บทที่ 3
การออกแบบและขั้นตอนการดำเนินงาน

3.1 รายละเอียดโปรแกรมที่จะพัฒนา

3.1.1 Input Specification

1) รูปภาพ หรือ ข้อความที่ต้องการที่จะนำมาแปลง เพื่อแสดงบนน้ำตกได้ โดยมี GUI ที่ง่ายต่อการใช้งาน
2) ควบคุมการทำงานฮาร์ดแวร์ด้วย GUI ให้แสดงภาพบนน้ำตกตามรูปแบบที่ต้องการ

3.1.2 Output Specification

1) แสดงรูปภาพที่ผ่านการแปลงแล้วบน GUI เพื่อให้ผู้ใช้งานสามารถตรวจสอบความพึงพอใจ ก่อนจะสั่งให้แสดงจริงบนน้ำตก
2) จำลองการทำงานของน้ำตกโดยแสดงเป็นภาพเคลื่อนไหวบน GUI
3) บันทึกรูปภาพที่ต้องการแสดงผลไว้ในลักษณะไฟล์เฉพาะ
4) ส่งภาพที่ต้องการแสดงบนน้ำตกเก็บไว้ในหน่วยความจำของฮาร์ดแวร์เพื่อใช้ในการแสดงผลต่อไป โดยควบคุมจากคอมพิวเตอร์(On PC Mode)หรือตัวฮาร์ดแวร์เอง (Stand Alone Mode)

3.1.3 Functional Specification

1) สามารถนำภาพที่ต้องการมาแปลงให้แสดงค่าด้วยน้ำตกได้
2) สามารถนำข้อความที่ต้องการมาแปลงเป็นภาพ และแสดงค่าด้วยน้ำตกได้
3) สามารถบันทึกภาพที่ได้แปลงแล้วสำหรับนำไปแสดงค่าบนน้ำตกลงบนเครื่องคอมพิวเตอร์ได้
4) สามารถเรียกใช้ภาพที่ได้บันทึกแล้ว เพื่อนำมาแสดงค่าบนน้ำตกได้
5) สามารถจำลองการทำงานบนเครื่องคอมพิวเตอร์ได้
6) สามารถกำหนดการทำงานของน้ำตกได้ทั้งทางซอฟท์แวร์บนเครื่องคอมพิวเตอร์และผ่านการทำงานแบบ Stand alone
7) สามารถเรียกใช้ข้อมูลภาพแบบต่างๆ จากบนบอร์ดได้
8) สามารถสั่งการทำงานน้ำตกได้ด้วยซอฟท์แวร์บนคอมพิวเตอร์ ได้แก่ การเล่น และหยุดเล่นน้ำตก
9) สามารถบันทึกข้อมูลภาพแบบต่างๆ สำหรับแสดงบนน้ำตกลงบอร์ดได้
10) บอร์ดแต่ละตัว รองรับการใช้งานน้ำตกได้จำนวน 16 วาล์ว

11) วาล์วแต่ละตัวสามารถแสดงภาพได้ความเร็วสูงสุด 25 ครั้งต่อวินาที
12) สามารถรองรับการทำงานของบอร์ดได้มากที่สุดจำนวน 255 บอร์ด
13) สามารถรองรับรูปแบบการแสดงผลบนบอร์ดได้สูงที่สุด 15 แบบ ซึ่งแต่ละแบบสามารถแสดงการทำงานได้ 40 วินาที

3.2 ขอบเขตและข้อจำกัดของโปรแกรมที่พัฒนา

3.2.1 ส่วนติดต่อกับผู้ใช้

1) สามารถเชื่อมต่อกับอุปกรณ์ควบคุมการทำงานของน้ำตกได้ด้วย Serial communication (RS232)
2) ผู้ใช้สามารถใช้งานโปรแกรมผ่าน GUI
3) สามารถนำภาพหรือข้อความที่ต้องการมาแปลงให้แสดงค่าด้วยน้ำตกได้ โดยที่จำกัดความสูงของภาพสูงสุด 1,000 pixel (แสดงผลด้วยความเร็ว 25 ครั้ง/วินาที โดยที่แสดงสูงสุด 40 วินาที) และความกว้างของภาพสูงสุด 256 pixel หรือ (16 บอร์ด x 16 วาล์ว)
4) สามารถบันทึกภาพที่ได้แปลงแล้วลงบนเครื่องคอมพิวเตอร์ในลักษณะไฟล์เฉพาะ
5) สามารถเรียกใช้ภาพที่ได้บันทึกแล้ว เพื่อนำมาแสดงค่าบนน้ำตกได้
6) สามารถจำลองการทำงานบนเครื่องคอมพิวเตอร์ได้
7) สามารถควบคุมอุปกรณ์ควบคุมการทำงานของน้ำตกได้ผ่านหน้าจอ GUI โดยที่สามารถสั่งคำสั่งแสดงผล (Play) , คำสั่งหยุดแสดงผล (Stop) เป็นต้น
8) สามารถบันทึกข้อมูลภาพแบบต่างๆ สำหรับแสดงบนน้ำตก ลงบอร์ดได้

3.2.2 โปรโตคอลที่ใช้สื่อสารระหว่างส่วนติดต่อกับผู้ใช้กับอุปกรณ์ควบคุมการทำงานของน้ำตก

1) สื่อสารด้วยรูปแบบ Serial communication (RS232)
2) สื่อสารในลักษณะ Stop and wait ARQ
3) รองรับการต่อพ่วงกับอุปกรณ์ควบคุมการทำงานของน้ำตกได้สูงสุด 255 จุด
4) รองรับการควบคุมรูปแบบของการแสดงผลได้สูงสุด 15 แบบ
5) รองรับรูปแบบของคำสั่งได้สูงสุด 16 คำสั่ง
6) รองรับขนาดของแพ็กเกตรวม header สูงสุด 1,275 bytes

3.2.3 อุปกรณ์ควบคุมการทำงานของน้ำตก

1) สามารถเชื่อมต่อกับคอมพิวเตอร์ได้ด้วย Serial communication (RS232)
2) สามารถทำงานได้โดยใช้คอมพิวเตอร์เป็นตัวควบคุม (On PC mode)

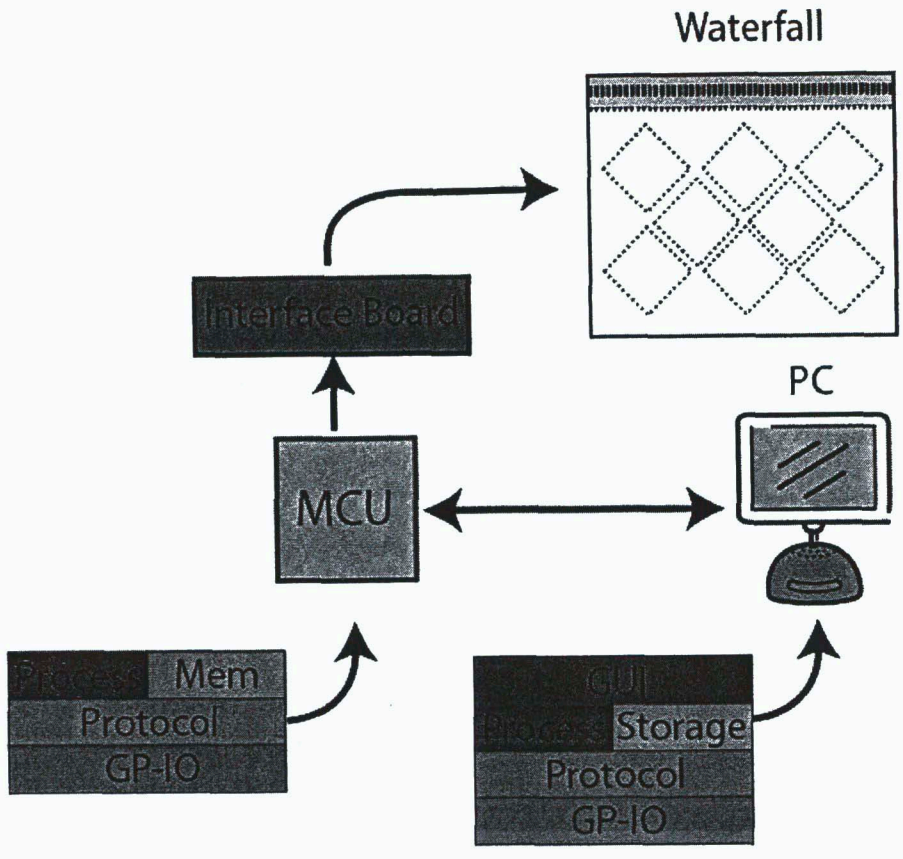
3) สามารถทำงานได้โดยไม่ต้องใช้คอมพิวเตอร์ควบคุม (Stand-alone mode) โดยที่แต่ละบอร์ดสามารถทำงานได้อย่างอิสระ หรือเชื่อมต่อการทำงานกับบอร์ดอื่นได้ เช่น คำสั่งแสดงผล (Play) , คำสั่งหยุดแสดงผล (Stop) เป็นต้น
4) สามารถจดจำรูปแบบการแสดงผลได้สูงสุด 8 รูปแบบ
5) สามารถแสดงผลแต่ละรูปแบบได้นานสูงสุด 40 วินาที
6) สามารถตัดน้ำด้วยความเร็วสูงสุด 25 ครั้งต่อวินาที
7) สามารถควบคุมน้ำตกได้สูงสุด 16 วาล์ว ต่อควบคุมการทำงานของน้ำตก 1 จุด
8) ขนาดของวาล์วแต่ละตัวเท่ากับ 2 หุน มีการเว้นระยะระหว่างวาล์วแต่ละตัว 4 หุน

