บทที่ 4

การวิเคราะห์และแสดงผลสัญญาณเสียงเต้นของหัวใจ

4.1 การใช้คณิตศาสตร์เพื่อการวิเคราะห์และแยกแต่ละครั้งของการเต้นของหัวใจ

เพื่อให้สามารถคำนวณค่าในเชิงปริมาณของสัญญาณที่จับได้ จึงต้องนำสัญญาณเข้าเก็บในเครื่องคอมพิวเตอร์ ในการทำวิจัยครั้งนี้ได้ส่งสัญญาณที่ออกจากด้านออกของวงจรขยายเข้าทางอินพุตไมโครโฟนของระบบคอมพิวเตอร์ (sound card) ซึ่งมี A/D ปรับอัตราการชักตัวอย่างและ bit ได้หลายค่าโดยในการบันทึกใช้อัตราการชักตัวอย่างที่ 22,050 Hz 16 bit แล้วเรียกใช้โปรแกรมบันทึกเสียงเก็บสัญญาณลงบนฮาร์ดดิสค์ในรูป .wav ไฟล์ (อาจเก็บใน format อื่น ๆ อีกได้หลายแบบ) ทั้งยังสามารถนำไฟล์ที่บันทึกได้นี้ออกมาเล่นกลับเพื่อฟังเสียงได้อีกด้วยซึ่งจะสามารถอ่านออกเพื่อใช้แสดงผลบนหน้าจอคอมพิวเตอร์และการคำนวณต่อไปได้

รูปคลื่นของสัญญาณเสียงที่ได้จากการเต้นของหัวใจแต่ละครั้งจะมีองค์ประกอบของเสียงที่สำคัญ 2 เสียงคือ first heart sound (S_1) และ second heart sound (S_2) โดยปกติเสียง S_1 จะมีขนาดใหญ่กว่าเสียง S_2 และเสียง S_1 จะอยู่หลังระยะห่างระหว่างกลุ่มของ S_1 และ S_2 และอาจจะมีเสียงอื่น ๆ ที่มีขนาดต่ำกว่าอยู่ระหว่างหรือด้านข้างของเสียงทั้งสองนี้ เมื่อทำการบันทึกเสียงเต้นของหัวใจที่มีจำนวนการเต้นหลายครั้งต่อเนื่องกันแล้วเขียนกราฟแสดงออกใน time domain จะมีลักษณะดังแสดงในภาพประกอบที่ 4-1 ในการตรวจดูกราฟด้วยสายตาจะสามารถบอกการเต้นแต่ละครั้งได้ทันที เพราะระยะห่างระหว่างกลุ่มของ S_1 และ S_2 จะน้อยกว่าระยะห่างระหว่างการเต้นสองครั้งที่อยู่ติดต่อกัน จากสเกลในแกนนอนของกราฟจะสามารถบอกอัตราการเต้นของหัวใจที่บันทึกไว้ได้

ได้ทำการศึกษาหาอัลกอลิทึมในการวิเคราะห์ข้อมูลเสียงที่เก็บบันทึกไว้ ตัวอัลกอลิทึมเองจะทำการประมวลผลข้อมูลที่ได้บันทึกไว้แล้วสามารถที่จะบ่งบอกตำแหน่งข้อมูลที่เกิดการเต้นของหัวใจในแต่ละครั้ง พร้อมกับบอกตำแหน่งของ S_2 ได้ เรียกว่าการทำ segmentation สัญญาณเสียงเต้นของหัวใจ ในการศึกษาครั้งนี้มุ่งที่จะใช้เพียงสัญญาณเสียงจากการเต้นของหัวใจเท่านั้นในการวิเคราะห์ จะไม่ใช้สัญญาณ ECG (electrocardiogram) ประกอบในการวิเคราะห์ วิธีการที่ได้ศึกษาและเห็นว่าได้ผลดีคือการคำนวณหาพลังงานของสัญญาณโดยการใช้ค่าเฉลี่ยที่นอร์มาไลท์แล้วของ shannon energy นอกจากนี้ยังได้ศึกษาวิธีการทำ segmentation โดยใช้ค่าเฉลี่ยที่ normalize แล้วของ energy square, shannon entropy และ absolute value

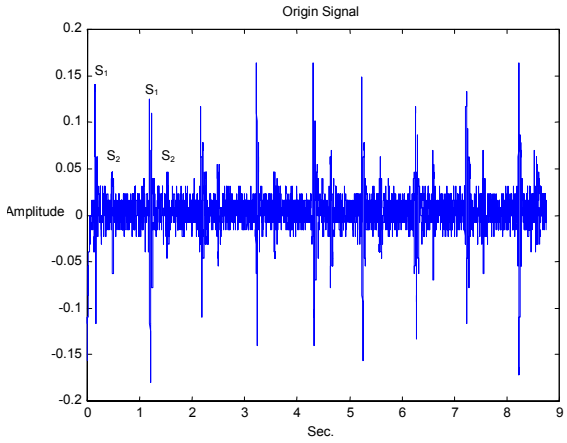
การศึกษาครั้งนี้ใช้โปรแกรม MATLAB เป็นเครื่องมือในการคำนวณและแสดงผล เริ่มจากอ่านข้อมูลที่บันทึกไว้ในไฟล์ .wav มาแสดงผลบนจอภาพคอมพิวเตอร์ ใช้ฟังก์ชันที่มีใน MATLAB เองและเขียนฟังก์ชันต่าง ๆ ขึ้นมาเอง ทำการประมวลผลสัญญาณดิจิตอลของสัญญาณที่บันทึกมาได้

4.2 การทำ segmentation โดยวิธีการคำนวณเชิงคณิตศาสตร์

จากภาพประกอบที่ 4-1 ขณะหัวใจเต้นแต่ละคาบจะเป็นพัลส์สูง ๆ 2 พัลส์ คือ S_1 และ S_2 จากการสังเกตจะเห็น S_1 ที่ช่วงเวลา 0.1, 1.1, 2.2, 3.1, 4.2 วินาที เราใช้วิธีการคำนวณเชิงคณิตศาสตร์เพื่อจะหา ตำแหน่ง S_2 ที่เวลาต่าง ๆ นอกจากนี้ยังสามารถคำนวณอัตราการเต้นของหัวใจได้ซึ่งมีวิธีการทำดังนี้

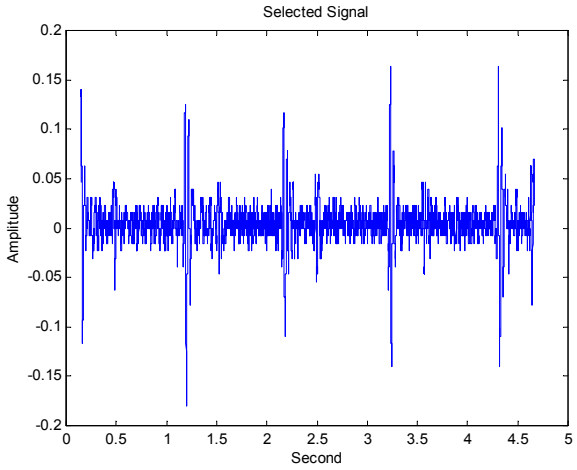
ให้สัญญาณเสียงเต้นของหัวใจที่ถูกเก็บบันทึกไว้แล้วคือลำดับ $x_i[n]$ โดยที่ n = 0,1,2, … และเป็นตัวเลขบอกลำดับที่ของสัญญาณ x_i แล้วดำเนินการตามลำดับขั้นตอนดังนี้

4.2.1 นำลำดับ $x_i[n]$ ที่เก็บบันทึกไว้แล้วแสดงบนจอภาพซึ่งมีค่า $x_i[n]$ ดังตัวอย่างคือ 0.0156, 0.0234, 0.0156, 0.0078, 0, -0.0078, 0.0018, 0, -0.0078, -0.0156, -0.0234, -0.0313, -0.0391, -0.0313, -0.0234, -0.0156, 0, 0.0078, 0.0156, 0.0234, 0.0313, 0.04, 0.05, 0.06, 0.08, 0.09, 0.1, 0.11, 0.12, 0.11, 0.12, 0.11, 0.09, 0.08, 0.07……



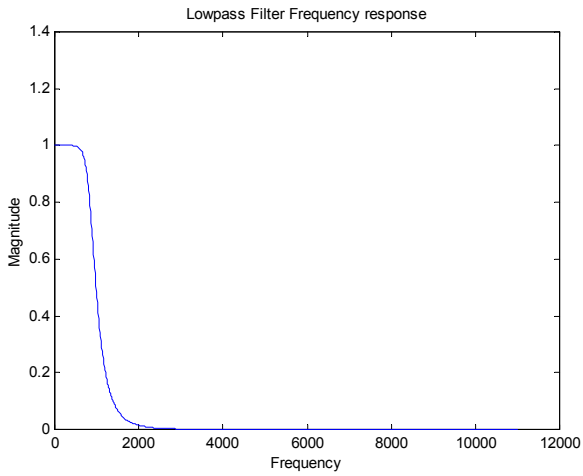
ภาพประกอบที่ 4-1 เสียงเต้นของหัวใจใน time domain ก่อนผ่านกระบวนการ

4.2.2 เลือกช่วงลำดับ $x_i[n]$ ที่จะนำไปวิเคราะห์ จะต้องเลือกช่วงของเวลาที่มีหลาย ๆ รอบของการเต้นของหัวใจ ให้เลือกมาทั้งหมด N_{select} ลำดับ และช่วงเวลาในการเลือกคือ t_{select} วินาที ซึ่งในแต่ละครั้งจะเห็นเสียง 2 ครั้ง คือ S_1 และ S_2

