บทที่ 3

การออกแบบโปรแกรม

(Software Design)

ในส่วนของ Software ที่ใช้ควบคุมการทำงานของวงจรที่ออกแบบขึ้น จะแบ่งออกเป็น 2 ส่วนหลักๆ ได้แก่ ส่วนของโปรแกรมที่ทำงานในระดับต่ำ คือติดต่อกับ 8255 เพื่อให้ทำหน้าที่เป็น Successive Approximation Register กับส่วนของโปรแกรมที่ทำงานในระดับสูง คือจะคำนวนและแสดงค่าน้ำหนัก สำหรับความสามารถพิเศษอื่นๆ หากต้องการนำไปประยุกต์ใช้ก็สามารถเพิ่มเติมได้ โดยการเขียน Software ให้ทำงานตามที่ต้องการ ซึ่งในงานวิจัยนี้จะแสดงเฉพาะแนวทางและตัวอย่างของการเขียนโปรแกรมควบคุมวงจรเท่านั้น

3.1 ส่วนของโปรแกรมในระดับต่ำ

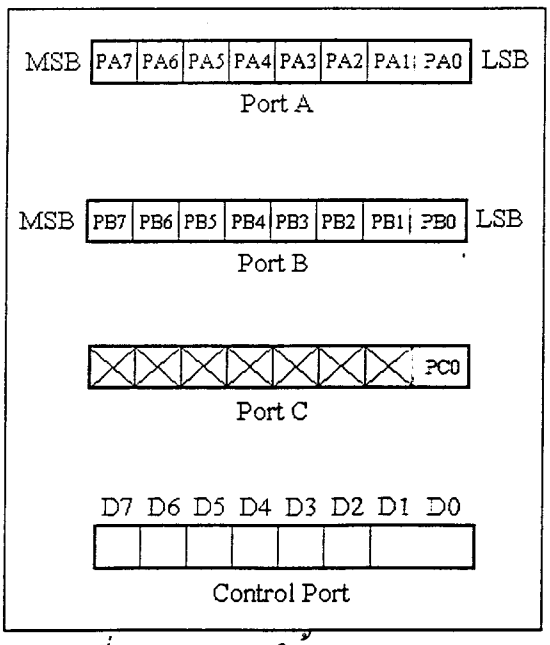
โปรแกรมในส่วนนี้จะต้องควบคุมให้ 8255 ทำหน้าที่เป็น SAR และต้องคอยตรวจสอบสัญญาณ EOC ว่าเกิดการเปลี่ยนแปลงระดับหรือไม่ สำหรับการติดต่อกับ 8255 จะมี Address ของ Port ต่างๆ ดังนี้

Address ของ Port A คือ 0300H

Address ของ Port B คือ 0301H

Address ของ Port C คือ 0302H

Address ของ Control Port คือ 0303H



ภาพที่ 3-1 แสดงการใช้งาน Port ของ 8255

Port A ถูกออกแบบให้ทำงานเป็น Output Port สำหรับส่งข้อมูล 8 บิตบนไปให้ DAC
Port B ถูกออกแบบให้ทำงานเป็น Output Port สำหรับส่งข้อมูล 8 บิตล่างไปให้ DAC
Port C ถูกออกแบบให้ทำงานเป็น Input Port สำหรับรับสัญญาณ EOC โดยใช้งานเฉพาะบิต PC0
สำหรับ Control Port แต่ละบิตจะมีความหมายดังนี้

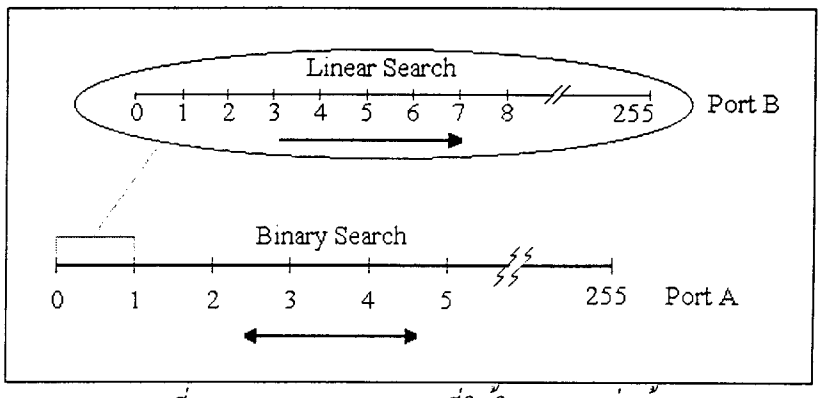
D7	**Mode Select Flag**
	1 = Active; 0 = Inactive
D6 D5	**Mode Selection (Group A)**
	00 = Mode 0
	01 = Mode 1
	1X = Mode 2
D4	**Port A**
	1 = Input; 0 = Output
D3	**Port C (Upper)**
	1 = Input; 0 = Output
D2	**Mode Selection (Group B)**
	0 = Mode 0; 1 = Mode 1
D1	**Port B**
	1 = Input; 0 = Output
D0	**Port C (Lower)**
	1 = Input; 0 = Output

