things
LCPC	Low Complexity Parity Check
LDPC	Low-Density Parity-Check
PCM	Parity-Check Matrix
RS	Reed-Solomon
SNR	Signal-to-Noise Ratio
VNC	Virtual Network Computing
WiMAX	Worldwide Interoperability for Microwave Access

ANALISIS PRESTASI DALAM AUTOMASI RUMAH BERDASARKAN IOT DENGAN MENGGUNAKAN HYBRID RS-LDPC CODES

ABSTRAK

Internet untuk segalanya (IoT) kini telah diumumkan sebagai perkara besar. Ia telah mengubahkan cara hidup orang lain. Ia menghubungkan peranti fizikal ke Internet dan menjadikannya lebih pintar. Dalam sistem automasi rumah berdasarkan IoT, operasi yang diminta oleh pengguna mungkin mengalami masalah kehilangan kawalam apabila kesilapan dalam bits ysng disebabkan oleh bunyi yang berat di saluran yang memudar. Untuk menangani masalah ini, kerja ini mula menilai prestasi kadar kod yang berlainan bagi kod LDPC yang digunakan dalam penyediaan kod kombinasi kod Reed-Solomon (RS) dan kod Low-Density Parity-Check (LDPC). Tesis ini mencadangkan algoritma pengesanan dan pembetulan ralat menggunakan kod Hybrid RS-LDPC untuk pengekodan dan pengekodan saluran dalam sistem automasi rumah. Prestasi kod Hybrid RS-LDPC dinilai dari segi kecekapan pengekodan dan masa pemprosesan menggunakan papan Raspberry Pi. Maklumat masukan adalah mesej teks yang dihantar dari laman web. Mesej yang dihantar akan diluluskan oleh pengekod RS dan diikuti dengan pengekod LDPC. Kemudian, codeword yang dihasilkan dihantar melalui saluran pudar dengan kurang daripada atau sama dengan 0.9 SNR. Oleh itu, codeword disahkod oleh dekoder LDPC, kemudian diikuti oleh dekoder RS untuk mendapatkan mesej yang diterima dengan bebas ralat. Kadar kod 1/3 dengan 27% dalam kecekapan pengekodan dan 11.9s dalam masa yang digunakan untuk proses penyahkodean adalah lebih sesuai daripada 1/4 untuk dilaksanakan ke dalam kod LDPC dalam kod Hibrid apabila SNR = 0.9. Pengesanan ralat dan algoritma pembetulan yang dilaksanakan dalam sistem automasi rumah berjaya dibangunkan dengan pengekodan dan penyahkodan mesej teks yang dihantar dari laman web ke papan Raspberry Pi menggunakan bahasa pemrograman Python dan HTML. Penyahkodan kecekapan untuk kod Hybrid RS-LDPC adalah 11% lebih banyak daripada kod LDPC apabila SNR = 0.9 dan 50 panjang aksara dihantar dengan sebanyak 100 sampel yang dianalisiskan. Kod Hybrid RS-LDPC memberikan prestasi penyahkodan yang lebih baik daripada kod FEC solo (seperti kod LDPC).

PERFORMANCE ANALYSIS IN IOT BASED HOME AUTOMATION USING HYBRID RS-LDPC CODES

ABSTRACT

Internet of Things (IoT) is no longer an idea that announces being the next big thing, but it is the thing. It has transformed the way people live. It connected the physical devices to the Internet and made them become more intelligent. In home automation system based on IoT, user-requested operations may have malfunction problems when bits error gets generated due to heavy noise in the fading channel. In order to address this problem, this work starts to evaluate the performance of different code rates for LDPC codes used in the preparation for Hybrid combination of Reed-Solomon (RS) and Low-Density Parity-Check (LDPC) codes. The thesis proposes an error detection and correction algorithms using Hybrid RS-LDPC codes for the channel encoding and decoding in the home automation system. The performance of Hybrid RS-LDPC codes is evaluated in term of decoding efficiency and processing time using Raspberry Pi board. The input information is the text messages that transmitted from webpage. The transmitted messages are passed though the RS encoder and followed by LDPC encoder. Then, the generated codeword is sent over a fading channel with less than or equal to 0.9 SNR. Hence, the codeword is decode by LDPC decoder , then followed by RS decoder to obtain the received messages with error-free. Code rate of 1/3 with 27% of decoding efficiency and 11.9s of time taken for decoding process is more suitable than 1/4 to implement into the LDPC codes in Hybrid codes when SNR=0.9. The error detection and correction algorithms implemented in home automation system is successfully developed by encoding and decoding the text messages that transmitted from webpage to Raspberry Pi board using Python and HTML programming language. The efficiency decoding for Hybrid RS-LDPC codes is 11% more than the LDPC codes when SNR=0.9 and 50 of characters length transmitted for total 100 samples analysed. The Hybrid RS-LDPC codes provide better decoding performance than solo FEC codes (such as LDPC codes).

CHAPTER ONE

INTRODUCTION

1.1 Background

The Internet of Things (IoT) has been a buzzword in recent years, but it is no longer to be the next big thing anymore, it is here now and driving an innovation across the way of people's lives and connecting everyone to everything. IoT is defined as a networking paradigm that provides seamless communication between various physical devices over the Internet [1]. IoT connects all web-enabled devices to the Internet to obtain, transfer and process on data from surrounding environment without any human intervention [2].

Smart home has started to take-off for IoT concept in several years, as it has played an important role at business for both technology and tech start-ups companies [3]. Smart home provides people live in a convenient and comfortable environment that contains various devices connected to the Internet. People can monitor the condition of their homes through their smartphones or computers. For example, people can set up devices to automatically control the home's condition based on the settings and alert house owner when an emergency happens.

Unfortunately, IoT based home automation has a latent catastrophe when transmitting information. The bit errors may occur when the operation requested by the user or the relevant script to be implemented on the server get corrupted due to noise, fading, dust, obstacles or any form of external interference in the communication channel [4, 5]. If the system executes the received bit error operation command, it will cause a high degree of risk, thereby impairing the user's safety and security. In home automation, any single data error cannot be acceptable. Hence, the aim of the thesis is to use error detection and correction algorithms to prevent this potential catastrophe.

The bit errors occurred that cause information corruption can be countered by implementing error concealment methods. One of the standard methods to achieve this is Forward Error Correction (FEC) codes [6]. FEC codes attempt to correct errors at the receiver without the need for requesting a retransmission. The error correction is completed based on the parity bits attached by the transmitter to the data. The presence of bit errors is detected at the receiver. If the number of bit errors is lower than the error correction capability of the codes, then the bit errors are corrected. FEC codes are an established solution that can be used to overcome data loss and increase the reliability of data [7].

Different types of FEC codes are used in practice, such as Viterbi codes, Reed-Solomon codes, Golay, BCH, Multidimensional parity, Hamming codes and LDPC codes [8]. Through various research about digital communication in the past [9-11], the hybrid codes have shown its high potential and outstanding performance, which may be applied in wireless communication to provide superior and robustness error correction performance. In the thesis, the Hybrid combination of RS-LDPC codes is chosen as the error detection and correction algorithms for information transmission in the home automation system. The performance of Hybrid RS-LDPC codes is evaluated and compared with LDPC codes, which is a single error correcting code.

