การออกแบบระบบรู้จำเสียงพูดคำโดดบนเอฟพีจีเอ

Design an FPGA-based Isolated Word Recognition System

ณรงค์ฤทธิ์ แซ่งั่น[1)] และ ณรงค์ บวบทอง[2)]

Narongrit Sae-ngan[1)] and Narong Buabthong[2)]

[1)] บริษัท เวสเทิร์น ดิจิตอล (ประเทศไทย) จำกัด ปทุมธานี 12121

[2)] ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยธรรมศาสตร์ ศูนย์รังสิต ปทุมธานี 12121

บทคัดย่อ

บทความนี้นำเสนอการออกแบบระบบรู้จำเสียงพูดคำโดดบนอุปกรณ์ FPGA ซึ่งมีกระบวนการทำงานสามขั้นตอน คือ การประมวลสัญญาณเบื้องต้น การดึงค่าลักษณะสำคัญด้วยวิธีการหาค่าสัมประสิทธิ์เคปสตรัมบนความถี่เมล และการรู้จำรูปแบบโดยใช้วิธีของโครงข่ายประสาทเทียม สำหรับขั้นตอนการออกแบบระบบบนเครื่องคอมพิวเตอร์ใช้โปรแกรม MATLAB ในการสร้างและฝึกโครงข่ายประสาทเทียมเพื่อเก็บค่าถ่วงน้ำหนัก ส่วนขั้นตอนการออกแบบระบบบนอุปกรณ์ FPGA ใช้โปรแกรม Xilinx Platform Studio สร้างหน่วยประมวลผล MicroBlaze ขึ้นภายในอุปกรณ์ FPGA โดยระบบของ MicroBlaze จะเชื่อมต่อกับชุดวงจรแปลงสัญญาณอนาล็อกเป็นดิจิตอลและชุดหน่วยความจำซึ่งใช้เก็บค่าถ่วงน้ำหนัก ในการเก็บตัวอย่างเสียงพูดคำโดดจากผู้พูด 15 คน จำนวน 1,750 คำ แบ่งเป็นสามกลุ่ม คือ กลุ่มตัวอย่างสำหรับการฝึกจำนวน 700 คำ กลุ่มทดสอบการรู้จำแบบขึ้นกับผู้พูดจำนวน 700 คำ และกลุ่มทดสอบการรู้จำแบบไม่ขึ้นกับผู้พูดจำนวน 350 คำ ในขั้นตอนการทดสอบประสิทธิภาพการรู้จำ พบว่าการรู้จำแบบขึ้นกับผู้พูดได้ผลการรู้จำเฉลี่ยร้อยละ 99.57 ส่วนการรู้จำแบบไม่ขึ้นกับผู้พูดได้ผลการรู้จำเฉลี่ยร้อยละ 91.99

คำสำคัญ : ระบบรู้จำเสียงพูดคำโดด โครงข่ายประสาทเทียม เอฟพีจีเอ ไมโครเบลส

Abstract

This paper proposes a design of an FPGA-based isolated word recognition system which consists of three processes: (i) preprocessing (ii) feature extraction using Mel-Frequency Cepstral Coefficients (MFCC) and (iii) pattern recognition using Artificial Neural Networks (ANN). MATLAB is used to train the neural networks and calculate their weight and bias. Xilinx Platform Studio (XPS) is also used to create a MicroBlaze system in an FPGA, which connects to both an Analog to Digital Converter (ADC) module and a flash memory used for storing the weight and bias. Speech samples of 1,750, recorded from 15 speakers, are used. These samples are divided into three sets: the training set (700 samples), the speaker-dependent testing set (700 samples) and the speaker-independent testing set (350 samples). The experiment result shows that the recognition performance of the speaker-dependent testing is 99.57%, while the speaker-independent testing is 91.99%.

Keywords : isolated word recognition system, Artificial Neural Networks, FPGA, MicroBlaze

1. บทนำ

การรู้จำเสียงพูด (Speech Recognition) คือ การทำให้คอมพิวเตอร์เข้าใจความหมายคำพูดของมนุษย์ โดยการแปลงสัญญาณเสียงพูดให้เป็นข้อมูลซึ่งมีรูปแบบที่แตกต่างกันตามการออกเสียง และมีกระบวนการแบ่งแยก (Classification) รวมทั้งจดจำรูปแบบของข้อมูลซึ่งแทนคำพูดแต่ละคำ ด้วยเหตุนี้ระบบรู้จำเสียงพูดจึงทำให้มนุษย์สามารถควบคุมและสั่งงานคอมพิวเตอร์รวมถึงอุปกรณ์รอบข้างด้วยเสียงพูดได้ เมื่อนำระบบรู้จำเสียงพูดไปใช้งานร่วมกับอุปกรณ์รับข้อมูลอย่างแป้นพิมพ์และเมาส์จะทำให้คอมพิวเตอร์มีช่องทางในการติดต่อกับมนุษย์มากขึ้น จึงมีการพัฒนาวิธีรู้จำเสียงพูดอย่างต่อเนื่องเพื่อเพิ่มความถูกต้องในการรู้จำ จนกระทั่งปัจจุบันในต่างประเทศมีบริษัทผลิตซอฟท์แวร์เกี่ยวกับการรู้จำเสียงพูด เช่น โปรแกรมพิมพ์ตามเสียงพูด โปรแกรมที่ทำงานโดยใช้คำสั่งเสียง ซึ่งเป็นการเพิ่มความสะดวกสบายให้กับผู้ใช้ทั่วไป และยังช่วยให้ผู้ใช้ที่พิการทางด้านร่างกาย เช่น แขนหรือมือสามารถใช้คอมพิวเตอร์ได้ง่ายขึ้น

เทคโนโลยีเกี่ยวกับการรู้จำเสียงพูดส่วนใหญ่ถูกสร้างมาในรูปของซอฟท์แวร์เพื่อใช้กับคอมพิวเตอร์ส่วนบุคคลที่มีหน่วยความจำขนาดใหญ่และตัวประมวลผลกลางที่มีความเร็วสูงซึ่งออกแบบมาเพื่อใช้ทำงานหลายๆ อย่าง แต่ในกรณีที่ต้องการสร้างเครื่องมือที่ใช้งานเฉพาะอย่างการนำเอาคอมพิวเตอร์ส่วนบุคคลมาประมวลสัญญาณจะทำให้ใช้งานได้ไม่สะดวกเนื่องจากภายในคอมพิวเตอร์แบบนี้ประกอบด้วยอุปกรณ์ที่มีขนาดใหญ่และมีน้ำหนักมาก ดังนั้น หากมีการออกแบบระบบสมองกลฝังตัว (Embedded system) เพื่อใช้ในการรู้จำเสียงพูด ซึ่งเป็นการลดขนาดของหน่วยความจำและหน่วยประมวลผลจะทำให้เทคโนโลยีการรู้จำเสียงพูดสามารถนำไปประยุกต์ใช้งานได้หลากหลายยิ่งขึ้น

FPGA (Field Programmable Gate Array) [1] เป็นวงจรรวมที่โครงสร้างภายในประกอบด้วยวงจรลอจิกพื้นฐาน วงจรลอจิกที่สามารถกำหนดการทำงานของวงจรได้ (Configurable Logic Blocks) และหน่วยความจำ ดังนั้นจึงสามารถออกแบบให้ FPGA ทำงานเหมือนวงจรดิจิตอลได้ตั้งแต่ระดับลอจิกพื้นฐานไปจนถึงวงจรที่มีความซับซ้อนมากๆ เช่น ซีพียู ได้

