หอสมุดกลาง จุฬาลงกรณมหาวิทยาลัย

บทที่ 4

ระบบโปรแกรมการสุ่มค่าความเข้มของภาพ

จากที่ได้กล่าวถึงการเกิดภาพของระบบโทรทัศน์ขาวดำตลอดจนระบบของเครื่องไมโครคอมพิวเตอร์ที่ใช้ในการทำวิทยานิพนธ์นี้ พร้อมทั้งส่วนต่าง ๆ อันเกี่ยวเนื่องกับระบบของเครื่องไมโครคอมพิวเตอร์ที่ใช้โดยละเอียดแล้วนั้น เพื่อผู้อ่านจะได้นำเกล็ดความรู้ที่ได้รับมาทำความเข้าใจได้กับระบบของโปรแกรมซึ่งเป็นหัวใจสำคัญของวิทยานิพนธ์เรื่องนี้

ระบบของโปรแกรมที่ใช้ในการทำวิทยานิพนธ์นี้ได้แบ่งออกเป็น 2 ตอนด้วยกันคือ

1. ระบบโปรแกรมการสุ่มค่าความเข้มของภาพ
2. ระบบโปรแกรมการจัดตัวอักษรที่ใช้และการพิมพ์ภาพ

ระบบโปรแกรมทั้งสองตอนนี้มีลักษณะการทำงานต่อเนื่องกันโดยระบบโปรแกรมแรกจะทำการสุ่มค่าความเข้มของภาพจากสัญญาณภาพในแต่ละเส้นมาเก็บไว้ในหน่วยความจำจนครบตามต้องการแล้ว ข้อมูลเหล่านั้นจะถูกแปลงให้อยู่ในรูปของตัวอักษรและพิมพ์เป็นรูปภาพที่สมบูรณ์ต่อไปโดยอาศัยระบบโปรแกรมตอนที่สองนั้น แต่เพื่อให้ผู้อ่านทำความเข้าใจกับระบบโปรแกรมทั้งสองได้ง่ายและกระจ่างชัดขึ้น จึงได้แยกอธิบายเป็นเรื่อง ๆ ไป ในบทนี้จะกล่าวถึงระบบโปรแกรมการสุ่มค่าความเข้มของภาพโดยเริ่มจากที่มาและสมมติฐานของการสร้างระบบโปรแกรมนี้ตลอดจนถึงการทำงานของระบบโปรแกรมโดยละเอียด ส่วนระบบโปรแกรมการจัดตัวอักษรและการพิมพ์ภาพจะกล่าวถึงในบทต่อไป

ที่มาและสมมติฐานเบื้องต้นในการสุ่มค่าความเข้มของภาพ

ก. การใช้โปรแกรมการสุ่มค่าความเข้มของภาพอย่างง่าย ๆ เพื่อทดสอบการทำงานของแผง D+7A

เมื่อได้จัดระบบของเครื่องไว้เรียบร้อยแล้วตามบทที่ 3 ก็เริ่มจับภาพที่ต้องการโดย

ผ่านกล้องถ่ายโทรทัศน์วงจรปิดนั้น จะได้สัญญาณภาพรวมเป็นสัญญาณอนาลอกและเมื่อสัญญาณภาพรวมนี้ผ่านแผง D+7A ก็จะถูกเปลี่ยนเป็นสัญญาณดิจิตอลตามคุณสมบัติของแผงนี้ ดังนั้นเมื่อหน่วยควบคุมมีคำสั่งให้สุ่มค่าความเข้มของภาพเข้ามา สัญญาณภาพรวมที่ผ่านเข้ามาก็จะถูกเปลี่ยนเป็นสัญญาณดิจิตอลซึ่งคือค่าความเข้มของภาพในขณะนั้น และค่านี้จะถูกเก็บไว้ในหน่วยความจำเพื่อเปลี่ยนเป็นค่าของตัวอักษรที่สัมพันธ์กับค่าความเข้มนั้นต่อไป เพราะฉะนั้นในขั้นแรกนี้จึงต้องสร้างโปรแกรมเพื่อทดสอบว่าสามารถทำงานได้หรือไม่เมื่อได้ต่อแผง D+7A นี้เรียบร้อยแล้ว