1.2 Problem Statements

Today, in this smart and intelligent technology era, IoT based home automation is getting more closer to human life. Now, human has begun to rely on all these "smart" things at home. If these "smart" things are malfunction or out of control, this can cause huge damage to the human's life and property. One of the factors is the operation invoked by user gets corrupted due to the severe noise in fading channel. This provides a good opportunity to explore the performance of the Hybrid codes in heavy noise.

In previous studies [4], RS codes are introduced for error detection and correction before any corrupted operation commands invoked by the user are executed. However, the error correcting capability of a single code is lower than that the capability of hybrid codes. Nowadays a promising technique has been introduced to merge RS and LDPC codes together to generate Hybrid codes to increase the error correcting capability [12]. However, there is lack of discussion on the implementation of Hybrid RS-LDPC codes for IoT-enabled Communication Systems. In addition, most of the research papers are normally simulated in a multi-paradigm numerical computing environment (such as Matlab software) to evaluate the performance of FEC codes [13]. The performance evaluation of Hybrid RS-LDPC codes can be analysed in more realistic way instead of using Matlab. The suggestion is to use Python programming language for Raspberry Pi board implementation.

1.3 Research Objectives

The objectives of the thesis are listed below:

1. To evaluate different code rate performances for LDPC codes that used in the preparation for Hybrid combination of RS and LDPC codes.
2. To design and implement Hybrid RS-LDPC codes on the error detection and correction algorithms in IoT based home automation system using Python programming language.
3. To evaluate the performance of the developed program in term of decoding efficiency and processing time using Raspberry Pi board.

1.4 Scope of Research

The scope of the thesis is to develop a program for Hybrid RS-LDPC codes in IoT based home automation. The program developed is written by using Python programming language in Raspberry Pi board. Virtual Network Computing (VNC) viewer software is used to remotely access the graphical desktop that is running on the Raspberry Pi board. Therefore, user can easily develop the program and obtain results.

In the program, only text message with a length of less than 50 characters will be dealt with. The developed program permits the user to send text messages to the Raspberry Pi board utilizing Flask framework in the web browser in computer, tablet or smartphone that are connected to the same Wi-Fi connection. During data transmission, the sent message is encoded to become codeword by the outer code and then followed by the inner code. When passed through the fading channel, the heavy noise is added to the message. The decoders are applied in reverse order on the received message, in order to attempt to correct the bit errors and obtain back the original sent message.

1.5 Thesis Organization

This thesis is made up of five chapters. Chapter 1 provides an introduction to this project which includes the research background, problem statement, research objectives, scope of research, and thesis organization.

Chapter 2 consists of the literature review which points out the previous review and research on the concept of IoT, home automation system, the structure of RS and LDPC codes, and schematic depiction of Hybrid RS-LDPC codes.

Chapter 3 explains the methodology of this project. In this chapter, the error correction algorithms are proposed by using Hybrid codes in channel encoding and decoding. The overall procedures flow for the algorithms is introduced to make sure the research is done step by step.

Chapter 4 presents the results and discussion pf the thesis in detail. The results for decoding performance analysis are recorded with proper graphs by using Python environment and Raspberry Pi. The explanations and discussions on the results are done.

Chapter 5 gives a conclusion of this final year project. The suggestions for future improvements are described in this final chapter.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

In this chapter, the review of the literature includes valuable sources from various thesis, conference papers, journals, books, and website pages. The content of Chapter 2 is organized as follows. Section 2.2 describes the IoT concept and provides deep research for smart home system. Section 2.3 describes the FEC codes, and emphasise on RS codes, LDPC codes, and Hybrid RS-LDPC codes. The structure and architecture for encoder and decoder of channel codes are discussed at here. Finally, Section 2.4 and 2.5 show the related works and conclusion of this chapter.

2.2 Concept of IoT

Internet of Things (IoT) is establishing a socially convergent environment through the Internet. This connected environment has brought about a model change in the people's professional and personal life. IoT makes all things in the world are linked together with the Internet to achieve more complex and dynamic methods of managing production and life. In recent years, technological innovation has promoted the application of the IoT concept in everywhere which is of human interest, such as healthcare, smart grid, home automation, smart city and others [14, 15].

2.2.1 Home Automation

Before the early of the 20th Century, home automation, or called smart home defines as a characteristic of science fiction writing [16]. However, due to the introduction of electricity and the high-speed improvement in information technology, home automation has become more functionary and practical.

Home automation grows up rapidly in the recent years and affects more and more homeowner adopted this smart home technology. The reasons to understand the explosive growth of home automation are because of its affordability, simplicity, and convenience using smartphone connectivity. All the electrical devices in the house connect with each other using the Internet and programming codes to complete the time-sucking household tasks [17].

Home automation provides centralized access to control physical devices to automatic complete household activities. It can consist of air conditioning, lighting control, heating, appliances, and safety and security system [18]. It can improve convenience, security, comfort, and energy efficiency of the houses.

Chris Woodford [19] mentions that the smart home system meets the needs of those with special needs, such as the elderly and disabled people to handle household tasks and decrease the dependent on caregiver. For example, people with dementia who live in a home that equipped with automated sensors can help them to check whether the tap has overflowed and the cooker has left on to avoid danger. In addition, elderly people live in the home which is installed automated lighting system using motion sensors, can help them to avoid stumbling into darkness when getting up at night. Blind people can live independently in a smart home, which can monitor their home appliances to meet their individual needs with simple commands.

2.2.2 Home Automation System with Raspberry Pi board

When developing new IoT solutions in smart homes, the hobbyist hardware platforms such as Raspberry Pi and Arduino have increased the development speed and provided rapid prototyping and improvement. All of these off-the-shelf boards are allowed to customise a prototype by connecting various sensors or components. Based on the experience of dealing with these hardware boards, Raspberry Pi board has more stable performance in Wi-Fi connection than the Arduino board. This is because Raspberry Pi board has its own built-in Wi-Fi module, while Arduino board does not. Arduino board needs to connect an additional Wi-Fi module to complete the smart home system [20].

In addition, most of the researchers are used Raspberry Pi board as the central control unit in their smart home projects instead of using Arduino board. The reason is that the Raspberry Pi board is a single-board computer that can provide more powerful functions to accomplish complex tasks. The papers presented in [21-23] show the application of Raspberry Pi board in IoT solution projects for home automation, consisting of connectivity, data acquisition and control, data processing and storage, and power management.

The Raspberry Pi foundation sold the Zero board version of the Raspberry Pi board in 2016, which is smaller and much cheaper [24]. This latest version allows users to automate their homes using this board as a central hub without taking up a lot of space. In the thesis, Raspberry Pi Zero W ("W" stands for Wireless) is selected as the hardware board to develop the program that can implement error correcting algorithms, and present it in IoT based home automation system, as Zero W version provides convenient wireless connectivity and its volume is smaller [25].

