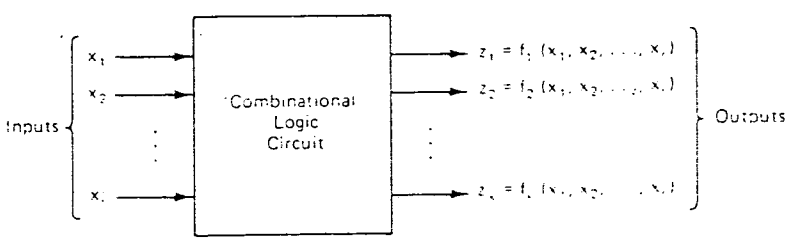
บทที่ 2

แนวคิดทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 บทนำ

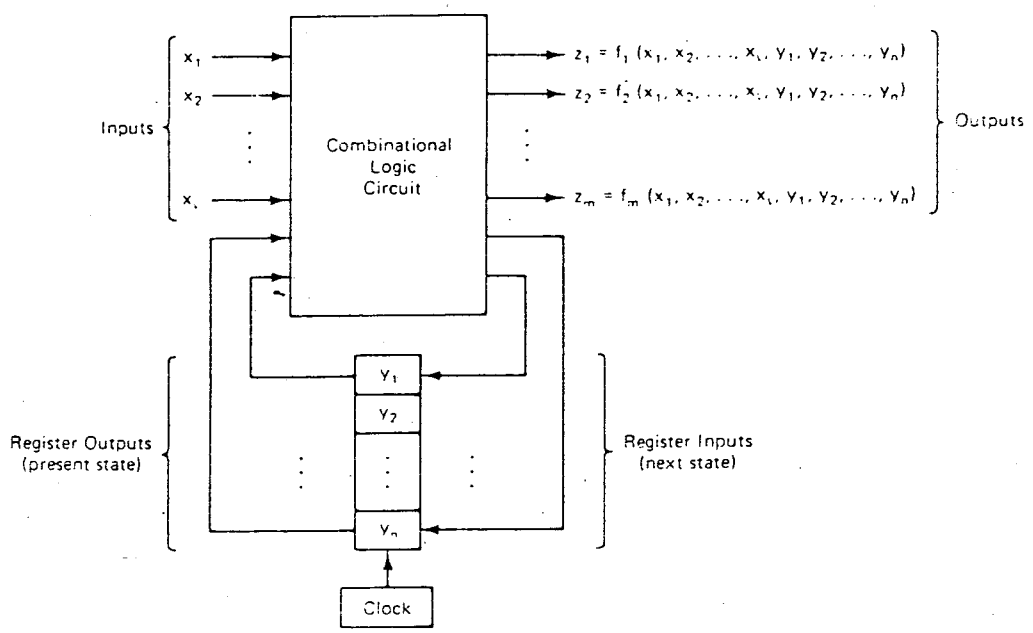
PLD (Programmable Logic Device) คือ อุปกรณ์ที่ใช้ในวงจรทางด้านลอจิกที่ผู้ใช้สามารถเลือกโปรแกรมโครงแบบ (Configuration) ได้ตามต้องการ โดยที่โครงสร้างภายในจะประกอบไปด้วยเกตแบบแอนด์ และเกตแบบออร์บนชิปหนึ่งตัว การใช้งานจะทำการโปรแกรมเกตแบบแอนด์ หรือเกตแบบออร์ หรือโปรแกรมทั้งด้านเกตแบบแอนด์และเกตแบบออร์ เพื่อให้วงจรลอจิกเชิงผสม (Combinational Logic) ทำงานตามที่ได้ออกแบบไว้ นอกจากนี้ยังมี PLD บางตัวมีเรจิสเตอร์ (Register) ซึ่งโครงสร้างภายในนอกจากจะมีเกตแบบแอนด์ และเกตแบบออร์แล้วยังมีฟลิปฟลอป (Flip-flop) ซึ่งจะช่วยในการออกแบบวงจรลอจิกแบบเชิงลำดับ (Sequential Logic) ดังนั้นในการใช้งานวงจรทางลอจิก จึงเป็นข้อดีในด้านความน่าเชื่อถือสูง และยังช่วยลดต้นทุนในการผลิตอีกด้วย [5]

จากรูปที่ 2.1 (ก) แสดงแผนภาพบล็อกของวงจรลอจิกเชิงผสม ผลของเอาต์พุตจะขึ้นอยู่กับฟังก์ชันทางด้านอินพุต การเขียนผลจะเขียนอยู่ในรูปของสมการบูลีน ซึ่งเป็นผลของวงจรแอนด์-ออร์ อยู่ในรูปแบบผลบวกของผลคูณ (Sum Of Product) ส่วนรูปที่ 2.1 (ข) แสดงแผนภาพบล็อกของ วงจรเชิงลำดับผลของเอาต์พุตที่มาจากวงจรเชิงลำดับนี้ขึ้นกับฟังก์ชันของอินพุตในสถานะปัจจุบัน (Present State) และสถานะต่อไป (Next State) โดยมีสัญญาณนาฬิกา (Clock) เป็นตัวกำหนดการทำงานของฟลิปฟลอป



(ก)

รูปที่ 2.1 (ก) แสดงแผนภาพบล็อกของวงจรลอจิกแบบเชิงผสม (Combinational Logic Circuit)



(ข)

รูปที่ 2.1 (ข) แสดงแผนภาพบล็อกของวงจรลอจิกแบบเชิงลำดับ (Sequential Logic Circuit)

2.2 ทฤษฎีของ PLD

2.2.1 พื้นฐานของ PLD

จากโครงสร้างภายในของ PLD แสดงได้ดังรูปที่ 2.2 (ก) - (ค) แบ่งประเภทของการโปรแกรม PLD เป็นดังนี้

ก. Programmable AND-fixed OR : การโปรแกรมทางด้านแอนด์ ได้แก่

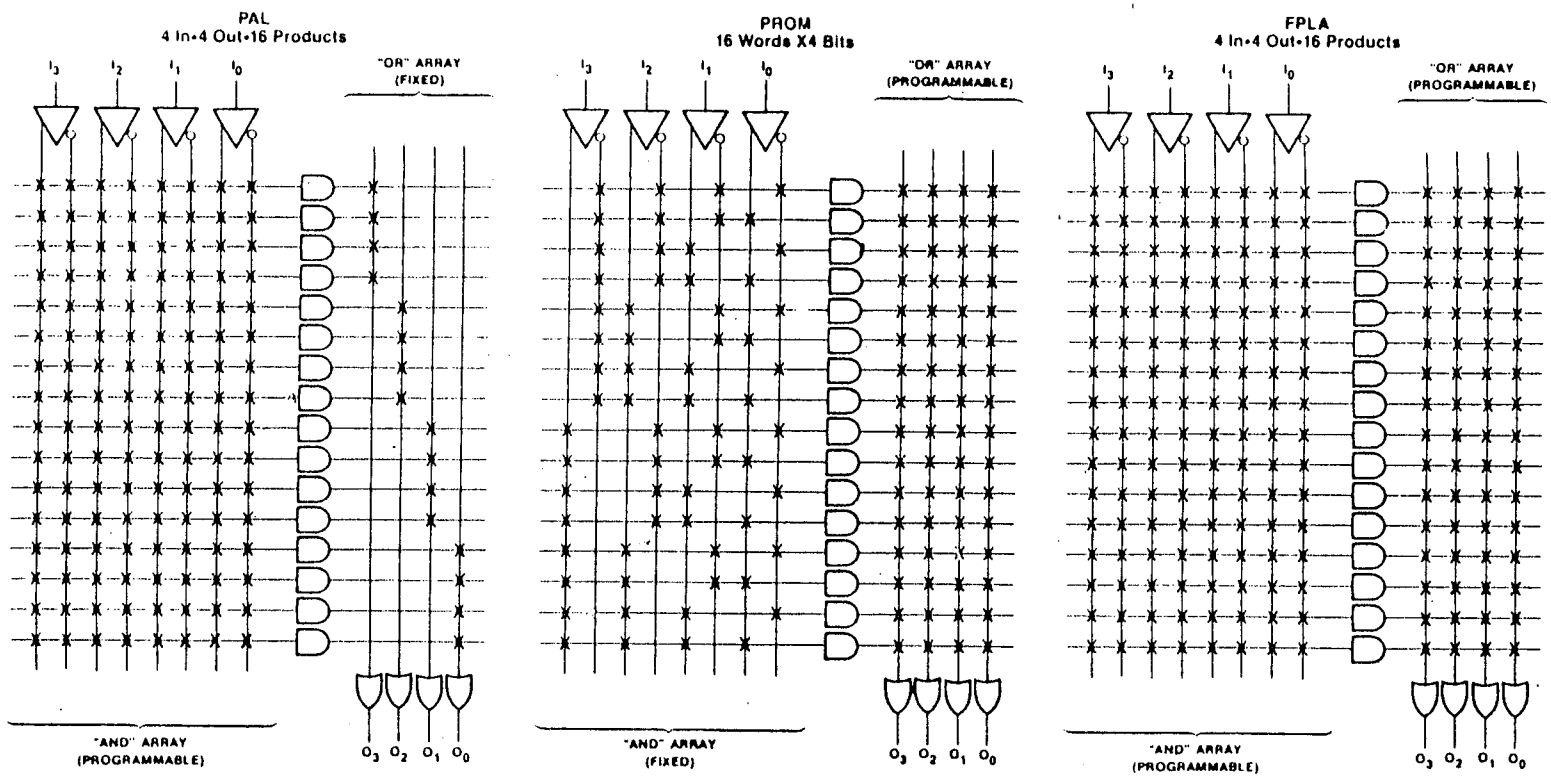
- อุปกรณ์ PAL : Programmable Array Logic
- อุปกรณ์ EPLD : Erasable Programmable Logic Device
- อุปกรณ์ GAL : Gate Array Logic

ข. Fixed AND-programmable OR : การโปรแกรมทางด้านออร์ ได้แก่

- อุปกรณ์ PROM : Programmable Read Only Memory

ค. Programmable AND-programmable OR : การโปรแกรมทั้งด้านแอนด์ และด้านออร์ ได้แก่

- อุปกรณ์ FPLA : Field Programmable Logic Array



(ก) (ข) (ค)

รูปที่ 2.2 แสดงโครงสร้างของ Programmable Logic Device : (ก) Programmable AND-fixed OR; (ข) Fixed AND-programmable OR; (ค) Programmable AND-fexed OR.