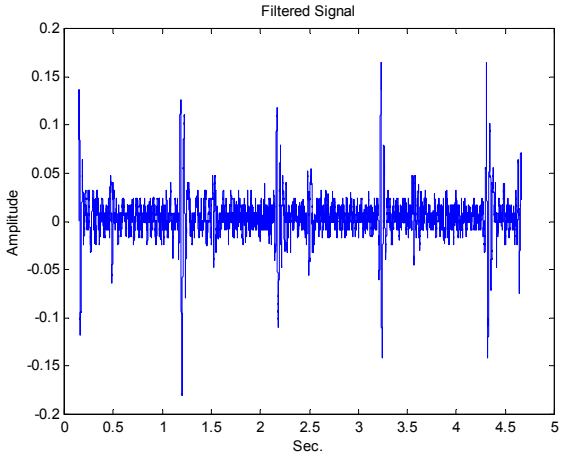


ภาพประกอบที่ 4-2 เสียงที่ผ่านการเลือกช่วงเวลาที่ต้องการ

4.2.3 นำลำดับ $x_i[n]$ ไปผ่าน chebyshev lowpass filter ที่มี cutoff frequency 882 Hz จากลำดับ $x_i[n]$ เมื่อนำไปผ่าน filter จะได้ลำดับใหม่คือ $x_f[n]$

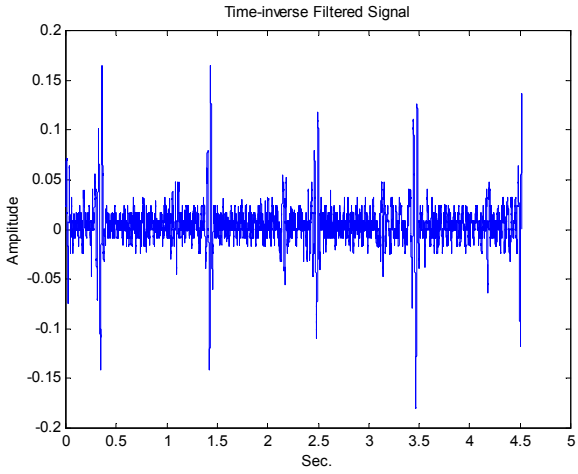


ภาพประกอบที่ 4-3 คุณสมบัติของ lowpass filter ที่ใช้



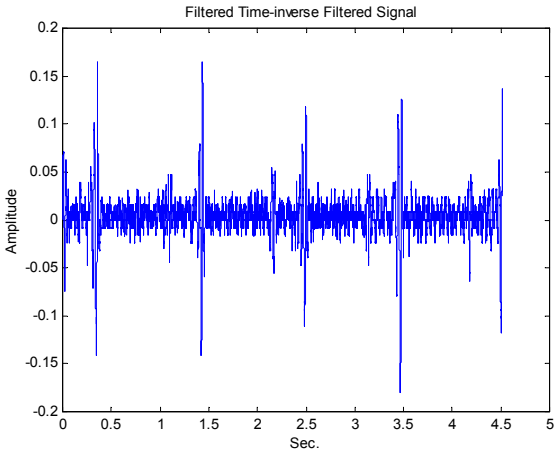
ภาพประกอบที่ 4-4 เสียงที่ผ่านการ filter แล้ว

4.2.4 นำลำดับ $x_f[n]$ มาพลิกกลับจากหน้าเป็นหลัง การทำเช่นนี้จะได้ลำดับที่มีความเพี้ยนด้านเฟสเกือบเป็นศูนย์



ภาพประกอบที่ 4-5 เสียงที่ผ่านการพลิกกลับ

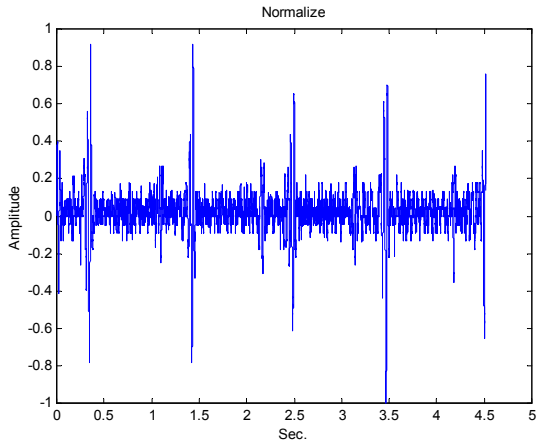
4.2.5 นำลำดับ $x_f[n]$ ไปผ่าน filter เดิมอีกครั้ง



ภาพประกอบที่ 4-6 เสียงเมื่อผ่าน filter เดิมอีกครั้ง

4.2.6 นำลำดับ $x_f[n]$ มาทำการ normalize เพื่อให้ค่าที่มากที่สุดในลำดับมีค่าไม่เกินจาก –1 ถึง 1 โดยหารทุก ๆ พจน์ของลำดับด้วยค่า absolute ที่ใหญ่ที่สุดของลำดับดังสมการ โดย x[n] คือ ลำดับที่ผ่านการ normalize แล้ว

$$x[n] = x_f[n] / \text{Max}(\{ \ | x_f[n] | \}) \qquad (4.1)$$



ภาพประกอบที่ 4-7 เสียงเมื่อผ่านการ normalize

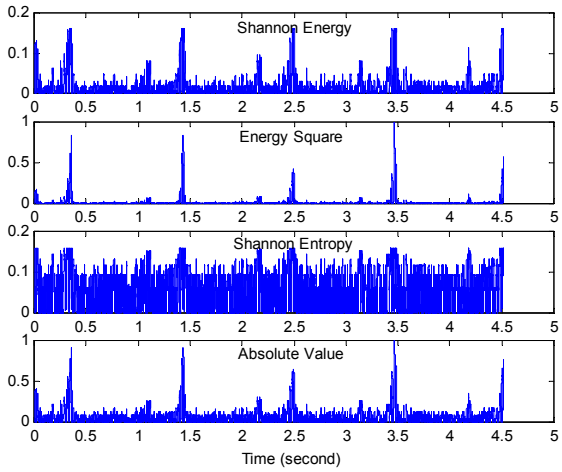
4.2.7 คำนวณหาพลังงานของลำดับ x[n] ซึ่งวิธีการคำนวณค่าพลังงานมี 4 วิธีดังนี้

$$\text{shannon energy :} \quad E = -x^2[n].\log x^2[n] \tag{4.2}$$

$$\text{energy square :} \quad E = x^2[n] \tag{4.3}$$

$$\text{shannon entropy :} \quad E = -|x[n]|.\log.|x[n]| \tag{4.4}$$

$$\text{absolute value :} \quad E = |x[n]| \tag{4.5}$$



ภาพประกอบที่ 4-8 การคำนวณค่าพลังงานในแต่ละวิธี

กราฟในภาพประกอบที่ 4-8 แสดงพลังงานที่คำนวณได้ในแต่ละวิธีดังกล่าว จะสังเกตได้ว่าค่าพลังงานและลักษณะรูปร่างของพลังงานเมื่อเทียบกับเวลาในแต่ละวิธีจะมีลักษณะแตกต่างกัน การใช้ shannon energy, energy square, absolute value จะสามารถมองเห็นพลังงานมีค่าสูงเป็น peak ค่อนข้างชัดเจนในตำแหน่งที่เกิดเสียง S_1 และ S_2 แต่ shannon entropy กราฟที่ปรากฏจะไม่แสดงเป็น peak ที่ชัดเจน นอกจากนี้ค่าพลังงานที่คำนวณได้จะมีค่าเป็นบวกเท่านั้น

4.2.8 ทำการแบ่งลำดับทั้งหมดออกเป็นวินโดว์สั้น ๆ แล้วคำนวณหาค่าเฉลี่ยของพลังงานในแต่ละวินโดว์ ให้ความกว้างแต่ละวินโดว์คือ T_w วินาที และให้วินโดว์ที่ติดกันเหลื่อมกัน $T_w/2$ วินาที ถ้าให้ $T_w = 0.02$ วินาที ทำการคำนวณค่าเฉลี่ยของพลังงานในแต่ละวินโดว์ซึ่งคำนวณได้ตามสมการต่อไปนี้

$$\text{average shannon energy : } E_s = -1/N_w \cdot \sum_{n=1}^{Nw} x^2[n] \cdot \log x^2[n] \quad (4.6)$$

$$\text{average energy square : } E_s = 1/N_w \cdot \sum_{n=1}^{Nw} x^2[n] \quad (4.7)$$

$$\text{average shannon entropy: } E_s = -1/N_w \cdot \sum_{n=1}^{Nw} |x[n]| \cdot \log |x[n]| \quad (4.8)$$

$$\text{average absolute value: } E_s = 1/N_w \cdot \sum_{n=1}^{Nw} |x[n]| \quad (4.9)$$

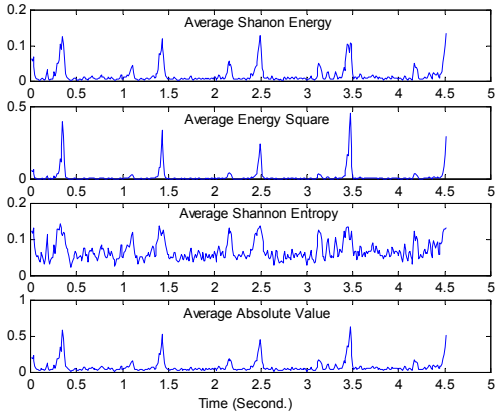
ซึ่ง x[n] คือลำดับที่ผ่านการ normalize แล้ว

