

การรับส่งสัญญาณ โดยใช้ชีวิธี Frequency-shift keying ในย่านความถี่หลายชั้นที่ไม่ทับซ้อนกัน

วรุฒ มาศสุวรรณ

ภาควิชาศึกษาศาสตร์ สาขาวิชคอมพิวเตอร์ มหาวิทยาลัยเกษตรศาสตร์

เลขที่ 50 ถนนรามคำแหง แขวงลาดยาว เขตจตุจักร กรุงเทพฯ 10900

บทคัดย่อ

โครงการเรื่องการรับส่งสัญญาณโดยใช้ชีวิธี Frequency-shift keying ในย่านความถี่หลายชั้นที่ไม่ทับซ้อนกันมีวัตถุประสงค์เพื่อศึกษาวิธีการรับส่งสัญญาณจากข้อมูล Digital เป็นสัญญาณ Analog โดยใช้การซ้อนทับของการส่งข้อมูลแบบFSK ในย่านความถี่ที่ไม่ทับซ้อนกันเพื่อเพิ่มปริมาณของข้อมูลที่ส่งไปในหนึ่งหน่วยเวลาให้มากขึ้นโดยในขั้นตอนการทำงานจะมีการสร้างสัญญาณโดยเริ่มจากชั้นที่มีความถี่ต่ำที่สุดก่อนและค่อยเพิ่มชั้นของสัญญาณที่จะใช้ในการซ้อนทับโดยจะมีพารามิเตอร์ $p[i]$ ไว้กำหนดระยะเวลาของคุณภาพสี่ในแต่ละชั้นหลังจากการสร้างสัญญาณแล้ว จะมีการนำสัญญาณที่ได้มาทำการถอดรหัสให้เป็นข้อมูล Binary โดยการเปรียบเทียบค่า magnitude ที่ได้จากการคำนวณใน Goertzel algorithm จากนั้นจะนำข้อมูลที่ได้รับมาหาค่า Bit Error Rate ในแต่ละชั้นของสัญญาณ และในค่า $p[i]$ ที่ต่างกัน รวมไปถึงทดสอบสัญญาณกับ SNR_db โดยใช้ Additive White Gaussian Noise (AWGN) และคำนวนหา modulation efficiency (bit/sec/Hz) ของวิธีการส่งสัญญาณในรูปแบบนี้

คำสำคัญ: Goertzel algorithm, DFT, AWGN, Bit Error Rate, Frequency shift keying

Abstract

In this research, about data Transmission using Frequency-shift keying in multi-frequencies band with non overlap aims to study about the method of data transmission from digital data to analog signal by overlaying FSK data transmission with non overlap frequency band in each layer for increasing a mount of data transmitted in time. In the working process, the signal will be created start of lowest frequency layer and gradually adding the signal layer to be used in the overlay with parameter $p[i]$ for determine distance between each frequency pair in layer. After signal was created, the obtain signal will decode to be binary data by comparing the magnitude value obtained from Goertzel algorithm then the obtained binary data will be used to find the Bit Rate Error in each signal layer and with different $p[i]$ including test the signal with Additive White Gaussian Noise (AWGN) in various SNR_db and find modulation efficiency (bit/sec/Hz) of this data transmission method.

Keywords: Goertzel algorithm, DFT, AWGN, Bit Error Rate, Frequency shift keying

1. บทนำ

1.1 ที่มาและความสำคัญ

ในยุคปัจจุบันนี้การสื่อสาร และ การส่งข้อมูลเป็นสิ่งจำเป็นมากสำหรับเราทุกคนไม่ว่าจะเป็นการสื่อสาร ผ่านทางโทรศัพท์ หรือการส่งข้อมูลผ่านอินเทอร์เน็ต ดังนั้นการพัฒนาวิธีการส่งข้อมูลอย่างมีประสิทธิภาพจึงเป็นสิ่งที่มีการศึกษาเกี่ยวกับมาอย่างยาวนานและหนึ่งในสิ่งที่เป็นปัจจัยสำคัญในการส่งข้อมูลคือความเร็วในการส่งข้อมูลที่ปัจจุบันมีการขยายตัวของขนาดข้อมูลที่มีการส่งมากขึ้นเรื่อยๆ