2.2.2 Programmable AND-fixed OR : การโปรแกรมทางด้านแอนด์

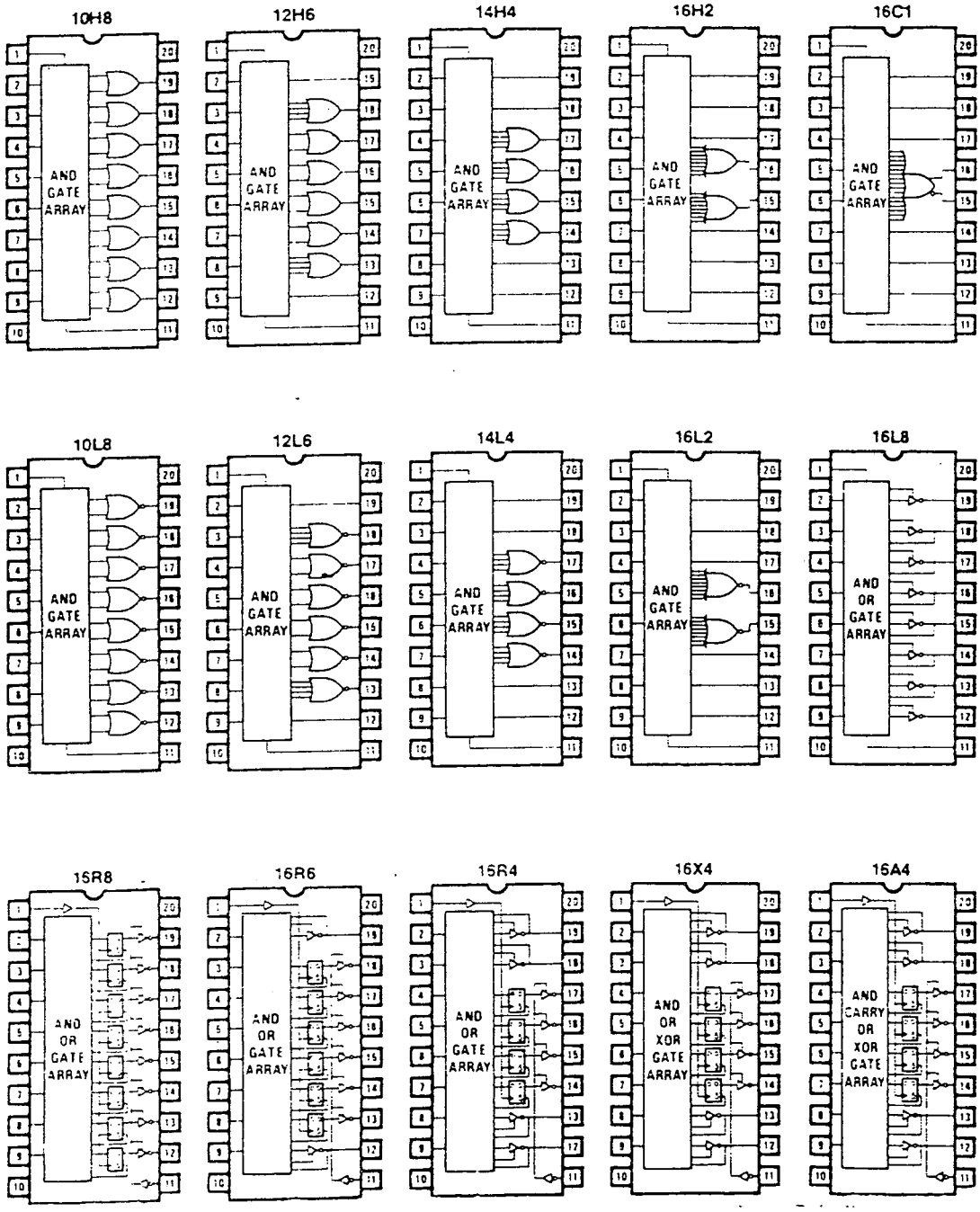
2.2.2.1 อุปกรณ์ PAL

บริษัท MMI (Monolithic Memories Inc.) เป็นต้นกำเนิดในการผลิตอุปกรณ์ที่เรียกว่า Programmable Array Logic (PAL) โครงสร้างภายในจะทำการโปรแกรมทางด้านเกตแบบแอนด์ มีลักษณะที่เป็นทั้งแบบ TTL และ CMOS ประกอบด้วย PAL แบบ 20 ขา ดังแสดงในรูปที่ 2.3 [5] สำหรับเอาต์พุตที่มาจาก PAL จะมีทั้งแบบแอกทีฟสูง, แอกทีฟต่ำ, แบบเติมเต็ม (Complementary), แบบมีการป้อนกลับ และแบบมีเรจิสเตอร์

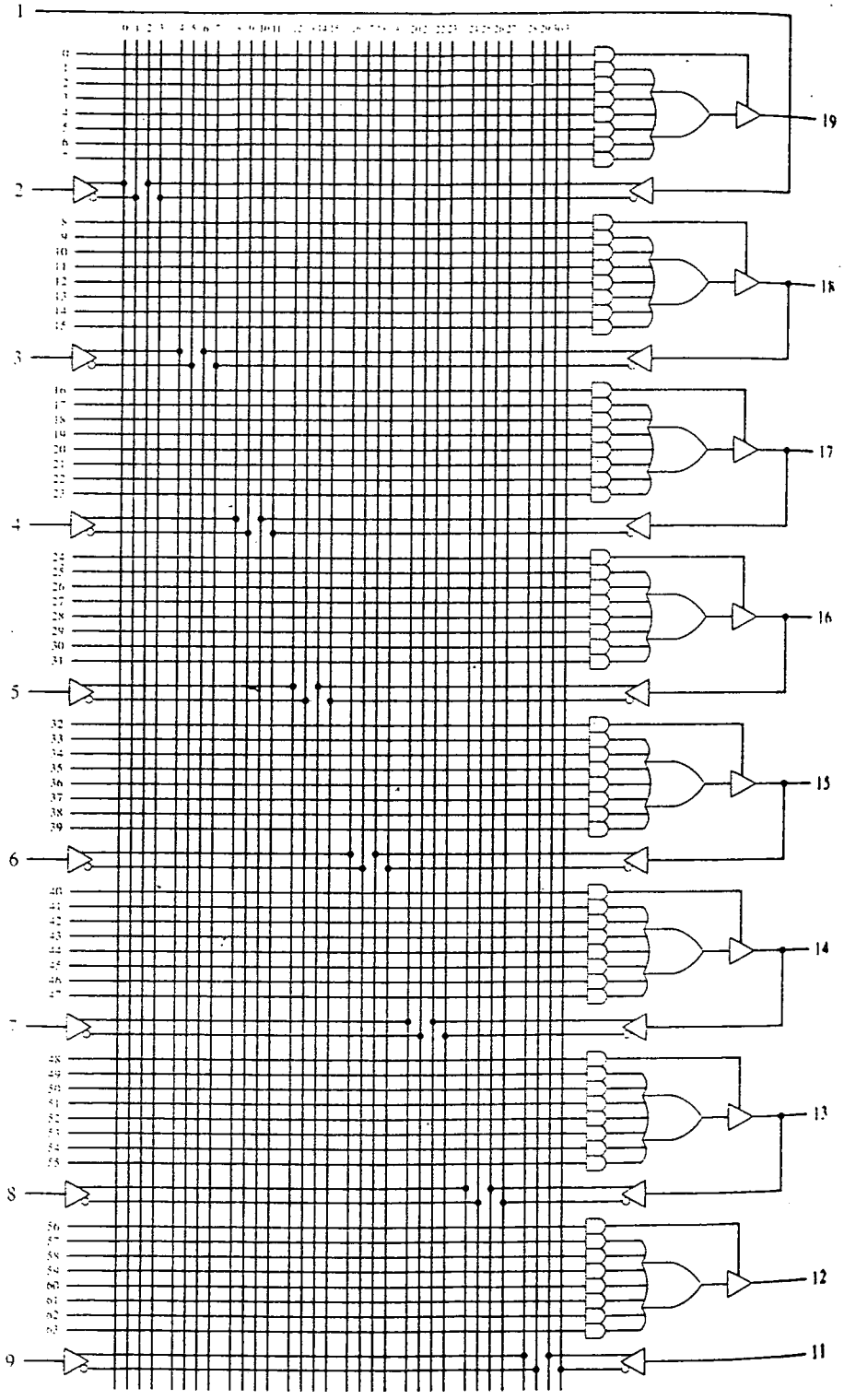
รูปที่ 2.4 แสดงวงจรลอจิกภายในโครงสร้างของ PAL ที่ประกอบด้วย 8 อินพุต 8 เอาต์พุต ส่วนรูปที่ 2.5 แสดงการอธิบายรูปพื้นฐานของวงจรลอจิกที่ตัดมาส่วนหนึ่ง โดยที่ภายในจะมีการต่อถึงกันในแต่ละเส้น เพื่อทำการโปรแกรมระหว่างเส้นในแนวดิ่ง 32 เส้น ซึ่งเราเรียกว่าเส้นอินพุต (Input Line) และเส้นที่อยู่ในแนวระดับของอินพุตจะต่อเข้ากับเกตแบบแอนด์ และเส้นที่ต่อเข้ากับเกตแบบแอนด์ทั้งหมดจะต่อเข้ากับเกตแบบออร์ สำหรับขาอินพุตที่ใช้จะไปต่อกับวงจรภายนอกเรียกว่าเป็นตัวขับ (Driver) โดยที่ขาอินพุตนี้จะถูกกำหนดให้เป็นได้ทั้งระดับสูง (High) และระดับต่ำ (Low) และกรณีขาเอาต์พุตจากรูปจะถูกใช้ให้เป็นได้ทั้งอินพุต และเอาต์พุต เพราะตัวขับเอาต์พุตเป็นแบบสามสถานะ (Tristate) โดยการโปรแกรมผ่านเกตแบบแอนด์ของทางด้านอินพุตของ PAL ถ้าตัวขับเอาต์พุตเปิดทาง (Enable) แล้วทางด้านเอาต์พุตจะทำหน้าที่เป็นเอาต์พุต ถ้าตัวขับเอาต์พุตปิดทาง (Disable) แล้วขาเอาต์พุตจะทำหน้าที่เป็นอินพุตผ่านทางตัวขับทางด้านอินพุต [6]