N_w คือจำนวนลำดับในแต่ละวินโดว์

F_s คือ อัตราการชักตัวอย่าง

ดังนั้น $N_w = F_s \times T_w$ (4.10)

ค่าเฉลี่ยพลังงานในแต่ละวินโดว์ของลำดับ x[n] ที่คำนวณได้จะเป็นลำดับใหม่คือ E_s [k] ซึ่งผลการคำนวณค่าเฉลี่ยพลังงานของ E_s [k] ตามวิธีดังกล่าวแสดงดังภาพประกอบที่ 4-9



ภาพประกอบที่ 4-9 ผลการพล็อต E_s [k] ที่คำนวณได้ในแต่ละแบบ

E_s [k] ที่ได้จากการคำนวณในแต่ละแบบ จะมีความแตกต่างกันจะเห็นได้ว่าการคำนวณโดยใช้ average shannon energy และ average energy square จะแยกทำให้เห็นสัญญาณที่มีค่าสูงและสัญญาณที่มีค่าต่ำเห็นแตกต่างกันเด่นชัดขึ้น ส่วนการใช้ average shannon entropy

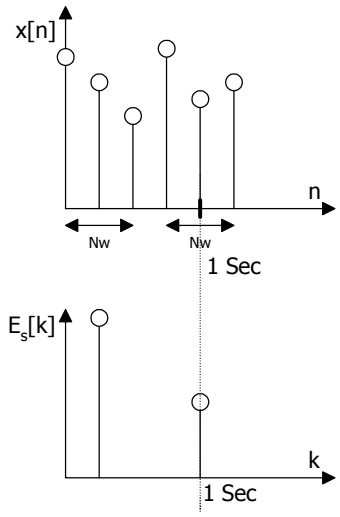
จะทำให้สัญญาณรบกวนที่ต่ำกลับมีค่าสูงขึ้นและการใช้ average absolute value ผลก็คือไม่ได้เพิ่มหรือลดความแตกต่างระหว่างสัญญาณที่มีค่าสูงและสัญญาณที่มีค่าต่ำ

เนื่องจากลำดับ E_s [k] หนึ่งค่าได้จากการใช้ลำดับ x[n] จำนวน N_w ค่า ความสัมพันธ์ระหว่างเวลา t และ n คือ $t = n / F_s$ ในการใช้ลำดับ E_s [k] มีความจำเป็นต้องหาความสัมพันธ์ระหว่างเวลา k และ t ในภาพประกอบที่ 4-10 แสดงความสัมพันธ์กรณีแต่ละวินโดว์ที่ทำการหาค่าเฉลี่ยพลังงานไม่เหลื่อมกันในกรณีนี้จะพบว่า

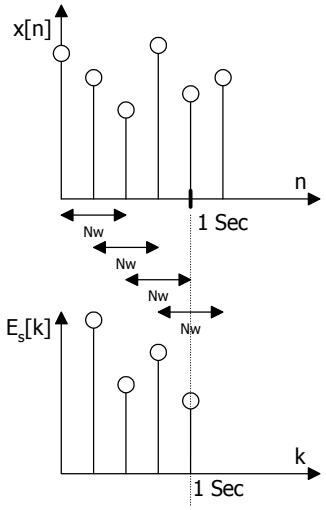
$$t = (k N_w + 0.5 \times N_w) / F_s \qquad (4.11)$$

และในภาพประกอบที่ 4-11 แสดงความสัมพันธ์กรณีแต่ละวินโดว์ที่ทำการหาค่าเฉลี่ยพลังงานเหลื่อมกัน กรณีนี้จะพบว่า

$$t = (k +1) \times 0.5 \times N_w / F_s \qquad (4.12)$$



ภาพประกอบที่ 4-10 กราฟแสดงตัวอย่างการหาค่าเฉลี่ยพลังงานของแต่ละวินโดว์โดยความกว้างแต่ละวินโดว์คือ N_w ภาพบนคือลำดับ x[n], ภาพล่างคือ E_s [k] ที่ได้จากการคำนวณโดยที่แต่ละวินโดว์ไม่เหลื่อมกัน



ภาพประกอบที่ 4-11 กราฟแสดงตัวอย่างการหาค่าเฉลี่ยพลังงานของแต่ละวินโดว์โดยความกว้างแต่ละวินโดว์คือ N_w ภาพบนคือลำดับ x[n], ภาพล่างคือ E_s [k] ที่ได้จากการคำนวณโดยที่แต่ละวินโดว์เหลื่อมกันด้วยเวลา T_w / 2

เมื่อพิจารณาลำดับ E_s [k] ที่ได้ว่าเป็นลำดับที่คำนวณจากการชักตัวอย่างสัญญาณเวลาต่อเนื่อง x(t) เดิมแล้วลำดับ E_s [k] เกิดจากการใช้อัตราการชักตัวอย่าง F_{sn} กรณีแต่ละวินโดว์ที่ทำการหาค่าเฉลี่ยพลังงานไม่เหลื่อมกัน ดังนั้นถ้าให้ F_{so} คืออัตราการชักตัวอย่างของกรณีแต่ละวินโดว์ที่ทำการหาค่าเฉลี่ยพลังงานเหลื่อมกัน

เพื่อหาความสัมพันธ์ระหว่าง F_{sn} และ F_{so} จากภาพประกอบที่ 4-10 และภาพประกอบที่ 4-11 สมมุติว่า F_s มีค่าเท่ากับ 5 ลำดับ / วินาที, F_{sn} ก็จะมีค่าเท่ากับ 2 ลำดับ / วินาที และ F_{so} มีค่าเท่ากับ 4 ลำดับ / วินาที จะสรุปเป็นความสัมพันธ์ดังสมการ

$$F_{so} = 2 \times F_{sn} \qquad (4.13)$$

4.2.9 เพื่อที่จะบอกว่า E_s [k] ที่ตำแหน่งใดเกิดเสียง S_1 และ S_2 ทำการตั้ง thershold โดยกำหนดให้ มีค่า P_a ลำดับ E_s [k] ใดตรงกับการเกิดเสียง นั่นคือ E_s [k] ที่สูงกว่า P_a การกำหนดค่า P_a ทำได้ดังนี้

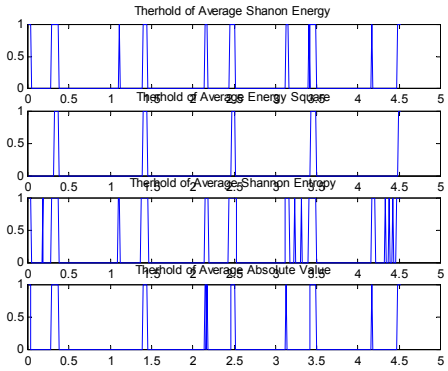
$$P_a = M\{E_s[k]\} + S\{E_s[k]\} \quad (4.14)$$

$M\{E_s[k]\}$ = ค่าเฉลี่ย (mean) ของ $E_s[k]$

$S\{E_s[k]\}$=ค่าเบี่ยงเบนมาตรฐานของ (standard deviation) ของ $E_s[k]$

สร้างลำดับใหม่ $E_p[k]$ จาก $E_s[k]$ โดยลำดับใหม่มีค่าเท่ากับ 1 เมื่อ $E_s[k]$ มากกว่าหรือเท่ากับ P_a และเท่ากับ 0 เมื่อ $E_s[k]$ น้อยกว่า P_a

$$E_p[k] = \begin{cases} 1 \text{เมื่อ} E_s[k] \geq P_a \\ 0 \text{เมื่อ} E_s[k] < P_a \end{cases} \quad (4.15)$$



ภาพประกอบที่ 4-12 การพล็อต $E_p[k]$ จากลำดับ $E_s[k]$ โดยใช้เงื่อนไขตามสมการ (4.15)

ในภาพประกอบที่ 4-12 แสดง $E_p[k]$ โดยการคำนวณ $E_p[k]$ แต่ละแบบตามสมการ (4.6) - (4.9) จากกราฟที่พล็อตได้ จะเห็นว่ากราฟบริเวณใดมีค่า $E_p[k]$ เป็น 1 และบริเวณใดมีค่า$E_p[k]$ เป็น 0 จากการคำนวณเมื่อใช้สมการแตกต่างกันก็จะให้กราฟลักษณะที่แตกต่างกัน ซึ่งในแต่ละวิธีพอจะบอกตำแหน่งที่เกิดเสียง S_1 และ S_2 ได้ เช่น การใช้ average shannon energy จะเห็นกลุ่มของ $E_p[k]$ เป็น 1 ในบริเวณที่เกิด peak แต่การใช้ average energy square บางช่วงที่เกิด peak ค่า $E_p[k]$ ไม่ได้เป็น 1 และการใช้ average shannon entropy บางช่วงที่เกิด peak จะมีมากกว่าการใช้สมการอื่น ๆ ส่วนการใช้ average absolute value บางช่วงที่เกิด peak แต่ค่า $E_p[k]$ ไม่ได้เป็น 1

4.2.10 เมื่อพล็อตค่า E_p [k] ที่คำนวณได้สามารถสังเกตเห็นตำแหน่งของ k ในช่วงที่เกิดเสียง S_1 และ S_2 เพื่อพัฒนาอัลกอลิทึมให้สามารถรับลำดับ E_p [k] แล้วบ่งบอกตำแหน่งที่เกิดเสียง S_2 ได้ ทำการกำหนดค่าตัวเลขขึ้นใหม่ดังนี้