โดยหนึ่งในวิธีในการส่งข้อมูลที่ใช้ในปัจจุบันคือ Frequency-shift keying (FSK) ซึ่งเป็นวิธีการในการส่งข้อมูล Digital โดยใช้สัญญาณ Discrete โดยข้อมูล binary สองสถานะคือ '0' (ต่ำ) และ '1' (สูง) ซึ่งแต่ละค่าจะแสดงในรูปแบบของ Analog waveform และทั้งสองค่าแสดงโดยใช้ความถี่ที่ต่างกัน [1] (แม้กระนั้นในการส่งสัญญาณในลักษณะนี้ก็อาจจะประสบปัญหาในด้านความเร็วสำหรับการส่งข้อมูลจำนวนมากในระยะเวลา สั้นๆ ถ้าให้ช่วงเวลาของแต่ละ bit มีช่วงที่ยาว

ด้วยเหตุนี้เองผู้จัดทำจึงมีแนวคิดในการศึกษาและพัฒนาประสิทธิภาพของการส่งสัญญาณโดยใช้วิธี FSK โดยการเพิ่มชั้นของช่วงสัญญาณใหม่จำนวนที่มากขึ้น และรวมสัญญาณกันเป็นสัญญาณเดียวเพื่อเพิ่มจำนวนของข้อมูลที่ส่งไปในหนึ่งสัญญาณ ให้มากขึ้น

1.2 วัตถุประสงค์หลักของโครงการ

1. เพื่อศึกษาวิธีการในปรับปรุงประสิทธิภาพในด้านความเร็วในการส่งข้อมูล สำหรับการส่งข้อมูล แบบ FSK

2. เพื่อศึกษาการแปลงข้อมูลจากสัญญาณ Analog จากการสร้างสัญญาณด้วย วิธี FSK เป็นข้อมูล Binary

1.3 ขอบเขตของโครงการ

สัญญาณนำเข้า คือ ชุดสัญญาณผสมที่สร้างขึ้น โดยใช้วิธีการ FSK ผ่าน library numpy ในโปรแกรม python

ชนิดของสัญญาณ คือ สัญญาณสมรรถะห่วงสัญญาณ Sinusoidal ในหลายๆ ยานความถี่ที่ไม่ทับซ้อนกัน

การวิเคราะห์สัญญาณ คือ การใช้ Goertzel Algorithm สำหรับ เพื่อหาความถี่เฉพาะ สำหรับแปลงสัญญาณ เป็นข้อมูล Binary

2. วิธีที่นำเสนอ

2.1 แนวทาง และทฤษฎีต่าง

การแปลงฟูเรียร์แบบบันท่อเนื่อง (Discrete Fourier Transform—DFT) เป็นการแปลงที่ทำกับสัญญาณที่ไม่ต่อเนื่อง และมีขนาดที่จำกัดซึ่งเทียบเท่ากับการทำการทำการแปลงฟูเรียร์แบบบันท่อเนื่อง(CFT)สำหรับข้อมูลNช่วงโดยมีช่วงเวลาระหว่างตัวอย่างเป็น T_s (กล่าวคือสำหรับลำดับข้อมูลที่มีขนาดจำกัด)แม้ว่าผลตอบสนองของการทำ CFT จะเพียงพอ แต่ในการทำงานจริงสัญญาณที่ใช้ในการแปลงมักเป็นสัญญาณที่ไม่ต่อเนื่องผ่านการทำ sampling สัญญาณด้วย ADCs ผ่านทาง sensors นอกจากนี้ CFT จะทำกับสัญญาณที่มีระยะเวลา ตั้งแต่ $-\infty$ ถึง ∞ แต่ DFT จะคำนวณในช่วงที่จำกัดโดยการใช้ window สำหรับลูกคื่นที่เป็น periodic [2]

