บทที่ 6

สรุปและข้อเสนอแนะ

6.1 สรุปผลงานวิทยานิพนธ์

วิทยานิพนธ์ฉบับนี้ได้พัฒนาโปรแกรมการให้กำเนิดรูปแบบสัญญาณทดสอบโดยอัตโนมัติ โดยใช้วิธีรายการลดทอน สำหรับวงจรดิจิตอล ซึ่งสามารถทดสอบวงจรได้ทั้งวงจรประกอบ และวงจรลำดับ ตามวัตถุประสงค์

จากผลการทดสอบวงจรประกอบ มีค่าครอบคลุมจุดเสียอยู่ในช่วง 95-99% จุดเสียบางจุดไม่สามารถทดสอบได้ เพราะว่าจุดเสียอยู่ในส่วนของวงจรที่เป็น Redundancy ในผลการทดสอบของ [14] ทำให้ไม่สามารถทดสอบหารูปแบบสัญญาณได้ ส่วนเวลาที่ใช้ในการทดสอบน้อยมาก ในทดสอบวงจรประกอบทั้งหมด 10 วงจรใช้เวลาไปทั้งหมดเท่ากับ 39.849 วินาที ผลการทดสอบเปรียบเทียบกับที่ใช้หลักวิธี PODEM เป็นพื้นฐาน จำนวนรูปแบบสัญญาณ ค่าครอบคลุมจุดเสีย และเวลาที่ใช้ในการทดสอบของ ATOM (1998) กับ หลักวิธีรายการลดทอนมีค่าใกล้เคียงกัน

สำหรับวงจรลำดับ มีค่าครอบคลุมจุดเสียสูงสุด 98% ต่ำสุด 4% เพราะว่าวงจรนี้มีจำนวนฟลิปฟลอปมากถึง 32 ตัว และมีข้อจำกัดอยู่ที่กรอบเวลาในการทดสอบจุดเสียแต่ละจุด ถ้าตั้งจำนวนกรอบเวลาไว้มากจะทำให้ใช้เวลามากในการทดสอบจุดเสีย และต้องการใช้หน่วยความจำของเครื่องมากขึ้นด้วยตามลำดับ แต่โอกาสที่จะทดสอบจุดเสียได้ก็มีมาก (หลักวิธีรายการลดทอนตั้งกรอบเวลาไว้ที่ 10) เวลาทั้งหมดที่ใช้ในการทดสอบวงจรลำดับจำนวน 8 วงจรเท่ากับ 28.14 นาที เมื่อนำผลการทดสอบไปเปรียบเทียบ กับ STPG (1997) ที่ใช้หลักวิธี PODEM ผลการทดสอบใกล้เคียงกัน

หลักวิธีกำเนิดรูปแบบสัญญาณทดสอบโดยอัตโนมัติ โดยใช้วิธีรายการลดทอน สำหรับวงจรดิจิตอล ก็เป็นหลักวิธีการหนึ่งที่สามารถให้กำเนิดรูปแบบสัญญาณทดสอบโดยอัตโนมัติได้

6.2 ปัญหาและข้อเสนอแนะ

ปัญหาหลักที่สำคัญที่พบก็คือ ตัวระบบปฏิบัติการ UNIX และตัวแปลภาษาซี เช่น ถ้าเป็นเครื่อง SUN รุ่น SPARCstation IPX, II, 5 ไม่สามารถแปลโปรแกรมได้ผ่านทั้งหมด เพราะว่าบางฟังก์ชัน คำสั่งมีไม่เหมือนกันของเครื่องแต่ละรุ่น ถ้าเป็นเครื่อง PC ใช้ระบบปฏิบัติการ LINUX ความเร็ว 200MHz ขึ้นไป หน่วยความจำไม่น้อยกว่า 128 MB ใช้ได้ไม่มีปัญหา

การพัฒนาโปรแกรมเอทีพีจี สามารถพัฒนาได้บนเครื่อง SUN, HP, PC ทำงานภายใต้ระบบปฏิบัติงานที่เป็น UNIX หรือ LINUX (สำหรับเครื่อง PC) ภาษาที่ใช้ คือ ภาษาซี หรือ C++ เป็นพื้นฐาน แต่มีอีกภาษาหนึ่งใช้เขียนโปรแกรมเอทีพีจีได้ คือ VHDL ทำงานได้ทั้งระบบ UNIX และ WINDOWS

ตัวระบบ UNIX มีการทำงานแบบหลายงาน และหลายโพรเซสเซอร์พร้อมกัน การพัฒนาโปรแกรมเอทีพีจีควรจะดึงเอาความสามารถอันนี้มาใช้งาน ตัวอย่างเช่น โปรแกรมเอทีพีจีกับโปรแกรมจำลองการทำงานจุดเสียสามารถทำงานพร้อมกัน เมื่อได้รูปแบบสัญญาณทดสอบมา 1 รูปแบบ โปรแกรมจำลองการทำงานจุดเสียเริ่มทำงาน โปรแกรมเอทีพีจีไม่จำเป็นต้องรอให้โปรแกรมจำลองการทำงานจุดเสียทำงานให้เสร็จก่อน สามารถทำงานต่อได้เลย โดยใช้หลักวิธีการเขียนโปรแกรมแบบขนาน (Parallel Programming) และต้องศึกษาระบบปฏิบัติการ UNIX (UNIX Operating System) การจัดการโปรเซสในระบบ UNIX ก็จะช่วยให้การทำงานของโปรแกรมเอทีพีจีมีประสิทธิภาพมากยิ่งขึ้น

มหาวิทยาลัยเชียงใหม่ Chiang Mai University