N_R คือจำนวนลำดับ E_p [k] ที่อยู่ระหว่าง S_1 และ S_2

N_{21} คือจำนวนลำดับ E_p [k] ที่อยู่ระหว่าง S_2 และ S_1

N_{22} คือจำนวนลำดับ E_p [k] ที่อยู่ระหว่าง S_2 และ S_2

โดยที่

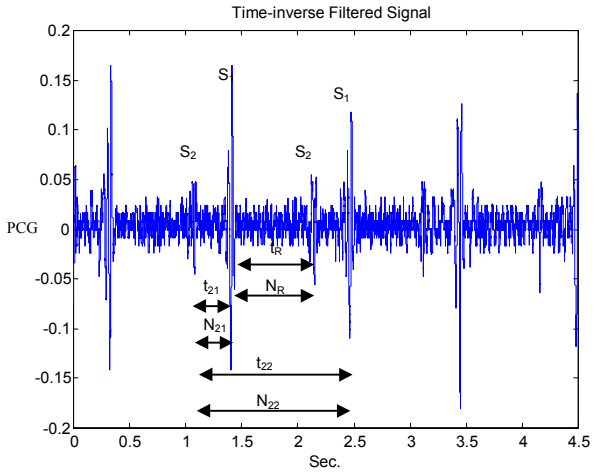
$$N_{21} < N_R < N_{22} \tag{4.16}$$

เมื่อทำการสังเกต N_R จะพบว่ามีค่าประมาณ 2 เท่าของ N_{21} เมื่อเขียนเป็นความสัมพันธ์จะได้ดังสมการ

$$N_R = 2 \times N_{21} \tag{4.17}$$

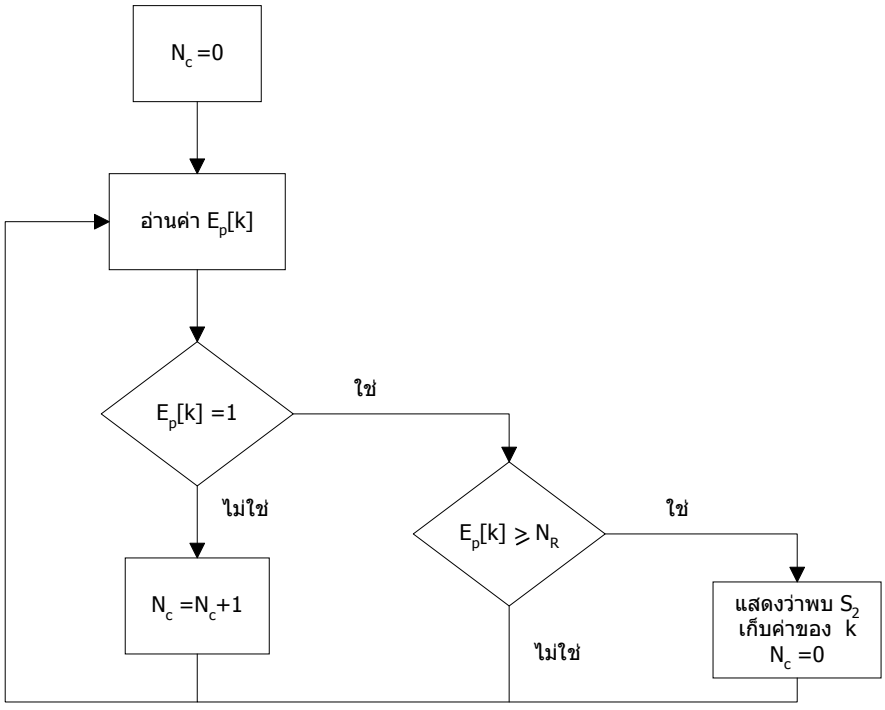
โดยการประเมินจากเสียงหัวใจคนปกติจะพบว่า t_{21} จะมีค่าประมาณ 0.25 วินาที เมื่อรู้ค่า F_{so} ก็จะสามารถหา N_{21} ได้จากความสัมพันธ์

$$N_{21} = t_{21} \times F_{so} \tag{4.18}$$



ภาพประกอบที่ 4-13 กราฟแสดงรายละเอียด t_R , t_{21} , t_{22} , N_R, N_{21} , N_{22} , S_2 , S_1

หลักการในอัลกอลิทึมที่สร้างขึ้นจะรับค่า E_p [k] เรียงกันต่อเนื่องเข้ามาทีละค่า เมื่อพบว่า E_p [k] มีค่าเท่ากับ 1 ให้ถือว่าเริ่มเกิดมีเสียง S_1 หรือ S_2 แล้วทำการบันทึกค่า k ที่ตำแหน่งนี้ การที่จะรู้ได้ว่าเป็น S_1 หรือ S_2 นั้นจะต้องดูระยะห่างของเสียงที่เกิดขึ้นถัด ๆ ไป E_p [k] ค่าถัด ๆ ไปยังคงเป็น 1 ตลอดช่วงความกว้างของเสียง S_1 และ S_2 แต่เมื่อถัดจากเสียง S_1 และ S_2 แล้ว E_p [k] จะมีค่าเป็น 0 ให้เริ่มนับจำนวน E_p [k] ที่เท่ากับ 0 จำนวน 0 ที่นับได้คือ N_C ให้นับจำนวน N_C ต่อไปจนกว่าจะพบว่า E_p [k] มีค่าเท่ากับ 1 อีกครั้ง จึงถือได้ว่ามีเสียง S_1 หรือ S_2 รอบใหม่เกิดขึ้น การที่จะรู้ได้ว่าเสียงรอบใหม่และเสียงที่ผ่านไปเป็น S_1 หรือ S_2 ได้นั้นให้เอาจำนวน N_C ไปเปรียบเทียบกับ N_R แต่ในที่นี้เราต้องการเสียง S_2 อย่างเดียว ดังนั้นถึงแม้ว่า E_p [k] จะมีค่าเท่ากับ 1 ก็ตาม แต่ถ้า N_C น้อยกว่า N_R แสดงว่าเสียงในรอบถัดมาไม่ใช่ S_2 ให้ตรวจสอบ E_p [k] ในลำดับต่อไปอีก และต้องนับจำนวน E_p [k] ที่เท่ากับ 0 ให้ได้จำนวน N_C มากกว่า N_R เมื่อ N_C มากกว่า N_R แล้วหลังจากนั้น ถ้า E_p [k] มีค่าเท่ากับ 1 เมื่อไร แสดงว่าเริ่มเกิดเสียง S_2 ให้เก็บค่า k ที่ตำแหน่งนี้ไว้ แล้วกำหนดให้ N_C มีค่าเท่ากับ 0 จากนั้นให้กลับไปดำเนินกระบวนการเดิมในการรับค่า E_p [k] ใหม่ ภาพประกอบที่ 4-14 แสดง flow chart ของอัลกอลิทึมในการคำนวณหาตำแหน่ง S_2 ดังที่ได้กล่าว



ภาพประกอบที่ 4-14 flow chart แสดงขั้นตอนการหาตำแหน่งที่เกิดเสียง S_2

ได้ทำการทดสอบอัลกอลิทึมนี้โดยใช้ MATLAB เขียนโปรแกรม ซึ่งมีชุดคำสั่งรายการดังนี้

ตัวอย่าง โปรแกรมการตรวจหาตำแหน่งที่เกิด S_2

```
Ts=N*T/2;%divided by 2,because of overlappling
t=Ny/FS
Nn=Ny/(N)
FSn=Nn/t
FSo=2*FSn
N21=FSo*0.25
NR =2*N21
Ny=length(Pa1);
PP1=[];
NC=0;
for  n=1:Ny
  if Pa1(n)==1
    if NC >= NR
      NC = 0;
      PP1=[PP1,(n-1)*Ts];
    end;% if NC
  else
    NC = NC+ 1;
  end;%if Pa

end;%for n

Peak1=PP1
Period1=[(PP1(3)-PP1(2))+(PP1(2)-PP1(1))]/2
Heartbeat1=60/Period1
```

4.2.11 คำนวณอัตราการเต้นของหัวใจจากค่า k ที่เก็บบันทึกโดยหาจากระยะห่างของค่า k ที่อยู่ติดกันคือ T วินาทีหลังจากนั้นคำนวณอัตราการเต้นของหัวใจใน 1 นาที โดย

อัตราการเต้นของหัวใจใน 1 นาที = 60 / T (4.19)

4.3 โปรแกรมคอมพิวเตอร์ที่ได้พัฒนาขึ้น

ในการวิจัยครั้งนี้ได้ใช้ MATLAB พัฒนาโปรแกรมขึ้นมาชุดหนึ่ง โปรแกรมจะอ่านข้อมูลเสียงเต้นของหัวใจที่ถูกบันทึกในไฟล์รูปของ wav โปรแกรมสามารถให้ผู้ใช้ทำการเลือกไฟล์ที่ต้องการ ข้อมูลทั้งหมดในไฟล์ที่ถูกเลือกจะอ่านเข้ามาในตัวแปรของโปรแกรม จึงทำให้สะดวกที่จะเรียกใช้สัญญาณที่ตำแหน่งใดตำแหน่งหนึ่งได้ ภาพประกอบแสดงผลการคำนวณในหัวข้อ 4.2 เป็นผลลัพธ์ที่ได้จากการใช้โปรแกรมนี้ ส่วนรายละเอียดทั้งหมดได้รวบรวมไว้อยู่ในภาคผนวก ลำดับต่อไปในหัวข้อนี้จะแสดงผลที่ได้จากการคำนวณบางส่วนที่ได้จากการใช้โปรแกรมนี้ โดยใช้ไฟล์ที่เก็บเสียงเต้นของหัวใจของคนปกติเก็บบันทึกเสียงเป็นเวลา 4.52 วินาที