ในการวิเคราะห์ทางคณิตศาสตร์ ข้อมูลมีจำนวนจำกัด ภายใต้ window DFT จะเห็นชุดสัญญาณเป็น Periodic signal ซึ่งเราสามารถแสดง DFT ในรูปของสัมการณ์ ได้ดังนี้

$$F(k) = C \cdot \sum_{n=0}^{N-1} f(n) \cdot e^{-j \frac{2\pi}{N} kn} \quad (1)$$

$f(n)$ คือ สัญญาณไม่ต่อเนื่อง

$F(k)$ คือ เฟสเซอร์ของาร์มอนิกที่ k

N คือ จำนวน sampling ต่อ cycle.

k คือ ค่า Harmonic

C คือ Scaling factor:

Goertzel Algorithm เป็น algorithm ที่ใช้ในการคำนวณหา DFT coefficient และ spectra ของสัญญาณ โดยใช้ digital filtering method. โดยการใช้ Goertzel Algorithm ทำให้เราสามารถ คำนวณหา spectra ของสัญญาณได้โดยไม่ต้อง ใช้ complex algebra แบบ DFT หรืออาจจะกล่าวได้ว่า Goertzel Algorithm เป็น filtering method สำหรับการคำนวณหา DFT coefficient $X(k)$ ของความถี่ในช่วงที่ k สำหรับ Digital data $x(0), x(1), \dots, x(N-1)$ [3]

โดยความสามารถแสดง Goertzel Algorithm ได้โดยการใช้ the second-order IIR digital Goertzel filter, โดยมี transfer function ดังนี้

$$H_R(z) = \frac{Y_k(z)}{X(z)} = \frac{1 - W_N^{kz^{-1}}}{1 - 2\cos\left(\frac{2\pi k}{N}\right)z^{-1} + z^{-2}} \quad (2)$$

k คือ Frequency index ที่สนใจ

N คือ จำนวน sampling ต่อ cycle

$$W_N^k \text{ คือ } e^{-j\frac{2\pi k}{N}}$$

Spectrum Efficiency แนวคิดเรื่องประสิทธิภาพของ

スペクトรัม (spectral efficiency) [4] มีความสำคัญในระบบวิทยุดิจิทัลซึ่งมีรากฐานมาจาก Shannon's theorem ที่แสดงถึงความสามารถในการรับส่งข้อมูลในช่องสัญญาณหนึ่ง ดังสมการ

$$\bar{C} = B_c \log_2(1 + SNR) \quad (3)$$

\bar{C} คือ ขนาดของ bit ต่อ วินาที

B_c คือ ความกว้างของช่องสัญญาณ (Hz)

SNR คือ อัตราส่วน สัญญาณ กับ เสียงรบกวน

ในส่วนของ ประสิทธิภาพในด้านปริมาณในการส่งข้อมูล จากการ modulate สัญญาณ เรียกว่า modulation efficiency เราสามารถวัดประสิทธิภาพได้จากการ ดังนี้

$$\eta_c = \frac{R_c}{B_c} \quad (4)$$

R_c คือ ความเร็วในการส่งบิต ในช่องสัญญาณ (bit/s)

η_c คือ ประสิทธิภาพในการส่งสัญญาณ ($\text{bit s}^{-1} \text{ Hz}^{-1}$)

2.2 ขั้นตอนการดำเนินการ

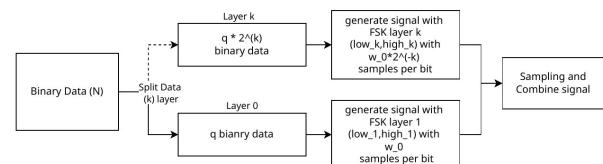
2.2.1 กระบวนการในการสร้างและถอดรหัสสัญญาณ