2.3 Forward Error Correction (FEC) Codes

The performance of wireless communication systems is often difficult to predict. As it is mainly controlled by the dynamic and unpredictable characteristics of the wireless channel environment. Forward error correction (FEC) codes, also known as error correcting codes or channel codes are in the one-way communication medium, hence when the error is found, the receiver has no capacity to apply for retransmission. FEC codes are widely used in wireless communication to ensure the quality of communication. Fig. 2.1 presents the process flow of FEC mechanism [26]. A k -bits data block with additional redundancy is mapped into an n-bits block called a codeword (where n is greater than k) using the FEC encoder. The codeword is transmitted over the unreliable and noisy channel, which may generate bit errors in transmission. The incoming data block is passed through the FEC decoder for error correction.

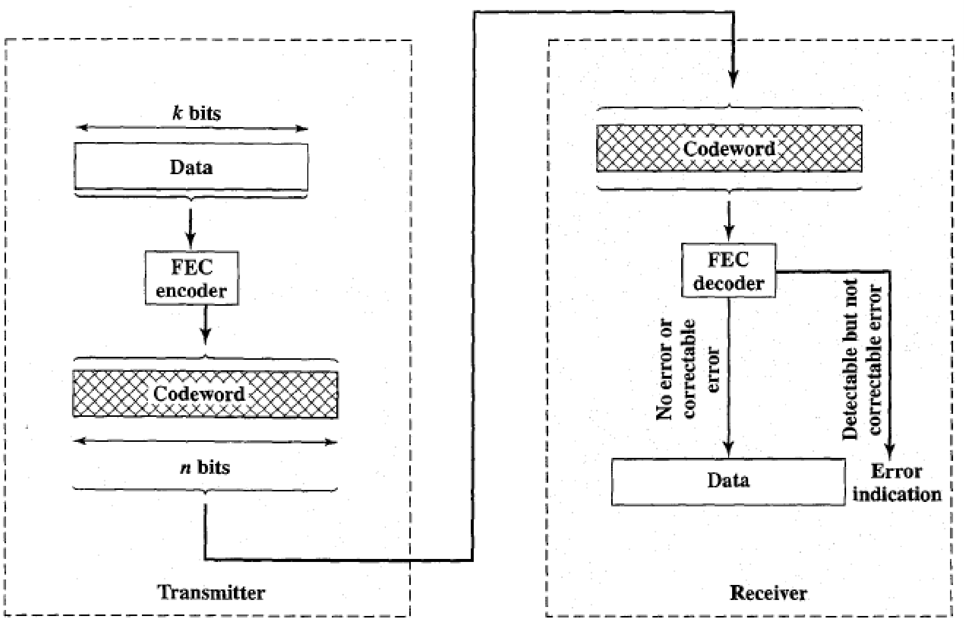

Figure 2.1: FEC Process Flow [26]

In a wireless communication channel, the superior error correcting codes are used in practice, such as Reed-Solomon (RS) codes, Low-Density Parity-Check (LDPC) codes, and turbo codes. All these are powerful and effective FEC codes that add the constant size of redundancy to correct random and burst errors generated due to noise and other factors without the need for requesting a retransmission [7].

2.3.1 Reed-Solomon (RS) Codes

The Reed-Solomon (RS) codes were announced in 1960 by Irving S. Reed and Gustave Solomon. Encoding data using RS codes is relatively straightforward, but decoding is time-consuming. RS codes regard as one with cyclic BCH codes, which proposes an efficient decoding algorithm. RS codes are suitable for medium sized message protection. RS codes are block-based error correcting codes based on the use of Galois field (GF) arithmetic. RS codes encode a group of bits instead of one bit at a time and adds extra redundant bits. Besides that, the RS codes decoder deals with each block and overtures to correct errors and recover the original data [27].

RS codes are typically written as $RS\ (n, k)$, where n is the codeword length, and k is the data length. RS encoder includes k data symbols added by $2t$ parity symbols to form an n symbol codeword. Moreover, RS decoder can correct up to t symbols that contain errors in the codeword by using the relationship as

$$t = \frac{(n-k)}{2} \quad \ldots(2.1)$$

Fig. 2.2 displays a structure of basic RS codeword as below [11, 28].

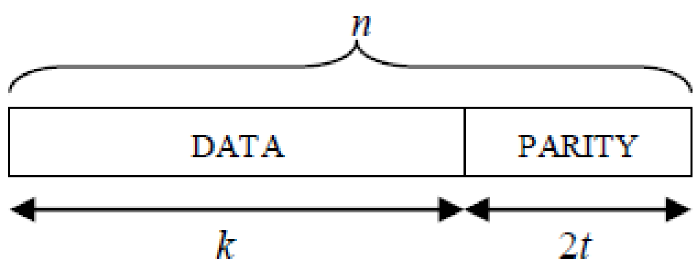

Figure 2.2: A typical RS codeword [28]

The codeword is created using a special polynomial by RS encoder, which can exactly divisible by the generator polynomial. The general equation of the generator polynomial $g(x)$ is given by

$$g(x) = (x + a^{i})(x + a^{i+1}) \ldots (x + a^{i+2t-2})(x + a^{i+2t-1}) \qquad \ldots(2.2)$$

The relation between resulting codeword and generator polynomial is defines as

$$c(x) = g(x).i(x) \qquad \ldots(2.3)$$

, where $c(x)$ is the generated codeword, $g(x)$ is the generator polynomial, and $i(x)$ corresponds to the information block [29].

The structure of an RS encoder as shown in Fig. 2.3 below is a $2t$ tap shift register, where the width of each register is n bits [30]. The objective of the encoding process is to find the remainder or parity symbols based on the message divided by the generator polynomial $g(x)$. Then these parity symbols are added to the end of data symbols to generate a codeword which can accurately divisible by $g(x)$. Thus, decoder divides the received message block with RS generator polynomial to determine the value of the remainder. If the value equal to zero, this means no errors are found, else it shows the presence of bit errors.