เนื่องจากต้องการให้ Port A และ B เป็น Output Port โดยมีการทำงานแบบ Simple I/O หรือ Mode 0 และ ให้ Port C เป็น Input Port ทำงานแบบ Simple I/O ดังนั้นจะได้ Control Word ของ 8255 ดังนี้

D7	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**		
1	0	0	0	1	0	0	1	=	89H

สำหรับ Port C เนื่องจากมีการใช้งานเฉพาะบิต PC0 ดังนั้นข้อมูลที่อ่านได้จาก Port C จึงมีค่าที่เป็นไปได้เพียง 2 ค่าได้แก่ FFH เมื่อข้อมูลที่ส่งให้กับ 8255 มีค่ามากกว่าน้ำหนักจาก Load cell และ FEH เมื่อข้อมูลที่ส่งให้กับ 8255 มีค่าน้อยกว่าน้ำหนักจาก Load cell เท่านั้น

Algorithm ของการหาค่าน้ำหนักจะใช้ 2 วิธีคือ Binary search และ Linear search มีหลักการในการทำงานดังนี้



ภาพที่ 3-2 แสดง Algorithm ที่ใช้ในการหาค่าน้ำหนัก

เนื่องจาก Port B ทำงานเป็นตัวส่งข้อมูล 8 บิตล่าง และ Port A ทำงานเป็นตัวส่งข้อมูล 8 บิตบน ดังนั้นข้อมูลทั้งหมดจะเริ่มต้นจาก 0000H ไปจนถึง FFFFH มีจำนวนทั้งสิ้น 2^{16} ค่า ซึ่งหากใช้วิธีการแบบ Linear search จะทำให้การหาค่าน้ำหนักเป็นไปได้ช้ามาก วิธีการแก้ไขคือ แบ่งการส่งข้อมูลออกเป็น 2 วิธีคือ การส่งข้อมูลให้กับ Port A จะเป็นแบบ Binary search ส่วน ข้อมูลที่ส่งให้กับ Port B เป็นแบบ Linear search ซึ่งก็คือทำการเข้าถึงค่าอย่างหยาบก่อน จากนั้นหากพบว่าสัญญาณเปรียบเทียบมีการเปลี่ยนแปลง จึงจะทำการเข้าถึงค่าอย่างละเอียด ด้วยวิธีการนี้จะช่วยให้เร็วขึ้นเพราะไม่ต้องส่งค่าที่เป็นไปได้ทุกค่าออกไปทาง Port A และ B

3.2 ส่วนของโปรแกรมในระดับสูง

เมื่อทราบค่าที่ส่งให้กับ Port A และ B ที่ทำให้เกิดการเปลี่ยนแปลงค่าน้ำหนัก ก็จะสามารถคำนวนหาน้ำหนักที่แท้จริงได้ดังนี้

น้ำหนักสูงสุดที่วัดได้คือ 500 กก.

ความละเอียดของการแปลงคือ $1/2^{16} = 1/65536$

Port B เป็น LSB ความละเอียดต่ำสุดที่เป็นไปได้คือ 1

ดังนั้นน้ำหนักที่อ่านได้จาก Port B จะมีค่า = 500 * Port B * 1/65536 กก.

Port A เป็น MSB ความละเอียดต่ำสุดที่เป็นไปได้คือ 256

ดังนั้นน้ำหนักที่อ่านได้จาก Port A จะมีค่า = 500 * Port A * 1 / 256 กก.

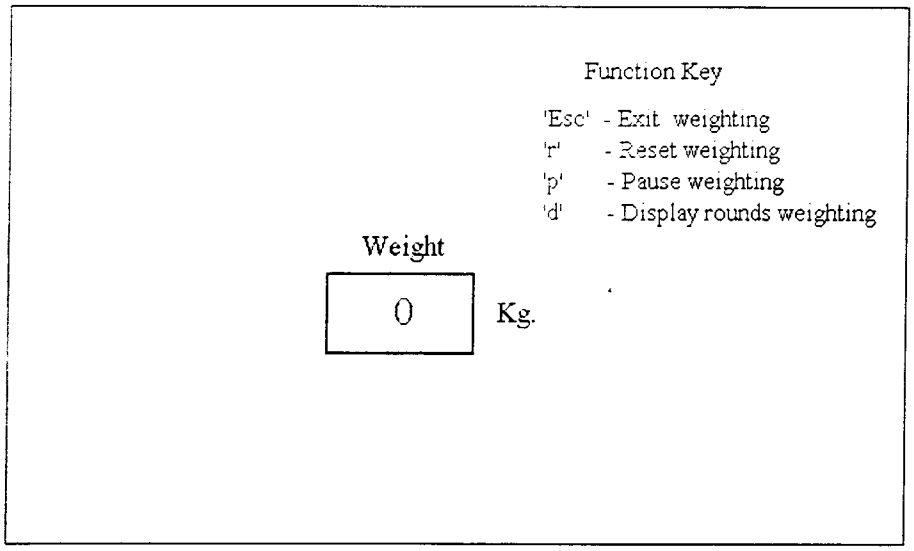
เพราะฉะนั้น น้ำหนักที่แท้จริงคือ ผลรวมของน้ำหนักที่อ่านได้จาก Port A และ B