ในขั้นตอนการสร้างสัญญาณเราจะทำโดยการรับข้อมูลนำเข้า (binary data $b_0 b_1 b_2 b_3 \dots b_n$) ที่ต้องการจะแปลงเป็นสัญญาณมาก่อน จากนั้นนำ ข้อมูล ที่ได้มาแยก เป็นชั้นๆ (l_0, l_1, \dots, l_k) โดยให้ชั้นที่ 0 จะเก็บ bit ออยู่ q ตัว จากบิตที่ 0 ถึง $q-1$ และในชั้นถัดจะเป็น บิตที่ q ถึง $3q-1$ ซึ่งจะมีการแบ่งโดยให้จำนวนของ บิตในชั้นถัดไปเป็น 2 เท่าของชั้นก่อนหน้า จนถึงชั้นที่ k ซึ่งจะเก็บข้อมูล เป็นจำนวน $q * 2^k$ บิต เป็นต้น โดยจำนวนชั้นของ บิต จะมีจำนวนอยู่ $\lceil \log_2 n \rceil$ ชั้น

เมื่อแยกข้อมูลเสร็จแล้วเราจะมีการนำข้อมูลมาแปลงเป็นสัญญาณ analog ให้มีลักษณะคล้ายกับการทำ FSK โดย ให้ชั้นที่

i มี $f_h[i]$ และ $f_l[i](f_h$ คือ ความถี่ที่ใช้ในบิต '1' และ f_l คือ ความถี่ที่ใช้ในบิต '0') โดยที่ จะมีระยะห่าง ระหว่าง $f_h[i]$ กับ $f_l[i+1]$ เป็น $2 * p[i]$ Hz เมื่อ $p[i]$ ผลต่างความถี่ของ ชั้นสัญญาณที่ i ($f_h[i] - f_l[i]$)

นอกจากนี้ ชั้นที่ i จะมีจำนวนแซมเพลิ สำหรับ 1 บิตอยู่ที่ $w_0/2^i$ จุด เมื่อ w_0 เป็นความกว้างเริ่มต้น ของ bit ในชั้นแรก หลังจากนั้น เราจะ sampling ชั้นสัญญาณ Cosine ของทุกชั้น ของข้อมูล โดยใช้ ความถี่ในการ sampling เป็น $2 * f_h[K]$ และทำการผสานสัญญาณ ออกมาเป็นหนึ่งชุดสัญญาณ



รูปที่ 1 การในการสร้างสัญญาณ ด้วย FSK ในหลายชั้นสัญญาณ

2.2.1 ขั้นตอนในการถอดรหัสสัญญาณ

ในส่วนของการถอดรหัสสัญญาณเป็น Binary number สำหรับการทดลองในครั้งนี้ เราให้ค่าของช่วงความถี่ของสัญญาณ แต่ละชั้นมาก่อนแล้ว รวมถึงข้อมูล เช่น ระยะเวลาของสัญญาณ หรือระยะเวลาใน 1 บิต ของสัญญาณในชั้นแรก เป็นต้น จากนั้นเราจะมาคำนวณหาจำนวน samples ที่ใช้ในแต่ละบิต ของแต่ละชั้นสัญญาณ โดยมาจาก

$$W(i) = \left\lceil \frac{t_B \cdot f_S}{2^i} \right\rceil \quad (5)$$

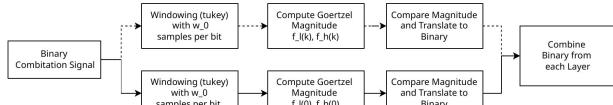
t_B คือ เวลาที่ใช้สำหรับ 1 บิตในชั้นแรก

f_S คือ sampling frequency

จากนั้นเราจะเลือก samples สำหรับ 1 บิตจากสัญญาณของแต่ละชั้น i ให้ จำนวน $W(i)$ จุด นำไปใช้ turkey window (alpha = 0.5) และมาเข้าใน Goertzel Algorithm โดยให้ใช้ $f_h[i]$ ต่อด้วย $f_l[i]$ สำหรับการคำนวณสำหรับหา Magnitude (M) (แสดงถึงความเหมือนของสัญญาณ ในชุดข้อมูลกับความถี่ที่ใช้ คำนวณ) จากนั้น นำขนาดที่ได้มาเปรียบเทียบ ระหว่าง M_h จาก $f_h[i]$ กับ M_l จาก $f_l[i]$ โดยถ้า $M_l < M_h$ แสดงถึงการประกูลของความถี่ $f_h[i]$ สูง จะแสดงเป็นข้อมูล binary "1" และ "0" สำหรับ $M_l > M_h$ อย่างไรก็ตามในชั้นสุดท้ายของสัญญาณเป็นไปได้ว่าไม่มีข้อมูลเต็มทั้งหมดสัญญาณหรือขนาด M จะจะใกล้เคียงกัน ทำให้เรามีการใช้ threshold ออยู่ที่ $\text{Max}(M_h, M_l)/\text{Min}(M_h, M_l)$