แผนภาพในรูปที่ 2.4 จะประกอบด้วยอินพุตและเอาต์พุต โดยที่มีขา 2 เป็นอินพุตเหมือนกับการอธิบายในรูปที่ 2.5 รวมทั้งขา 1 ก็จะเป็นอินพุตด้วย และให้ผลเอาต์พุตผ่านทางขา 19 และในกรณีของขา 9 และ 11 ก็จะแสดงเป็นอินพุต และส่งเอาต์พุตผ่านทางขา 12 สรุปแล้ว PAL นี้มีจำนวนของอินพุตทั้งหมด 10 ตัว เป็นแบบการติดต่อจากภายนอกให้ทำหน้าที่เป็นได้ทั้งอินพุตและเอาต์พุตอยู่ 8 ตัว

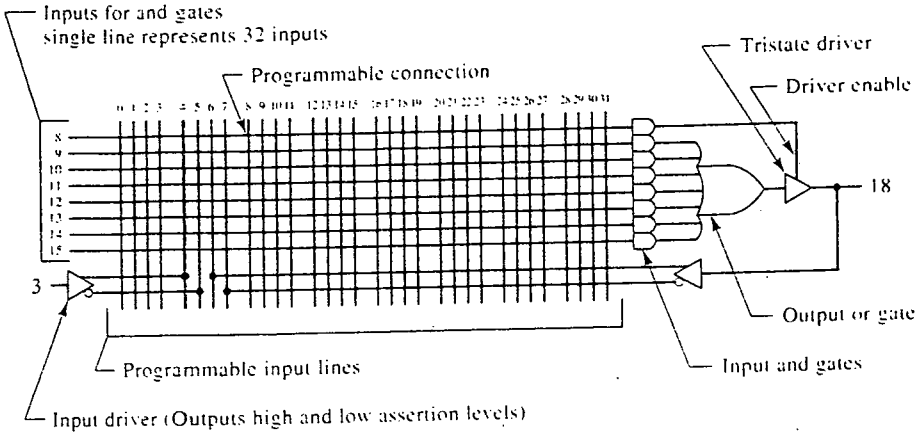
อุปกรณ์ PAL โดยทั่วไปจะมีความแตกต่างกันทั้งจำนวนและขนาดของวงจรลอจิกที่ประกอบอยู่ภายในตัว PAL รวมทั้งโครงแบบของเอาต์พุตด้วย รูปที่ 2.6 (ก)-(ง) แสดงโครงแบบของเอาต์พุตในแบบต่างๆ สำหรับรูปที่ 2.6 (ก) เป็นวงจรที่ได้อธิบายแล้วในตอนต้น แต่ต่างกันตรงที่เอาต์พุตจะมีบัฟเฟอร์เป็นแบบระดับต่ำ กล่าวคือ จะให้ค่ากลับกันกับ



รูปที่ 2.3 แสดง PAL ที่เป็นแบบ 20 ขา

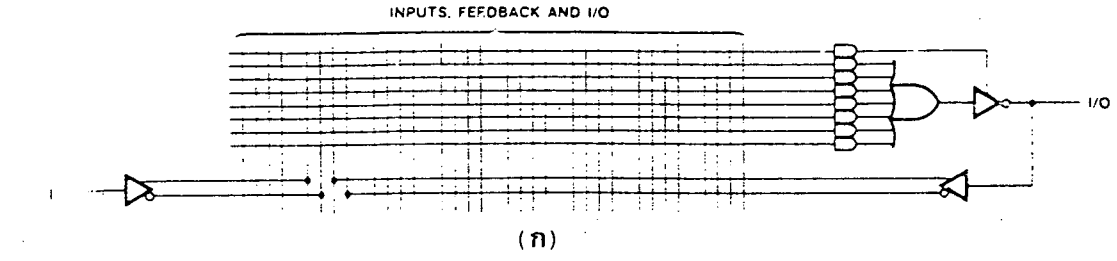


รูปที่ 2.4 แสดงวงจรลอจิกภายในโครงสร้างของ PAL

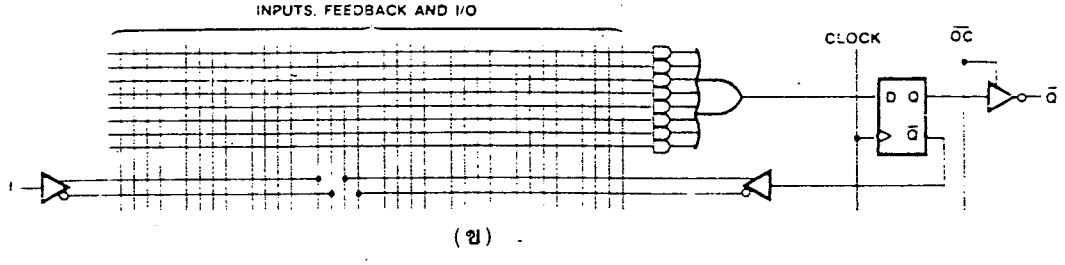


รูปที่ 2.5 แสดงส่วนหนึ่งของรูปวงจรลอจิก PAL

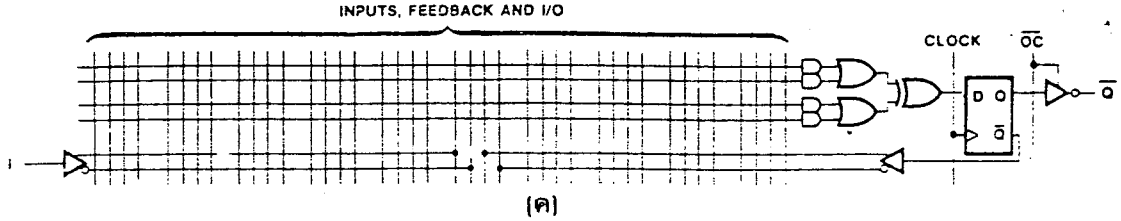
เอาต์พุตที่มีบัฟเฟอร์เป็นแบบระดับสูง ในกรณีของ PAL ที่ประกอบด้วยเรจิสเตอร์ที่เอาต์พุต ดังแสดงในรูปที่ 2.6 (ข) โดยที่มีสัญญาณอินพุต 1 เส้น ควบคุมสัญญาณนาฬิกาซึ่งต่อถึงกันหมด และมีอินพุตอื่นอีก 1 เส้นที่ควบคุมเอาต์พุตแบบสามสถานะของฟลิปฟลอปทั้งหมด เอาต์พุต Q ของฟลิปฟลอปจะถูกป้อนกลับผ่านบัฟเฟอร์ไปที่อินพุต สำหรับเอาต์พุตแบบอื่นๆ ก็จะมีโครงสร้างเป็นแบบเอกคลูซีฟออร์ที่มีเรจิสเตอร์ด้วย ดังแสดงในรูปที่ 2.6 (ค) นอกจากนี้แล้วยังมีโครงสร้าง PAL ที่มีการคำนวณด้วย กล่าวคือ มีการบวก, การลบ มีการเปรียบเทียบมากกว่าน้อยกว่า เพื่อเพิ่มประสิทธิภาพการทำงานของอุปกรณ์ รูปที่ 2.6 (ง) แสดงโครงสร้างของ PAL ที่เป็นแบบการป้อนกลับที่เกตสามารถทำการคำนวณได้ (Arithmatic Gated Feedback)



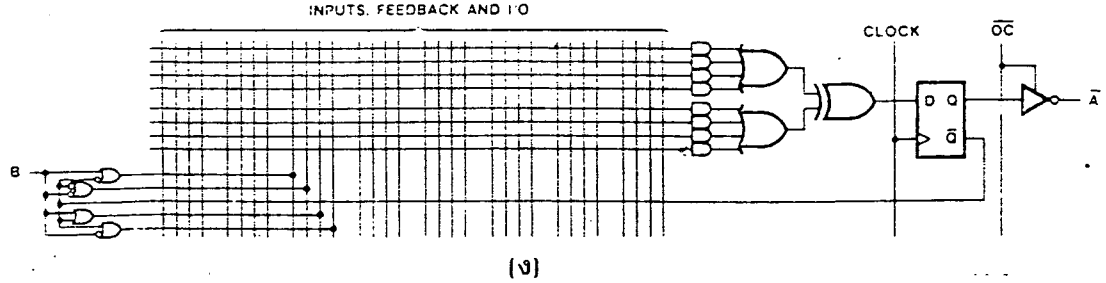
(ก)



(ข)



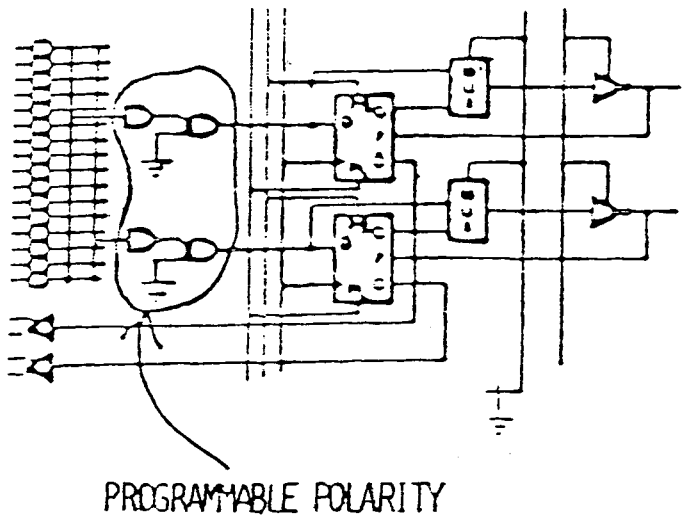
(ค)



(ง)

รูปที่ 2.6 แสดงโครงสร้างของ PAL : (ก) แสดงโครงสร้างที่โปรแกรมทางด้านอินพุต-เอาต์พุต; (ข) ทางด้านเอาต์พุตมีเรจิสเตอร์ด้วย; (ค) โครงสร้างแบบเอกคลูซีฟออร์; (ง) โครงสร้างแบบป้อนกลับที่เกตสามารถทำการคำนวณได้ (Arithmatic Gated Feedback)