3.3 ภาพรวมของระบบ

โครงการวิจัยนี้จึงศึกษาและพัฒนาระบบการแสดงผลด้วยภาพและตัวอักษรผ่านทางสายน้ำ ซึ่งประกอบไปด้วยการศึกษา ออกแบบ และพัฒนา ทั้งหมด 3 ส่วนดังนี้

1) ส่วนติดต่อผู้ใช้ (UI)
2) ส่วนชุดควบคุมการแสดงผลผ่านทางสายน้ำ
3) ส่วนโปรโตคอลที่ใช้ในการส่งผ่านข้อมูลระหว่างคอมพิวเตอร์ กับฮาร์ดแวร์ที่ควบคุมการทำงานของน้ำตก

ในการศึกษาได้ทำการออกแบบส่วนต่างๆ ส่วนติดต่อผู้ใช้ ให้รองรับการทำงานที่หลากหลาย ส่วนชุดควบคุมการแสดงผล และโปรโตคอลเพื่อรับส่งข้อมูลระหว่างคอมพิวเตอร์กับชุดควบคุมการแสดงผล



รูป 3.1 ภาพรวมของระบบ

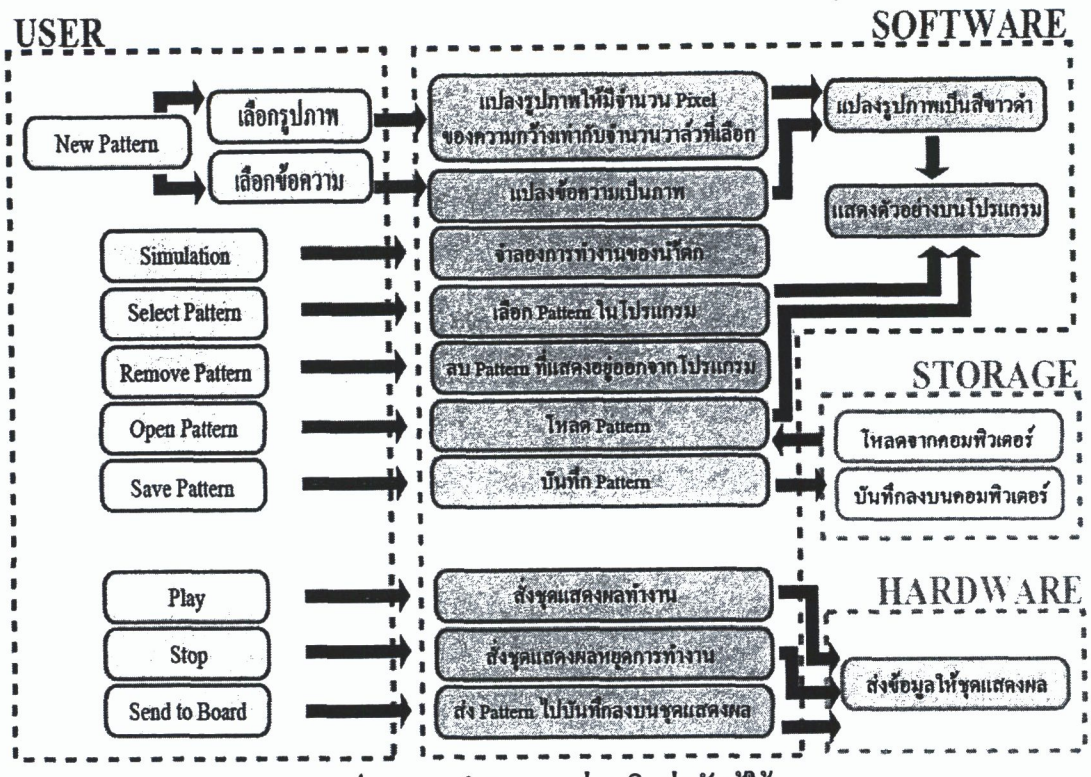
3.4 ส่วนประกอบของระบบ

ในระบบแสดงผลภาพและตัวอักษรผ่านทางสายน้ำประกอบไปด้วย ชุดการแสดงผลผ่านทางสายน้ำ ส่วนของคอนโทรลเลอร์ ส่วน Interface Board และส่วนการแสดงผลบนหน้าจอคอมพิวเตอร์

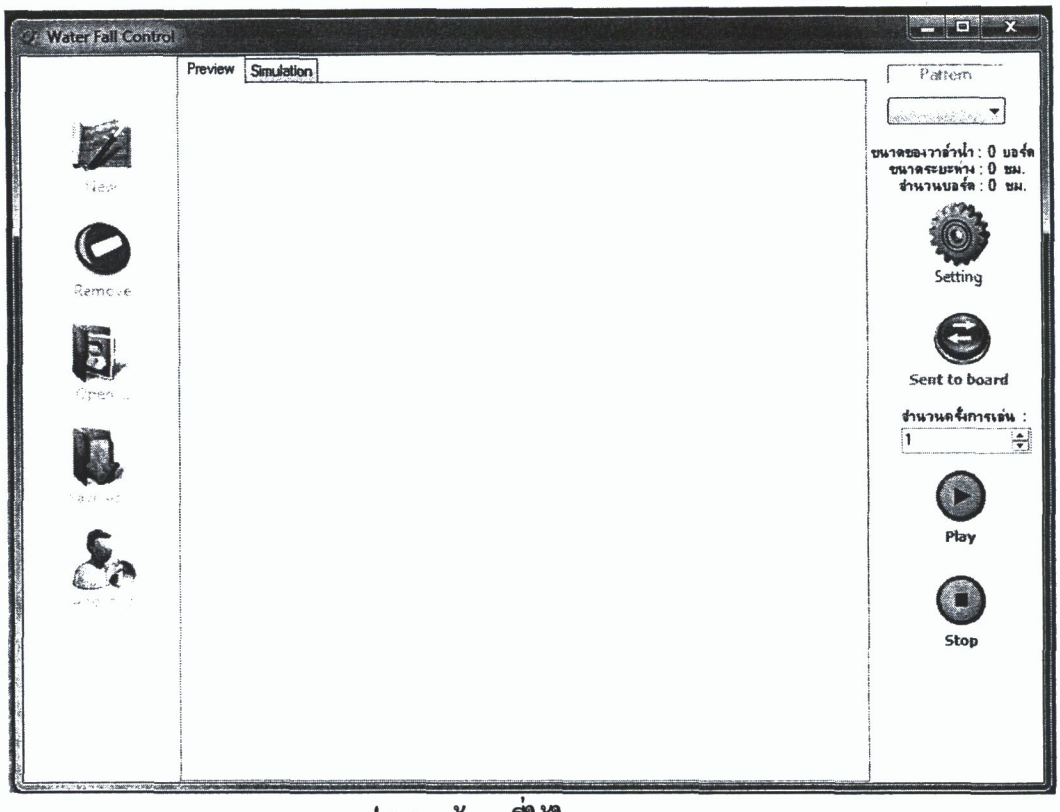
3.4.1 ส่วนที่ใช้ในการแสดงผลบนหน้าจอคอมพิวเตอร์

รองรับให้ผู้ใช้สามารถออกแบบแก้ไขรูปแบบของสายน้ำได้อย่างอิสระ โดยมีการจำลองก่อนทำการบันทึกลงหน่วยความจำของบอร์ดจริง หรือในกรณีที่ผู้ใช้ไม่ต้องการบันทึกรูปแบบของสายน้ำลงบนหน่วยความจำของบอร์ดทันที ก็สามารถบันทึกเก็บไว้ในคอมพิวเตอร์ เพื่อเปิดดูและนำไปบันทึกได้ในภายหลัง และสามารถนำภาพหรือตัวอักษรมาออกแบบเป็นรูปแบบที่ต้องการพร้อมทั้งแสดงตัวอย่างและจำลองการทำงานให้กับผู้ใช้ อีกทั้งผู้ใช้ยังสามารถกดแสดงหรือหยุดสายน้ำได้จากหน้าจอคอมพิวเตอร์ทันที