< 1.2 และแสดงค่าเป็น “2” เพื่อง่ายในการทดสอบและหาจุดที่ผิดพลาด สำหรับงานในครั้งนี้

เมื่อเราทดสอบรหัสจากสัญญาณทั้งหมดแล้วเราจะนำข้อมูลที่ได้มาต่อ กันโดยเริ่มจากข้อมูลในชั้นแรกสุดก่อน จนครบทุกชั้น



รูปที่ 2 การในการสร้างสัญญาณด้วย FSK ในหลายชั้นสัญญาณ

2.2.1 วิธีการประเมินประสิทธิภาพในการส่งสัญญาณ

เราจะประเมินประสิทธิภาพของสัญญาณ โดยการหา Bit Error Rate [5] และ NULL Bit Error Rate (ค่า BER ของ bit ที่ไม่ได้ส่งแต่ตรวจสอบ) ของวิธีการส่งข้อมูลในรูปแบบนี้ เมื่อทดสอบร่วมกับ AWGN [6] ในหลายช่วงของ SNR_{db}

และต่อมาเราจะมาคำนวณ modulation efficiency โดยใน 1 วินาที เราจะสามารถส่งข้อมูลได้จำนวน $q^k (2^k - 1)$ bit (R_C) สำหรับสัญญาณที่มี k ชั้น และชั้นแรกส่งข้อมูล q บิตในหนึ่งวินาที สำหรับชั้นแรกที่ส่งข้อมูลจำนวน 1 บิต และเราใช้ ของความถี่ของสัญญาณ (B_C) เป็น $2 * p[0] + 2 * p[0] * 2^1 + \dots + 2 * p[0] * 2^{k-1}$ ได้ $2 * p[0] (2^k - 1)$ จะได้

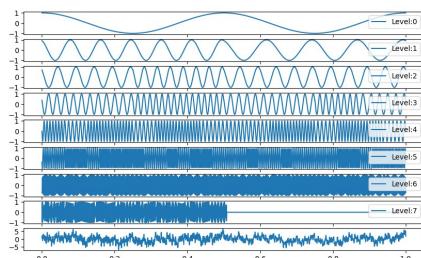
$$\eta_C = \frac{q \cdot (2^k - 1)}{2 \cdot p[0] \cdot (2^k - 1)} = \frac{q}{2 \cdot p[0]} \quad (6)$$

3. ผลการทดลอง (Experimental Results)

3.1 การทดสอบโดยไม่เพิ่มสัญญาณรบกวนใน

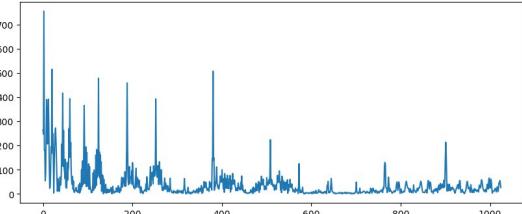
3.1.1 การนำเข้าข้อมูลและการแสดงผล

ในการทดสอบสำหรับการสร้างสัญญาณโดยการสร้างข้อมูล Binary จาก string ในจาก "Hello, How Are You Doing?" ที่มี บิตจำนวน 192 ตัว โดยกำหนดให้มี $p[0] = 2, 4, 8, 16, \dots, 512$ Hz โดยให้ชั้นแรกสุดเก็บข้อมูลจำนวน 1 บิต ในระยะเวลา 1 วินาที เราจะได้สัญญาณผสมดังรูป สำหรับ $p[0] = 2$ Hz