จากลำดับ x[n] เมื่อทำการหาค่าเฉลี่ยพลังงานในแต่ละวินโดว์ค่าที่คำนวณได้จะเป็นลำดับใหม่คือ E_s [k]

จากสมการ

$$E_s = -1/N_w \cdot \sum_{n=1}^{Nw} x^2[n] \cdot \log x^2[n]$$

กำหนดให้

F_s = 22,050 ลำดับ/ วินาที

T_w = 0.02 วินาที

T_w / 2 คือระยะห่างของวินโดว์ที่ติดกันเหลื่อมกัน

t_{select} = 4.52 วินาที

N_n คือ จำนวนวินโดว์ทั้งหมดของกรณีแต่ละวินโดว์ที่ทำการหาค่าเฉลี่ยพลังงาน ไม่เหลื่อมกันดังนั้น

t_{21} = 0.25 วินาที

t_{22} = 1.10 วินาที

ดังนั้น

$$N_w = T_w \times F_s$$
$$= 0.02 \times 22,050$$
$$= 441 \text{ ลำดับ}$$

$$N_{select} = t_{select} \times F_s$$
$$= 4.52 \times 22,050$$
$$= 99,660 \text{ ลำดับ}$$

$$N_n = N_{select} / N_w$$
$$= 99,660 / 441$$
$$= 226 \text{ ลำดับ}$$

$$F_{sn} = N_n / t_{select}$$
$$= 226 / 4.52$$
$$= 50 \text{ ลำดับ/ วินาที}$$

จากสมการที่ 4.13 จะได้

$$F_{so} = 2 \times F_{sn}$$
$$= 2 \times 50$$
$$= 100 \text{ ลำดับ/ วินาที}$$

จากภาพประกอบที่ 4-12 ทำการประเมินค่า t_{21} ด้วยสายตาจะพบว่า t_{21} ประมาณ 0.25 วินาที และ t_{22} ประมาณ 1.1 วินาที

ดังนั้น

$$N_{21} = t_{21} \times F_{so}$$
$$= 0.25 \times 100$$
$$= 25 \text{ ลำดับ}$$

$$N_{22} = t_{22} \times F_{so}$$
$$= 1.1 \times 100$$
$$= 110 \text{ ลำดับ}$$

จากสมการ 4.17 จะได้

$$N_R = 2 \times N_{21}$$
$$= 2 \times 25$$
$$= 50$$

เนื่องจาก $N_{21} < N_R < N_{22}$

ดังนั้น $25 < N_R < 110$

ทำการตั้ง thershold โดยกำหนด thershold คือ P_a ดังสมการ

$$P_a = M\{E_s[k]\} + S\{E_s[k]\}$$

จากการคำนวณของโปรแกรม MATLAB จะได้

$$M\{E_s[k]\} = 0.017$$
$$S\{E_s[k]\} = 0.023$$

ดังนั้น

$$P_a = M\{E_s[k]\} + S\{E_s[k]\}$$
$$= 0.04$$

และ

$$E_p[k] = \begin{cases} 1 \text{เมื่อ} E_s[k] \geq 0.04 \\ 0 \text{เมื่อ} E_s[k] < 0.04 \end{cases}$$

ทำการตรวจจับเฉพาะค่าพีคของ S_2โดยเงื่อนไขการจับค่าพีค คือ E_p [k] มีค่าเท่ากับ 1 และ N_C มากกว่า N_R การทำงานของโปรแกรมเริ่มเมื่อ E_p [k] มีค่าเท่ากับ 1 ก็จะทำการเก็บพีคและทำการเพิ่มค่า N_c จาก 0, 1, 2, 3.....จนกระทั่ง E_p [k] มีค่าเท่ากับ 1 อีกครั้งและ N_C มากกว่า N_R ตัวอย่างนี้ตั้งค่า N_R = 50 และจากการคำนวณข้างต้น N_{21}= 25 ลำดับ และ N_{22}= 110 ลำดับ

ที่เวลา t=0 วินาที เริ่มต้นเพิ่มค่า N_C ไปเรื่อย ๆ จนกระทั่งพบ E_p [k] มีค่าเท่ากับ 1 ที่ตำแหน่ง S_1 แต่ N_C น้อยกว่า N_R เพราะ N_{21}= 25 ลำดับ ดังนั้นจึงทำการเพิ่มค่า N_C ไปอีกจนกระทั่งพบ E_p [k] มีค่าเท่ากับ 1 ที่ตำแหน่ง S_2 และ N_C มากกว่า N_R เพราะ N_{22}= 110 ลำดับ ดังนั้นจึงทำการเก็บพีคของ S_2 หลังจากนั้นก็เริ่มต้นเพิ่มค่า N_C ใหม่จนกระทั่งเข้าเงื่อนไข E_p [k] มีค่าเท่ากับ 1 และ N_C มากกว่า N_R จึงจะทำการเก็บพีคในลำดับถัดไป

จากตัวอย่างข้อมูลที่ได้จากไฟล์เสียงดังกล่าวทำการเก็บพีคของ S_2 ได้ที่ตำแหน่ง 0, 1.08, 2.12, 3.11, 4.15 วินาที จะได้ T เฉลี่ย คือ 1.01 วินาที ดังนั้น

อัตราการเต้นของหัวใจใน 1 นาที = 60 / T

= 60 / 1.01

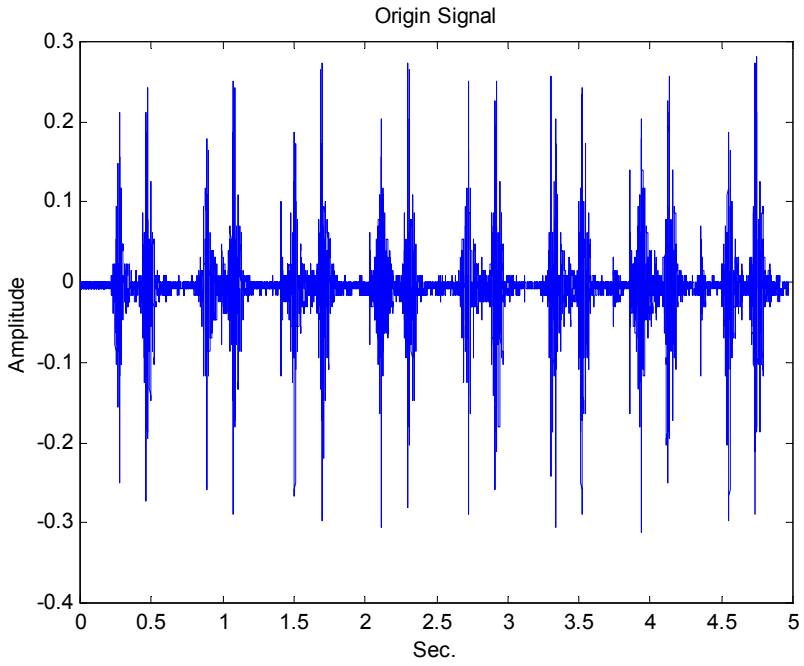
= 59.11 ครั้ง / นาที

ถ้าเราทำการตั้งค่า N_R ให้น้อยกว่า N_{21} เมื่อ N_{21} มีค่าเท่ากับ 25 ถ้าให้ N_R มีค่าเท่ากับ 15 เริ่มต้นเมื่อ E_p [k] มีค่าเท่ากับ 1 และ N_C มีค่าเท่ากับ 0 ทำการเพิ่มค่าของ N_C จนกระทั่งมากกว่า N_R จะพบว่าโปรแกรมจะทำการเก็บพีคของ S_1 เนื่องจาก E_p [k] มีค่าเท่ากับ 1 และ N_C มากกว่า N_R ซึ่งที่ตำแหน่ง S_1 นี้ N_C มีค่าเท่ากับ 25 ซึ่งมากกว่า N_R จึงทำการเก็บพีคของ S_1 ซึ่งไม่ถูกต้องเพราะเราต้องการเก็บพีคของ S_2 เท่านั้น

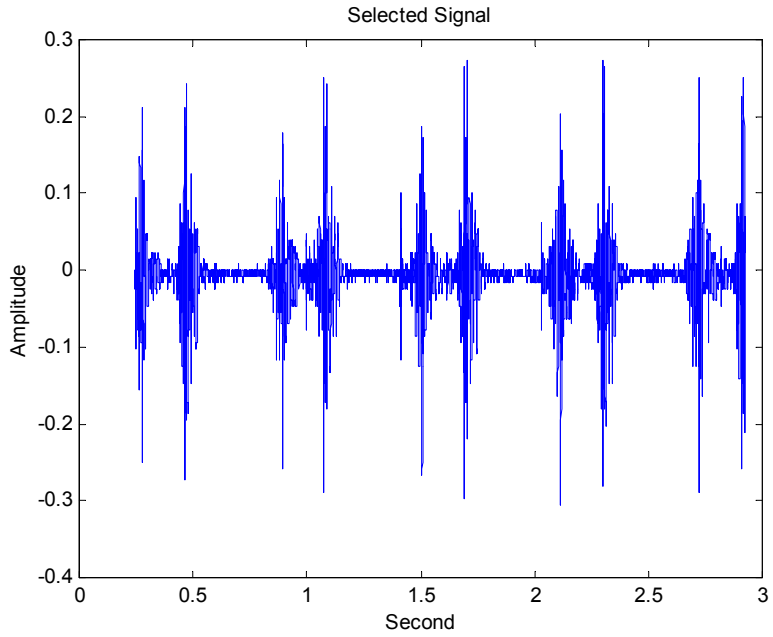
ถ้าเราตั้งค่า N_R ให้มากกว่า N_{22} เมื่อ N_{22} มีค่าเท่ากับ 110 ถ้าให้ N_R มีค่าเท่ากับ 120 เริ่มต้นเมื่อ E_p [k] มีค่าเท่ากับ 1 และ ทำการเพิ่มค่าของ N_C จนกระทั่งพบเสียง S_1 และพบเสียง S_2 ที่เราต้องการเก็บพีคแต่ไม่สามารถเก็บเสียง S_2 ได้เพราะค่าของ N_C ยังน้อยกว่า N_R เนื่องจากที่ตำแหน่ง S_2 นี้ถึงแม้ E_p [k] มีค่าเท่ากับ 1 แต่ N_C ซึ่งมีค่าเท่ากับ 110 ยังมีค่าน้อยกว่า N_R อยู่เพราะโปรแกรมนี้จะเก็บพีคได้ก็ต่อเมื่อ N_C มากกว่า N_R และ E_p [k] มีค่าเท่ากับ 1 เท่านั้น