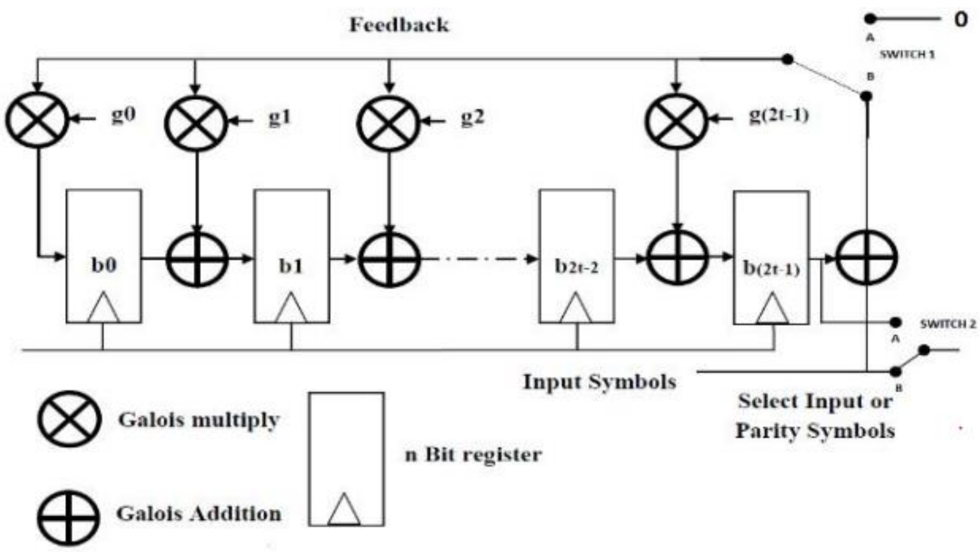

Figure 2.3: Block diagram of RS encoder [30]

The RS decoder computes the syndromes for each received codeword to determine the number of errors. If any bit errors are introduced in the message block, the decoder attempts to figure out the positions of the errors and correct it [4]. Fig. 2.4 given below is the block diagram of RS decoder. RS decoder attempts to determine the location and magnitude of up to t errors and to correct the errors. The received codeword $r(x)$ is the transmitted codeword $c(x)$ plus errors, as follows:

$$r(x) = c(x) + e(x) \quad \ldots(2.4)$$

Besides that, Si is syndromes, $L(x)$ is error locator polynomial, Xi is error locations, and Yi is error magnitudes.

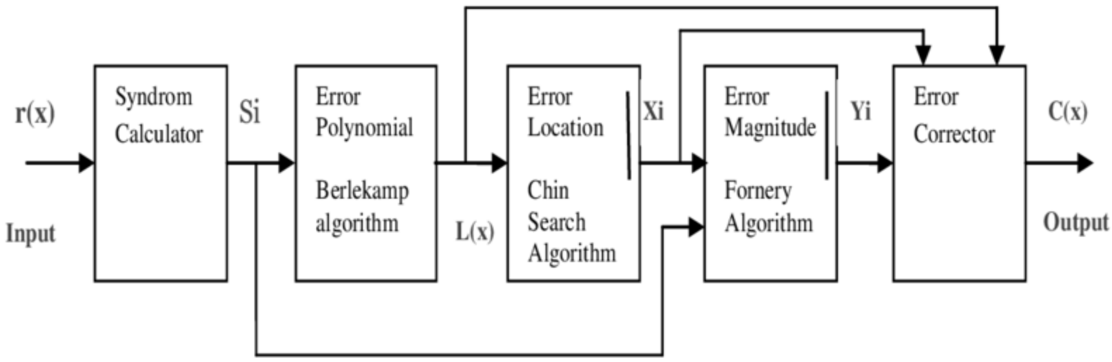

Figure 2.4: Block diagram of RS decoder [4]

To summary the decoding process for RS decoder, the steps involved show in below [31]:

a) Calculate syndrome
b) Generate error locator polynomial
c) Calculate roots of the polynomial
d) Determine symbol error values

RS codes are especially suitable to correct burst errors, where errors are determined in the received codeword for a series of bits [11].

2.3.2 Low-Density Parity-Check (LDPC) Codes

Low-Density Parity-Check (LDPC) codes are an error correcting code developed by Robort Gallager in 1960. But it is not popular for a few decades due to the trade-off between high implementation complexity and decoding performance. After reintroduced by D. MacKay and M. Neal in 1997, the popularity of LDPC codes gain as It can achieve a BER performance which is closer to the Shannon limit. LDPC codes can provide outstanding performance which is among the best channel coding for error control in systems which needs high reliability and efficiency[12]. On other hands, LDPC codes are achieved low complexity in the decoding technique compared with other correction codes [32, 33]. LDPC codes are widely used in many standards such as Wi-Fi/IEEE 802.11, WiMAX/IEEE 802.16e, DVB-S2, and etc.

LDPC codes classify as the class of block codes, it can represent the sparse parity check matrix (PCM) H, which consists of very few numbers of non-zero elements. The number of 1s in the H matrix is very less contrasted to the number of 0s. LDPC encoder encodes the useful information bits as codeword by adding redundant bits. The number of non-zero elements in each row and each column is called the degree of distribution. The matrix is introduced as regular matrix if the degree of distribution is the same, otherwise, it is an irregular matrix. Besides that, the H matrix can be expressed graphically by using Tanner graph. Tanner graph is a bipartite graph, which means the graph is separated into variable nodes and check nodes [34]. Fig. 2.5 is an example of parity check matrix and Fig. 2.6 is its corresponding Tanner graph representation [35].

$$
\begin{array}{c} \\ H = \end{array}
\begin{array}{c}
\begin{array}{ccccccccc} V_1 & V_2 & V_3 & V_4 & V_5 & V_6 & V_7 & V_8 & V_9 \end{array} \\
\left[\begin{array}{ccccccccc}
0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1
\end{array}\right]
\end{array}
\begin{array}{c} \\ C_1 \\ C_2 \\ C_3 \\ C_4 \\ C_5 \\ C_6 \end{array}
$$

Figure 2.5: An example of regular LDPC matrix [35]

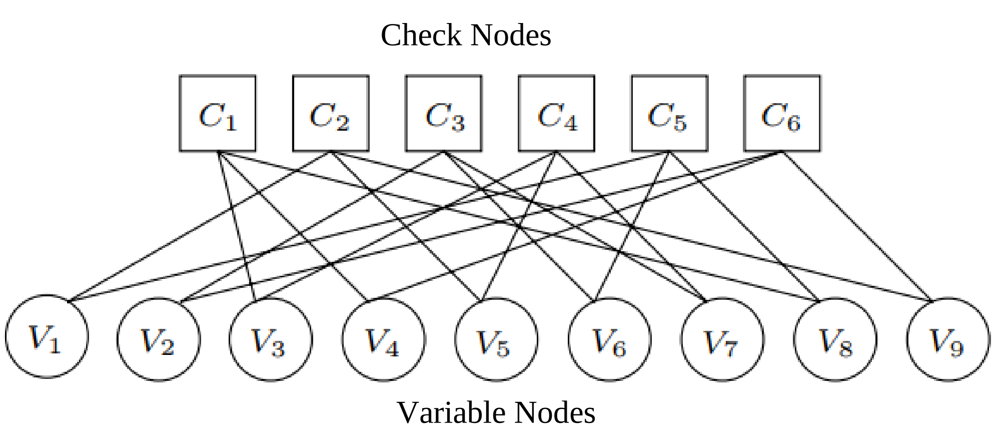

Figure 2.6: Tanner graph of LDPC matrix mentioned in Fig. 2.5 [35]

In LDPC encoder, the form of parity-check matrix H can be determined by performing Gauss-Jordan elimination in the form of

$$H = [A, I_{n-k}] \qquad \ldots(2.5)$$

Where, A is the $(n-k) \times k$ binary matrix, I_{n-k} is the size $(n-k)$ identity matrix, n is the number of output bits, and k is the number of data bits. Meanwhile, the form of generator matrix is

$$G = [I_k, A^k] \qquad \ldots(2.6)$$

Where, I_k is the identify matrix with size k and A^k is the binary matrix with size k. Codeword is created by modulo-2 addition using

$$C = mG \qquad \ldots(2.7)$$

Where, C is the codeword, m is the input message bits and G is the generator matrix [36].