สำนักงานคณะกรรมการวิจัยแห่งชาติ ห้องสมุดงานวิจัย



รูป 3.2 การทำงานของส่วนติดต่อกับผู้ใช้

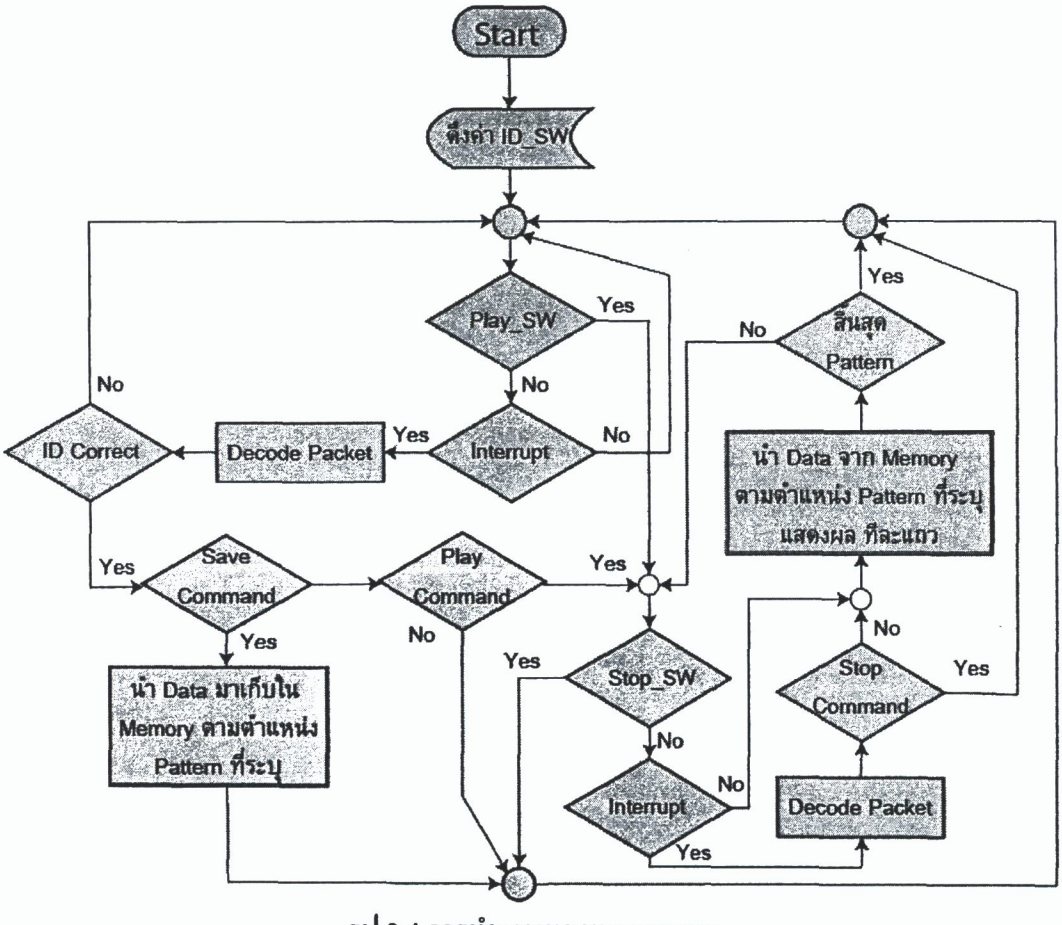


รูป 3.3 หน้าจอที่ใช้ในการแสดงผล

3.4.2 ชุดควบคุมการแสดงผลผ่านทางสายน้ำ

ชุดการแสดงผลนั้นสามารถทำงานได้โดยผ่านการควบคุมจากคอมพิวเตอร์หรือสามารถทำงานได้ด้วยตนเอง ซึ่งชุดการแสดงผลจะถูกแยกออกเป็นบอร์ดย่อยๆ โดยที่แต่ละบอร์ดจะมีหน้าที่ควบคุมสายน้ำบอร์ดละ 16 เส้น และแต่ละบอร์ดสามารถทำงานแยกกันอย่างอิสระหรือต่อพ่วงกันได้

ชุดการแสดงผลจะทำการรับข้อมูล (รูปแบบการแสดงผล) จากคอมพิวเตอร์มาเก็บไว้ภายในหน่วยความจำของชุดควบคุมการแสดงผล หลังจากนั้นจึงนำข้อมูลที่ได้รับมากำหนดการเปิด-ปิดน้ำด้วยโซลินอยด์วาล์ว โดยใช้ความเร็วสูงสุดในการเปิด-ปิดน้ำ 25 ครั้งต่อวินาที ซึ่งการปล่อยน้ำลงมาแสดงผลออกเป็นภาพและข้อความตัวอักษรนั้น แต่ละรูปแบบจะมีความยาวในการแสดงผลสูงสุด 40 วินาที



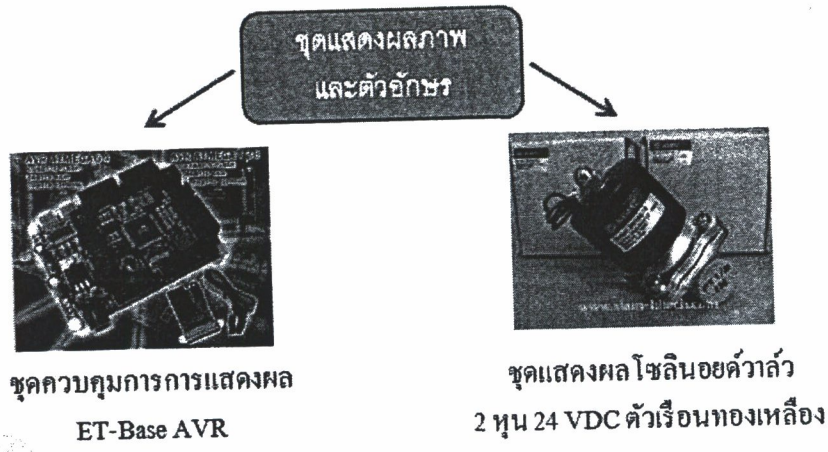
รูป 3.4 การทำงานของชุดแสดงผล

3.4.2.1 ส่วนประกอบของชุดควบคุมการแสดงผลผ่านทางสายน้ำ

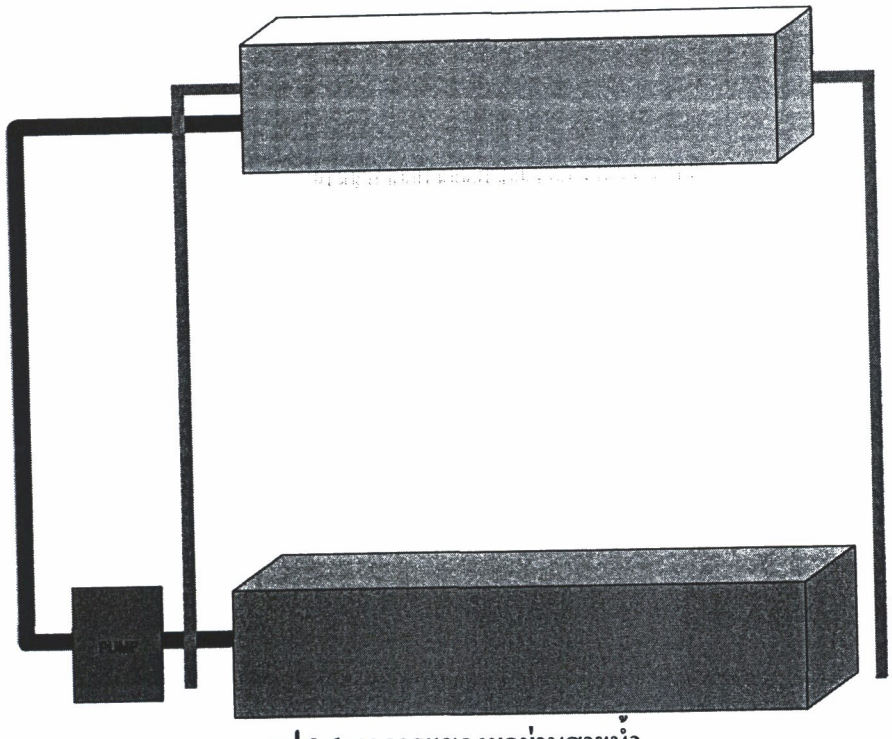
ส่วนประกอบของชุดแสดงผลภาพและตัวอักษรมี 2 ส่วน ดังนี้