4.4 ตัวอย่างการแสดงผลบนหน้าจอคอมพิวเตอร์โดยใช้โปรแกรมที่พัฒนา

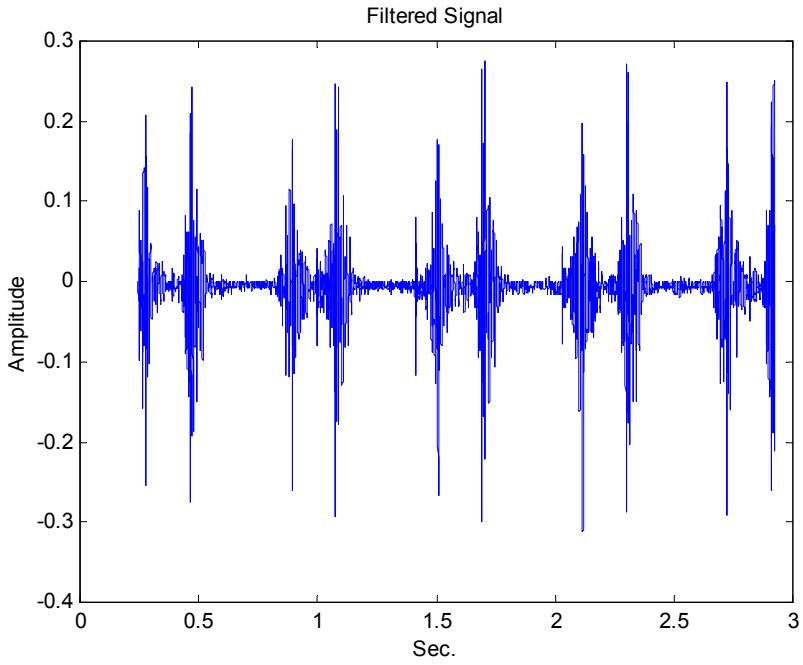
จากโปรแกรมที่พัฒนาขึ้นดังกล่าวข้างต้นสามารถแสดงผลการคำนวณบนหน้าจอคอมพิวเตอร์ในหัวข้อนี้จะแสดงภาพที่ได้จากหน้าจอคอมพิวเตอร์ โดยเลือกใช้ไฟล์เสียงที่เก็บบันทึกได้ในเวลาประมาณ 5 วินาที และเลือกสัญญาณมาประมาณ 2.75 วินาที ภาพประกอบที่ 4-15 แสดงสัญญาณเสียงทั้งหมดที่บันทึกได้ ส่วนภาพประกอบที่ 4-16 แสดงสัญญาณที่เลือกออกมาจากสัญญาณในภาพประกอบที่ 4-15 ในภาพประกอบที่ 4-17 ถึง 4-20 แสดงผลที่ได้จากการคำนวณตามขั้นตอนโดยเอาสัญญาณที่เลือกผ่านกระบวนการต่าง ๆ ก่อนการคำนวณหาพลังงาน เมื่อทำการคำนวณค่าพลังงานโดยใช้ shannon entropy, absolute value, shannon energy และ energy square จะได้ผลลัพธ์ดังภาพประกอบที่ 4-21, 4-22, 4-23 และ 4-24 ตามลำดับการหาค่าเฉลี่ยพลังงานเลือกใช้ average shannon energy ดังภาพประกอบที่ 4-25 นอกจากนั้นยังแสดงการพล็อตค่า $E_p[k]$ โดยทำการตั้ง thershold ที่ให้เท่ากับค่าเฉลี่ยบวกด้วยหนึ่งค่าเบี่ยงเบนมาตรฐานดังภาพประกอบที่ 4-26



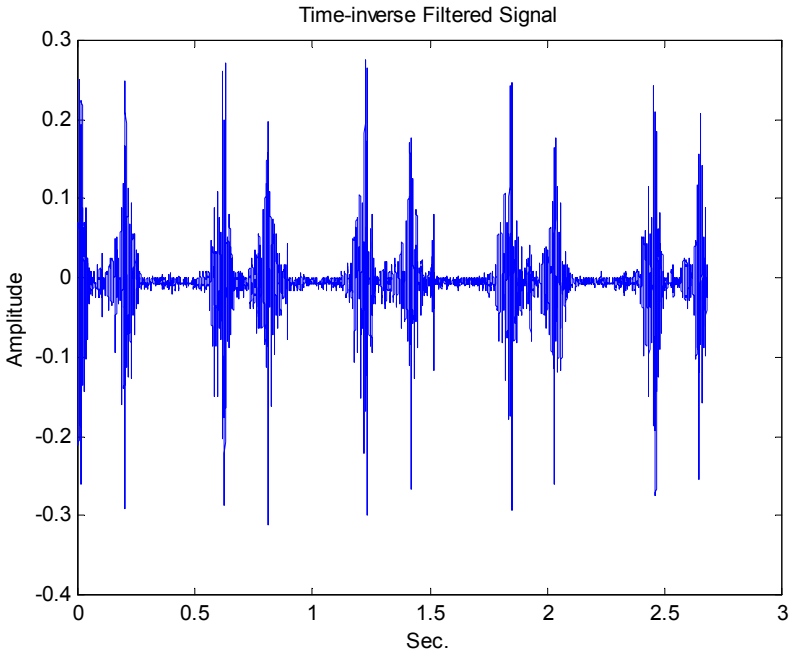
ภาพประกอบที่ 4-15 เสียงเต้นหัวใจใน time domain ก่อนผ่านกระบวนการ



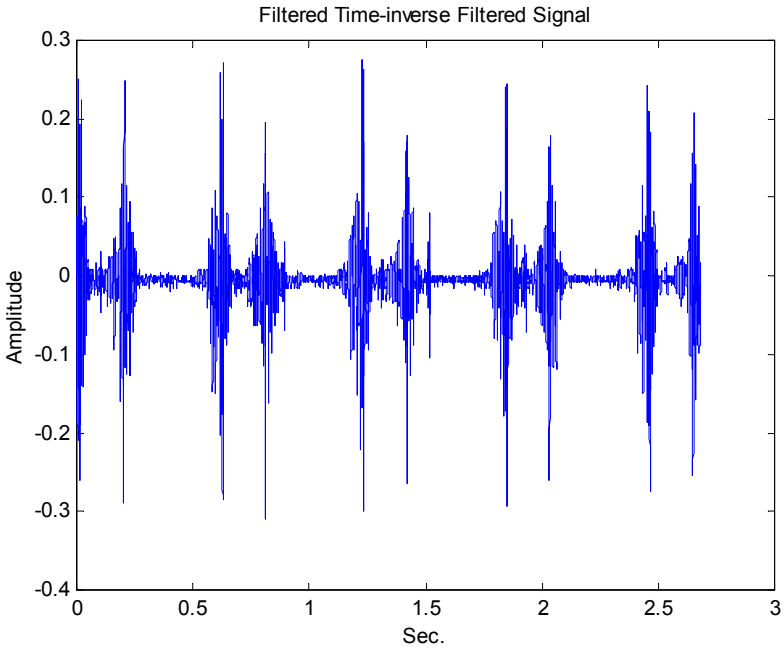
ภาพประกอบที่ 4-16 เสียงที่ผ่านการเลือกช่วงเวลาที่ต้องการ



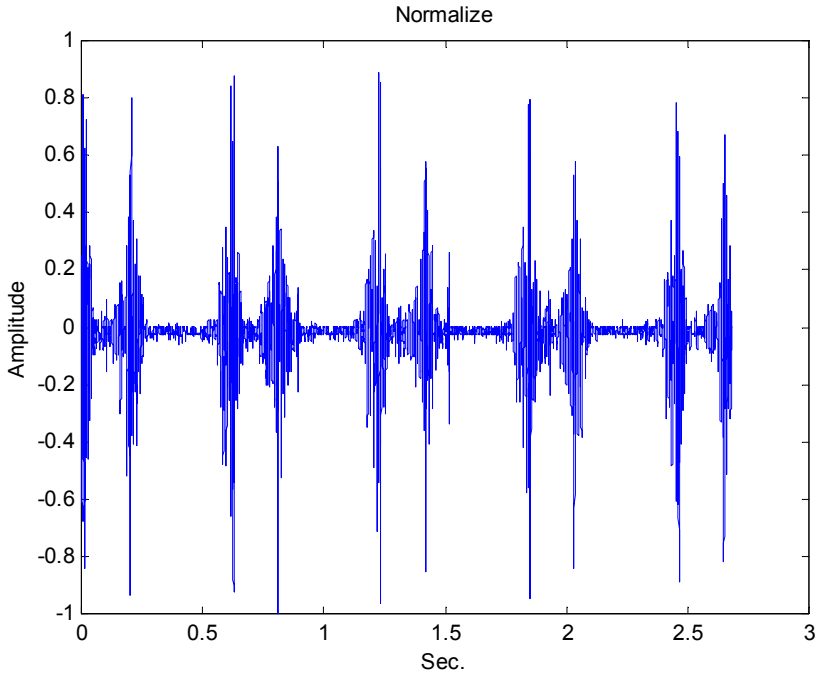
ภาพประกอบที่ 4-17 เสียงที่ผ่านการ lowpass filter แล้ว