For LDPC decoder, it is carried out through iterative processing based on the Tanner graph to fulfill the parity check conditions. Parity check condition is the condition that $CH^T = 0$. If CH^T equals to zero, then the received codeword is valid to prove that the received codeword is similar to the transmitted codeword. Iterative decoding has two variations, which is hard decision and soft decision decoding algorithms. Hard-decision is the decoder able to make the decision based on the received information if the value of a single bit can either be 0 or 1. While soft-decision is the decoder that able to distinguish between a set of quantized values between 0 and 1. These values provide the probability of a particular bit in a node [36].

One of the widely used soft-decision decoding algorithms in LDPC decoder is belief propagation (BP) algorithms, which analyses each decoding bit in reliability based on parity-cheek-sum. Since BP algorithms employ too much multiplication, so log-BP algorithms are introduced, which converts the multiplication into addition to reduce the complexity of decoding proceeding [37, 38].

2.3.3 Hybrid combination of RS and LDPC Codes

Hybrid code or product code is a promising technique for the next generation digital terrestrial broadcasting transmission system, due to its superior error correction performance. Hybrid code merges two different codes together, which can cover their strong features and eliminate their weakness. The concept of the product code is first introduced in 1954, which offers remarkable the performance of error correction, transmission reliability, and robustness against noise at higher data rates [12].

To achieve high reliability and efficiency, LDPC codes are proposed to Hybrid codes and RS codes are chosen as the other component [11, 12]. RS-LDPC codes is an interleaved serial concatenated code, thus it may have benefits in burst error cases. The concept of ‘inner coder-out coder’ is applied to combine both RS codes and LDPC codes together for encoding and decoding procedures of communication systems [12, 39]. The inner LDPC codes take full advantage of the channel’s soft information and have good performance in combating short random errors. The outer RS codes are effective against burst errors. Besides that, if the number of residue erroneous symbols are within the capability of outer RS codes, the inner LDPC codes iteration number can be terminated.

The schematic depiction of the Hybrid RS-LDPC codes is presented in Fig. 2.7. First, the message is first encoded by RS encoder. Galois field is used in RS encoder to generate the codeword. Then, the codeword is encoded second time by LDPC encoder. Then, the codeword is sent over a fading channel after modulation. After that, the codeword is decoded by using LDPC decoder using log-BP algorithms, then followed by RS decoder.

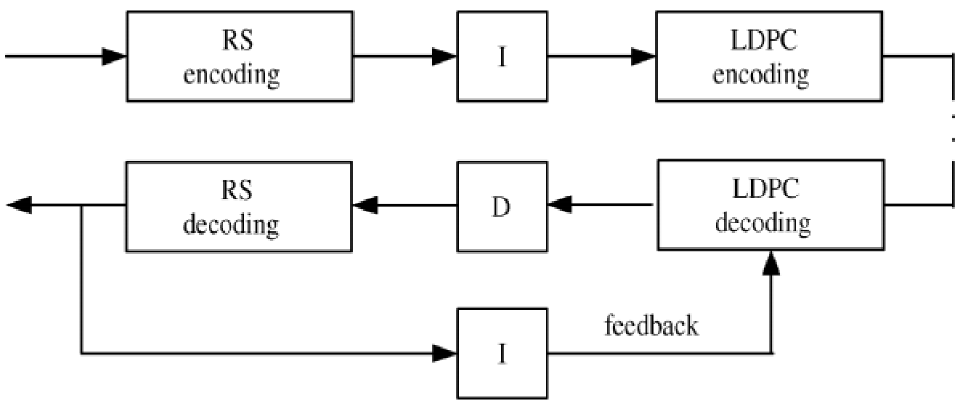

Figure 2.7: Schematic depiction of Hybrid code using the concept of 'inner coder-outer coder'[39]

2.4 Related Works

The work in [4] implements RS codes in error detection and correction for loT based home automation, in order to prevent any possible disaster before the user-invoked operation gets executed. The paper only displays the flowchart of the proposed scheme, but there is no result to prove the success of the proposed scheme. In [5], Cyclic Redundancy Check (CRC) codes are introduced to the energy constrained short-range IoT applications, such as Bluetooth low energy and IEEE 802.15.4 standards, which utilise data redundancy provided to detect erroneous packets. Since CRC codes are only used for error detection, therefore FEC codes are more suitable than CRC codes to apply in IoT applications. This is because FEC codes can correct the error at the receiver without retransmission to save transmission energy.

In the paper [40], Reed Solomon (255,239) codes are successfully implemented in the WiMAX network application for encoder and decoder by using VHDL codes. The incoming data stream which is obtained from the RS encoder is divided into blocks and by adding redundancy. Then, the operation is performed by the RS decoder and it corrects the errors based on the redundancy present in the received data.

The paper [12] discusses the performance of the LDPC-RS product codes that can provide superior error correction performance and robustness for a high SNR or tolerant to high BER for the systems. The paper highlights the problem of long delay due to long LDPC codes code length in Hybrid RS-LDPC codes structure. Two possible solutions that may deal with the problem are raised out, which are to fold the long LDPC codes or to use relatively short RS component codes.

In the paper [10], a new hybrid iterative decoding scheme and rate compatible RS-LDPC codes are proposed, which lowers the decoding threshold and reduces the decoding complexity for the next generation of digital terrestrial broadcasting systems. A new RS soft-decision decoder algorithms are needed to increase the accurate error estimation with lower complexity in RS-LDPC codes.

From the related works previously mentioned, it can be observed that the FEC codes are evaluated in various communication system but little to nothing is done in IoT based system. Therefore, in the thesis, the performance evaluation of Hybrid RS-LDPC codes in IoT based home automation system will be carried out using Python programming language in Raspberry Pi board.

2.5 Summary

In this chapter, the concept of IoT and smart home system are discussed. Raspberry Pi board is chosen as the central hub to build a home automation system. Besides, FEC codes are discussed. Different FEC codes have their own advantages and disadvantages. Based on the reviews, Hybrid combination of RS and LDPC codes are chosen, as it may raise the probability that the original message can be recovered from possibly corrupted data that is received. In next chapter, the procedures flow of proposed mechanism and the method to analyse its performance will be presented.

CHAPTER THREE

METHODOLOGY

3.1 Introduction

This chapter discusses the overall research method to accomplish the objectives of the research. In the research work, the goal is to analyse the performance of Hybrid RS-LDPC codes encoder and decoder that can improve the reliability and accuracy of the received data in IoT based home automation system. The entire research consists of five main stages, as shown in Fig. 3.1 below.