HDL หรือ Hardware Description Language เป็นเครื่องมือที่ช่วยให้การออกแบบวงจรดิจิตอลด้วย FPGA ที่มีประสิทธิภาพสูง ทำให้เกิดการออกแบบระบบที่มีการทำงานร่วมกันระหว่างฮาร์ดแวร์กับซอฟท์แวร์ (Hardware-Software Codesign) ขึ้นภายใน FPGA โดยส่วนของฮาร์ดแวร์จะเป็นการออกแบบไมโครโปรเซสเซอร์ขึ้นภายใน FPGA และส่วนของซอฟท์แวร์ที่ทำงานในไมโครโปรเซสเซอร์จะเขียนเป็นภาษา C ดังนั้นระบบที่ออกแบบจะมีความยืดหยุ่นสูง และใช้เวลาในการพัฒนาสั้น

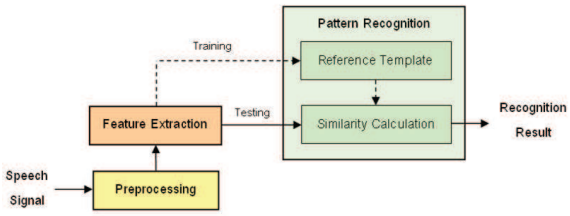
แนวทางของงานวิจัยเกี่ยวกับการรู้จำเสียงพูดภาษาไทยที่เป็นคำโดดที่ผ่านมาส่วนใหญ่เป็นการศึกษาค้นคว้าและปรับปรุงวิธีการในส่วนต่างๆ เพื่อเปรียบเทียบประสิทธิภาพของระบบ [2-4] ต่อมาเมื่อระบบมีอัตราการรู้จำที่สูงขึ้น แนวทางของงานวิจัยจึงเป็นการพัฒนาระบบเพื่อนำไปประยุกต์ใช้ในชีวิตประจำวัน [5,6] ซึ่งเป็นการพัฒนาโปรแกรมคอมพิวเตอร์และส่วนเชื่อมต่อกับอุปกรณ์ที่ต้องการควบคุม

สำหรับงานวิจัยนี้มุ่งเน้นที่การออกแบบและพัฒนาระบบรู้จำเสียงพูดคำโดดบนอุปกรณ์ FPGA โดยสร้างเป็นระบบสมองกลฝังตัวขึ้นภายในอุปกรณ์ ในส่วนของซีพียูสร้างด้วยไมโครโปรเซสเซอร์แบบ Sof-core ที่เรียกว่า MicroBlaze [9] และส่วนของการรู้จำ ใช้โครงข่ายประสาทเทียม ที่พัฒนาขึ้นด้วยภาษา C

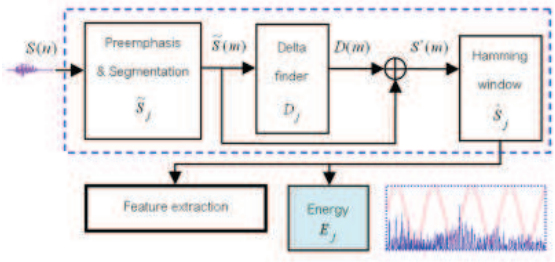
2. ทฤษฎีในการสร้างระบบรู้จำเสียงพูดคำโดด

ระบบรู้จำเสียงพูดคำโดดที่ใช้ในงานวิจัยนี้สามารถจำแนกกระบวนการต่างๆ ออกเป็น 3 ส่วนใหญ่ๆ คือ การประมวลสัญญาณเบื้องต้น (Preprocessing) การดึงค่าลักษณะสำคัญ (Feature Extraction) และการรู้จำรูปแบบ (Pattern Recognition)

ในการรู้จำรูปแบบ แบ่งออกเป็น 2 ขั้นตอน ขั้นตอนแรก คือ การฝึกฝน (Training) เป็นการสร้างข้อมูลอ้างอิง (Reference Template) ขึ้นจากคำในชุดฝึกฝน และขั้นตอนที่สองคือการทดสอบเพื่อหาผลลัพธ์ของระบบ (Testing) เป็นการหาค่าความคล้ายคลึงกันของคำ (Similarity Calculation) โดยอาศัยข้อมูลอ้างอิงที่เก็บไว้ในขั้นตอนแรก ตามรูปที่ 1



รูปที่ 1 การทำงานของระบบรู้จำเสียงพูดคำโดด



รูปที่ 2 การประมวลสัญญาณเบื้องต้น (Preprocessing)

2.1 การประมวลสัญญาณเบื้องต้น (Preprocessing)

เนื่องจากสัญญาณเสียงที่รับเข้ามาในระบบจะมีสัญญาณรบกวนรวมอยู่ด้วย จึงต้องมีขั้นตอนการลดสัญญาณรบกวน รวมถึงการแบ่งสัญญาณเสียงซึ่งเป็นวิธีการจัดเตรียมสัญญาณให้เหมาะกับการประมวลผลในขั้นตอนต่อไป ซึ่งประกอบด้วยขั้นตอนต่างๆ ตามรูปที่ 2 โดยมีรายละเอียดดังนี้

• สัญญาณเสียง (Speech signal) ที่เข้ามาในระบบจะถูกปรับให้เป็นระดับปกติ (Amplitude normalization) โดยการทำให้ค่าสูงสุดของสัญญาณมีค่าเท่ากับ 1 ดังสมการ

$$S(n) = \frac{Speech(n)}{\max(Speech)} \qquad (1)$$

• การเน้นล่วงหน้า (Preemphasis) [2,4,6] คือ การใช้วงจรกรองดิจิตอลอันดับหนึ่ง (first-order preemphasis filter) เพื่อลดผลกระทบจากสัญญาณรบกวนความถี่ต่ำทำให้มีอัตราส่วนสัญญาณเสียงต่อสัญญาณรบกวน (Signal to Noise Ratio: SNR) สูงขึ้น

$$\tilde{S}(n) = S(n) - a \cdot S(n-1) \qquad (2)$$

• การแบ่งสัญญาณออกเป็นช่วงสั้นๆ (Frames) โดยทั่วไปในงานวิจัยเกี่ยวกับเสียงพูด [4-6] จะแบ่งสัญญาณให้มีขนาดเฟรมละ 10-30 มิลลิวินาที เพราะถือว่าสัญญาณภายในช่วงนี้ค่อนข้างคงที่ (Stationary)

• การเพิ่มค่าความแตกต่างของเฟรมรอบข้าง (Delta)

รูปที่ 3 ขั้นตอนการดึงค่าลักษณะสำคัญ MFCC

ให้สัญญาณ [7] เพื่อทำให้พลังงานของเสียงในช่วงเริ่มต้นและสิ้นสุดของคำเพิ่มขึ้น ซึ่งช่วยให้การตรวจหาขอบเขตของคำแม่นยำยิ่งขึ้น

• การวางกรอบสัญญาณ (Windowing) เพื่อลดการเปลี่ยนแปลงอย่างรวดเร็วที่เกิดขึ้นบริเวณปลายแต่ละข้างของเฟรม

2.2 การดึงค่าลักษณะสำคัญ (Feature Extraction)