โปรแกรมทดสอบการสุ่มค่าเข้มของภาพ

ตำแหน่งของคำสั่งในหน่วยความจำ (Address)	รหัส (OP-Code)	คำสั่ง (Mnemonic)
01 00	3E CA	LD A, CA
01 02	D3 03	OUT 03
01 04	3E 27	LD A, 27
01 06	D3 03	OUT 03
01 08	21 00 02	LD HL, (0002)
01 0B	01 1F A0	LD BC, (1FA0)
01 0E	ED 32	INIR
01 10	3E FF	LD A, FF
01 12	D3 FF	OUT FF
01 14	76	HALT

การทำงานของโปรแกรมนี้เริ่มต้นด้วยการกำหนดจุดเริ่มต้นของหน่วยความจำที่จะไว้ใช้เก็บข้อมูลและให้รีจิสเตอร์ C มีค่าเป็น 1F คือเป็นหน่วยรับสัญญาณอนาลอกหน่วยที่ 7 ของแผง D+7A ที่ใช้อยู่ และให้รีจิสเตอร์ B เป็นที่เก็บจำนวนข้อมูลที่ต้องการรับเข้ามาทั้งหมดว่าเป็นจำนวนเท่าใด ในตัวอย่างนี้ให้รีจิสเตอร์ B มีค่าเป็น 160 ค่าหรือ A0 ค่าเมื่อใช้เป็นเลขฐาน 16 หลังจากนั้นแล้วโปรแกรมจะทำการรับค่าของข้อมูลซึ่งคือค่าความเข้มของภาพ

ในลักษณะสัญญานดิจิตอลเข้ามาเรื่อย ๆ และเก็บค่าลงในหน่วยความจำต่อเนื่องกันไปจนครบ 160 ค่า แล้วจึงหยุด และแสดงผลที่หน้าปัดของเครื่องไมโครคอมพิวเตอร์คือหน่วย FF เป็นไฟดับหมด แสดงว่าโปรแกรมนี้ถูกเอคเซคคิวท์เรียบร้อยแล้ว

ผลของโปรแกรมจะปรากฏดังนี้

ตำแหน่งในหน่วยความจำของเก็บข้อมูล	ค่าความเข้มของภาพที่สุ่มเข้ามาเก็บไว้ในหน่วยความจำ															
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0200	05	00	C5	05	00	01	04	C1	04	05	00	05	C4	01	05	01
0210	04	C5	05	04	00	05	C0	04	00	05	05	C0	00	05	04	00
0220	C4	05	04	04	04	C5	04	05	05	05	C1	05	00	01	00	C1
0230	00	04	04	00	C4	04	04	04	00	C1	04	04	04	05	C4	01
0240	00	04	01	C0	00	05	05	05	C0	04	05	05	00	C5	04	05
0250	04	01	C0	05	00	01	05	C0	00	05	04	04	C0	05	05	04
0260	04	C5	00	05	04	05	C0	05	00	01	01	C1	00	01	04	00
0270	C4	04	04	04	00	C1	04	05	05	05	C0	05	00	01	04	C1
0280	00	05	04	05	C0	05	05	04	00	C5	01	05	04	05	C5	05
0290	00	01	05	C5	04	04	04	01	C0	05	05	04	04	C5	01	05

ตารางที่ 4.1 ค่าความเข้มของภาพที่สุ่มเข้ามาเก็บไว้ในหน่วยความจำและพิมพ์ออกมาทางเครื่องพิมพ์ในรูปของเลขฐาน 16

การแสดงผลหรือการสั่งให้นำค่าที่เก็บไว้ในหน่วยความจำออกมาพิมพ์กระทำโดยใช้คุณสมบัติของแซปโมนิเตอร์ (Zap monitor) ซึ่งเป็นส่วนของโปรแกรมโมนิเตอร์ (Program monitor) ที่ช่วยให้ข้อมูลในหน่วยความจำปรากฏได้ทางเครื่องพิมพ์เมื่อผู้ใช้ต้องการและเมื่อใช้คำสั่ง D0200 029F ที่เครื่องพิมพ์แล้วข้อมูลที่เก็บไว้ในหน่วยความจำที่ตำแหน่งดังกล่าวจะปรากฏดังตารางที่ 4.1 ผลที่ได้ออกมานั้นคือค่าความเข้มของภาพ ณ ขณะที่มีคำสั่งให้รับข้อมูลเข้ามาเป็นสัญญานดิจิตอล และปรากฏค่าในรูปของเลขฐาน 16 ดังตัวอย่างข้างต้นซึ่งช่วยให้อ่านและทำความเข้าใจได้ง่ายกว่าที่จะให้ผลออกมาในรูปของเลขฐาน 2