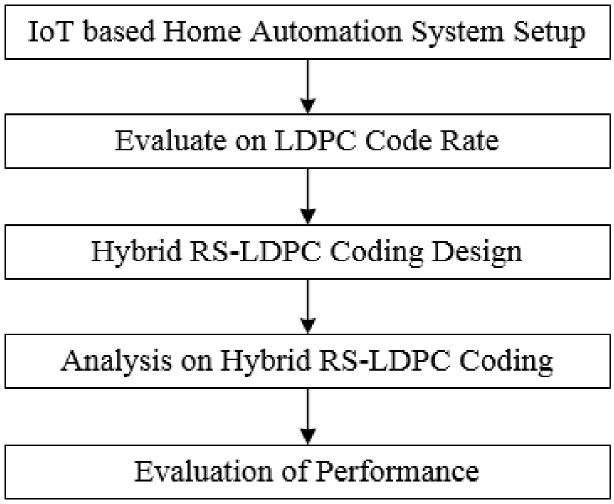

Figure 3.1: General Flowchart of This Research Work

Fig. 3.2 displays the block diagram of the communication system used in the thesis. The transmitter sends a message through a channel to a receiver. The channel is air when using a wireless network for IoT enabled communication systems. Heavy noise with less than or equal to 0.9 SNR may appear on the fading channel, so in order to receive the message with error-free as possible, channel coding is introduced in encoder and decoder. The encoder encodes the message with redundancy to form a codeword. While the decoder decodes the corrupted codeword to original message by using the redundancy. The Hybrid RS-LDPC codes are used in the project by applying concept ‘inner coder-outer coder’, which RS codes are outer code and LDPC codes are inner code, to provide superior error correction performance in IoT based home automation.

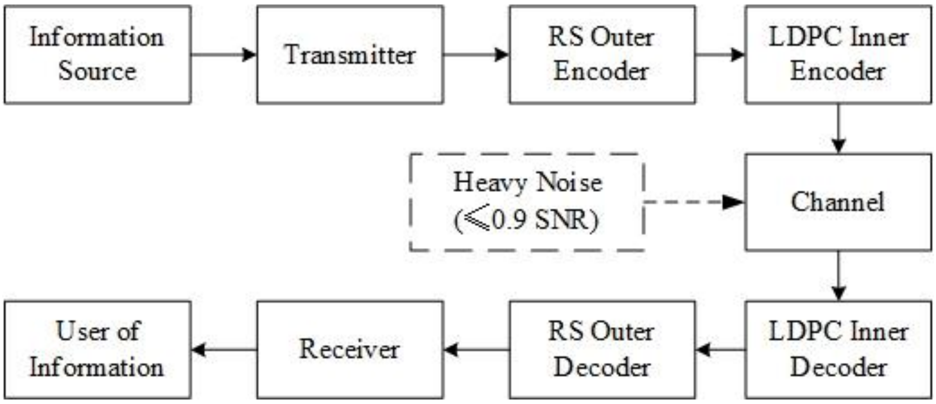

Figure 3.2: Block diagram of communication system

In addition, a detailed flowchart that can be used to introduce the entire procedures in channel encoder and decoder is demonstrated in Fig. 3.3.

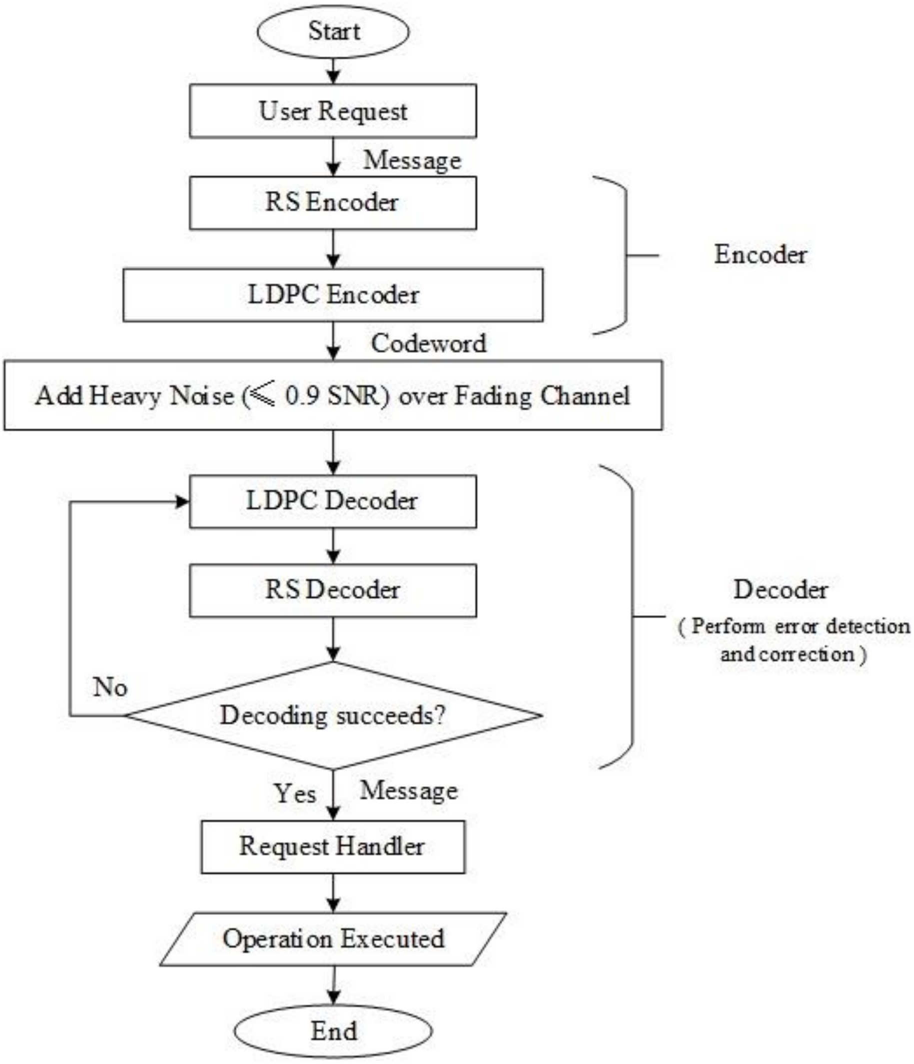

Figure 3.3: Flowchart for the entire procedures

First of all, the home automation system is implemented on the Raspberry Pi board, which accepts the user requests or messages generated from the web servers on computers, tablets or smartphones. In order to facilitate the analysis of decoding performance in the work, text messages up to 50 characters in length can be transmitted, which represents the user request.

At the beginning of the procedures, once the submit button is clicked on the web page, the text messages are sent. The generated messages are first passed to the RS encoder (outer coder) and then passed to the LDPC encoder (inner coder) to encode the messages into codeword. The codeword is then transmitted over the fading channel into the receiver. The codeword may get corrupted due to heavy noise with less than or equal to 0.9 SNR. The LDPC decoder (inner coder) and followed by the RS decoder (outer coder) at the receiver side decode the corrupted codeword to regenerate messages back.