การดึงค่าลักษณะสำคัญของเสียงพูด คือ การหาค่าลักษณะเด่นที่เป็นตัวแทน (Representation) ของสัญญาณเพื่อใช้ในการลดจำนวนข้อมูล โดยเสียงพูดแบบเดียวกันก็จะมีลักษณะเด่นของเสียงที่เหมือนหรือคล้ายกัน ซึ่งเป็นขั้นตอนสำคัญที่ส่งผลต่อประสิทธิภาพการรู้จำของระบบ วิธีการหาลักษณะสำคัญของเสียงพูดมีอยู่หลายวิธี ในงานวิจัยนี้ใช้วิธีการหาค่าสัมประสิทธิ์เคปสตรัมบนความถี่เมล (Mel-Frequency Cepstral Coefficients: MFCC) [5,6]

โดยสัญญาณ $\hat{S}$ ในแต่ละเฟรมจะถูกแปลงฟูเรียร์ เพื่อหาค่าสเปกตรัมกำลัง (Power spectrum) ส่งเข้าไปในชุดของตัวกรอง (Filter bank) ที่อยู่บนสเกลความถี่แบบเมล (Mel-scale) ได้เป็นค่าสเปกตรัมกำลังแบบเมล (Mel-power spectrum) จากนั้นใส่ค่าลอการิทึม ก่อนที่จะแปลงโคซายน์ (DCT) ให้เป็นค่าสัมประสิทธิ์เคปสตรัมบนความถี่เมล ขั้นตอนทั้งหมดนี้แสดงในรูปที่ 3

2.2.1 ค่าความแตกต่างระหว่างสัมประสิทธิ์ในแต่ละเฟรมกับค่าเฉลี่ยของสัมประสิทธิ์ในช่วงเงียบ

หลังจากหาค่าลักษณะสำคัญของสัญญาณในแต่ละเฟรมแล้ว จากนั้นจึงคำนวณหาค่าระยะห่างแบบยูคลีเดียน (Euclidian distance) ระหว่างสัมประสิทธิ์ในแต่ละเฟรมกับค่าเฉลี่ยของสัมประสิทธิ์ในช่วงเงียบ (3 เฟรมแรกและ 3 เฟรมสุดท้าย) [7,8]

ค่าระยะห่างแบบยูคลีเดียนแสดงถึงความแตกต่างระหว่างสัญญาณในแต่ละเฟรมกับสัญญาณอ้างอิง (ค่าเฉลี่ยของสัมประสิทธิ์ในช่วงเงียบ) ซึ่งจะแยกระหว่างสัญญาณเสียงช่วงที่มีพลังงานต่ำ เช่น เสียงพ่นลมในพยัญชนะ ฟ ส

ซ (Fricative) กับสัญญาณในช่วงเงียบ (Background noise) ได้เด่นชัดยิ่งขึ้น

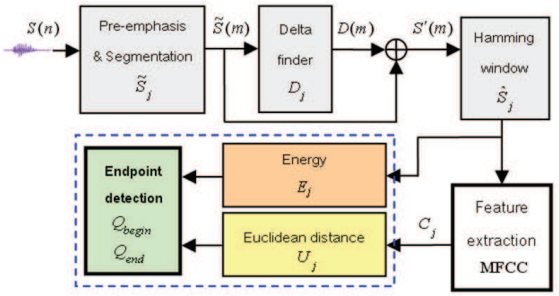
2.2.2 การตรวจหาขอบเขตของคำ

สัญญาณในช่วงเงียบ (Silence) จะมีค่าพลังงานต่ำและค่อนข้างเรียบ ส่วนในช่วงเสียงพูดแบบโฆษะ (Voiced) สัญญาณจะมีค่าพลังงานสูง ในงานวิจัยนี้จึงนำค่าพลังงานของเสียง (E) พร้อมด้วยค่าความแตกต่างระหว่างสัมประสิทธิ์ในแต่ละเฟรมกับค่าเฉลี่ยของสัมประสิทธิ์ในช่วงเงียบ (U) มาใช้ในการตรวจหาขอบเขตของคำ

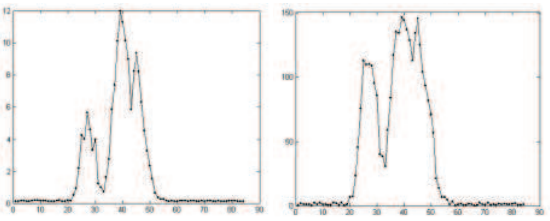
การหาค่าพลังงานของเสียง E และค่าระยะห่างแบบยูคลีเดียน U แสดงในรูปที่ 4 และลักษณะของ ทั้ง E และ U แสดงในรูปที่ 5

การตรวจหาขอบเขตของคำจะให้สัญญาณในเฟรมที่เป็นเสียงพูด (มีค่าเป็น 1) ถ้าเฟรมนั้นมีค่า E_j และ U_j ตรงตามเงื่อนไขข้อใดข้อหนึ่ง ต่อไปนี้

- $U_j > \alpha * U_{sil}$ & $E_j > \beta * E_{sil}$
- $U_j > U_{sil}$ & $E_j > \eta * E_{sil}$
- $E_j > \gamma * E_{sil}$



รูปที่ 4 การหาค่า E และ U ที่ใช้ในการตรวจหาขอบเขตของคำ



(ก) (ข)

รูปที่ 5 (ก) ค่าพลังงานของเสียง (E)
(ข) ค่าระยะห่างแบบยูคลีเดียน (U)

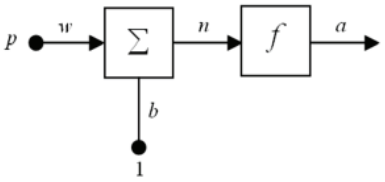
โดยที่ $\alpha = 3$; $\beta = 0.75$; $\eta = 1.1$; $\gamma = 1.3$ เป็นค่าจากการทดลอง [8] ส่วน E_{sil} เป็นค่าเฉลี่ยของ E ในช่วงเงียบและ U_{sil} เป็นค่าเฉลี่ยของ U ในช่วงเงียบ

2.3 การรู้จำรูปแบบ (Pattern Recognition)

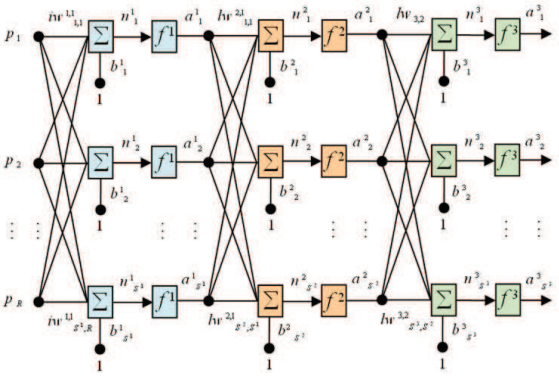
เมื่อได้ลักษณะสำคัญของเสียงแล้วจะนำค่าที่ได้มาคำนวณเทียบเคียงกับข้อมูลอ้างอิงเพื่อหาคำตอบว่าค่าลักษณะสำคัญนั้นตรงหรือคล้ายคลึงกับเสียงพูดคำใด ซึ่งขั้นตอนในการฝึกฝนเพื่อให้ได้ข้อมูลอ้างอิงนั้นขึ้นอยู่กับวิธีในการรู้จำของระบบ เช่น วิธี Dynamic Time Warping: DTW [2] หรือวิธี Hidden Markov Model: HMM [2,5] หรือ วิธีการทางโครงข่ายประสาทเทียม (Artificial Neural Networks: ANN) [2] ซึ่งจาก [2] ได้แสดงให้เห็นว่าวิธีการของโครงข่ายประสาทเทียมใช้กรรมวิธีที่ธรรมดาที่สุดโดย ใช้เงื่อนไขความคลาดเคลื่อนต่ำสุดในการเปรียบเทียบ แต่ให้ผลลัพธ์ของการรู้จำใกล้เคียงกับ 2 วิธีแรก ดังนั้นในงานวิจัยนี้จึงเลือกใช้วิธีการของโครงข่ายประสาทเทียมในการรู้จำรูปแบบของเสียงพูด เพื่อให้ระบบมีความซับซ้อนน้อยที่สุดเหมาะสมกับการนำไปพัฒนาเป็นชิพสำหรับการรู้จำคำสั่งภาษาไทยในโอกาสต่อไป