1) ชุดควบคุมการแสดงผล ET-Base AVR

2) ชุดแสดงผลโซลินอยด์วาล์ว 2 หุน 24VDC ตัวเรือนทองเหลือง



รูป 3.5 ส่วนประกอบของชุดแสดงผล



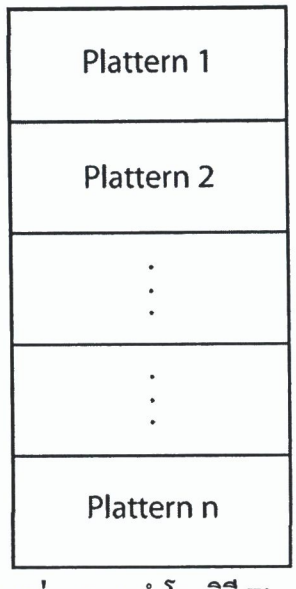
รูป 3.6 ชุดการแสดงผลผ่านสายน้ำ

3.4.2.2 ส่วนการกำหนดตำแหน่งในหน่วยความจำของไมโครคอนโทรลเลอร์

สำหรับการจัดการข้อมูลในหน่วยความจำของไมโครคอนโทรลเลอร์นั้น เพื่อให้ได้วิธีที่เหมาะสม ในด้านการใช้งาน ความสะดวกสบาย และความสมเหตุสมผลของงบประมาณ

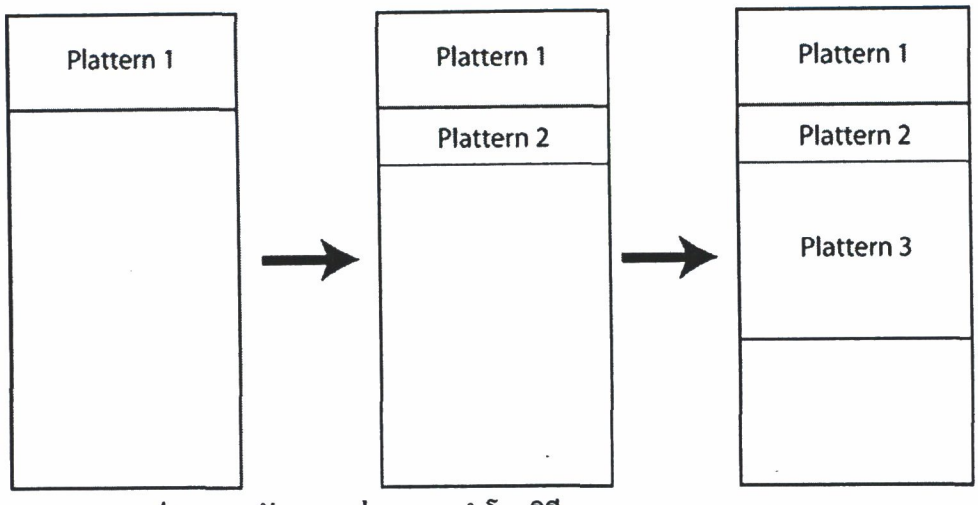
ที่ใช้ วิธีแรกคือ Fixed-block Memory การกำหนดตำแหน่งในหน่วยความจำของไมโครคอนโทรลเลอร์ วิธีที่สอง คือ Variable-block Memory

1) การจัดการตำแหน่งโดยวิธี Fixed-block Memory เป็นการกำหนดตำแหน่งของหน่วยความจำที่กำหนดขนาดไว้ โดยการกำหนดเป็นตำแหน่งของหน่วยความจำแบบตายตัว ซึ่งจะทำให้สามารถเก็บรูปแบบของการแสดงผลไว้เป็นสัดส่วน แต่จะทำให้ไม่สามารถใช้พื้นที่ในหน่วยความจำได้เป็นประโยชน์สูงสุดอีกทั้งความยาวของรูปแบบก็จะถูกกำหนดตายตัวตามขนาดของหน่วยความจำในแต่ละ block



รูป 3.7 การจัดการหน่วยความจำโดยวิธี Fixed-block Memory

2) การจัดการตำแหน่งโดยวิธี Variable-block Memory เป็นการกำหนดตำแหน่งของหน่วยความจำที่กำหนดตำแหน่งของหน่วยความจำแบบไม่ตายตัวทำให้สามารถรองรับรูปแบบขนาดต่างๆ ได้ แต่ต้องเพิ่มเรื่องการจัดการหน่วยความจำเข้ามา



รูป 3.8 การจัดการหน่วยความจำโดยวิธี Variable-block Memory

3.4.2.3 วิธีที่เลือกใช้

เลือกวิธีการจัดการหน่วยความจำโดยวิธี Fixed-block Memory ซึ่งเป็นการกำหนดตำแหน่งของหน่วยความจำที่กำหนดขนาดไว้

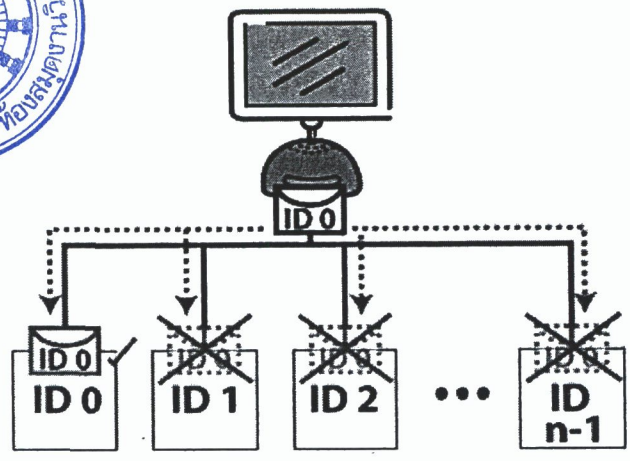
3.4.3 ส่วนโปรโตคอลที่ใช้ในการส่งผ่านข้อมูลระหว่างคอมพิวเตอร์ กับฮาร์ดแวร์ที่ควบคุมการทำงานของน้ำตก

เนื่องจากระบบมีการทำงานที่หลายหลายรูปแบบจึงต้องมีการกำหนดโปรโตคอลที่ใช้ในการส่งผ่านข้อมูลระหว่างคอมพิวเตอร์ กับฮาร์ดแวร์ที่ควบคุมการทำงานของน้ำตกซึ่งมีข้อจำกัดในการพิจารณาดังนี้

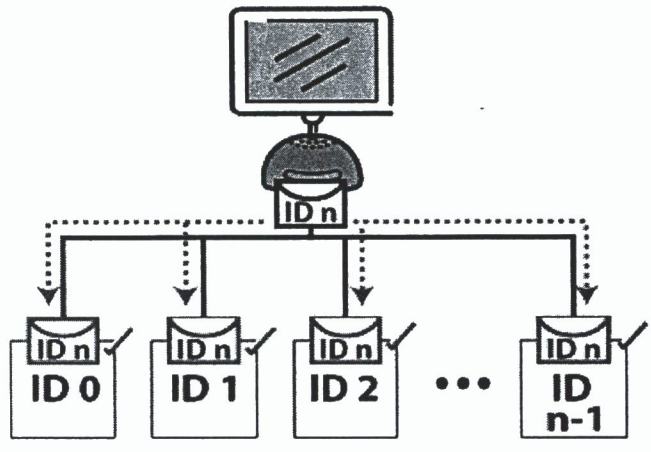
1) โปรโตคอลต้องสามารถรองรับการควบคุมบอร์ดจำนวนมากหรือเฉพาะเจาะจงได้
2) โปรโตคอลต้องสามารถบรรจุคำสั่งที่ใช้ในการควบคุมบอร์ดได้
3) โปรโตคอลต้องสามารถรองรับการระบุหมายเลขของรูปแบบการแสดงผล (รูปแบบ) จำนวนมากหรือเฉพาะเจาะจงได้
4) โปรโตคอลต้องสามารถรับ-ส่งข้อมูลระหว่างเครื่องคอมพิวเตอร์และตัวบอร์ดควบคุมได้และมีความน่าเชื่อถือ

3.4.3.1 รูปแบบโปรโตคอลที่เลือกใช้

รูปแบบการส่งผ่านข้อมูลที่ใช้ในโครงการจะเป็นการสื่อสารในรูปแบบอนุกรม (Serial communication) หรือมาตรฐาน RS-232 โดยที่จะสื่อสารด้วยโปรโตคอล Stop-and-Wait ARQ และมีรูปแบบการเชื่อมต่อระหว่างคอมพิวเตอร์กับบอร์ดต่างๆ ในลักษณะ Bus topology



รูป 3.9 การส่งข้อมูลด้วย Bus topology โดยไม่เจาะจงผู้รับ



รูป 3.10 การส่งข้อมูลด้วย Bus topology โดยเจาะจงผู้รับ

3.4.3.2 รูปแบบของแพ็กเกตของข้อมูล

แพ็กเกตของข้อมูลที่ออกแบบประกอบไปด้วย Header ขนาด 5 bytes และข้อมูลสูงสุด 1270 bytes ภายใน Header ประกอบไปด้วยส่วนประกอบต่างๆ ดังนี้