= 500*(Port A / 256 + Port B / 65536) กก.

จัดรูปใหม่จะได้

น้ำหนัก = (500 / 256) * (Port A + (Port B / 256))
= 1.95 * (Port A + (Port B / 256)) กก.

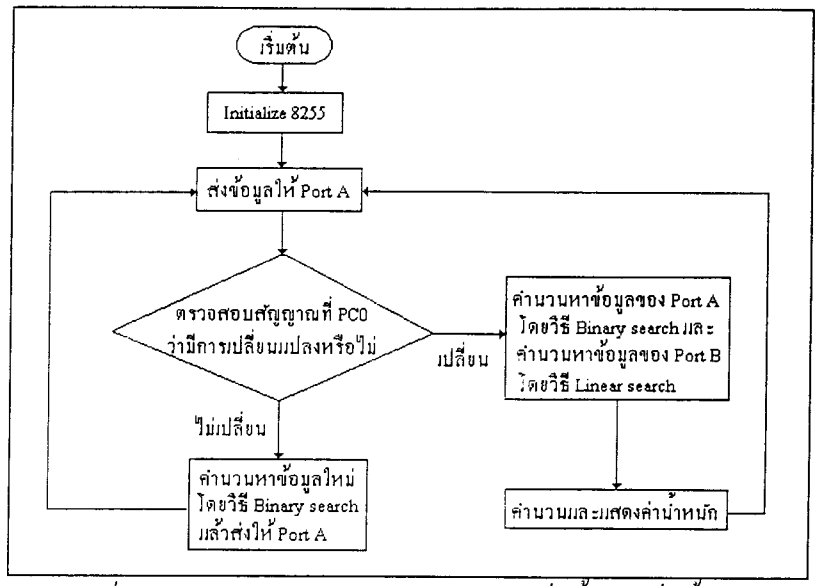
จากสมการข้างต้น ก็จะนำไปเขียนในโปรแกรมเพื่อทำการคำนวนและแสดงออกมาเป็นค่าของน้ำหนักในหน่วยกิโลกรัม ซึ่งหากมีการนำไปใช้กับระบบเครื่องชั่งที่มีคานช่วยรับน้ำหนักก็สามารถจะแก้ไขค่าของน้ำหนักที่แสดงได้โดยการคูณด้วยค่าคงที่ที่เหมาะสมลงในสมการคำนวนน้ำหนัก และส่วนอื่นๆ ของโปรแกรมก็ได้แก่การติดต่อกับผู้ใช้ โดยในการวิจัยนี้ได้เสนอเฉพาะการแสดงผลอย่างพื้นฐานเท่านั้น ซึ่งก็ได้ทำการออกแบบหน้าจอแสดงผลไว้ดังภาพที่ 3-3



ภาพที่ 3-3 แสดงหน้าจอของโปรแกรมตัวอย่างสำหรับการแสดงค่าน้ำหนัก

3.3 โครงสร้างการทำงานของโปรแกรม

จากรายละเอียดของโปรแกรมที่ได้กล่าวมาแล้วในหัวข้อ 3.1 และ 3.2 สามารถเขียนโครงสร้าง (Flowchart) การทำงานของโปรแกรมที่ใช้ในการแสดงค่าน้ำหนักได้ดังภาพที่ 3-4 ส่วนรายละเอียดทั้งหมดของโปรแกรมจะอยู่ในภาคภนวก ก.



ภาพที่ 3-4 แสดง Flowchart ของโปรแกรมที่ใช้แสดงค่าน้ำหนัก

โปรแกรมตัวอย่างพัฒนาโดยใช้ Turbo C V2.0 บนเครื่องคอมพิวเตอร์ PC 486 DX-33 แล้วนำ Executable file ที่ได้ไป Run บนเครื่อง PC/AT 286 จอภาพแบบ Monochrome ดังนั้นใน directory ของ Executable file จะต้องมี Driver ของจอภาพที่จะใช้งาน (*.BGI) และ file ตัวอักษรที่ใช้ในการแสดงผล (*.CHR) อยู่ด้วยเสมอจึงจะทำงานได้ถูกต้อง