รูปที่ 3 สัญญาณ "Hello, How Are You Doing?" ($p[0] = 2$ Hz)

สัญญาณที่สร้างขึ้นนั้นจะมี การผสมทั้งหมด 8 ชั้นสัญญาณจากการทำ FSK โดยมีความถี่อยู่ในช่วง 2 ถึง 1020 Hz ที่แสดงด้านล่างในรูปที่ 4



รูปที่ 4 FFT coefficient จากสัญญาณตัวอย่าง ($p[0] = 2$ Hz) ซึ่งจะใช้ความถี่ในการ sampling เป็น 2048 Hz จากนั้นเราจะเห็นได้ว่าจากนั้นเราจะนำสัญญาณที่ได้มาถอดรหัสเป็น binary ได้ดังนี้

0100100001100101011011000110110001101111
001011000100100001101111011101110010...

รูปที่ 5 ผลการถอดรหัสจากสัญญาณตัวอย่าง ($p[0] = 2$ Hz)

3.1.2 การวัดประสิทธิภาพ

ในส่วนของการสัดประสิทธิภาพเราทดสอบการถอดรหัสสัญญาณในหลายช่วง F_s และ $p[0]$ ดังตาราง

ตารางที่ 1 ค่า BER ของสัญญาณในหลาย $p[0]$, F_s

$P[0] \setminus F_s$	1024	2048	4096	8192	16384	32768	65536
1	40.62	25	20.31	19.79	19.27	19.27	18.75
2		0.52	0	0	0	0	0
4			0	0	0	0	0
8				5.73	0	0	0
16					0	0	0

จากการทดสอบพบว่า สำหรับ การเพิ่ม sampling frequency สำหรับ $p[i]$ คงที่ พบร่วมกับ $p[0]$ เป็น 1 Hz ค่า BER อยู่ในช่วง 20 - 40% และมีแนวโน้มที่ลดลง ประมาณ 11.46% ในขณะที่ $p[0]$ ที่ค่าอื่น พบร่วมกับ $p[0]$ เป็น 2, 8 Hz ซึ่งจากข้อมูลดังกล่าวเราสามารถ อธิบายได้ว่า F_s มีอิทธิพลต่อ BER สำหรับ $p[0]$ ที่น้อย หรือคุ้มค่าอยู่ใกล้กันมาก และสำหรับการส่งข้อมูลให้มี BER ที่น้อย ควรให้มี $p[0]$ อย่างน้อย เป็น 2 Hz เป็นต้นไป

3.2 การทดสอบที่มีการเพิ่มสัญญาณรบกวน

3.2.1 การวัดประสิทธิภาพ

จากชุดสัญญาณในขั้น 3.1.1 เราได้นำสัญญาณมาผสานกับ AWGN โดยให้มีระดับของ SNR_db อยู่ที่ 0.125, 0.5, 2, 8 ตามลำดับร่วมกับ Fs ที่มีค่าต่างกัน สำหรับ $p[0]$ เป็น 2 Hz

ตารางที่ 2 ค่า BER(%) ของสัญญาณในหลายๆ ค่า Fs ,SNR_db

SNR_db\Fs	2048	4096	8192	16384	32768
0.125	22.4	18.75	7.29	3.65	0
0.5	29.17	16.67	8.85	0.52	0
2	26.04	11.46	2.08	0.521	0
8	7.29	1.04	0	0	0

จากการทดสอบในตารางแสดงให้เห็นว่า เมื่อ SNR_db มีค่าที่น้อย จะทำให้ ค่า BER มีค่ามากขึ้น ขณะเดียวกับความถี่ในการ sampling มีมากขึ้น ทำให้ BER มีลดลง โดย จะมี BER ลดลง ประมาณ 25 % เมื่อ เพิ่ม Fs จาก 2048 เป็น 32768 Hz ในช่วง SNR_db เป็น 0.125 - 2 และ 7.29% เมื่อ SNR_db เป็น 8