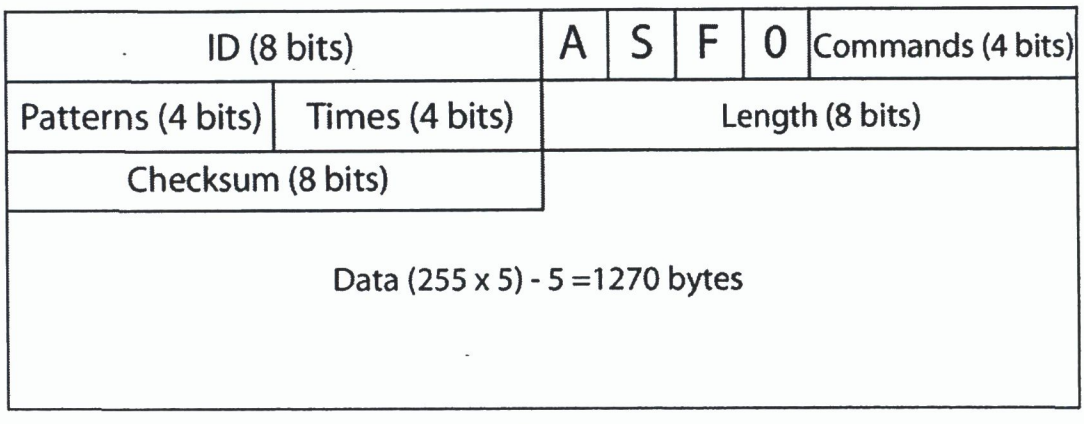
1) ID ขนาด 8 bits : ใช้ระบุหมายเลขชุดแสดงผล รองรับชุดแสดงผลได้สูงสุด 255 บอร์ด ในกรณีที่ ID = 0xFF จะเป็น Broadcast ID
2) ACK bit : ใช้สำหรับควบคุมการไหลของข้อมูล
3) SYN bit : ใช้สำหรับการทำ Handshaking
4) FIN bit : ใช้สำหรับระบุแพ็กเกตสุดท้าย
5) Commands ขนาด 4 bit : ใช้เพื่อระบุคำสั่ง (Command)

6) Patterns ขนาด 4 bits : ใช้ระบุหมายเลขรูปแบบ รองรับได้สูงสุด 15 รูปแบบ ในกรณีที่ รูปแบบ = 0x0F ให้เลือกทุกรูปแบบ

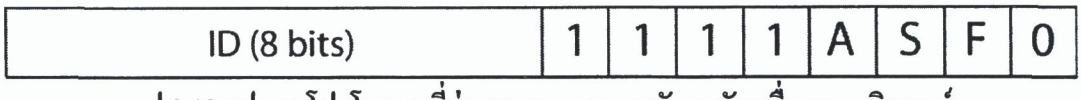
7) Times ขนาด 4 bits : ใช้เพื่อระบุจำนวนรอบในการแสดงผล

8) Length ขนาด 8 bits : ใช้เพื่อระบุขนาดของแพ็กเกต (Header+data) หน่วยเป็น Block (1 block = 5 byte)

9) Checksum ขนาด 8 bits : ใช้เพื่อตรวจสอบความถูกต้องของแพ็กเกต



รูป 3.11 แพ็กเกตของข้อมูล



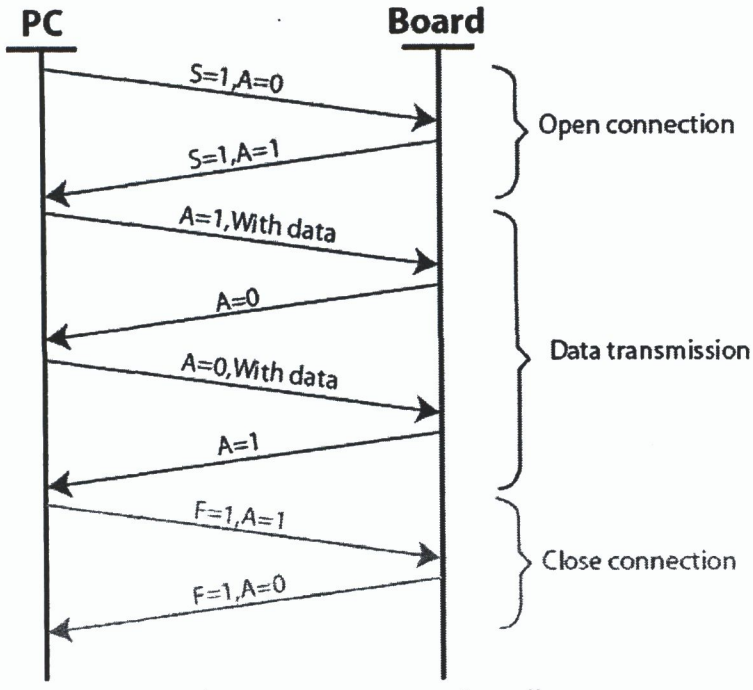
รูป 3.12 รูปแบบโปรโตคอลที่ส่งจากชุดควบคุมกลับมายังเครื่องคอมพิวเตอร์

3.4.3.3 การใช้งานโปรโตคอล

ใช้การสื่อสารระหว่างคอมพิวเตอร์กับบอร์ดควบคุม จะแบ่งรูปแบบของแพ็กเกตออกเป็น 2 รูปแบบคือ

1) แพ็กเกตควบคุม (Control packet) : เป็นแพ็กเกตที่ไม่บรรจุข้อมูลอยู่ จะไม่มีการ SYN และทำ Handshake เช่น Play, Stop เป็นต้น

2) แพ็กเกตข้อมูล (Data packet) : เป็นแพ็กเกตที่บรรจุข้อมูลอยู่ มีการ SYN และทำ Handshake เช่น SendToBoard เป็นต้น



รูป 3.13 การทำงานของแพ็กเกตข้อมูล

3.4.3.4 คำสั่งสำหรับควบคุมการแสดงผล

ในการใช้งานส่วนติดต่อกับผู้ใช้เพื่อควบคุมการทำงานของชุดควบคุมการแสดงผลนั้น จะใช้ฟิลด์ Commands ขนาด 4 bits เพื่อระบุคำสั่งต่างๆ ซึ่งมีรายละเอียดคำสั่งและขั้นตอนการทำงานของโปรโตคอลดังนี้

1) Commands = 0 : Test Device แพ็กเกตที่มีค่า Commands = 0 เป็นแพ็กเกตชนิดแพ็กเกตควบคุม คำสั่งนี้มีไว้สำหรับการทดสอบการเชื่อมต่อของอุปกรณ์ควบคุมการแสดงผล โดยที่แพ็กเกตชนิดนี้จะถูกส่งออกไปทีละบอร์ดเพื่อตรวจสอบการเชื่อมต่อและรอแพ็กเกตตอบกลับจากแต่ละบอร์ด ถ้าแพ็กเกตที่ตอบกลับมานั้นมี ID ตรงกับ ID ที่ส่งไปก็จะคืนค่าเป็น TRUE แต่ถ้าไม่มีแพ็กเกตตอบกลับมาหรือแพ็กเกตที่ตอบกลับมานั้นมีค่า ID ไม่ตรงกับที่ส่งไป จะคืนค่าเป็น FALSE

ID (8 bits)		0	0	1	0	0
0	0	1				
Checksum (8 bits)						

ID (8 bits)	1	1	1	1	0	0	1	0

รูป 3.14 แพ็กเกตสำหรับการทดสอบการเชื่อมต่อ

2) Commands = 1 : Play แพ็กเกตที่มีค่า Commands = 1 เป็นแพ็กเกตชนิดแพ็กเกตควบคุม คำสั่งนี้มีไว้เพื่อสั่งการให้อุปกรณ์ควบคุมการแสดงผลแสดงรูปแบบการแสดงออกมา โดยต้องทำการระบุ ID บอร์ดที่ต้องการแสดงผลในฟิลด์ ID, ระบุรูปแบบที่ต้องการให้แสดงผลในฟิลด์ Pattern และระบุรอบของการแสดงผลในฟิลด์ Timers ในกรณีที่ต้องการให้ทุกบอร์ดแสดงผลต้องกำหนดค่า ID = 0xFF และในกรณีที่ต้องการสั่งให้แสดงผลรูปแบบการแสดงผลทั้งหมดต้องกำหนดค่า Pattern = 0x0F แพ็กเกตนี้จะไม่มีการรอรับค่ากลับจากชุดควบคุมการแสดงผล

ID (8 bits)		0	0	1	0	1
Patterns (4 bits)	Times (4 bits)	1				
Checksum (8 bits)						