ข. สมมติฐานในการสร้างโปรแกรมการสุ่มค่าความเข้มของภาพ

เมื่อได้ผลปรากฏดังกล่าวแล้วจึงนำมาพิจารณาดูว่าการใช้โปรแกรมทดสอบข้างต้นนั้นสามารถรับค่าความเข้มของภาพจากสัญญานทางแนวนอนใน 1 เส้นมาได้เป็นจำนวนเท่าไร

เนื่องจากความถี่ของภาพเป็น 50 เฮิทซ์และ 625 เส้นดังนั้นจึงสามารถคำนวณได้ว่าการสะแกนของสัญญานทางแนวนอน 1 เส้นนั้นใช้เวลา 64 ไมโครวินาทีโดยรายละเอียดของการคำนวณมีปรากฏดังนี้

ด้วยเหตุที่ใน 1 ฟิลด์ประกอบด้วยจำนวนเส้นสะแกนทางแนวนอน $\frac{625}{2}$ เส้น

เมื่อการสะแกนทั้งหมด 50 ฟิลด์ใช้เวลา 1 วินาที

นั่นคือการสะแกนทั้งหมด $50 \times \frac{625}{2}$ เส้น ใช้เวลา 1 วินาที

ดังนั้นการสะแกนทั้งหมด 1 เส้น ใช้เวลา $\frac{1 \times 2}{50 \times 625}$ วินาที

นั่นคือการสะแกนของสัญญานทางแนวนอน 1 เส้น ใช้เวลา $\frac{2 \times 10^6}{50 \times 625}$ ไมโคร-วินาที

ซึ่งเท่ากับ 64 ไมโครวินาที

และเมื่อใช้เครื่องออสซิลโลสโคป (Oscilloscope) วัดค่าสัญญานแบลงค์กิงและสัญญานซิงค์ตามแนวนอนปรากฏว่าเวลาในช่วงนี้มีค่าเป็น 10.5 ไมโครวินาที และเวลาที่ใช้ในขณะที่เป็นสัญญานภาพมีค่าเป็น 52.5 ไมโครวินาที เมื่อรวมกันแล้วจะได้เป็นเวลาของการสะแกนของสัญญานทางแนวนอน 1 เส้น ใช้เวลาทั้งหมด 63 ไมโครวินาที ซึ่งมีค่าแตกต่างจากการคำนวณ 1 ไมโครวินาที ค่าที่แตกต่างไปนี้อาจเกิดขึ้นเนื่องจากเครื่องมือออสซิลโลสโคปและจากการวัดนั้นค่าที่วัดดูด้วยสายตาซึ่งค่าแตกต่างนี้ไม่ถือเป็นความสำคัญเพราะเป็นค่าที่น้อยมาก

จากโปรแกรมทดสอบข้างต้นนั้น สังเกตุได้ว่าการที่จะรับค่าของสัญญานภาพเข้ามาได้แต่ละค่านั้น เครื่องจะต้องทำคำสั่งการรับข้อมูลเข้ามาเก็บไว้ในหน่วยความจำแล้วทำการตรวจสอบค่าที่ได้ว่ามีจำนวนครบตามความต้องการหรือยัง นั่นคือคำสั่ง INIR ในโปรแกรมนั่นเอง ซึ่งเฉพาะคำสั่งนี้ต้องใช้เวลาในการเอคเซคคิวท์ทั้งสิ้น 20 ทีสเตท ซึ่งได้กล่าวไว้แล้วว่า 1 ทีสเตทมีค่าเท่ากับ 0.5 ไมโครวินาที ดังนั้นในการรับค่าแต่ละจุดจะใช้เวลาห่างกันประมาณ 10 ไมโครวินาที จากที่กล่าวมาแล้วว่าสัญญานภาพจริงที่ปรากฏในการสะแกนตามแนวนอน 1 เส้นนั้นจะใช้เวลา