โครงข่ายประสาทเทียม [2-4] คือ ระบบที่มีโครงสร้างและการทำงานเลียนแบบระบบประสาทของมนุษย์ซึ่งแบบจำลองเซลล์ประสาทในรูปที่ 6 ประกอบด้วยค่าอินพุท p คูณกับค่าน้ำหนัก w แล้วบวกกับค่าถ่วง b จากนั้นส่งผ่านฟังก์ชันกระตุ้น (activation function) f เพื่อจำกัดขอบเขตค่าเอาท์พุต a ให้อยู่ในช่วงที่ต้องการ

เมื่อมีเซลล์ประสาทมากกว่า 1 โหนด (node) เชื่อมต่อกันจะเรียกว่าชั้น (layer) ของเซลล์ประสาท ซึ่งภายในโครงข่ายสามารถมีจำนวนชั้นได้มากกว่า 1 ชั้น (Multiple layers of neurons) ตามรูปที่ 7 โดยเซลล์ประสาทนี้เรียกว่า เพอเซปตรอน (Perceptron) มีความสามารถในการแยกประเภทของข้อมูลที่ป้อนเข้ามาโดยใช้ค่าถ่วงน้ำหนักที่ได้จากการเรียน



รูปที่ 6 แบบจำลองเซลล์ประสาท



รูปที่ 7 เซลล์ประสาทที่เชื่อมต่อกันแบบหลายชั้น

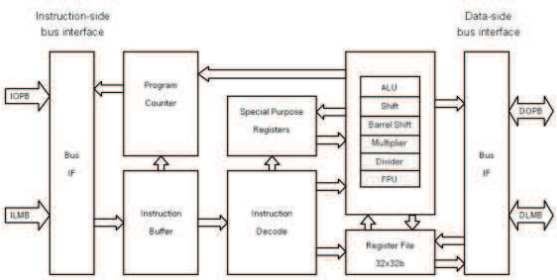
รู้แบบมีการสอน (Supervised learning) คือ มีการกำหนดชุดตัวอย่างที่ใช้ในการเรียนรู้ให้

2.3 MicroBlaze

MicroBlaze [9] เป็นไมโครโปรเซสเซอร์ที่สร้างขึ้นจาก Configurable Logic Blocks (CLBs) ภายใน FPGA (Soft-core microprocessor) สิ่งที่น่าสนใจ คือ การเป็นตัวประมวลผลที่มีความยืดหยุ่นในการใช้งาน เนื่องจากสามารถสร้าง MicroBlaze ให้มีความสามารถในการทำงานเท่าที่จำเป็น (เช่น Barrel shifter, Multiplier, Divider, Floating-Point Unit และ อื่นๆ) ความสามารถที่ไม่ได้เลือกใช้งานจะไม่ถูกนำไปสร้าง จึงไม่ใช้พื้นที่ของ FPGA

นอกจากนั้นยังสามารถปรับเพิ่ม-ลดอุปกรณ์เชื่อมต่อต่างๆ ได้ (เช่น UART, GPIO และอื่นๆ) ทำให้ผู้ใช้สามารถออกแบบระบบที่มีประสิทธิภาพการทำงานคุ้มค่ากับทรัพยากรที่เสียไป โครงสร้างพื้นฐานของ MicroBlaze แสดงในรูปที่ 8

MicroBlaze เป็นสถาปัตยกรรมแบบ 32-bit Reduced Instruction Set Computer (RISC) ที่ถูกพัฒนาเพื่อนำมาใช้



รูปที่ 8 โครงสร้างพื้นฐานของ MicroBlaze

ใน FPGA ของบริษัท Xilinx ซึ่งมีการแยก Local Memory Bus (LMB) ที่ใช้ติดต่อกับหน่วยความจำภายใน FPGA (Block RAM แบบ Dual port) ออกเป็นส่วนของชุดคำสั่ง (Instruction-side) และส่วนของข้อมูล (Data-side) นอกจากนั้นยังมี On-chip Peripheral Bus (OPB) ที่ใช้ติดต่อกับอุปกรณ์รอบข้างรวมถึงหน่วยความจำภายนอกด้วย สำหรับประเภทข้อมูลที่สนับสนุน คือ Word (32 บิต), Half word (16 บิต) และ Byte (8 บิต) โดยใช้รูปแบบ Big-Endian ในการแทนบิตของข้อมูล (Bit 0 คือ MSB และ Bit 31 คือ LSB)

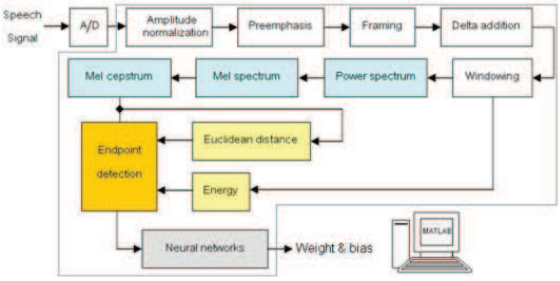
EDK (Embedded Development Kit) เป็นเครื่องมือที่ช่วยสร้าง MicroBlaze โดยจำเป็นต้องมี ISE (Integrated Software Environment) ซึ่งเป็นซอฟท์แวร์ที่ใช้พัฒนาการออกแบบอุปกรณ์ของ Xilinx ติดตั้งอยู่ด้วย สำหรับขั้นตอนการออกแบบระบบจะใช้ซอฟท์แวร์ XPS (Xilinx Platform Studio) ในการสร้างและปรับแต่งส่วนของ Hardware (Processor core, Memory-controller, I/O peripherals) นอกจากนั้นยังสามารถสร้างอุปกรณ์ที่นอกเหนือไปจากรายการที่มีอยู่ (Custom peripherals [10]) เพื่อใช้ทำงานที่แตกต่างออกไปตามความต้องการ และใช้ภาษา C ใน EDK เป็นคำสั่งควบคุมการทำงานของ MicroBlaze

3. การออกแบบระบบรู้จำเสียงพูด

ระบบรู้จำเสียงพูดที่นำเสนอได้ออกแบบเป็น 2 ส่วน ส่วนแรกเป็นส่วนการฝึกโครงข่ายประสาทเทียมเพื่อหาค่าถ่วงน้ำหนักที่จะนำไปใช้ในส่วนที่สองที่เป็นส่วนรู้จำ โดยส่วนแรกออกแบบไว้บนเครื่องคอมพิวเตอร์ส่วนบุคคล และส่วนรู้จำได้ออกแบบไว้บนอุปกรณ์ FPGA ดังมีรายละเอียดดังต่อไปนี้

3.1 การออกแบบระบบรู้จำเสียงพูดบนเครื่องคอมพิวเตอร์สำหรับการฝึกโครงข่ายประสาทเทียม