รูป 3.15 แพ็กเกตสำหรับสั่งการให้แสดงผล

3) Commands = 2 : Stop แพ็กเกตที่มีค่า Commands = 2 เป็นแพ็กเกตชนิดแพ็กเกตควบคุม คำสั่งนี้มีไว้เพื่อระงับการแสดงผลเท่านั้น ไม่สามารถระงับการทำงานอื่นๆ ของอุปกรณ์ได้ และแพ็กเกตนี้จะไม่มีการรอรับค่ากลับจากชุดควบคุมการแสดงผล

ID (8 bits)		0	0	1	0	2
0	0	1				
Checksum (8 bits)						

รูป 3.16 แพ็กเกตสำหรับระงับการแสดงผล

4) Commands = 3 : Send to board แพ็กเกตที่มีค่า Commands = 2 เป็นแพ็กเกตชนิดแพ็กเกตข้อมูล ดังนั้นจึงมีการ SYN และทำ Handshake เพื่อความถูกต้องของข้อมูล คำสั่งนี้มีไว้เพื่อส่งรูปแบบการแสดงผลที่ออกแบบไปจัดเก็บไว้ในชุดควบคุมการแสดงผล โดยต้องทำการระบุ ID บอร์ดที่ต้องการนำรูปแบบการแสดงผลไปเก็บในฟิลด์ ID และระบุหมายเลขของรูปแบบการแสดงผลที่ฟิลด์ Pattern เนื่องจากปริมาณข้อมูลที่ได้คำนวณไว้ต่อหนึ่งรูปแบบนั้นมีขนาด 2,000 bytes จึงได้ทำการแบ่งข้อมูลออกเป็น 2 ส่วน ส่วนละ 1000 bytes เพื่อส่งข้อมูลออกไป

ID (8 bits)		0	1	0	0	3
Patterns (4 bits)	0	1				
Checksum (8 bits)						

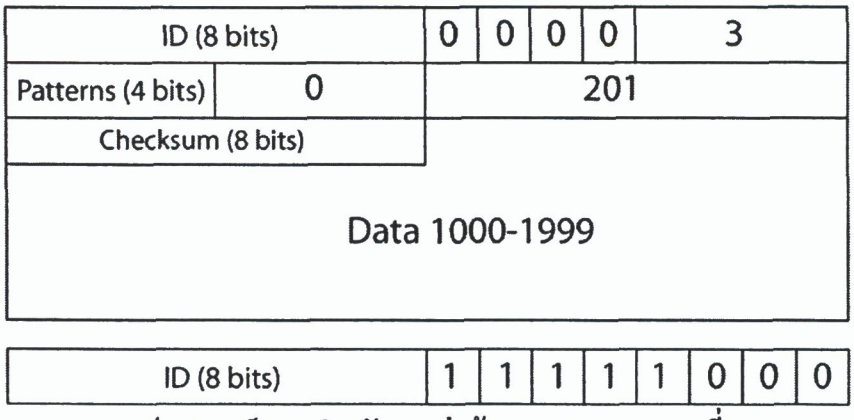
ID (8 bits)	1	1	1	1	1	1	0	0

รูป 3.17 แพ็กเกตสำหรับการเปิดการเชื่อมต่อ

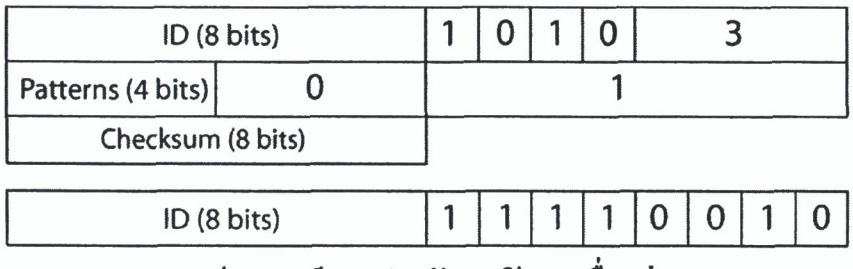
ID (8 bits)		1	0	0	0	3
Patterns (4 bits)	0	201				
Checksum (8 bits)						
Data 0-999						

ID (8 bits)	1	1	1	1	0	0	0	0

รูป 3.18 แพ็กเกตสำหรับการส่งข้อมูล 1000 bytes แรก



รูป 3.19 แพ็กเกตสำหรับการส่งข้อมูล 1000 bytes ชุดที่ 2



รูป 3.20 แพ็กเกตสำหรับการปิดการเชื่อมต่อ

3.5 การเก็บข้อมูลในหน่วยความจำอีอีพรอม

บอร์ดควบคุมการทำงานของชุดแสดงผลจะต้องนำรูปแบบการแสดงผลที่ได้รับมาจัดเก็บในหน่วยความจำอีอีพรอมก่อน หลังจากนั้นจึงทำการโหลดรูปแบบการแสดงผลออกมาเพื่อใช้แสดงผลต่อไป

รูปแบบการเก็บข้อมูลในหน่วยความจำที่นำมาใช้คือการจัดเก็บโดยระบุตำแหน่งที่แน่นอนของแต่ละรูปแบบ (Fixed-block memory) แบบตายตัว ซึ่งจะทำให้สามารถเก็บรูปแบบของการแสดงผลไว้เป็นสัดส่วน ทั้งความยาวของรูปแบบ ก็จะถูกกำหนดตายตัวตามขนาดของหน่วยความจำในแต่ละ block

3.5.1 การคำนวณพื้นที่หน่วยความจำที่ต้องใช้นั้นพิจารณาจากการใช้งาน

1) แต่ละบอร์ดใช้โซลินอยด์วาล์ว 16 วาล์ว (16 bits)
2) แต่ละบอร์ดจัดเก็บรูปแบบได้ 8 รูปแบบ
3) แต่ละรูปแบบแสดงผลนาน 40 วินาที
4) แต่ละรูปแบบตัดน้ำด้วยความเร็วสูงสุด 25 ครั้ง/วินาที

ซึ่งสามารถคำนวณได้จากสมการ

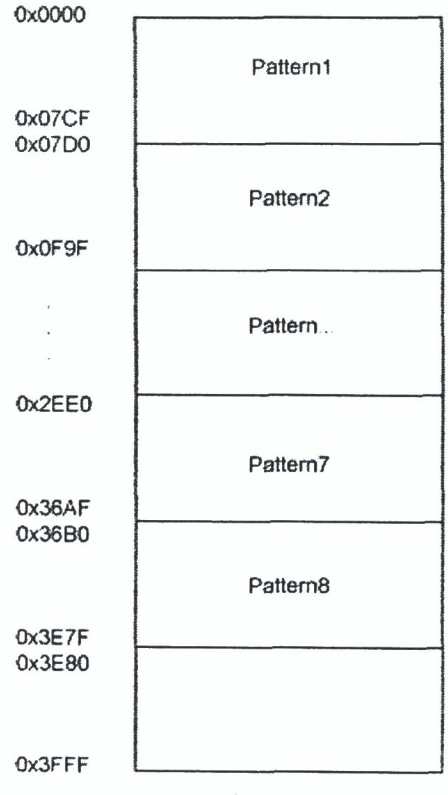
จำนวนวาล์ว x จำนวนรูปแบบ x เวลาในการแสดงผล x อัตราเร็วในการตัดน้ำ (3.1)

ซึ่งจากการคำนวณ = 16 x 8 x 40 x 25 = 128,000 bits (15,625 kbyte) เพราะฉะนั้น ขนาดของหน่วยความจำสูงสุดคือ 128,000 bits หรือ 15,625 bytes โดยแต่ละรูปแบบจะถูกกำหนดขนาดสูงสุดไว้ที่ 16,000 bits หรือ 1,953 bytes

3.5.1.1 การเก็บรูปแบบลงในหน่วยความจำ

ซึ่งในแต่ละรูปแบบเก็บดังนี้

1) รูปแบบที่ 1 เก็บ ณ ตำแหน่งที่ 0x0000 – 0x07CF
2) รูปแบบที่ 2 เก็บ ณ ตำแหน่งที่ 0x07D0 – 0x0F9F
3) รูปแบบที่ 3 เก็บ ณ ตำแหน่งที่ 0x0FA0 – 0x176F
4) รูปแบบที่ 4 เก็บ ณ ตำแหน่งที่ 0x1770 – 0x1F3F
5) รูปแบบที่ 5 เก็บ ณ ตำแหน่งที่ 0x1F40 – 0x270F
6) รูปแบบที่ 6 เก็บ ณ ตำแหน่งที่ 0x2710 – 0x2EDF
7) รูปแบบที่ 7 เก็บ ณ ตำแหน่งที่ 0x2EE0 – 0x36AF
8) รูปแบบที่ 8 เก็บ ณ ตำแหน่งที่ 0x36B0 – 0x3E7F