The RS-LDPC decoder will detect and correct any such errors that might have been produced when passing through the fading channel. If the number of bit errors is higher than the error correction capability of the Hybrid RS-LDPC codes, then the correction for bit errors fails. If no bit errors detected, the message regenerated by the LDPC-RS decoder is forwarded to the request handler.

Finally, the user's operation requested will be performed in the terminal on Raspberry Pi board once received the instruction from user handler. If the received text message matches the text message sent by the front end, the success message is displayed; otherwise, the failure message is displayed.

3.2 Design and Performance Setting

In this research, the hardware design is done using Raspberry Pi board as a central hub to receive the text messages from web server and display it on the terminal. The program is written using Python programming language to apply the Hybrid RS-LDPC codes algorithms on error correction for corrupted received information. Besides that, the web page is developed using HTML programming language. The web page contains text box to allow user type text messages on it and a submit button to transmit text messages to Raspberry Pi board.

The decoding performance evaluation is done using Python Environment in Windows and Raspberry Pi board as the platform for verifying the Hybrid RS-LDPC codes algorithms Python program. The decoding performance analysis is based on the percentage of success decoding of a corrupted message into the original error-free message. In addition, the decoding process is simulated to record its complete decoding time used by running a total of 100 samples with different data lengths, code rates, and SNRs.

3.3 Error Correction Coding Design

The algorithms for RS codes and LDPC codes are developed separately using Python programming language. The code rate used for both RS codes and LDPC codes which can provide more better performance in decoding process is determined. After confirming both codes that can be successfully decoded to obtain error-free original data, the combination of this two codes is developed to generate the product codes.

3.3.1 Reed-Solomon (RS) Codes

RS codes are specified as $RS\ (n, k)$, where n is codeword symbols, and k is data symbols. The number of burst errors that can be corrected is t, which can be obtained using parity symbols, $2t = (n - k)$. In the program, $GF(2^8) = GF(256)$ is used, thus each symbol is 8 bits long. The code rate is defined as $\frac{k}{n}$. Due to previous research on performance evaluation on RS codes, it is found that $1/3$ code rate outperforms the other code rates. Thus, the code rate used in this program is fixed as $1/3$ and will be concatenated with LDPC codes to form Hybrid codes. Since text messages are acted as input data, hence the number of data symbols are different based on the length of characters that transmitted by the sender. Only up to 50 characters can be sent to the receiver. The evaluation is conducted by following the 5 categories with different number of data symbols. The parameters of each category data symbols are shown below:

a) The characters length is 10, thus $k = 10$ and $n = 10 \times \frac{3}{1} = 30$ by using this formula $code\ rate = \frac{k}{n} = \frac{1}{3}$. Therefore, it is represented as $RS\ (30{,}10)$ and it can correct around 10 burst errors where $2t = n - k = 30 - 10 = 20$ and $t = \frac{20}{2} = 10$.

b) The characters length is 20, thus $k = 20$ and $n = 20 \times \frac{3}{1} = 60$. Therefore, it is represented as RS (60,20) and it can correct around 20 burst errors where $2t = n - k = 60 - 20 = 40$ and $t = \frac{40}{2} = 20$.

c) The characters length is 30, thus $k = 30$ and $n = 30 \times \frac{3}{1} = 90$. Therefore, it is represented as RS (90,30) and it can correct around 30 burst errors where $2t = n - k = 90 - 30 = 60$ and $t = \frac{60}{2} = 30$.

d) The characters length is 40, thus $k = 40$ and $n = 40 \times \frac{3}{1} = 120$. Therefore, it is represented as RS (120,40) and it can correct around 40 burst errors where $2t = n - k = 120 - 40 = 80$ and $t = \frac{80}{2} = 40$.

e) The characters length is 50, thus $k = 50$ and $n = 50 \times \frac{3}{1} = 150$. Therefore, it is represented as RS (150,50) and it can correct around 50 burst errors where $2t = n - k = 150 - 50 = 100$ and $t = \frac{100}{2} = 50$.

For RS encoder, the transmitted k -bit information block is encoded to generate a n -bit codeword by multiplying the generator polynomial. When the codeword passes through the fading channel, errors are introduced inside the codeword to cause bits error happened. Then, the flow of decoding process is shown as Fig. 3.4 by using RS decoder. RS decoder performs an error detection up to t errors and the error correction accordingly. First, the error syndromes are calculated by substituting the $2t$ roots of generator polynomial into received codeword. Then, Berlekamp-Massey algorithms are used to generate an error locator polynomial. The roots of the polynomial are calculated by using Chein search algorithms to determine the positions of the errors. Next, the symbol error values of errors are determined by using Forney algorithms. Finally, the mask and codeword are compared and then sequentially inverting all erroneous bits.

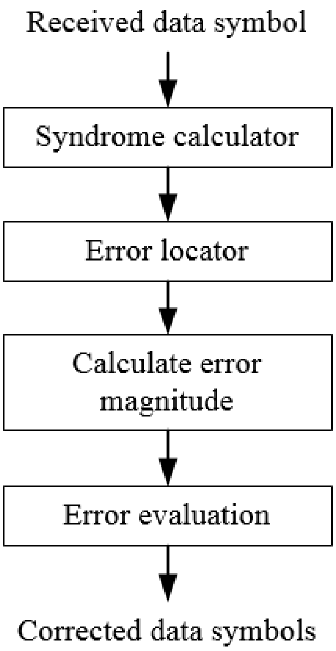

Figure 3.4: Flow chart of RS decoder

3.3.2 Low-Density Parity-Check (LDPC) Codes

The code rate for LDPC codes is equal to $\frac{k}{n}$, where n is number of coding bits, and k is number of information bits. The value of k and n are in term of bits, which is different compared with bytes used in RS codes. In order to build a regular Parity-Check Matrix H following Gallager's algorithms, the parameter of d_v and d_c need to be determined first, by using $\frac{k}{n} \geq 1 - \frac{d\_v}{d\_c}$, where n is also number of columns in parity-check matrix H, d_v is the number of ones per column (number of parity-check equations including a certain variable), and d_c is number of ones per row (number of variables participating in a certain parity-check equation).

The formula $HG^T = 0$ is used to obtain the coding matrix G using Gauss-Jordan algorithm on columns and rows of matrix in order to permute matrix change using matrix equivalence.