กระบวนการทำงานของระบบเสียงพูดบนเครื่องคอมพิวเตอร์สำหรับการฝึกโครงข่ายประสาทเทียม เป็นไปตามรูปที่ 9 โดยสัญญาณเสียงที่ผ่านการแปลงด้วย A/D จะถูกปรับให้เป็นระดับปกติ (Amplitude normalization) โดยการทำให้ค่าสูงสุดของสัญญาณมีค่าเท่ากับ 1 แล้วทำการเน้นล่วงหน้า (Preemphasis) และแบ่งออกเป็นเฟรม (Framing) เพื่อหาค่าความแตกต่างของเฟรมรอบข้าง (Delta) เพิ่มให้กับสัญญาณ ก่อนทำการวางกรอบหน้าต่าง ให้กับสัญญาณ



รูปที่ 9 การทำงานของระบบรู้จำเสียงพูด

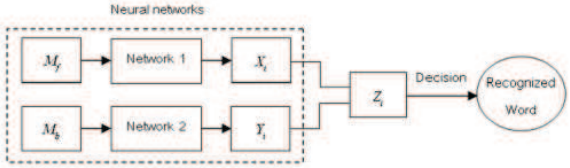
ในแต่ละเฟรม

ขั้นตอนต่อไปคือการหาค่าพลังงานเสียง และค่าเคปสตรัมบนความถี่เมล ของสัญญาณในแต่ละเฟรม เมื่อได้ค่าสัมประสิทธิ์ครบทุกเฟรมแล้วจึงหาค่าระยะห่างแบบยูคลิเดียน (Euclidean distance) ระหว่างค่าสัมประสิทธิ์ในแต่ละเฟรมกับค่าเฉลี่ยของสัมประสิทธิ์ในช่วงเงียบ จากนั้นใช้ค่าพลังงานเสียงพร้อมกับค่าระยะห่างของสัมประสิทธิ์ตรวจหาขอบเขตของคำ (Endpoint detection) เมื่อได้คู่ตำแหน่งที่ระบุตำแหน่งเฟรมเริ่มต้นและเฟรมสิ้นสุดของคำแล้ว จะสามารถหาตำแหน่งของสัมประสิทธิ์ช่วงที่เป็นเสียงพูดได้

เนื่องจากข้อมูลในชั้นอินพุท (Input layer) ของโครงข่ายประสาทเทียมจะต้องมีจำนวนข้อมูลเท่ากัน แต่ค่าสัมประสิทธิ์เคปสตรัมบนความถี่เมล ที่ได้จากกระบวนการหาขอบเขตของคำมีจำนวนไม่เท่ากันขึ้นอยู่กับการออกเสียงสั้น-ยาวของผู้พูด ดังนั้นจึงต้องมีวิธีการปรับจำนวนของ MFCC ให้เท่ากันก่อน โดยในระบบนี้ใช้วิธีแบ่งค่าลักษณะสำคัญของเสียงออกเป็น 2 ส่วน ส่วนละ 125 ค่า คือ ค่าลักษณะสำคัญช่วงต้น (M_f) และค่าลักษณะสำคัญช่วงท้าย (M_b) เพื่อป้อนให้โครงข่ายประสาทเทียม 2 โครงข่าย

จากนั้นจะทำการบันทึกค่า M_f และ M_b (ของสัญญาณเสียงในกลุ่มตัวอย่าง) ไว้เป็นชุดข้อมูลสำหรับใช้ในขั้นตอนการฝึกโครงข่ายประสาทเทียมทั้ง 2 โครงข่าย เพื่อให้ได้ค่าถ่วงน้ำหนัก (Weight & bias) สำหรับการนำไปใช้ในระบบรู้จำต่อไป

ในขั้นตอนนี้ได้มีการทดสอบเบื้องต้นว่าสามารถให้ผลการรู้จำเป็นอย่างไร ด้วยการใช้ค่า M_f และ M_b ป้อนให้ชั้นอินพุทของโครงข่าย และนำค่าถ่วงน้ำหนักที่ได้จากขั้นตอนการฝึกมาใช้คำนวณหาค่าผลลัพธ์ของทั้ง 2 โครงข่าย ตามรูปที่ 10



รูปที่ 10 การหาค่าผลลัพธ์ของโครงข่ายประสาทเทียม

ถ้ากำหนดให้ผลลัพธ์ของโครงข่ายที่ 1 คือ X_i และผลลัพธ์ของโครงข่ายที่ 2 คือ Y_i แล้ว ผลลัพธ์ที่ได้จากโครงข่ายประสาทเทียม (Z_i) คือ

$$Z_i = X_i \times Y_i \quad , \quad 1 \le i \le 7 \qquad (3)$$

โดยที่ i คือ จำนวนโหนดในชั้นเอาท์พุตของโครงข่ายเท่ากับจำนวนคำที่ต้องการรู้จำ

จากนั้น ระบบจะทำการตัดสิน (Decision) ว่าสัญญาณเสียงที่รับเข้ามาตรงกับคำพูดคำใด โดยใช้วิธีการตรวจเลือกค่าสูงสุดของผลลัพธ์ที่ได้จากโครงข่ายประสาทเทียม โดยค่าสูงสุดนั้นจะต้องมากกว่าค่าที่กำหนดไว้จึงจัดเป็นคำพูดในกลุ่มคำที่ต้องการรู้จำ

ระบบที่กล่าวมาในหัวข้อนี้ นอกจากส่วนA/D พัฒนาด้วยโปรแกรม MATLAB

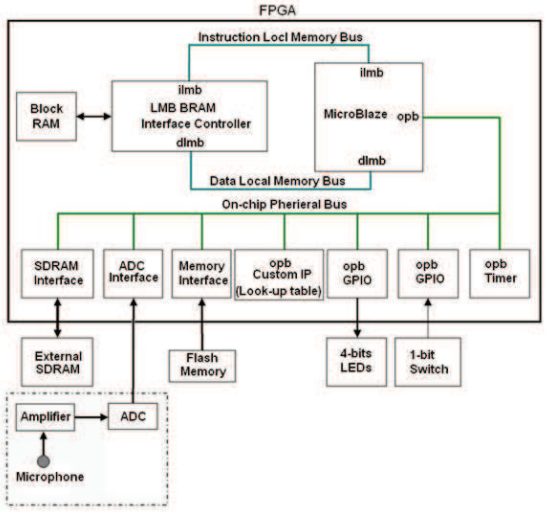
3.2 โครงสร้างของระบบรู้จำเสียงพูดที่ทำงานบนอุปกรณ์ FPGA

ระบบรู้จำเสียงพูดที่ทำงานบนอุปกรณ์ FPGA นี้ใช้ MicroBlaze เป็นหน่วยประมวลผลข้อมูล มีกระบวนการทำงานเหมือนกับระบบบนเครื่องคอมพิวเตอร์ในรูปที่ 9 เพียงแต่โครงข่ายประสาทเทียมจะนำค่าถ่วงน้ำหนัก จากหัวข้อ 3.1 มาใช้ในการรู้จำ

แผนผังบล็อกของระบบรู้จำเสียงพูดบนอุปกรณ์ FPGA แสดงในรูปที่ 11 โดยอุปกรณ์ต่างๆมีหน้าที่ดังนี้

- A/D ทำการแปลงสัญญาณเสียงพูดเก็บลงใน SDRAM
- หน่วยความจำแบบ SDRAM ใช้เก็บข้อมูลเสียงพูด
- Flash memory ใช้เก็บค่าค่าถ่วงน้ำหนัก
- LEDS (Output port) ใช้แสดงผลลัพธ์ของระบบ
- Switch (Input port) ใช้เลือกการแสดงผลการทำงานของระบบ