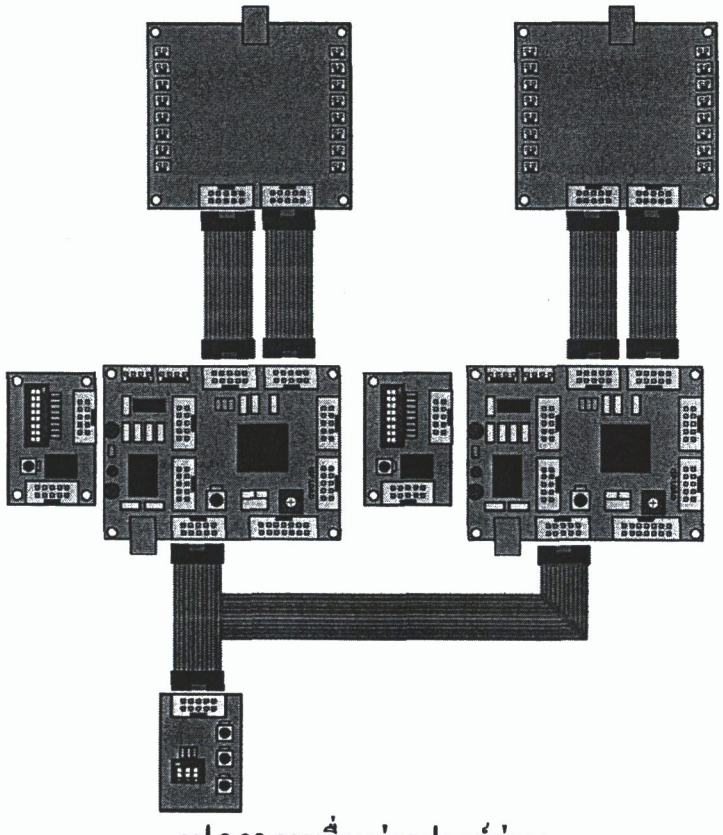


รูป 3.21 การกำหนดตำแหน่งของรูปแบบการแสดงผล

3.6 การเชื่อมต่ออุปกรณ์ต่างๆ

การออกแบบอุปกรณ์ควบคุมต่างๆ คำนึงเรื่องความต้องการที่หลากหลายของผู้ใช้เป็นหลัก จึงเป็นเหตุผลที่ทำการแบ่งการควบคุมการแสดงผลออกเป็นบอร์ดย่อยๆ ซึ่งผู้ใช้สามารถกำหนดขนาดของการแสดงผลได้ตามต้องการ (กำหนดจำนวนบอร์ดควบคุม) ดังนั้นจึงมีการออกแบบอุปกรณ์ต่อพ่วงต่างๆออกเป็นส่วนประกอบย่อยๆ มีรายละเอียดดังนี้

1) บอร์ดกลางสำหรับเชื่อมต่อกับคอมพิวเตอร์
2) บอร์ดสำหรับระบุ ID และเพิ่มหน่วยความจำ
3) บอร์ด ET base AVR
4) Interface board



รูป 3.22 การเชื่อมต่ออุปกรณ์ต่างๆ

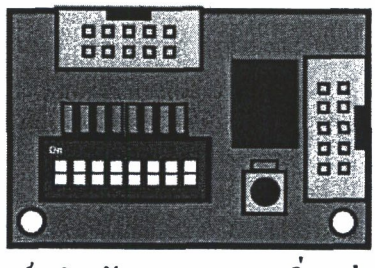
3.6.1 บอร์ดกลางสำหรับเชื่อมต่อกับ PC

เป็นบอร์ดที่ใช้เชื่อมต่อระหว่างคอมพิวเตอร์กับบอร์ด ET-Base AVR ผ่านพอร์ตอนุกรม ซึ่งจะเชื่อมต่อกันแบบ Bus topology ภายในบอร์ดมีการเพิ่ม Pattern_SW , PLAY_SW , STOP_SW, PLAYALL_SW เพื่อใช้ในการควบคุมในการทำงานแบบไม่เชื่อมต่อกับคอมพิวเตอร์

รูป 3.23 บอร์ดกลางสำหรับเชื่อมต่อกับ PC

3.6.2 บอร์ดสำหรับระบุ ID และ เพิ่มหน่วยความจำ

เนื่องจากมีการแยกการควบคุมโซลินอยด์วาล์วออกไปตามบอร์ดต่างๆ จึงต้องมีการระบุ ID ของแต่ละบอร์ดเพื่อใช้ในการติดต่อกับคอมพิวเตอร์ อีกทั้งหน่วยความจำภายใน ET-Base AVR ไม่เพียงพอสำหรับการจัดเก็บรูปแบบการแสดงผลทั้งหมดจึงจำเป็นต้องมีการเพิ่มหน่วยความจำเข้าไป และมี TEST_SW เพิ่มเข้ามาเพื่อทำการทดสอบน้ำตก



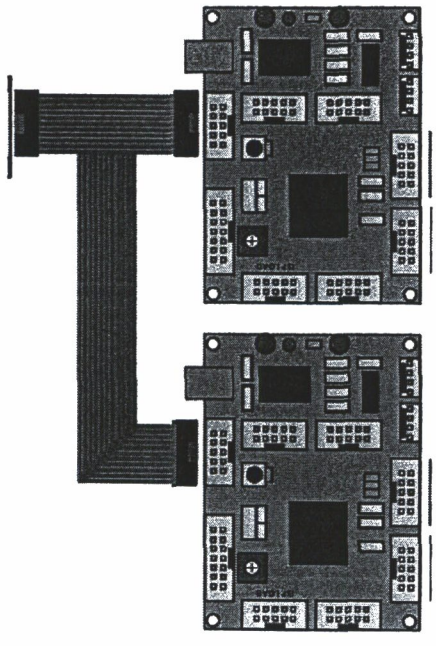
รูป 3.24 บอร์ดสำหรับระบุ ID และ เพิ่มหน่วยความจำ

3.6.3 บอร์ด ET base AVR

บอร์ดแต่ละบอร์ดถูกออกแบบมาให้สามารถควบคุมการทำงานของโซลินอยด์วาล์วได้บอร์ดละ 16 วาล์ว เหตุผลที่ทำการแบ่งการควบคุมวาล์วไปหลายบอร์ด คือต้องการให้ชุดควบคุมสามารถย่อ-ขยายอัตราส่วนของการแสดงผลได้ (ลด – เพิ่มจำนวนวาล์ว)

ไมโครคอนโทรลเลอร์ AVR ที่เลือกใช้ คือ ATmega128 เป็นไมโครคอนโทรลเลอร์ ที่ผลิตโดยบริษัท ATMEL (ผู้นำทางด้านไมโครคอนโทรลเลอร์ตระกูล MCS-51) AVR จัดเป็นไมโครคอนโทรลเลอร์ตระกูลใหม่จาก ATMEL มีสถาปัตยกรรมแบบ RISC (Advanced RISC architecture) คือหนึ่งคำสั่งทำงานใช้สัญญาณนาฬิกาเพียง 1 ลูก (instructions in a single clock cycle) เป็นไมโครคอนโทรลเลอร์ที่มีประสิทธิภาพและความสามารถสูง แบ่งออกเป็นหลายอนุกรม

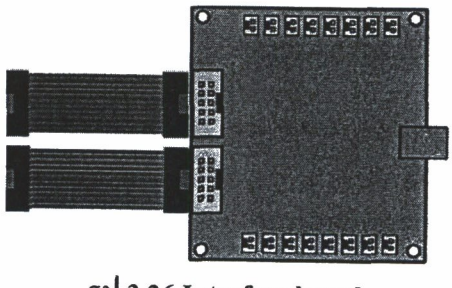
ในแต่ละอนุกรมยังแบ่งออกเป็นหลายเบอร์ เพื่อรองรับความต้องการที่แตกต่างของผู้ใช้งาน ในขณะที่ยังคงความประสิทธิภาพที่เท่ากัน



รูป 3.25 บอร์ด ET-Base AVR

3.6.4 Interface board

อุปกรณ์ที่ใช้ในการควบคุมการปล่อยน้ำ คือโซลินอยด์วาล์ว 24VDC ขนาด 2 หุน ตัวเรือนทองเหลือง เป็นอุปกรณ์ที่สามารถเปิด-ปิดน้ำได้โดยใช้ไฟฟ้าในการควบคุม แต่โซลินอยด์วาล์ว โดยปกติจะใช้กระแสโดยเฉลี่ยประมาณ 250-500 mA ดังนั้นจึงต้องมี Interface Board เพื่อเพิ่มกระแสและความต่างศักย์ให้กับตัวโซลินอยด์วาล์ว