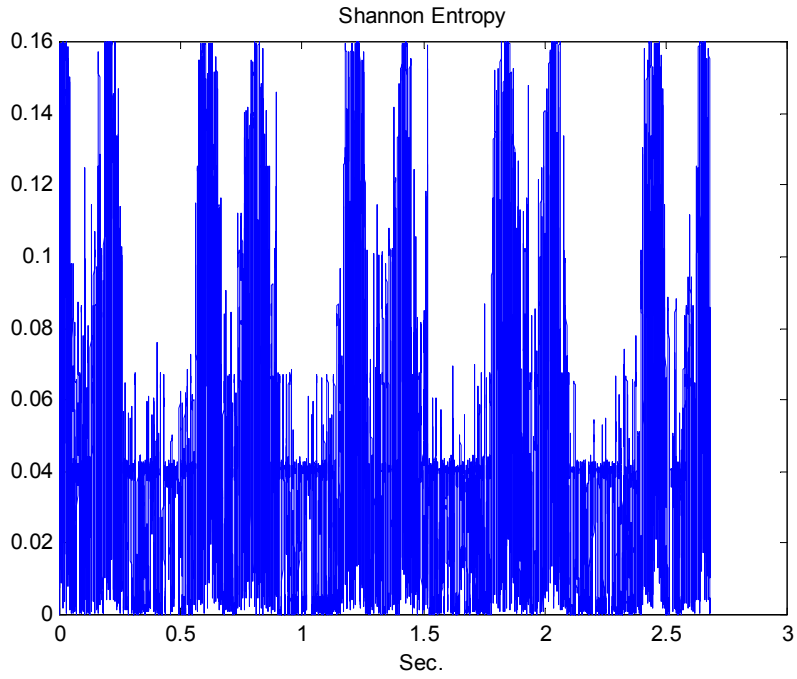
ภาพประกอบที่ 4-18 เสียงที่ผ่านการพลิกกลับ



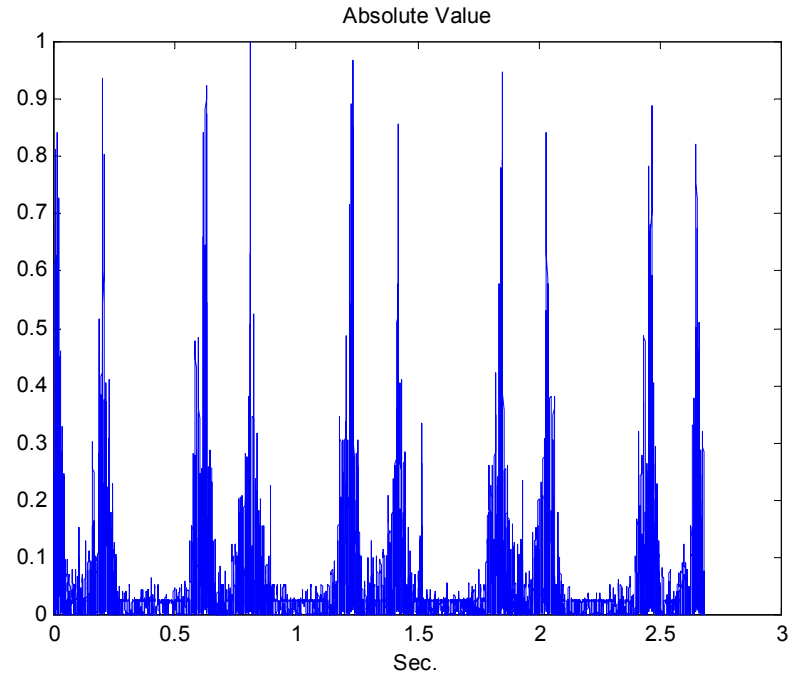
ภาพประกอบที่ 4-19 เสียงเมื่อผ่าน filter เดิมอีกครั้ง



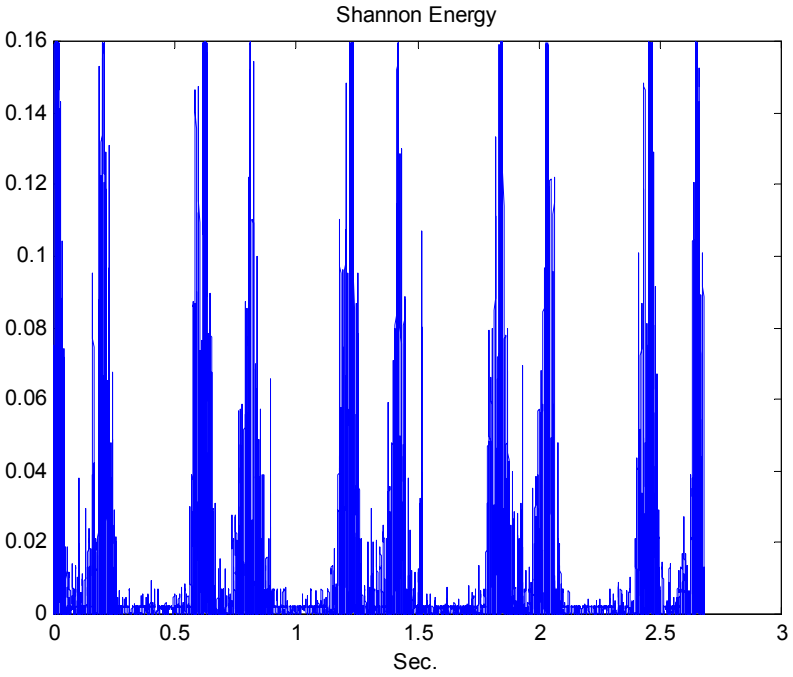
ภาพประกอบที่ 4-20 เสียงเมื่อผ่านการ normalize



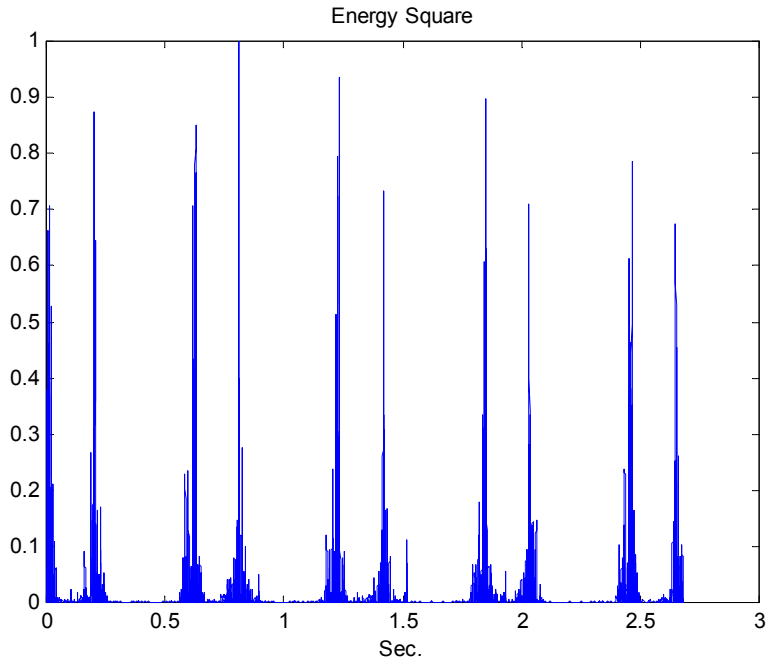
ภาพประกอบที่ 4-21 การคำนวณค่าพลังงานโดยใช้สมการ shannon entropy



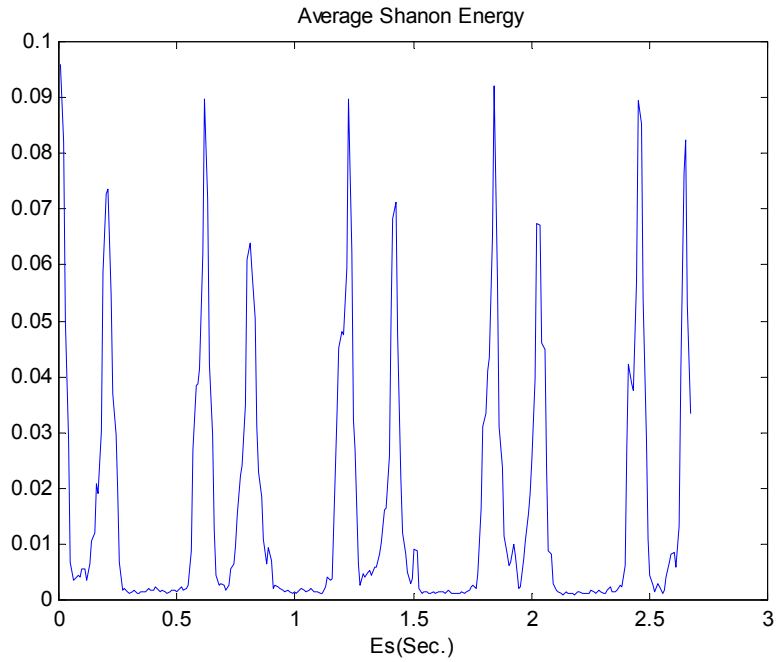
ภาพประกอบที่ 4-22 การคำนวณค่าพลังงานโดยใช้สมการ absolute value