ในส่วนของ MicroBlaze ที่สร้างขึ้นภายในอุปกรณ์ FPGA ได้กำหนดลักษณะพิเศษเพื่อเพิ่มประสิทธิภาพให้กับระบบ ดังต่อไปนี้



รูปที่ 11 ระบบรู้จำเสียงพูดบนอุปกรณ์ FPGA

• Barrel Shifter เป็นวงจรดิจิตอลที่สามารถเลื่อนข้อมูลตามจำนวนบิตที่กำหนดได้

• Floating-Point Unit (FPU) เป็นวงจรที่ออกแบบมาเพื่อปฏิบัติการทางคณิตศาสตร์ (การบวก ลบ คูณ หาร และเปรียบเทียบ) ของตัวเลขทศนิยม

• Integer Multiplier เป็นวงจรคูณเลขจำนวนเต็ม ช่วยเพิ่มประสิทธิภาพของระบบ

• Pattern comparator เป็นกลุ่มคำสั่งที่ใช้ในการเปรียบเทียบค่าใน register

การสร้างระบบรู้จำเสียงพูดภายใน MicroBlaze นั้นค่อนข้างยุ่งยากกว่าระบบที่สร้างจากโปรแกรม MATLAB เนื่องจากต้องคำนึงถึงพื้นที่ของหน่วยความจำซึ่งใช้เก็บข้อมูลที่นำมาประมวลผล โดยข้อมูลเป็นตัวเลขทศนิยมแบบ single precision (ข้อมูลขนาด 32 บิต) ดังนั้น เพื่อลดขนาดของทรัพยากรที่ต้องใช้ในระบบและเพิ่มความเร็วในการทำงาน จึงต้องเปลี่ยนจากการเขียนโปรแกรมภาษา C เพื่อคำนวณหาค่าโดยตรงเป็นการสร้างตาราง (Look-up Table) ที่มีข้อมูลจากการคำนวณไว้ก่อนแล้วใน 3 ขั้นตอน คือ

• การคำนวณค่าของฟังก์ชั่นหน้าต่างแบบแฮมมิ่ง

• การคำนวณค่าของฟังก์ชั่น cos และ sin สำหรับใช้คำนวณ FFT 256 จุด

• การคำนวณค่าของฟังก์ชันตัวกรองความถี่แบบเมลในแต่ละช่อง

สำหรับ DCT เพื่อให้การคำนวณเร็วขึ้น จึงเขียนโปรแกรมภาษา C จากแผนภาพการคำนวณ Fast DCT 8 จุด [11] ส่วนการทำงานของระบบรู้จำเสียงพูดในขั้นตอนอื่นๆ ออกแบบโดยการเขียนโปรแกรมภาษา C ทั้งหมด

4. การทดลอง

4.1 ข้อมูลที่ใช้ในการทดลอง

ข้อมูลที่ใช้ในการทดลองได้จากการเก็บตัวอย่างเสียงพูดคำโดดภาษาไทยของผู้พูดจำนวน 10 คน ในสภาพแวดล้อมที่ไม่มีเสียงรบกวน (Signal to Noise Ratio > 10 dB) โดยผู้พูดแต่ละคนทำการบันทึกเสียง 20 ครั้ง หลังจากนั้นแบ่งข้อมูลเสียงพูดออกเป็น 2 กลุ่ม กลุ่มแรก คือ กลุ่มตัวอย่าง (Training set) ที่ใช้ในการฝึกโครงข่าย เป็นเสียงคำพูด 10 ครั้ง จำนวน 700 คำ (10 คน x 10 ครั้ง x 7 คำ) โดยเสียงพูดในกลุ่มนี้จะถูกดึงค่าลักษณะสำคัญแล้วปรับจำนวนให้เท่ากัน (ช่วงต้นและช่วงท้าย) ก่อนจะบันทึกไว้เพื่อใช้เป็นฐานข้อมูลสำหรับฝึกโครงข่ายทั้ง 2 โครงข่าย และกลุ่มที่สอง คือ กลุ่มทดสอบ เป็นเสียงคำพูดอีก 10 ครั้งที่เหลือ จำนวน 700 คำ ใช้ในการทดสอบระบบรู้จำแบบขึ้นกับผู้พูด (Speaker-dependent)

นอกจากนั้นยังมีจากการเก็บตัวอย่างเสียงพูดของผู้พูดอีก 5 คน โดยผู้พูดในกลุ่มนี้จะทำการบันทึกเสียง 10 ครั้ง จำนวน 350 คำ เพื่อใช้ในการทดสอบระบบรู้จำแบบไม่ขึ้นกับผู้พูด (speaker-independent)

4.2 การฝึกโครงข่ายประสาทเทียมโดยใช้โปรแกรม MATLAB

เริ่มจากการอ่านข้อมูลที่ใช้ในการฝึกโครงข่ายแต่ละโครงข่ายซึ่งมีจำนวน 125 x 700 ค่า จากนั้นสร้างโครงข่ายประสาทเทียมแบบ Feed-forward แล้วนำข้อมูลทั้งหมดป้อนเป็นข้อมูลอินพุทของโครงข่ายซึ่งมีชั้นซ่อนตัว 3 ชั้น โดยในแต่ละชั้น ข้อมูลจะถูกส่งผ่านฟังก์ชันกระตุ้นแบบซิกมอยด์ (Sigmoid) จนถึงชั้นเอาท์พุต เพื่อหาค่าผิดพลาดระหว่างข้อมูลในชั้นเอาท์พุตกับค่าที่ต้องการ (ค่าที่ต้องการคือ ค่าที่กำหนดให้เป็นตัวแทนของรูปแบบข้อมูลที่ป้อนเข้ามา) โดยเลือกใช้ฟังก์ชัน trainscg (Scaled conjugate gradient algorithm) ในการฝึกฝนเพื่อปรับค่าถ่วงน้ำหนักของโครงข่าย เมื่อค่าเอาท์พุตจริงของโครงข่ายเข้าใกล้ค่าที่ต้องการ (ค่าผิดพลาดลดลงต่ำกว่าค่าที่กำหนดไว้) จะหยุด

ตารางที่ 1 ประสิทธิภาพการรู้จำแบบขึ้นกับผู้พูด

ครั้งที่	อัตราการรู้จำแบบขึ้นกับผู้พูด (%) ... mse สำหรับการฝึกโครงข่าย		
	0.00001	0.0001	0.001
1	99.14	99.57	99.57
2	99.28	99.28	99.85
3	99	99.14	99.28
เฉลี่ย	99.14	99.33	99.57

ตารางที่ 2 ประสิทธิภาพการรู้จำแบบไม่ขึ้นกับผู้พูด

ครั้งที่	อัตราการรู้จำแบบไม่ขึ้นกับผู้พูด (%) ... mse สำหรับการฝึกโครงข่าย		
	0.00001	0.0001	0.001
1	85.42	90.28	91.42
2	87.14	92.57	91.14
3	86.85	89.42	93.42
เฉลี่ย	86.47	90.76	91.99

การฝึกแล้วบันทึกค่าถ่วงน้ำหนัก (Weight & bias) ของโครงข่ายเก็บไว้

ตัวแปรที่ใช้ในการฝึกโครงข่ายประสาทเทียม คือ ค่าความผิดพลาดต่ำสุดที่ต้องการ โดยปกติจะเป็นค่าความผิดพลาดเฉลี่ยของโหนดในชั้นเอาท์พุต (Mean squared error: mse) และจำนวนรอบ (Epochs) ในการฝึกโครงข่าย โดยใช้จำนวนรอบในการฝึกไม่เกิน 250 รอบ ส่วนค่า Parameter อื่นๆ จะใช้ค่าเริ่มต้นของ Neural Networks Toolbox