รูป 3.26 Interface board

3.7 ปริมาณน้ำที่ต้องใช้หมุนเวียนในระบบ

ในกรณีที่ต้องการให้น้ำที่ไหลออกมาจากแต่ละวาล์วมีความสม่ำเสมอและมีปริมาณน้ำที่เท่ากัน นั้นจำเป็นต้องควบคุมแรงดันน้ำในแต่ละวาล์วให้คงที่ วิธีการควบคุมแรงดันน้ำเพื่อให้ได้แรงดันน้ำที่เกือบคงที่ ที่ง่ายและไม่จำเป็นต้องลงทุนมากนัก คือการทำระบบน้ำล้น โดยที่แรงดันของน้ำจะถูกรักษาระดับไว้ที่ผิวน้ำเสมอ ซึ่งต้องทำการคำนวณปริมาณน้ำที่ต้องใช้หมุนเวียนในระบบให้เพียงพอต่อการรักษาแรงดันของน้ำไว้ โดยทำการคำนวณจากกรณีที่น้ำจะไหลมากที่สุดนั้นคือกรณีที่วาล์วทั้ง 16 วาล์วเปิดพร้อมกัน

สามารถคำนวณปริมาตรของน้ำที่ไหลในแต่ละวาล์วได้จากสูตร

$$Q = Av \qquad (3.2)$$

Q คือ ปริมาตรน้ำที่ไหล

A คือ พื้นที่หน้าตัดของวาล์ว

V คือ ความเร็วของน้ำที่ไหล

และความเร็วของน้ำที่ไหลสามารถคำนวณได้จากสูตรการคลื่นที่ในแนวดิ่ง ดังนี้

$$v = u + gt \qquad (3.3)$$

v คือ ความเร็วปลาย

u คือ ความเร็วต้น

g คือ ความเร่งในแนวดิ่ง

t คือ เวลา

ซึ่งสมมติให้ ความเร็วต้นมีค่าเท่ากับ 0,น้ำเคลื่อนที่ด้วยความเร่งในแนวดิ่งมีค่าเท่ากับ 9.8 m^2/s, และเวลาที่ใช้มีค่าเท่ากับ 1 วินาที จะได้ 0+(9.8x1) = 9.8 เมตร/วินาที

และสามารถหาพื้นที่หน้าตัดของวาล์ว ได้จากสูตร

$$\text{พื้นที่วงกลม} = \pi r^2 \qquad (3.4)$$

ซึ่งวาล์วที่ใช้มีรัศมีขนาด 1 หุน หรือ 0.3125 เซนติเมตร ดังนั้นพื้นที่หน้าตัดของวาล์วมีขนาด $3.14 \times 0.3125^2 = 0.31$ ตารางเซนติเมตร

เมื่อได้พื้นที่หน้าตัดและความเร็วของน้ำที่ไหลมา ก็จะสามารถคำนวณหาค่าปริมาตรน้ำที่ไหลได้ดังนี้ 0.31x9.8 = 3.038 ลิตร/วินาที

ในกรณีที่ใช้วาล์วทั้งหมด 32 วาล์วจะต้องใช้น้ำ ทั้งหมด 3.3038x32 = 105.72 ลิตร/วินาที หรือ 6,343.2 ลิตร/นาที

3.8 การออกแบบวงจรสำหรับจ่ายไฟให้กับโซลินอยด์วาล์ว

โซลินอยด์วาล์วที่เลือกใช้ในการเปิด-ปิดน้ำนั้น ต้องใช้ไฟ 24 VDC 300-500 mA ต่อวาล์ว ดังนั้นจึงได้ออกแบบวงจรจ่ายไฟโดยใช้เฟท และทรานซิสเตอร์มาต่อร่วมกันเป็นวงจรผสม โดยทรานซิสเตอร์นั้นจะทำหน้าที่ขยายกระแสไฟฟ้าสัญญาณเอาต์พุตจากบอร์ด ET Base AVR เพื่อรักษาระดับแรงดันไฟฟ้าให้เพียงพอที่จะไปกระตุ้นการทำงานของเฟท ให้ขยายแรงดันจาก 5V เป็น 24V เพื่อจ่ายไฟให้กับโซลินอยด์วาล์ว ซึ่งในการใช้งานทรานซิสเตอร์นั้น จำเป็นต้องคำนวณหาค่าความต้านทานที่ขา B ของทรานซิสเตอร์ เพื่อให้ได้มาซึ่งกระแสไฟฟ้าที่ต้องการจากขา C ซึ่งมีสูตรการคำนวณดังนี้

$$I_C = \beta I_B \qquad (3.5)$$

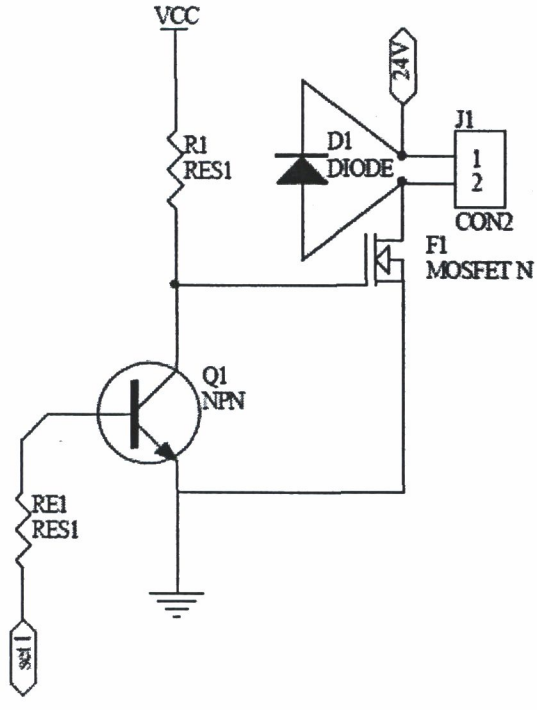
โดยที่ทรานซิสเตอร์ที่เลือกนำมาใช้คือ BD139 มีค่า β เฉลี่ยอยู่ที่ 200 และโซลินอยด์วาล์วที่นำมาใช้ต้องการกระแสไฟฟ้าประมาณ 300-500 mA ซึ่งในที่นี้จะจำกัดกระแสไว้ที่ 300 mA ซึ่งเพียงพอที่จะทำให้วาล์วทำงานได้ ดังนั้น จึงสามารถคำนวณหาค่ากระแสที่ขา B ได้จากสูตร

$$I_B = I_C/\beta \qquad (3.6)$$

ซึ่งจะได้กระแสไฟฟ้าที่ขา B มีค่าเท่ากับ 300/(1,000*200) = 1.5 mA หลังจากนั้นนำไปคำนวณหาค่าความต้านทานที่ต้องใช้ จากสูตร

$$R=V/I \qquad (3.7)$$

ดังนั้นจะได้ค่าความต้านทานที่ต้องใช้อย่างน้อยมีค่าเท่ากับ (5x1,000)/1.5 = 3333.33 Ω



รูป 3.27 วงจรสำหรับจ่ายไฟให้โซลินอยด์วาล์ว

3.9 ความเร็วในการตัดน้ำที่เหมาะสม

เนื่องจากรูปแบบการแสดงผลของระบบ เป็นการปล่อยให้น้ำร่วงลงมาด้วยความเร่งตามแรงโน้มถ่วงของโลก ดังนั้นจึงต้องคำนวณหาความเร็วในการตัดในที่เหมาะสมที่สามารถทำให้สามารถมองเห็นรูปแบบการแสดงผลได้ชัดที่สุด

ซึ่งสามารถคำนวณหาระยะเวลาที่น้ำจะตกถึงพื้นได้จากสูตร

$$s = ut+(gt^2)/2 \qquad (3.7)$$

s คือ การกระจัด

u คือ ความเร็วต้น

t คือ เวลา

g คือ ความเร่งในแนวดิ่ง

จากเวลาที่ได้สามารถนำมาคำนวณหาจำนวนครั้งที่สามารถเปิด-ปิดได้สูงสุด ก่อนที่น้ำจะตกถึงพื้น ได้ดังนี้

ตาราง 3.1 จำนวนครั้งที่สามารถเปิด-ปิดได้สูงสุด ก่อนที่น้ำจะตกถึงพื้น

ระยะความสูงจากพื้น	ระยะเวลาที่น้ำจะตกถึงพื้น	จำนวนครั้งที่สามารถเปิด-ปิดได้สูงสุด ก่อนที่น้ำจะตกถึงพื้น		
		ตัดน้ำด้วยความเร็ว 25 ครั้ง/วินาที	ตัดน้ำด้วยความเร็ว 40 ครั้ง/วินาที	ตัดน้ำด้วยความเร็ว 100 ครั้ง/วินาที
1	0.452	11	18	45
2	0.639	15	25	63
3	0.782	19	31	78
4	0.903	22	36	90