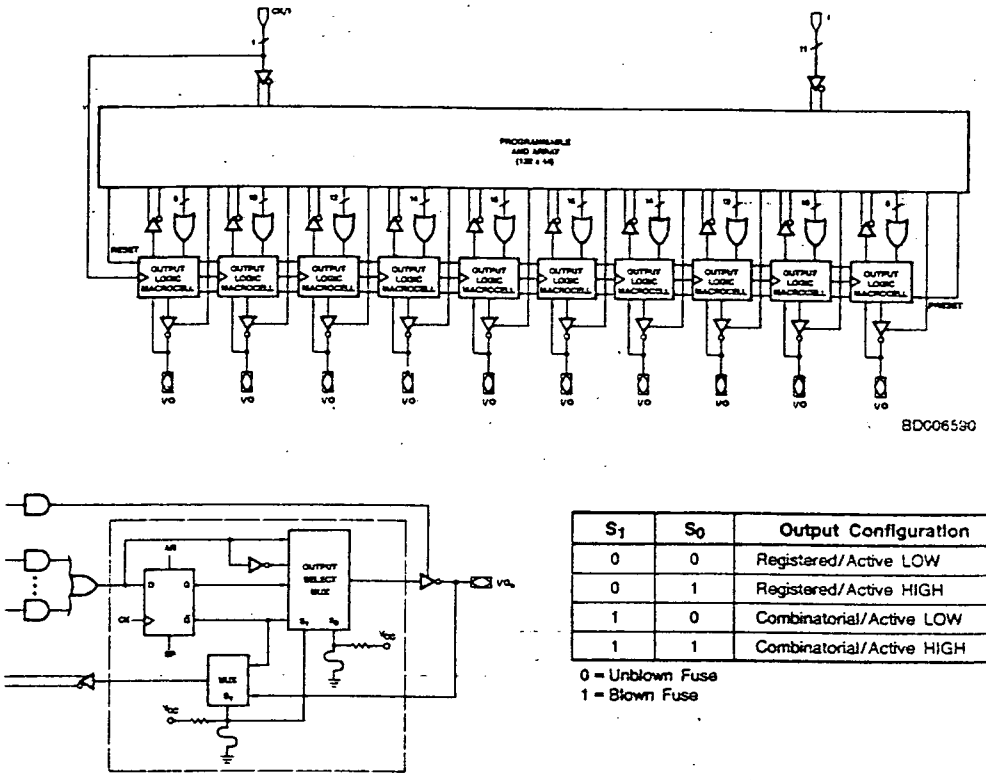
นอกจากคุณสมบัติตามที่ได้กล่าวมาแล้ว ยังมีคุณสมบัติแบบอื่นๆ อีก เช่น เอาต์พุตที่มีขั้ว (Output Polarity) ดังแสดงในรูปที่ 2.7 ในการใช้งานจะมีเอาต์พุต 1 เส้น ต่อลงกราวด์ และยังมี PAL ที่เป็นแบบ 24 และ 40 อีกด้วย



รูปที่ 2.7 แสดงโครงสร้างของ PAL ที่เป็นแบบเอาต์พุตที่มีขั้ว (Output Polarity)

สำหรับอุปกรณ์ PAL ที่ได้พัฒนาต่อมาจะมีโครงสร้างของเอาต์พุตเป็นแบบแมโครเซลล์ (Macrocell) เช่น AmPAL22V10 ที่มี 24 ขา ดังแสดงในรูปที่ 2.8 [5] โครงสร้างภายในของ PAL แบบนี้ประกอบด้วย 22 อินพุต และ 10 อินพุต ซึ่งที่เอาต์พุตนี้จะเลือกให้โปรแกรมให้เป็นแบบที่มีเรจิสเตอร์แอกทีฟต่ำหรือสูงได้ หรือเลือกให้เป็นแบบเชิงผสมแอกทีฟต่ำหรือสูง นั่นคือ ทำให้ผู้ใช้สามารถเลือกที่จะใช้แทน PAL ได้หลายชนิดโดยใช้ PAL22V10 เพียงตัวเดียว มีผลทำให้การออกแบบสะดวกสำหรับผู้ใช้และ ทำให้การออกแบบวงจรไม่ยุ่งยากซับซ้อน

ในปัจจุบันนี้ ได้มีการนำเอาอุปกรณ์ PAL มาใช้งานแทนวงจรลอจิกอย่างกว้างขวาง เนื่องจากว่าภายในโครงสร้างของตัว PAL จะประกอบไปด้วยเกตหลายตัวอยู่ในอุปกรณ์ PAL เพราะฉะนั้นจึงเป็นการลดจำนวนเกตที่ใช้ลงไปได้มาก ทำให้วงจรที่ออกแบบมีขนาดเล็กกะทัดรัด ซึ่งเป็นที่นิยมมากสำหรับนักออกแบบวงจรทางด้านเชิงเลข ต่อมาก็ได้มีการผลิต PAL รุ่นใหม่ที่สามารถทำการลบได้ คือ ทำการโปรแกรมซ้ำได้หลายๆ ครั้ง รวมทั้งยังมีซอฟต์แวร์ช่วยสนับสนุนในการออกแบบอีกด้วย ในการโปรแกรม PAL จะใช้เครื่องโปรแกรม (PAL Programmer) ซึ่งเครื่องนี้จะทำการต่อฟิวส์และการทำให้ฟิวส์ขาด ตามข้อมูลที่เป็นรูปแบบของการโปรแกรม



รูปที่ 2.8 แสดงรูปโครงสร้างภายในของ PAL ที่เป็นแบบแมโครเซลล์ (Macrocells)

2.2.2.2 อุปกรณ์ EPLD

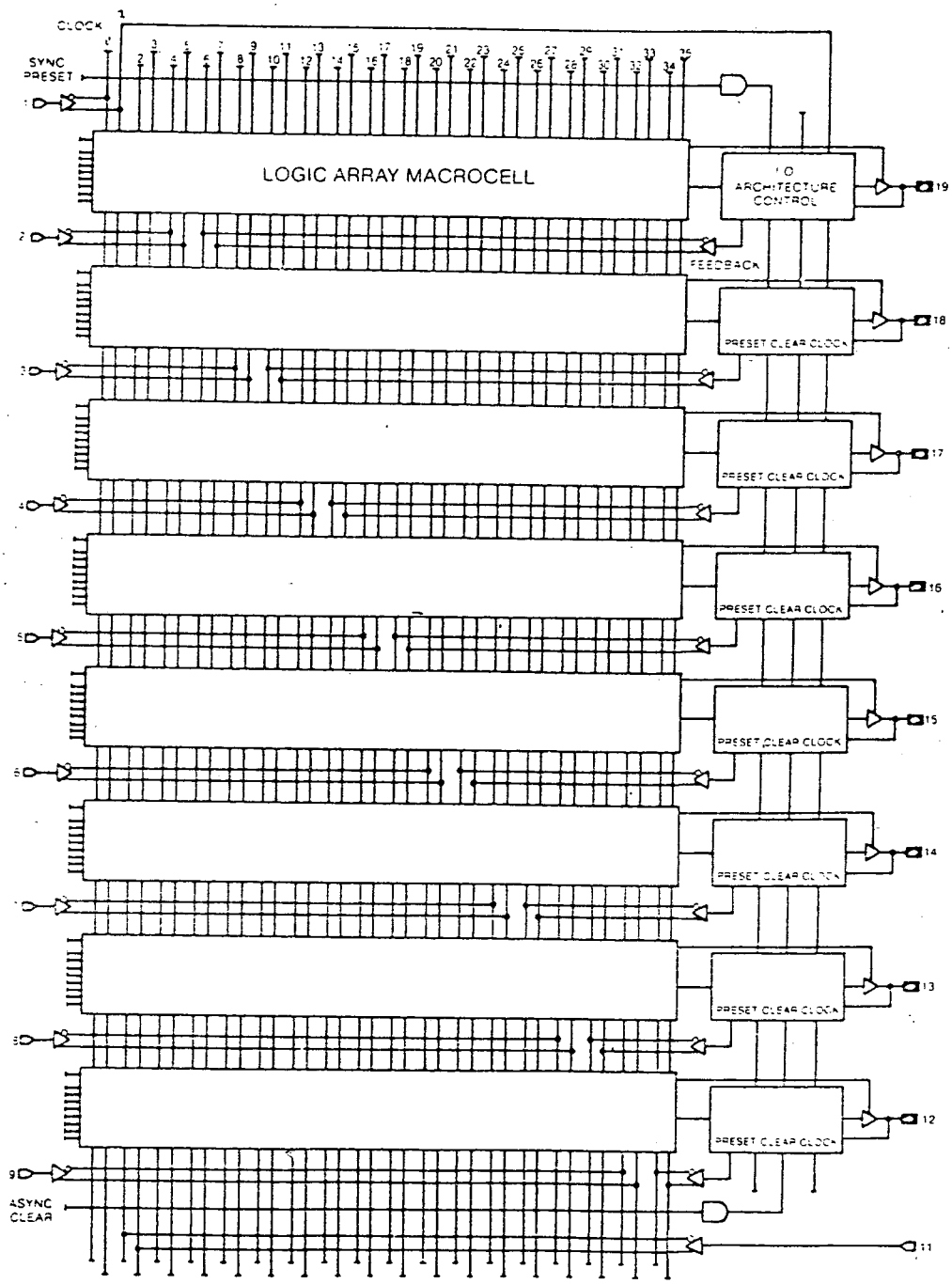
เป็นอุปกรณ์ที่ Programmable AND - fixed OR นอกจากนี้แล้วผู้ใช้ยังสามารถลบโปรแกรมได้โดยการใช้แสงอุลตร้าไวโอเลตเหมือนกับ EPROM บริษัท Altera Corporation ได้เริ่มทำการผลิต EPLD ขึ้นในปี 1984 โดยที่ EPLD ในรุ่นแรกจะเป็นแบบ CMOS 20 ขา และต่อมาก็ทำการผลิตแบบ 24 และ 28 ขา