4.3 ประสิทธิภาพการรู้จำโดยเฉลี่ย

เนื่องจากประสิทธิภาพการรู้จำของระบบเปลี่ยนแปลงตามค่าถ่วงน้ำหนักที่ได้จากการฝึกโครงข่าย ดังนั้นจึงทำการทดลองหาประสิทธิภาพการรู้จำทั้งแบบขึ้นกับผู้พูดและไม่ขึ้นกับผู้พูด 3 ครั้ง เพื่อหาค่าเฉลี่ยของอัตราการรู้จำที่ค่าการตัดสินใจเท่ากับ 0.01 ซึ่งได้ผลการทดลองดังแสดงในตารางที่ 1 และ 2

4.4 ประสิทธิภาพการรู้จำเมื่อปรับค่าการตัดสินใจ

การทดสอบประสิทธิภาพการรู้จำทั้งแบบขึ้นกับผู้พูดและไม่ขึ้นกับผู้พูดที่ผ่านมาได้กำหนดให้ค่าสูงสุดของโหนดในชั้นเอาท์พุตจะต้องมากกว่าค่าการตัดสินใจ (0.01) จึงเป็นผลการรู้จำที่ถูกต้อง ในบางครั้งค่าสูงสุดที่น้อยกว่าค่าการ

ตารางที่ 3 ประสิทธิภาพการรู้จำแบบขึ้นกับผู้พูดเมื่อมีการปรับค่าการตัดสินใจ

mse	..... 0.00001			..... 0.0001			..... 0.001		
ค่าการตัดสินใจ	0.01	0.1	0.5	0.01	0.1	0.5	0.01	0.1	0.5
คำที่รู้จำไม่ได้	12	21	42	5	26	48	0	14	50
คำที่รู้จำผิดพลาด	0	0	0	2	0	0	1	0	0
คำที่รู้จำถูกต้อง	688	679	658	693	674	652	699	686	650
อัตราการรู้จำแบบขึ้นกับผู้พูด (%)	98.3	97	94	99	96.3	93.1	99.9	98	92.9

ตารางที่ 4 ประสิทธิภาพการรู้จำแบบไม่ขึ้นกับผู้พูดเมื่อมีการปรับค่าการตัดสินใจ

mse	..... 0.00001			..... 0.0001			..... 0.001		
ค่าการตัดสินใจ	0.01	0.1	0.5	0.01	0.1	0.5	0.01	0.1	0.5
คำที่รู้จำไม่ได้	38	73	96	28	65	97	6	47	108
คำที่รู้จำผิดพลาด	6	2	0	13	4	2	26	12	2
คำที่รู้จำถูกต้อง	306	275	254	309	281	251	318	291	240
อัตราการรู้จำแบบไม่ขึ้นกับผู้พูด (%)	87.4	78.6	72.6	88.3	80.3	71.7	90.9	83.1	68.6

ตัดสินใจอาจเป็นคำที่รู้จำได้ถูกต้องแต่ระบบจะรวมคำนั้นเป็นการรู้จำที่ผิดพลาด ดังนั้นในการทดลองนี้จะแยกคำที่มีค่าสูงสุดน้อยกว่าค่าการตัดสินใจซึ่งเป็นคำที่ระบบรู้จำไม่ได้ออกแล้วหาเฉพาะคำที่ระบบรู้จำผิดพลาด จากนั้นทำการเปรียบเทียบประสิทธิภาพการรู้จำเมื่อกำหนดค่าการตัดสินใจ 3 ค่า คือ 0.01, 0.1 และ 0.5 ซึ่งได้ผลการทดลองดังแสดงในตารางที่ 3 และ 4

4.5 วิเคราะห์ผลการทดลอง

จากผลการทดลองในตารางที่ 1 และ 2 พบว่าระบบมีประสิทธิภาพการรู้จำโดยเฉลี่ยสูงสุดเมื่อกำหนดให้หยุดการฝึกโครงข่ายประสาทเทียมที่ค่า mse น้อยกว่า 0.001 เนื่องจากการฝึกโครงข่ายที่กำหนดให้ค่า mse ต่ำ (mse เท่ากับ 0.00001) จะทำให้โครงข่ายปรับค่าถ่วงน้ำหนักให้เหมาะสมกับรูปแบบที่ใกล้เคียงกับกลุ่มตัวอย่างเท่านั้น ทำให้เสียงพูดคำเดียวกันที่มีรูปแบบแตกต่างจากเสียงพูดในกลุ่มตัวอย่างมีโอกาสที่จะถูกปฏิเสธเมื่อค่าผลลัพธ์สูงสุดน้อยกว่าค่าการตัดสินใจ (0.01) ซึ่งทำให้ประสิทธิภาพการรู้จำลดลง ในขณะที่ค่าถ่วงน้ำหนักที่ได้จากการเรียนรู้อย่างพอเหมาะ (mse เท่ากับ 0.001) แม้จะทำให้ค่าผิดพลาดของผลลัพธ์เพิ่มขึ้นและอาจทำให้เกิดการรู้จำผิดพลาด แต่ก็ช่วยให้รูปแบบเสียงพูดที่แตกต่างจากกลุ่มตัวอย่างไม่ถูกปฏิเสธไปมากนัก ดังนั้นประสิทธิภาพการรู้จำโดยเฉลี่ยจึงมีค่าสูง

จากผลการทดลองในตารางที่ 3 และ 4 พบว่า การเพิ่มค่าการตัดสินใจมีผลทำให้อัตราการรู้จำลดลง เนื่องจากเป็นการกำหนดให้ระบบไม่ยอมรับคำที่มีค่าสูงสุดของโหนดในชั้นเอาท์พุตน้อยกว่าค่าการตัดสินใจ ทำให้เสียงพูดบางคำที่รู้จำได้ถูกต้องอาจถูกระบบปฏิเสธไปด้วย แต่ในขณะเดียวกันก็สามารถลดจำนวนคำที่ระบบรู้จำผิดพลาดลงได้

เมื่อพิจารณาการรู้จำแบบขึ้นกับผู้พูดพบว่า ประสิทธิภาพการรู้จำค่อนข้างสูง คือ มากกว่าร้อยละ 90 ในทุกการทดลอง ทั้งนี้อาจเป็นผลมาจากการที่ผู้พูดแต่ละคนบันทึกเสียงในช่วงเวลาที่ต่อเนื่องกัน ทำให้ข้อมูลที่ใช้ฝึกโครงข่ายและข้อมูลที่ใช้ทดสอบมีความแตกต่างกันไม่มากนัก ทำให้มีจำนวนคำที่รู้จำผิดพลาดน้อย สำหรับค่าการตัดสินใจที่เหมาะกับการรู้จำแบบขึ้นกับผู้พูด คือ 0.01 เนื่องจากสามารถปฏิเสธคำที่ระบบรู้จำผิดพลาดได้ส่วนหนึ่งและไม่ทำให้อัตราการรู้จำลดลงมากนัก