โดยความสามารถอธิบายได้ว่าการเพิ่มขึ้นของความถี่ ในการ sampling มีผลในการเพิ่มความถูกต้องของการส่งข้อมูลได้อย่างมีนัยยะสำคัญสำหรับสัญญาณในสภาพแวดล้อมที่มี เสียงรบกวน รุนแรง

3.3 การคำนวณ modulation efficiency

ตารางที่ 3 modulation efficiency(bit/sec/Hz) ในหลายๆ $p[0]$

p[0]	2	8	32	128	512
η	0.25	0.0625	0.015625	0.00390625	0.000976563

จากคำนวณในสัมการ ที่ 6 พบว่า modulation efficiency (bit/sec/Hz) มีค่าประมาณ ≤ 0.25 bit/sec/Hz ซึ่งแสดงถึง ประสิทธิภาพต่อ bandwidth ที่จะลดลงเมื่อเพิ่ม $p[i]$ ให้มากขึ้น

4. สรุป

การเพิ่มประสิทธิภาพในการส่งข้อมูลโดยใช้ FSK สามารถเพิ่ม ปริมาณของบิตที่ใช้ในการส่งข้อมูลได้จริงและสามารถลดรั้งสัญญาณออกมากได้ โดยมีค่า BER อยู่ที่ 20 -40% สำหรับ $p[0]$ เป็น 1Hz และ $< 6\%$ ในสำหรับ $p[0] \geq 2\text{Hz}$ ในกรณีที่สัญญาณไม่ได้รับเสียงรบกวน ในขณะเดียวกันเมื่อทดสอบกับ AWGN พบว่าค่า BER จะเปลี่ยนต่างๆ และแปรผันกับ SNR_db อย่างมีนัยยะสำคัญ โดย สำหรับในช่วง SNR_db ที่มี 0.125, 0.5, 2, 8

และมีความถี่ sampling ระหว่าง 2048 - 32768 Hz สำหรับ $p[0]=2\text{Hz}$ จะมี BER ช่วง $< 22.4\%$, $< 29.17\%$, $< 26.04\%$ และ $< 7.29\%$ ตามลำดับในส่วนของประสิทธิภาพ ของการใช้ bandwidth หรือ modulation efficiency พบว่าจะมีประสิทธิภาพจะขึ้นอยู่กับ $p[i]$ โดยที่ ในกรณี ที่ $p[0] \geq 2\text{Hz}$ จะมี ค่าประมาณ ≤ 0.25 bit/sec/Hz สำหรับ $p[0] \geq 2\text{Hz}$

5. กิตติกรรมประกาศ

โครงการฉบับนี้สำเร็จลุล่วงได้ด้วยดีเนื่องจากได้รับความอนุเคราะห์จากบุคคลหลายฝ่ายทั้ง ดร.กิตติพล ไทรเทพ, คณะอาจารย์ในภาควิชาวิศวกรรมศาสตร์ สาขาวิชาคอมพิวเตอร์ มหาลัยเกษตรศาสตร์ รวมไปถึงคณะผู้บริหารที่สนับสนุนในการทำโครงการในครั้งนี้ ผู้จัดทำขอขอบคุณบุคคลและสถาบัน ทั้งกล่าวอย่างยิ่งตลอดจนบุคคลอื่นๆ ผู้จัดทำหวังว่ารายงานฉบับนี้ จะเป็นประโยชน์กับผู้ที่สนใจต่อไป

6. เอกสารอ้างอิง

[1] Andrew Froehlich. (7 March 2025).

frequency-shift keying. [Online] Available :

<https://www.techtarget.com/>

[2] Jorge García. Encyclopedia of Electrical and Electronic Power Engineering Reference Work.

Processes and Information Techno-Logy. University of Oviedo, Gijon, Spain. 2023

[3] Li Tan and Jean Jiang. Digital Signal Processing Fundamentals and Applications Book. Purdue University North Central. 2013

[4] Michael Steer. (8 March 2025).

Spectrum Efficiency. [Online] Available :

<https://eng.libretexts.org/>

[5] David Large, James Farmer. Modern Cable Television Technology, Second Edition. 2004.