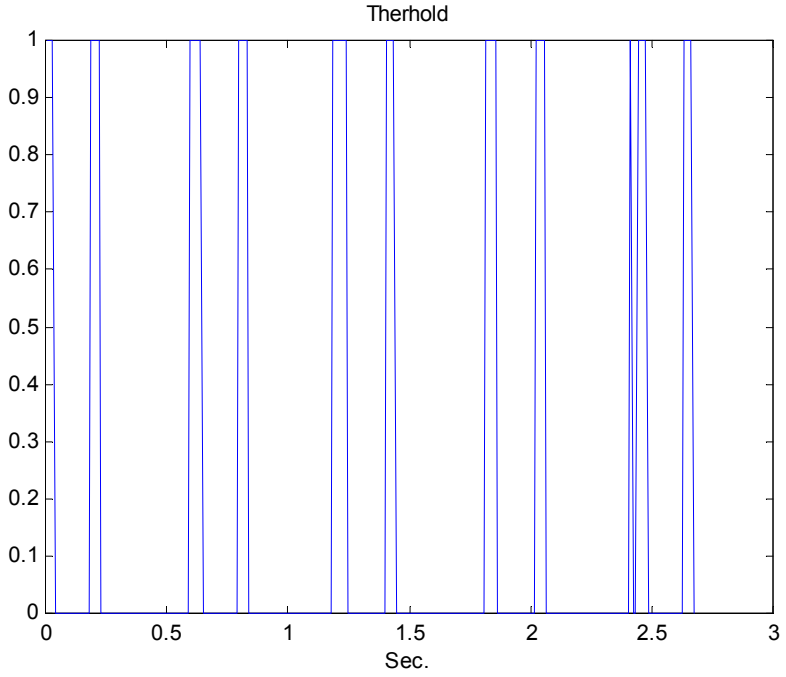
ภาพประกอบที่ 4-23 การคำนวณค่าพลังงานโดยใช้สมการ shannon energy



ภาพประกอบที่ 4-24 การคำนวณค่าพลังงานโดยใช้สมการ energy square



ภาพประกอบที่ 4-25 ผลการพล็อต $E_s[k]$ โดยใช้สมการ average shannon energy



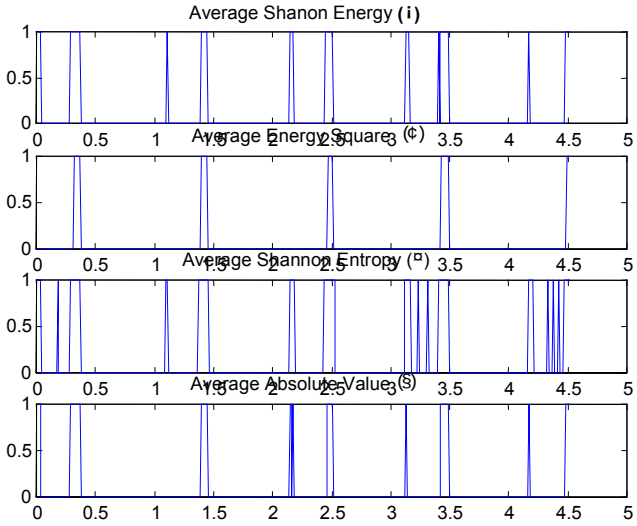
ภาพประกอบที่ 4-26 การพล็อต $E_p[k]$ จากลำดับ $E_s[k]$ โดยทำการตั้งค่า threshold ที่ให้เท่ากับค่า mean บวกด้วยหนึ่ง standard deviation

4.5 วิเคราะห์ความถูกต้องโดยเปลี่ยนพารามิเตอร์บางตัวเพื่อทดสอบผลการคำนวณ

ได้ทำการบันทึกเสียงเต้นของหัวใจของคนปกติ 4 ครั้ง แต่ละครั้งใช้เวลาเก็บประมาณ 5 วินาที เก็บบันทึกเป็นไฟล์ไว้จำนวน 4 ไฟล์ แล้วใช้โปรแกรมที่เขียนขึ้นทำการหาอัตราการเต้นของหัวใจในแต่ละไฟล์ โดยกำหนดให้ความกว้างของวินโดว์ T_w = 0.02 วินาที และ T_w = 0.04 ที่ความกว้างของ T_w ค่าหนึ่ง ๆ ได้ปรับขนาดของ N_R เพื่อทดสอบว่าอัตราการเต้นของหัวใจที่ได้จากการคำนวณโดยโปรแกรมถูกต้องหรือไม่

ในการทดสอบการคำนวณแต่ละครั้ง จะทำการสังเกตกราฟ $E_p[k]$ ที่พล็อตได้ถ้าบริเวณที่เกิดกลุ่มของ $E_p[k]$ เท่ากับ 1 ตรงกับการเกิดเสียง S_1 และ S_2 แล้วถือว่าเป็นลักษณะกราฟที่ถูกต้อง เช่น ดังภาพประกอบที่ 4-27 (ก) ให้ถือว่าเป็นลักษณะที่ถูกต้อง แต่ถ้าบริเวณที่ควรเกิด $E_p[k]$ เท่ากับ 1 แต่การคำนวณแล้วกราฟที่แสดงไม่ปรากฏ ดังภาพประกอบที่ 4-27 (ข) และภาพประกอบที่ 4-27 (ง) หรือบริเวณที่ไม่มีเสียง S_1 และ S_2 แต่ปรากฏว่ามี $E_p[k]$ เท่ากับ 1 ดังภาพประกอบที่ 4-27 (ค) ให้ถือว่าเป็นลักษณะที่ไม่ถูกต้อง

ส่วนอัตราการเต้นของหัวใจแต่ละไฟล์ที่ได้จากการคำนวณโดยโปรแกรม ให้นำไปเทียบกับค่าที่ประมาณได้จากการสังเกตจากกราฟ PCG ของไฟล์นั้น ถ้าตัวเลขทั้งสองต่างกันไม่เกิน 25 % ให้ถือว่าค่าที่ได้จากการคำนวณโดยโปรแกรมนั้นถูกต้อง



ภาพประกอบที่ 4-27 แสดงลักษณะกราฟในรูปแบบ (ก), (ข), (ค) และ (ง)

4.5.1 ใช้ พารามิเตอร์ คือ ความกว้างวินโดว์ (T_w) = 0.02 วินาที และ N_R = 0.2 วินาที

	ลักษณะกราฟที่ถูกต้องดังภาพประกอบที่ 4-27(ก)	การคำนวณอัตราการเต้นของหัวใจที่ถูกต้อง
1.average shannon energy	4 file	0 file
2. average energy square	0 file	0 file
3. average shannon entropy	0 file	0 file
4. average absolute value	2 file	0 file

ตารางที่ 4-1 เปรียบเทียบเมื่อใช้ T_w= 0.02 วินาที และ N_R = 0.2 วินาที

4.5.2 ใช้ พารามิเตอร์ คือ ความกว้างวินโดว์ (T_w) = 0.02 วินาที และ N_R = 0.46 วินาที

	ลักษณะกราฟที่ถูกต้องดังภาพประกอบที่ 4-27(ก)	การคำนวณอัตราการเต้นของหัวใจที่ถูกต้อง
1.average shannon energy	4 file	4 file
2. average energy square	0 file	3 file
3. average shannon entropy	0 file	3 file
4. average absolute value	2 file	3 file

ตารางที่ 4-2 เปรียบเทียบเมื่อใช้ T_w= 0.02 วินาที และ N_R = 0.46 วินาที

4.5.3 ใช้ พารามิเตอร์ คือ ความกว้างวินโดว์ (T_w) = 0.02 วินาที และ N_R = 1.33 วินาที

	ลักษณะกราฟที่ถูกต้องดังภาพประกอบที่ 4-27(ก)	การคำนวณอัตราการเต้นของหัวใจที่ถูกต้อง
1.average shannon energy	4 file	0 file
2. average energy square	0 file	1 file
3. average shannon entropy	0 file	0 file
4. average absolute value	2 file	0 file

ตารางที่ 4-3 เปรียบเทียบเมื่อใช้ T_w= 0.02 วินาที และ N_R = 1.33 วินาที

4.5.4 ใช้ พารามิเตอร์ คือ ความกว้างวินโดว์ (T_w) = 0.04 วินาที และ N_R = 0.46 วินาที

	ลักษณะกราฟที่ถูกต้องดังภาพประกอบที่ 4-27(ก)	การคำนวณอัตราการเต้นของหัวใจที่ถูกต้อง
1.average shannon energy	0 file	1 file
2. average energy square	0 file	1 file
3. average shannon entropy	3 file	1 file
4. average absolute value	0 file	1 file

ตารางที่ 4-4 เปรียบเทียบเมื่อใช้ T_w= 0.04 วินาที และ N_R = 0.46 วินาที

สรุป

จากผลการคำนวณตามวิธีดังกล่าวข้างต้นสรุปว่า

1. ควรใช้วิธีการคำนวณโดยใช้ average shannon energy

2.ในการทำ segmentation ควรใช้ T_w เท่ากับ 0.02 วินาที

3. เงื่อนไขค่า N_R คือ

$$0.25 \text{ วินาที} < N_R < 1.1 \text{ วินาที}$$