Since no previous research can provide a confident code rate that has the best BER performance, thus 5 types of common code rates will be analysed to determine their BER performance. The code rates analysed are $1/2$, $1/3$, $2/3$, $1/4$, and $3/4$. The code rate with the best BER performance based on the results will be concatenated with RS codes to form Hybrid codes. The characters length is fixed to 20 to obtain the decoding performance of the code rates. So, $k = 20 \times 8 \approx 160$. The parameters of each code rate are shown below:

a) The $code\ rate = \frac{k}{n} = \frac{1}{2}$, therefore $n = k \times \frac{2}{1} = 160 \times 2 = 320$. Based on $\frac{k}{n} \geq 1 - \frac{d\_v}{d\_c}$, $d\_v = 1$ and $d\_c = 2$.

b) The $code\ rate = \frac{k}{n} = \frac{1}{3}$, therefore $n = k \times \frac{3}{1} = 160 \times 3 = 480$. Based on $\frac{k}{n} \geq 1 - \frac{d\_v}{d\_c}$, $d\_v = 2$ and $d\_c = 3$.

c) The $code\ rate = \frac{k}{n} = \frac{2}{3}$, therefore $n = k \times \frac{3}{2} = 160 \times \frac{3}{2} = 240$. Based on $\frac{k}{n} \geq 1 - \frac{d\_v}{d\_c}$, $d\_v = 1$ and $d\_c = 3$.

d) The $code\ rate = \frac{k}{n} = \frac{1}{4}$, therefore $n = k \times \frac{4}{1} = 160 \times 4 = 640$. Based on $\frac{k}{n} \geq 1 - \frac{d\_v}{d\_c}$, $d\_v = 3$ and $d\_c = 4$.

e) The $code\ rate = \frac{k}{n} = \frac{3}{4}$, therefore $n = k \times \frac{4}{3} = 160 \times \frac{4}{3} = 216$. Based on $\frac{k}{n} \geq 1 - \frac{d\_v}{d\_c}$, $d\_v = 1$ and $d\_c = 4$.

For LDPC encoder, the parity check matrix (H) is first generated. Then, a generator matrix is derived. The value of n should be greater than k, where n is the codeword bits and k is the data bits. The generated H matrix should have least number of non-zero elements. H matrix is considered on the weight of the row where all the weight in the row should be alike. In addition, H matrix also focus on the weight of the column where all the weight in the column should be identical.

The algorithm used in LDPC decoder is Belief Propagation Decoding Algorithm with Log-domain. It is based on Tanner graph principle. Tanner graph is basically a two-part graph which contains the two divisions (variable nodes and check nodes). If the tanner graph is assumed to be cycle free then the message passing is ideal.

3.3.3 Hybrid RS-LDPC Codes

RS codes and LDPC codes are merged to form a Hybrid codes, which can cover their strong features and eliminate their weakness. The structure of Hybrid RS-LDPC codes is presented by using concept ‘ inner coder outer coder’. RS codes are chosen as outer codes due to its superior in burst errors. While LDPC codes are chosen as inner codes as it has a good performance in combating short random errors. Thus, Hybrid RS-LDPC codes are increased the overall error correction capability compared to the single code. RS codes with $1/3$ code rate are used in developing the error correction algorithms program, while the code rate used for LDPC codes can confirm after doing the decoding performance analysis.

The difficulty in combining RS codes with LDPC codes is the different scale used inside both codes. Information is stored in bytes for RS codes, while is in bits for LDPC codes. Hence, by wisely using the open-source add-on modules of Python can reduce the time taken to convert between bits and bytes. The useful modules in Python are NumPy and SciPy, which can provide common mathematical and numerical routines in pre-compiled, fast functions.

After trading off the decoding performance and processing time, $1/3$ code rate for LDPC codes are selected to be concatenated with $1/3$ code rate for RS codes to form Hybrid codes. The evaluation is conducted by following the 5 categories with different number of data symbols. Due to code rate for LDPC codes is $1/3$, the value of $d\_v = 2$ and $d\_c = 3$ based on $code\ rate\ \geq 1 - \frac{d\_v}{d\_c}$. The parameters of each category data symbols are shown below (the label '1' in both $k1$ and $n1$ stands for outer coder, while '2' in both $k2$ and $n2$ stands for inner coder):

a) The characters length is 10, thus $k1 = 10\ bytes$ and $n1 = 10 \times \frac{3}{1} = 30bytes$ by using the formula $code\ rate = \frac{k1}{n1} = \frac{1}{3}$ for outer coder (RS codes). In addition for inner coder (LDPC codes), $n1$ is equal to $k2$, but due to different scale used, $n1$ needs to be converted from $bytes$ to $bits$ by multiplying with 8 ($1bytes = 8bits$), therefore $k2 = 30bytes \times 8 \approx 241bits$. The $code\ rate = \frac{k2}{n2} = \frac{1}{3}$ for LDPC codes, therefore $n2 = k2 \times \frac{3}{1} = 240 \times 3 = 720bits$. Hence, it is represented as RS (30,10) for outer coder, and $k2 = 241$ and $n2 = 720$ for inner coder.
b) The characters length is 20, thus $k1 = 20$ and $n1 = 20 \times \frac{3}{1} = 60$ for RS codes. While, $k2 = 60 \times 8 \approx 481$ and $n2 = k2 \times \frac{3}{1} = 480 \times 3 = 1440$. Therefore, it is represented as RS (60,20) for outer coder, and $k2 = 481$ and $n2 = 1440$ for inner coder.

c) The characters length is 30, thus $k1 = 30$ and $n1 = 30 \times \frac{3}{1} = 90$ for RS codes. While, $k2 = 90 \times 8 \approx 721$ and $n2 = k2 \times \frac{3}{1} = 720 \times 3 = 2160$. Therefore, it is represented as RS (90,30) for outer coder, and $k2 = 721$ and $n2 = 2160$ for inner coder.

d) The characters length is 40, thus $k1 = 40$ and $n1 = 40 \times \frac{3}{1} = 120$ for RS codes. While, $k2 = 120 \times 8 \approx 961$ and $n2 = k2 \times \frac{3}{1} = 960 \times 3 = 2880$. Therefore, it is represented as RS $(120{,}40)$ for outer coder, and $k2 = 961$ and $n2 = 2880$ for inner coder.

e) The characters length is 50, thus $k1 = 50$ and $n1 = 50 \times \frac{3}{1} = 150$ for RS codes. While, $k2 = 150 \times 8 \approx 1201$ and $n2 = k2 \times \frac{3}{1} = 1200 \times 3 = 3600$. Therefore, it is represented as RS (150,50) for outer coder, and $k2 = 1201$ and $n2 = 3600$ for inner coder.

3.4 Analysis on Error Correction Coding Design

The decoding performance analysis is computed using Python environment on Windows. Windows is chosen instead of direct run the program for analysis in Raspberry Pi board. This is due to a large number of samples used to analyse decoding performance, the processor in Raspberry Pi board cannot support it and take very long process time. Firstly, the program for RS codes, LDPC codes and also Hybrid RS-LDPC codes are run in both version of Python environment to verify them can be successfully recover back the original sent message. Secondly, total 100 samples for each type of code rate are analysed to obtain the best code rate used for decoding performance in LDPC codes. The graph of decoding efficiency versus SNR from range 0 to 0.9 is plotted.

3.5 Hardware Design to implement Hybrid RS-LDPC Codes

The most common project in IoT based home automation is controlling AC appliances with the click of buttons on a webpage via the Internet. But the information in the user request command is too simple, which cause difficulties to analyse the performance of error correction algorithms on home automation. Hence, the web controlled display is