สำหรับ EPLD ในรุ่นแรกแบบ 20 ขา แสดงไว้ในรูปที่ 2.9 คือ EP300 จะประกอบด้วย 10 อินพุต และทางด้านที่จะทำการโปรแกรมอินพุต-เอาต์พุตมี 8 ตัว [7] EP300 มีประสิทธิภาพการทำงานเท่ากับเกต 300 ตัว สำหรับรายละเอียดโครงสร้างของแมโครเซลล์เพียงตัวเดียวได้แสดงไว้ในรูปที่ 2.10 โดยที่แต่ละแมโครเซลล์จะประกอบด้วยการโปรแกรมทางด้านแอนด์ ส่วนทางด้านเอาต์พุตมาจากเกตแบบออร์ และป้อนกลับไปยังบล็อกที่ควบคุมโครงสร้างภายในของไอโอ (I/O Architecture Control) และการควบคุมการทำงานเป็น

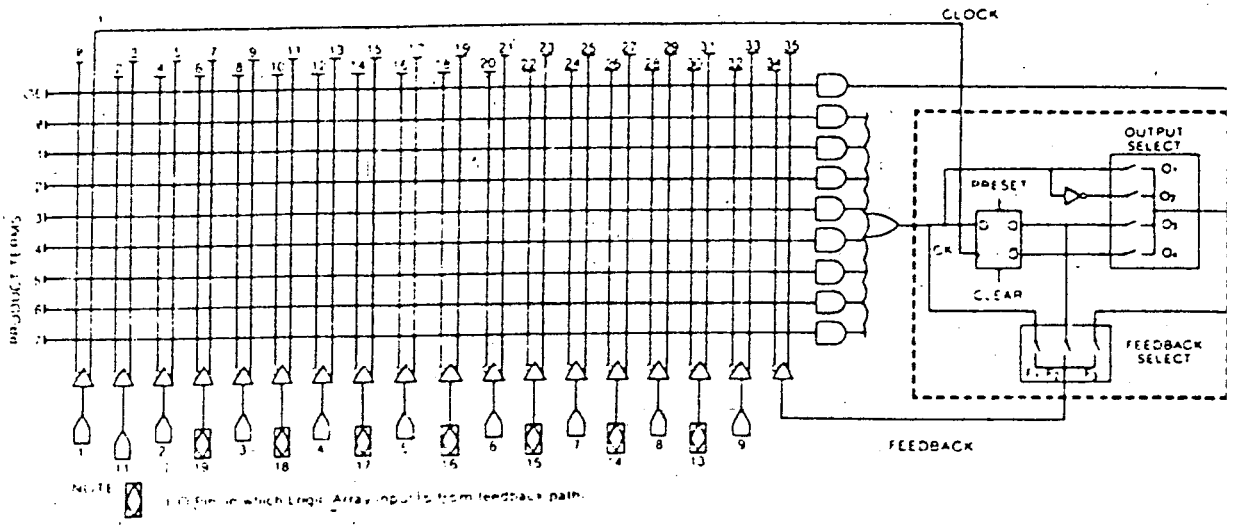
แบบสามสถานะ เพื่อทำการขับขาไอโอของแมโครเซลล์ ขาอินพุตจะเป็นทั้งแบบจริงและแบบเติมเต็ม โดยมีทั้งแบบที่ต่อมาจากข้างนอกและมีการป้อนกลับมาจากบล็อกควบคุมโครงสร้างภายในของไอโอ ส่วนสัญญาณนาฬิกามาจากอินพุตซึ่งมาต่อเข้ากับขาทริกเกอร์ของฟลิปฟลอปแบบดี ซึ่งอยู่ในบล็อกควบคุมโครงสร้างภายในของไอโอในแต่ละแมโครเซลล์จะประกอบด้วย 9 เกตแบบแอนด์ และมีเส้นอินพุตอยู่ 36 เส้น และเส้นที่ตัดกันระหว่างเส้นอินพุตกับเส้นที่มาต่อเข้าเกตแบบแอนด์ จะเป็นทรานซิสเตอร์แบบ CMOS ในกระบวนการโปรแกรม และกระบวนการลบโปรแกรมจะใช้การผ่านกระแสเข้าไปที่ทรานซิสเตอร์เพื่อทำการต่อวงจร หรือทำการเปิดวงจร

จากวงจรรูปที่ 2.10 ในส่วนที่เป็นบล็อกที่ควบคุมโครงสร้างภายในของไอโอจะมีส่วนที่เป็นตัวเลือกเอาต์พุตซึ่งจะมาด้วยกัน 4 ทาง ด้วยกัน คือ เอาต์พุตที่มาจากเกตแบบออร์โดยตรง, มาจากเอาต์พุตของอินเวอร์เตอร์ของเกตแบบออร์, เอาต์พุต Q ของฟลิปฟลอปและเอาต์พุต /Q ของฟลิปฟลอป และการเลือกการป้อนกลับจะมาจากเอาต์พุตที่ออกจากเกตแบบออร์, เอาต์พุตของฟลิปฟลอป และมาจากขาไอโออีกส่วนหนึ่ง

ในกรณีที่ใช้ฟลิปฟลอปจะมีสัญญาณสมวารเพื่อทำการเซต ในกรณีที่ต้องการเซตให้ฟลิปฟลอปทำงานตามขาขึ้นของสัญญาณนาฬิกา และสัญญาณอสมวารเพื่อทำการเคลียร์ฟลิปฟลอปให้ทำงานโดยไม่ต้องขึ้นกับสัญญาณนาฬิกา ในกรณีที่เมื่อเริ่มเปิดเครื่อง EP300 จะถูกเคลียร์ฟังก์ชันทั้งหมดอย่างอัตโนมัติ



รูปที่ 2.9 แสดงโครงสร้างภายในของ EP300 PLD



รูปที่ 2.10 แสดงรายละเอียดของส่วนหนึ่งของอุปกรณ์ EP300

2.2.2.3 อุปกรณ์ GAL

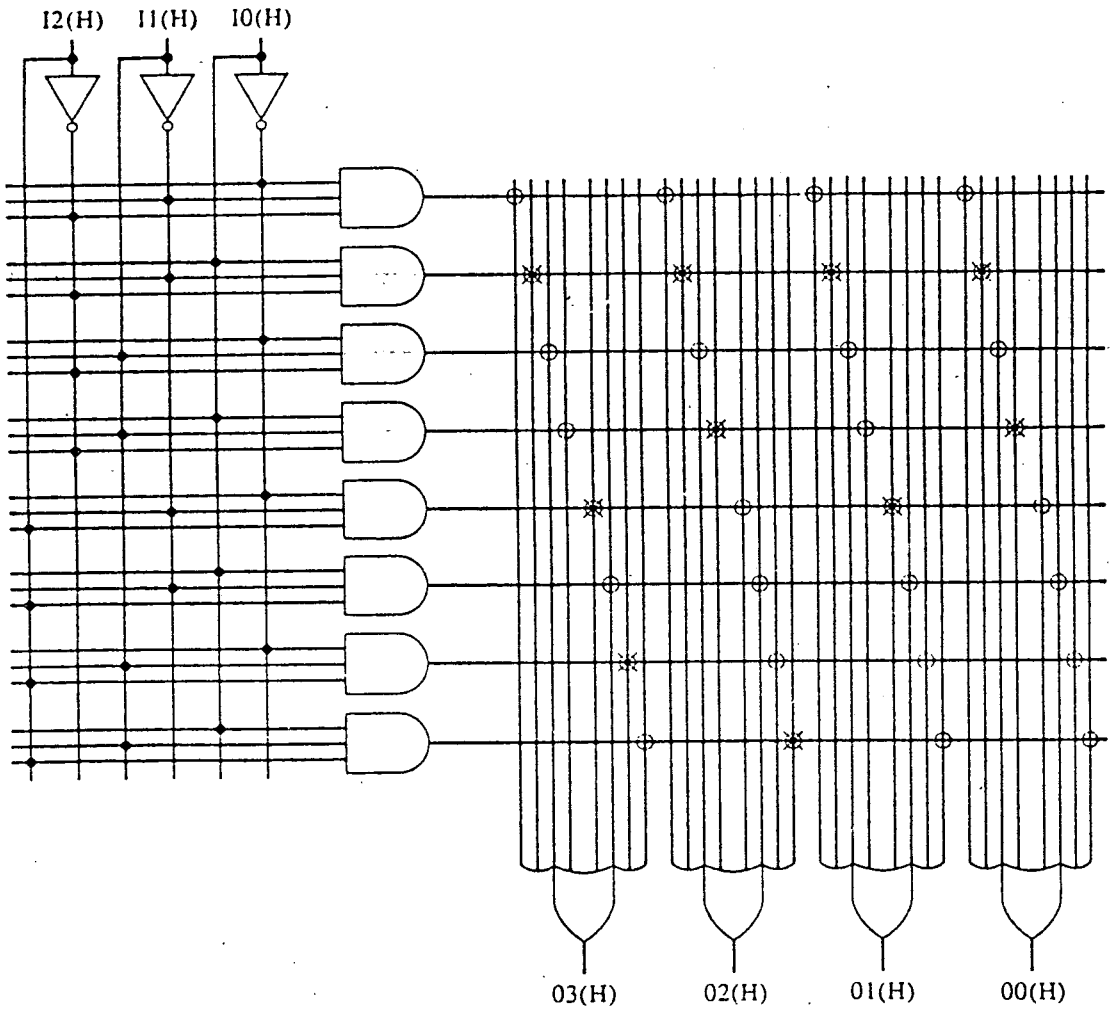
เป็นอีกรูปแบบหนึ่งของ EPROM ที่มีการใช้ทรานซิสเตอร์ในการกำหนดการเปลี่ยนแปลงของโปรแกรมในรูปแบบของโครงสร้างของ PAL โดยที่บริษัทที่ทำการผลิต GAL คือ บริษัท Lattice Semiconductor Corporation ซึ่งได้เริ่มผลิต GAL16V8 [5] โดยที่ GAL16V8 เป็นอุปกรณ์ CMOS ที่มี 20 ขา ใช้กระแสไฟฟ้าลบโปรแกรมได้ แต่ละเอาต์พุตเป็นแบบแมโครเซลล์มีโครงสร้างเหมือนกับ EP300 สามารถทำการลบโปรแกรมได้ประมาณ 100 ครั้ง