สำหรับการรู้จำแบบไม่ขึ้นกับผู้พูดพบว่า ประสิทธิภาพการรู้จำค่อนข้างดี คือ ร้อยละ 90.9 เมื่อกำหนดให้หยุดการฝึกโครงข่ายที่ค่า mse น้อยกว่า 0.001 และกำหนดให้ค่าการตัดสินใจเท่ากับ 0.01 แต่จะเห็นว่าจำนวนคำที่รู้จำผิดพลาดค่อนข้างมาก เมื่อเทียบกับประสิทธิภาพการรู้จำเมื่อกำหนดให้หยุดการฝึกโครงข่ายที่ค่า mse น้อยกว่า 0.00001 ที่ค่าการตัดสินใจค่าเดียวกัน ซึ่งมีอัตราการรู้จำใกล้เคียงกัน คือ ร้อยละ 87.4 แต่จำนวนคำที่รู้จำผิดพลาดน้อยกว่า สาเหตุมาจากค่าถ่วงน้ำหนักที่ได้จากการฝึกโครงข่ายที่กำหนดค่าผิดพลาดไว้ต่ำ (0.00001) มีโอกาสทำให้ระบบได้ผลลัพธ์ที่แม่นยำมากกว่าการฝึกโครงข่ายที่กำหนดค่าผิดพลาดไว้สูง (0.001)

ดังนั้น ในการสร้างค่าถ่วงน้ำหนักให้กับระบบรู้จำเสียงพูดบนอุปกรณ์ FPGA ได้กำหนดให้หยุดการฝึกโครงข่ายที่ค่า mse น้อยกว่า 0.00001 และกำหนดค่าการตัดสินใจเท่ากับ 0.01 เพราะระบบสามารถปฏิเสธการรู้จำที่ผิดพลาดได้และเหลือจำนวนคำที่รู้จำผิดไม่มากนัก โดยมีอัตราการรู้จำไม่ต่ำเกินไป

5. สรุปผลการวิจัย

งานวิจัยนี้ได้ออกแบบระบบรู้จำเสียงพูดบนอุปกรณ์ FPGA โดยใช้ Soft Processor ตระกูล MicroBlaze รวมกับการสร้างตารางข้อมูลขึ้นภายในอุปกรณ์ FPGA เพื่อใช้เป็นข้อมูลแทนการคำนวณค่าต่างๆเช่น ค่าของฟังก์ชันหน้าต่างแบบแฮมมิ่งในขั้นตอนการวางกรอบหน้าต่าง การคำนวณค่าของฟังก์ชัน cos และ sin ในขั้นตอนการคำนวณ FFT และการคำนวณค่าของฟังก์ชันตัวกรองความถี่ในขั้นตอนการคำนวณสเปคตรัมกำลังแบบเมล ซึ่งช่วยลดภาระการคำนวณของ MicroBlaze ทำให้ระบบมีประสิทธิภาพการทำงานเพิ่มขึ้น โดยแลกกับการเสียพื้นที่ logic ของ FPGA

ระบบรู้จำเสียงพูดนี้สร้างขึ้นใน Xilinx FPGA ตระกูล Spartan-3E เบอร์ XC3S500E ซึ่งความถี่ในการทำงานของ MicroBlaze อยู่ที่ 75 MHz โดยใช้ทรัพยากรในการสังเคราะห์วงจร ดังนี้

• จำนวนของ Slices ใช้ไป 2,912 จาก 4,656 คิดเป็น 62%

• จำนวนของ Slices Flip Flops ใช้ไป 1,938 จาก 9,312 คิดเป็น 20%

• จำนวนของ 4 input LUTs ใช้ไป 4,363 จาก 9,312 คิดเป็น 46%

• จำนวนของ MULT18X18SIOs ใช้ไป 7 จาก 20 คิดเป็น 35%

• จำนวนของ Block RAMs ใช้ไป 16 จาก 20 คิดเป็น 80%

ประสิทธิภาพการรู้จำของระบบ พบว่าการรู้จำแบบขึ้นกับผู้พูดได้ผลการรู้จำเฉลี่ยร้อยละ 99.57 ส่วนการรู้จำแบบไม่ขึ้นกับผู้พูดได้ผลการรู้จำเฉลี่ยร้อยละ 91.99

ดังนั้นระบบรู้จำเสียงพูดนี้ สามารถนำไปประยุกต์ใช้ในงานต่างๆ ได้หลากหลาย โดยเฉพาะงานที่เกี่ยวกับระบบสมองกลฝังตัว ซึ่งข้อได้เปรียบของ FPGA ทำให้เกิดแนวทางในการพัฒนาระบบที่ไม่ตายตัว สามารถปรับเปลี่ยนการทำงานได้ โดยมีข้อจำกัดอยู่ที่จำนวน logic block ของอุปกรณ์ FPGA

6. เอกสารอ้างอิง

[1] Pong P. Chu, FPGA Prototyping By Verilog Examples: Xilinx Spartan-3 Version, John Wiley & Sons, Inc.,2008.

[2] สมชาย จิตพันธ์กุล, การรู้จำเสียงพูดภาษาไทยระยะที่หนึ่ง: การรู้จำเสียงพูดคำไทยโดดๆ โดยไม่ขึ้นกับผู้พูด, โครงการวิจัย, สถาบันวิจัยและพัฒนาของคณะวิศวกรรมศาสตร์, จุฬาลงกรณ์มหาวิทยาลัย, 2540.

[3] ไชยันต์ สุวรรณชีวะสิริ, การรู้จำเสียงพูดตัวเลขภาษาไทยแบบขึ้นกับผู้พูดด้วยนิวรัลเน็ตเวอร์ค, วิทยานิพนธ์ปริญญาโท, สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ, 2540.

[4] กาญจนา ทองบุญนาค, การรู้จำเสียงคำโดดด้วยโครงข่ายประสาทเทียม, วิทยานิพนธ์ปริญญาโท, มหาวิทยาลัยเชียงใหม่, 2544.

[5] ฐนียา สัตยพานิช, ระบบรู้จำเสียงภาษาไทยต่อเนื่องแบบเฉพาะบุคคลสำหรับการใช้งานอีเมลล์, วิทยานิพนธ์ปริญญาโท, มหาวิทยาลัยเกษตรศาสตร์, 2546.

[6] เอกรินทร์ แซ่เฮ้ง, การประยุกต์ใช้ระบบรู้จำเสียงพูดคำไทยสำหรับงานพิมพ์เอกสารโดยใช้เทคนิควิเคราะห์สเปกตรัมและโครงข่ายประสาทเทียม, วิทยานิพนธ์ปริญญาโท, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2547.

[7] N. Sae-ngan and N. Buabthong, “Improvement of Endpoint Detection for Thai Isolated Word Recognition,” International Conference on ECTI-CON 2008, Vol. 1, pp. 537-540, May 2008.

[8] S.E. Bou-Ghazale and K. Assaleh, “A Robust Endpoint Detection of Speech for Noisy Environments with Application to Automatic Speech Recognition,” Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing 2002, Vol. 4, pp. 3808-3811, May 2002.

[9] Xilinx, Inc, MicroBlaze Processor Reference Guide, from http://www.xilinx.com/support/documentation/sw_manuals/edk81i_mb_ref_guide.pdf

[10] R. Jesman, F. M. Vallina and J. Saniie, MicroBlaze Tutorial Creating a Simple Embedded System and Adding Custom Peripherals Using Xilinx EDK Software Tools, from http://ecasp.ece.iit.edu/mbtutorial.pdf

[11] M.E. Rizkalla, M. Ei-Sharkawy, P. Salama and B. Dukel, “Implementation of Floating Point Fast Discrete Cosine Transform,” The 2002 45th Midwest Symposium on Circuits and Systems, Vol. 2, pp. 17-20, Aug. 2002.

[12] L. Rabiner and R. Schafer, Introduction to Digital Speech Processing, from http://www.nowpublishers.com/getpdf.aspx?doi=2000000001&product=SIG