2.2.3 Fixed AND-Programmable OR : การโปรแกรมทางด้านออร์

2.2.3.1 อุปกรณ์ PROM

รูปที่ 2.11 แสดงไดอะแกรมวงจรลอจิกของ PROM ที่มีขนาดเล็กมีอินพุต 3 อินพุต คือ I_0, I_1 และ I_2 และมีเอาต์พุต 4 เอาต์พุต คือ O_0 , O_1 , O_2 และ O_3 โดยที่สัญลักษณ์ที่เป็นจุดดำเป็นการกำหนดที่ถาวร สัญลักษณ์วงกลม คือ สามารถทำการโปรแกรมได้ ถ้าต้องการต่อฟิวส์หรือไม่ต้องการต่อฟิวส์ ตามการออกแบบวงจรที่ได้ออกแบบไว้

จากแผนภาพในรูปที่ 2.11 ตัว PROM มีอินพุตที่เป็นเกตแบบแอนด์ 3 อินพุตถูกกำหนดด้วยค่าที่แน่นอน คือ ค่าตั้งแต่ $I_2I_1I_0$ ถึง $\bar{I}_2\bar{I}_1\bar{I}_0$ ส่วนที่ทำการโปรแกรมได้ คือ เอาต์พุตที่มีเกตแบบออร์ ซึ่งเราจะได้ว่า



รูปที่ 2.11 แสดงวงจรลอจิกของ PROM

$$O_3(H) = (\bar{I}_2\bar{I}_1I_0 + I_2\bar{I}_1\bar{I}_0 + I_2I_1\bar{I}_0)(H) = (\bar{I}_2\bar{I}_1I_0 + I_2\bar{I}_0)(H) \quad \text{(สมการที่ 2.1)}$$

$$O_2(H) = (\bar{I}_2\bar{I}_1I_0 + \bar{I}_2I_1I_0 + I_2I_1I_0)(H) = (\bar{I}_2\bar{I}_1I_0 + I_1I_0)(H) \quad \text{(สมการที่ 2.2)}$$

$$O_1(H) = (\bar{I}_2\bar{I}_1I_0 + I_2\bar{I}_1\bar{I}_0)(H) \quad \text{(สมการที่ 2.3)}$$

$$O_0(H) = (I_2\bar{I}_1\bar{I}_0 + \bar{I}_2I_1I_0)(H) \quad \text{(สมการที่ 2.4)}$$

PROM ก็คือ อุปกรณ์ที่เป็นหน่วยความจำอีกรูปแบบหนึ่งที่มีการกำหนดค่าตำแหน่งที่แน่นอนในรูปแบบของเลขไบนารีไม่สามารถทำการเปลี่ยนค่าได้ เช่น การกำหนดค่าตามตารางที่ 2.1 ค่า $I_2I_1I_0$ ถูกกำหนดค่าดังแสดง ในตารางและนำไปแทนค่า $I_2I_1I_0$ ในสมการที่ 2.1 - สมการที่ 2.4 จากข้างต้นจะได้ค่าเอาต์พุต $O_0O_1O_2O_3$ ดังแสดงในตาราง

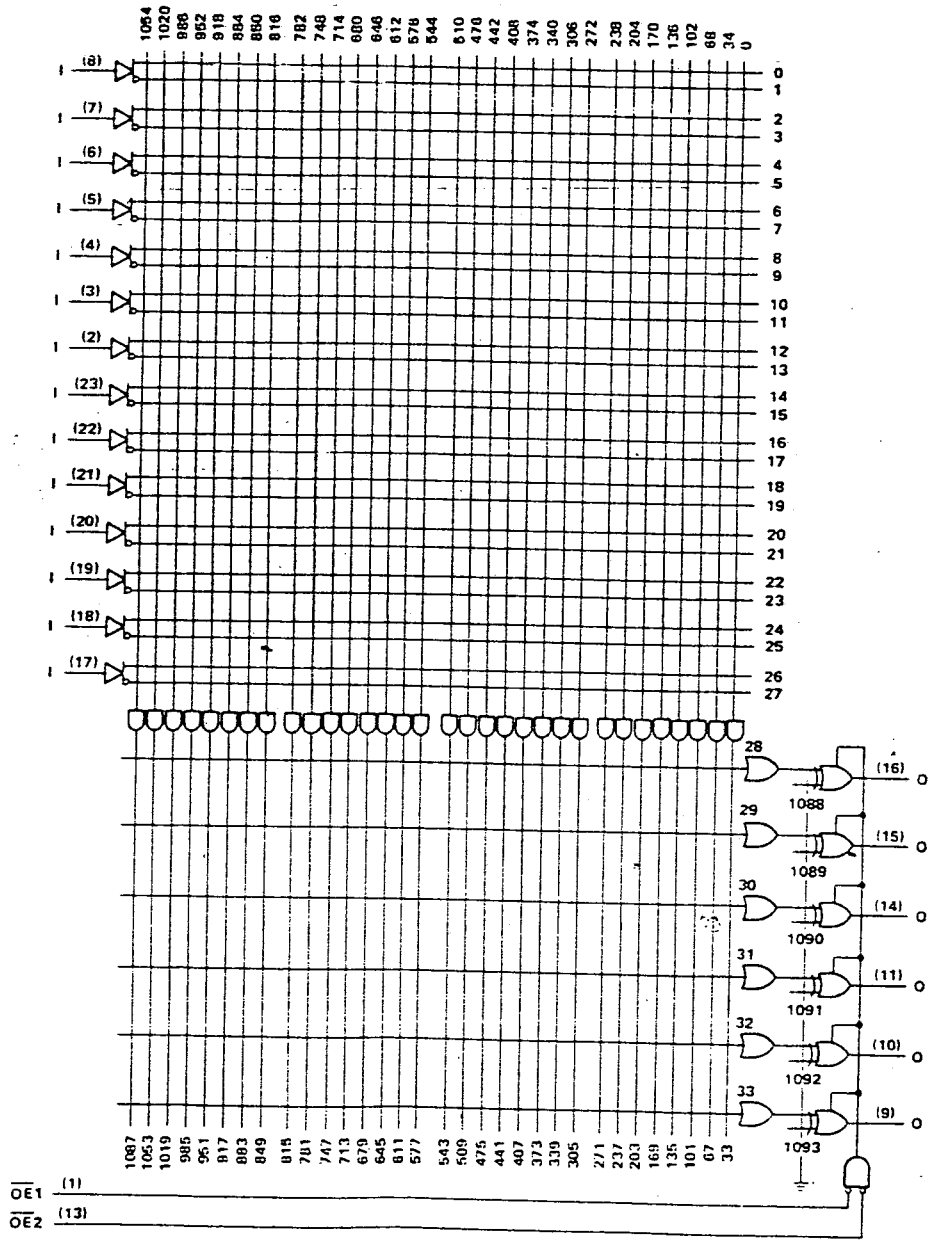
Adresses (Input) : I	Contents (Output) : O
000	0000
001	1110
010	0000
011	0100
100	1011
101	0000
110	1000
111	0100

รูปที่ 2.12 แสดงแถวลำดับของเรจิสเตอร์

2.2.4 Programmable AND-programmable OR : การโปรแกรมทั้งด้านแอนด์และออร์

2.2.4.1 อุปกรณ์ FPLA

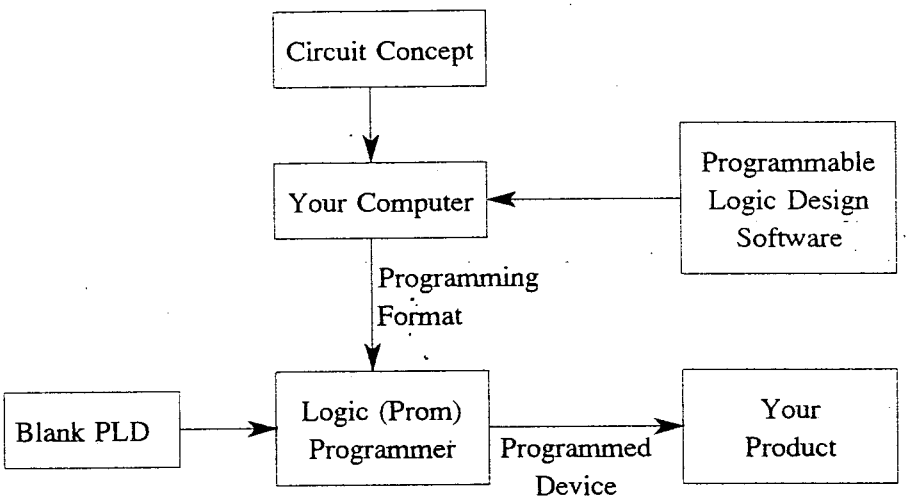
รูปที่ 2.13 แสดงไดอะแกรมลอจิกของ 14 x 32 x 6 FPLA, TIFPLA839 ประกอบด้วย 14 อินพุต, 32 Producterm และ 6 เอาต์พุต



รูปที่ 2.13 แสดงไดอะแกรมลอจิกของ FPLA

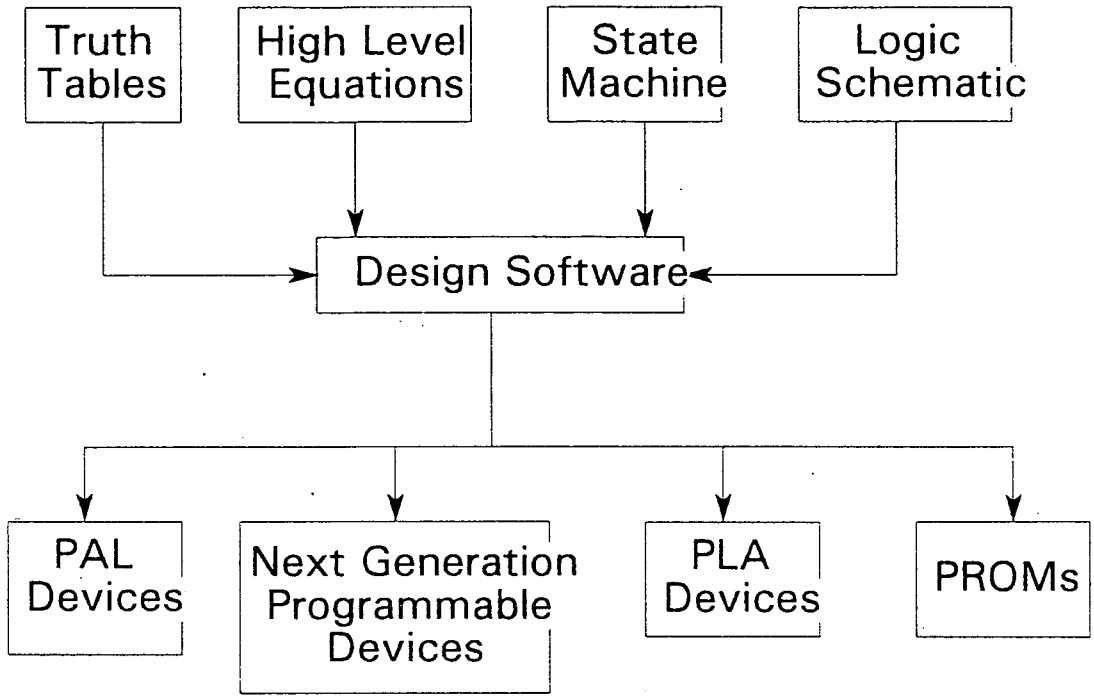
2.3 ซอฟต์แวร์ที่ช่วยพัฒนา PLD (PLD Development Software)

สำหรับซอฟต์แวร์ที่ใช้ออกแบบวงจรทางด้านลอจิก จะแปลงจากรูปแบบที่เครื่องโปรแกรมสามาะถยอมรับเพื่อทำการโปรแกรมไปที่ตัวอุปกรณ์ PLD ได้ รูปที่ 2.14 แสดงแผนภาพบล็อกของการพัฒนาลอจิกที่สามารถโปรแกรมได้



รูปที่ 2.14 แสดงแผนภาพบล็อกของการพัฒนาลอจิกที่สามารถโปรแกรมได้

ซอฟต์แวร์นี้จะใช้ช่วยในการออกแบบอุปกรณ์ที่เป็น PLD โดยการเขียนอินพุตที่ออกแบบไว้ สำหรับอินพุตที่ป้อนให้กับซอฟต์แวร์จะมีรูปแบบที่เป็นทั้งสเตตแมชีน (State Machine), สมการบูลีน (Boolean Equation), ตารางความจริง และแผนภาพตรรกะ หลังจากคอมไพล์ออกมาแล้วก็จะนำไปทำการโปรแกรมอุปกรณ์ PLD โดยเครื่องโปรแกรม PLD ต่อไป ดังแสดงในรูปที่ 2.15 [9]



รูปที่ 2.15 แสดงรูปแบบในการใช้งานตัวคอมไพเลอร์

เครื่องมือที่ใช้พัฒนาซอฟต์แวร์สำหรับ PLD จะต้องสามารถเป็นทั้งตัวคอมไพล์ สามารถทำการจำลองโปรแกรมได้ (Simulation) และสามารถทำการตรวจสอบซอฟต์แวร์ได้ (Test Software) [9] โดยที่ตัวคอมไพล์ PLD จะทำการเปลี่ยนรูปแบบที่ออกแบบให้อยู่ในรูปแบบที่เป็นฟิวส์แมปพร้อมที่จะนำไปโปรแกรมอุปกรณ์ต่างๆ เช่น อุปกรณ์ PAL, อุปกรณ์ PROM ต่อไป ส่วนการจำลองโปรแกรมเพื่อทำการตรวจสอบว่าการออกแบบวงจรนั้นถูกต้อง ก่อนที่จะไปทำการโปรแกรม PLD เพื่อป้องกันการผิดพลาดที่เกิดขึ้น และสำหรับการตรวจสอบซอฟต์แวร์ว่าโปรแกรมที่ได้ใส่เข้าไปที่อุปกรณ์นั้นถูกต้อง โดยการสร้างรูปแบบเพื่อทำการตรวจสอบ ตารางที่ 2.1 แสดงคอมไพเลอร์ PLD ที่ใช้บนเครื่องคอมพิวเตอร์ IBM PC Compatible [10]

		Entry Methods							
Manufacture	Product	Boolean Equation	Truth Table	State Machine	Waveform	Schematic	Logic Minimizer	Editor	PLD Supported (nonexaustive)
Accel Technologies	Tango-PLD	Yes	Yes	Yes	No	Yes	Espresso	Optional	PAL,GAL, PEEL,FPLA, FPL,EPLD, Atmel ATV
Advanced Micro Devices	Palasm 90	Yes	No	Yes	No	No	Yes	Yes	PAL,FPL
Data I/O	ABEL	Yes	Yes	Yes	No	Optional	Presto, Espresso	No	PAL,GAL, PEEL,FPLA, FPL,EPLD, Atmel ATV
Inlab	proLogic	Yes	Yes	Yes	No	No	No	No	EPLD
ISDATA	Log/iC	Yes	Yes	Yes	No	Optional	Fact,Bruno	No	PAL,GAL, PEEL,FPLA, FPL,EPLD, Atmel ATV
Logical Devices	CUPL	Yes	Yes	Yes	No	Otional	Redundancy minimization Presto, Espresso, Quine-McClusky	Yes	PAL,GAL, PEEL,FPLA, FPL,EPLD, Atmel ATV
Minc	PLDesign	Yes	Yes	Yes	Yes	Yes	Sum-of-Products, Espresso, Quine-McCluskey	Yes	PAL,GAL, PEEL,FPLA, FPL,EPLD
	PGADesign	Yes	Yes	Yes	Yes	Yes	Sum-of-Products, Espresso, Quine-McCluskey	Yes	PAL,GAL, PEEL,FPLA, FPL,EPLD, Altera MAX, Actel ACT-1, Xilinx
National Semiconductor	Plan II	Yes	No	No	No	No	No	No	PAL,GAL
Omation	Schema-PLD	Yes	Yes	Yes	No	Yes	Quick-min,	No	PAL, GAL

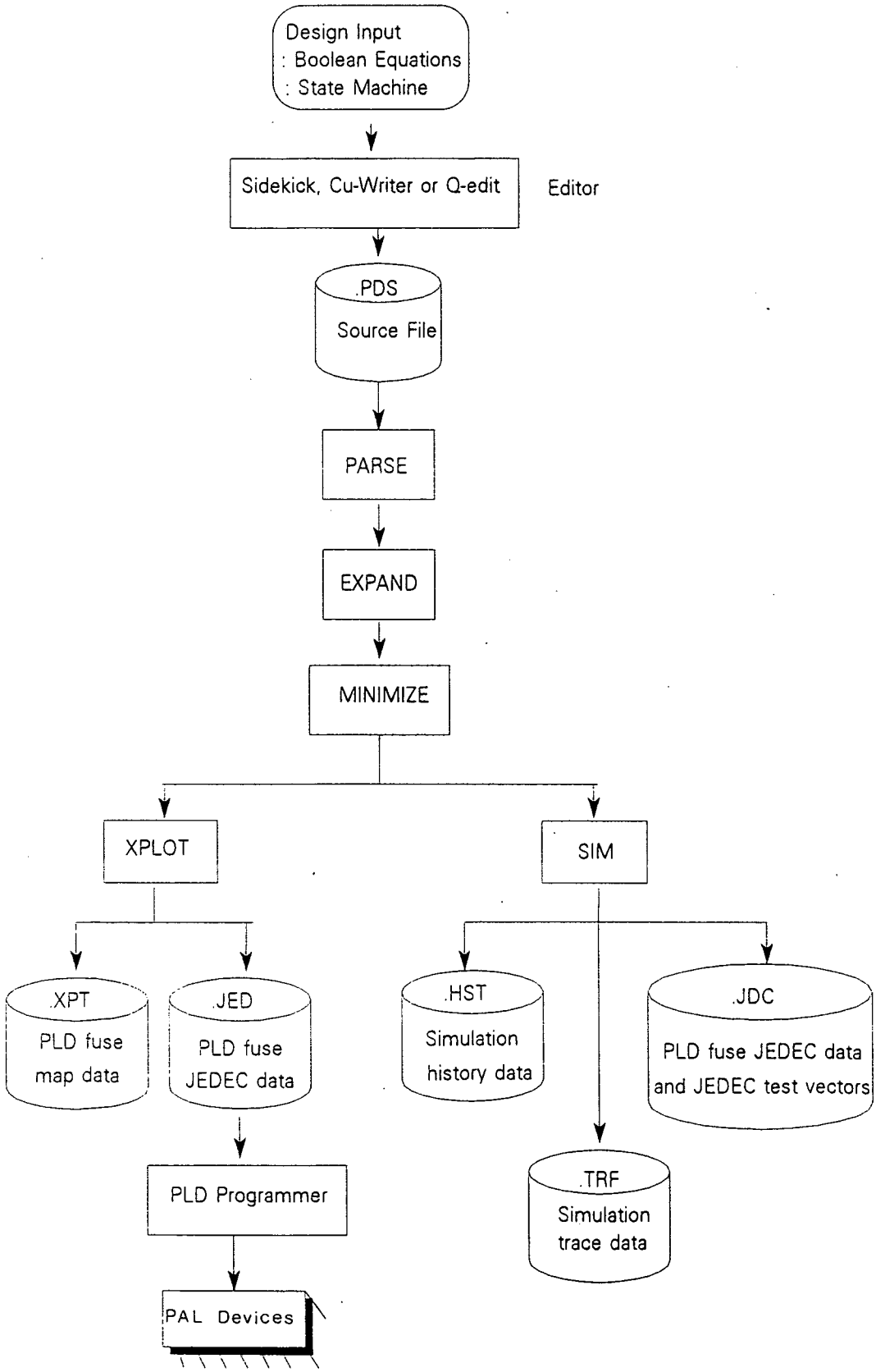
							Presto Quine-McCluskey proprietary method		
Orcad Systems	ORCAD/PLD	Yes	Yes	Yes	No	Yes	Quine-McCluskey McBoole	No	PAL,GAL, EPLD
Pistrol Electronic Tool	SP11	Yes	No	No	No	No	No	Yes	GAL, EPLD

ตารางที่ 2.1 แสดง PLD Compilers ที่ใช้บนเครื่องคอมพิวเตอร์ IBM Compatible

จากตารางเป็นการสรุปถึงซอฟต์แวร์ที่ช่วยในการออกแบบ PLD และบริษัทที่เป็นผู้สร้างโปรแกรมซึ่งอยู่ที่ต่างประเทศ เช่น บริษัท Data I/O มีคอมไพเลอร์ชื่อ ABEL มีการป้อนอินพุตได้เป็นแบบบูลีน, สเตตแมชีน, ตารางความจริง, และอาจจะป้อนอินพุตในรูปแบบที่เป็นแผนภาพตรรกะได้อีก และอุปกรณ์ที่สามารถใช้กับคอมไพเลอร์ได้จะมี PAL, GAL, PEEL, FPLA etc. สำหรับซอฟต์แวร์ที่ใช้ในการพัฒนาในงานวิจัยนี้ เพื่อสร้างรูปแบบของฟิวส์แมปได้เลือกใช้ซอฟต์แวร์ที่มีชื่อว่า PALASM2 ซึ่งเป็นโปรแกรมสำเร็จรูปเป็นเครื่องมือซอฟต์แวร์ที่ช่วยสนับสนุนงานวิจัยนี้ สาเหตุที่เลือกใช้ซอฟต์แวร์ตัวนี้ก็เพราะว่าเป็นโปรแกรมที่ใช้ง่าย เนื่องจากมีเมนูที่ติดต่อกับผู้ใช้ รูปที่ 2.16 แสดงแผนภาพบล็อกของโปรแกรมสำเร็จรูป PALASM2

PALASM2 คือ โปรแกรมสำเร็จรูปที่ใช้งานทางด้านวงจรลอจิก และมีเมนูโต้ตอบกับทำให้สะดวกต่อผู้ใช้ซึ่งเป็นข้อดีของโปรแกรม PALASM2 ในการโปรแกรมอุปกรณ์ PAL นั้นเป็นการออกแบบโดยกำหนดการตัดต่อของฟิวส์ ซึ่งตำแหน่งของการโปรแกรมในอุปกรณ์นั้นจะใช้ได้ทั้งสมการบูลีน และแบบสเตตแมชีน โดยที่โปรแกรมนี้ [17]

1. จะทำการตรวจสอบว่าแฟ้มอินพุตที่ป้อนเข้าไปมีวากยสัมพันธ์ (Syntax Check) ถูกต้องหรือไม่
2. สามารถแปลแฟ้มข้อมูลอินพุตให้เป็นแฟ้มข้อมูลเอาต์พุตต่างๆ ได้
3. สร้างรูปแบบของการโปรแกรมตัวอุปกรณ์ PLD ซึ่งเป็นรูปแบบ JEDEC ได้
4. จำลองการทำงานของวงจรก่อนที่จะนำไปโปรแกรมให้กับอุปกรณ์จริง



รูปที่ 2.16 แสดงแผนภาพบล็อกของโปรแกรมสำเร็จรูป PALASM

จากรูปที่ 2.16 ในการสร้างแฟ้มข้อมูลอินพุตของ PALASM2 จะเริ่มจากสร้างการออกแบบที่เป็นสมการบูลีนหรือสเตตแมชีน โดยการใช้เอดิเตอร์สร้างแฟ้มข้อมูลอินพุตที่เป็น .PDS ต่อจากนั้นจึงมาผ่านเข้าคอมไพเลอร์ของ PALASM2 คือ

PARSE : การตรวจสอบวากยสัมพันธ์

EXPAND : ขยายสมการอินพุตและเปลี่ยนรูปแบบสเตตแมชีนให้เป็นแบบสมการบูลีน

MINIMIZE : ลดรูปสมการลง

XPLOT : เป็นตัวสร้างแฟ้มข้อมูลที่มีรูปแบบที่จะนำไปโปรแกรมตัวอุปกรณ์ PAL ได้ โดยเครื่องโปรแกรม PAL

SIM : จำลองการออกแบบอุปกรณ์ PAL ว่าการออกแบบนั้นถูกต้องหรือไม่

เมื่อเสร็จจากการคอมไพล์ด้วยโปรแกรม PALASM2 แล้ว จะได้แฟ้มข้อมูลที่เป็นแฟ้มข้อมูลเอาต์พุตด้วยกันดังต่อไปนี้

.XPT : ข้อมูลฟิวส์แมปของ PLD

.JED : ข้อมูลรูปแบบ JEDEC ของ PLD

.HST : การจำลองข้อมูลของการออกแบบ

.JDC : ข้อมูลรูปแบบ JEDEC และทดสอบเวคเตอร์

.TRF : ข้อมูลการจำลองการ Trace

สำหรับการโปรแกรมตัวอุปกรณ์ PAL เราจะใช้แฟ้มข้อมูลเอาต์พุตที่เป็น JEDEC เพื่อนำไปโปรแกรมอุปกรณ์ PAL ต่อไป

2.4 การพัฒนาเกี่ยวกับ PLD ในปัจจุบัน

FPGA ย่อมาจากคำว่า Field Programmable Gate Array ซึ่งเป็นอุปกรณ์ทาง อิเล็กทรอนิกส์ที่สามารถโปรแกรมให้เกิดวงจรลอจิกตามแต่ผู้ใช้ต้องการ

2.4.1 FPGA : Field Programmable Gate Array

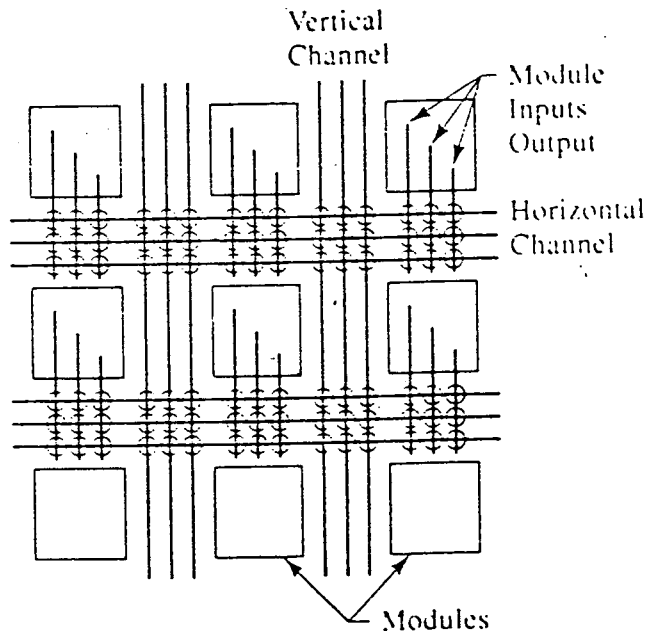
ส่วนประกอบของ FPGA มีส่วนประกอบหลัก ๆ อยู่ 3 ส่วน ดังแสดงในรูปที่ 2.17 คือ

1. Configuration Logic Blocks (CLB) เป็นกลุ่มของเกตและเรจิสเตอร์เล็ก ๆ ที่มีโครงสร้างภายในเหมือนๆ กัน ซึ่งมีอยู่เป็นจำนวนมากวางเรียงอยู่เป็นแถวทั้งแนวตั้ง และแนวนอน มีหน้าที่สำหรับสร้างวงจรลอจิกให้มีหน้าที่การทำงานต่างๆ ตามผู้ใช้กำหนด

2. Input/Output Blocks (IOB) ทำหน้าที่รับ และ/หรือส่งสัญญาณระหว่างวงจรภายในไปยังภายนอก FPGA

3. Routing Resource ทำหน้าที่เชื่อมต่อสัญญาณภายในระหว่าง Block ต่างๆ (CLB กับ CLB หรือ CLB กับ IOB)

สำหรับ FPGA จะมีการต่อภายในที่กำหนดแน่นอนน้อยมาก สามารถทำการโปรแกรมได้โดยการเชื่อมต่อโมดูลแต่ละโมดูลที่เป็นอิสระต่อกัน เพื่อเพิ่มให้เป็นวงจรที่มีขนาดใหญ่ขึ้นมาได้ตามแต่ที่ผู้ใช้ต้องการ ทำให้มีความยืดหยุ่นกว่า PLD มาก และภายในประกอบด้วยเกตตั้งแต่ 1,000 เกตขึ้นไป ซึ่งเป็นข้อได้เปรียบของ FPGA ในการโปรแกรม FPGA นั้นจะใช้โปรแกรมที่เป็นตัวคอมไพเลอร์ของ FPGA ทำการเปลี่ยนรูปแบบในการเขียนที่เป็นการโปรแกรม FPGA ก็คือการต่อภายในระหว่างโมดูลแต่ละโมดูลในโครงสร้างของ FPGA เพื่อให้ได้แฟ้มข้อมูลที่เป็นฟิวส์แมป ต่อจากนั้นจึงผ่านเข้าเครื่องโปรแกรม FPGA ต่อไป



รูปที่ 2.17 แสดงโครงสร้างภายในของ FPGA

2.4.2 การพัฒนาเครื่องโปรแกรม PLD (PLD Programmer)

สำหรับเครื่องโปรแกรม PLD ที่ใช้กันในบ้านเรานั้นจะใช้เป็นแบบเครื่องโปรแกรมแบบเอนกประสงค์ Universal Programmer ทำให้มีราคาแพง และยังไม่จำเป็นสำหรับบางกรณีที่จะต้องใช้เครื่องรุ่นนั้น เพราะสำหรับงานออกแบบทางด้านวงจรลอจิกจริงๆ แล้วใช้แค่เครื่องโปรแกรม PLD ก็น่าจะได้แล้ว เช่น เครื่องโปรแกรมเอนกประสงค์ยี่ห้อ Unisite40 ของบริษัท DATA I/O CORP. หรือเครื่องโปรแกรมเฉพาะ PLD ของบริษัท LOGICAL DEVICES INC. ซึ่งจะโปรแกรมได้เฉพาะอุปกรณ์ PAL เหล่านี้เป็นต้น ดังนั้นในการทำวิจัยเกี่ยวกับเครื่องโปรแกรม PLD ภายในประเทศยังมีไม่มากนัก เพราะเนื่องจากยังขาดข้อมูลในการสร้างเครื่องโปรแกรมจึงทำให้ยังจำเป็นต้องสั่งซื้อเข้ามาจากต่างประเทศ