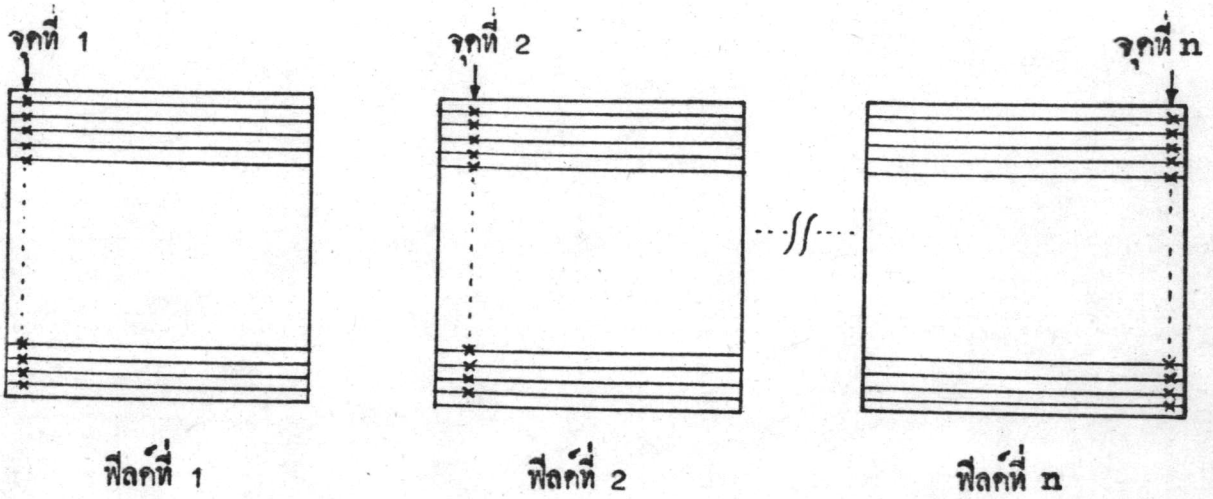
เพียง 52.5 ไมโครวินาทีเท่านั้น ถ้าคำนวณอย่างคร่าว ๆ ดูจะเห็นว่าในการสะแกนของสัญญาณทางแนวนอน 1 เส้นนั้นจะสามารถรับสัญญาณภาพมาเก็บในหน่วยความจำได้เพียง 5 จุดเท่านั้น ซึ่งเมื่อพิจารณาจากตารางที่ 4.1 ก็จะมีผลสอดคล้องกับการคำนวณด้วย นั่นคือค่าความต่างศักย์ในช่วงที่เป็นสัญญาณแบลงค์กิงและซิงค์ทั้งตามแนวนอนและตามแนวตั้งจะมีค่าโดยเฉลี่ยอยู่ระหว่าง −1 โวลท์ ถึง −0.6 โวลท์ (วัดด้วยเครื่องออสซิลโลสโคป) ซึ่งแทนได้ด้วยค่า CC ถึง EO เมื่อแปลงให้อยู่ในรูปของดิจิตอลตามคุณสมบัติของแผง D+7A และพิมพ์ออกมาเป็นเลขฐาน 16 แล้วค่าความเข้มของภาพที่จับด้วยเครื่องออสซิลโลสโคปในขณะที่เก็บข้อมูลตามตารางที่ 4.1 นั้นมีค่าอยู่ระหว่าง 0 โวลท์ ถึง 0.5 โวลท์ หรือ 00 ถึง 19 (เลขฐาน 16) ซึ่งตามตารางที่ 4.1 จะเห็นได้ว่าในการสะแกนของสัญญาณทางแนวนอน 1 เส้นนั้นจะสุ่มข้อมูลมาได้ 5 ค่าจริง ๆ คือเป็นข้อมูลในช่วงของสัญญาณภาพจริง 4 ค่า และข้อมูลในช่วงของสัญญาณแบลงค์กิงและซิงค์อีก 1 ค่า ซึ่งเป็นที่แน่นอนว่าค่าของสัญญาณภาพที่ได้มา 5 จุดนั้นไม่เพียงพอที่จะประกอบกันเป็นภาพที่สมบูรณ์ได้ และค่าที่เหมาะสมของสัญญาณภาพในแต่ละเส้นของการสะแกนตามแนวนอนที่รับเข้ามาควรมีค่าประมาณ 100 จุดขึ้นไป ดังนั้นจึงจำเป็นต้องหาวิธีแก้ไขเพื่อให้ได้ค่าของข้อมูลตามต้องการ

เนื่องจากจุดมุ่งหมายของการทำวิทยานิพนธ์เรื่องนี้ต้องการนำกล้องถ่ายโทรทัศน์วงจรปิดมาเชื่อมต่อเข้ากับระบบเครื่องไมโครคอมพิวเตอร์แล้วให้เครื่องพิมพ์พิมพ์ภาพได้โดยอาศัยระบบโปรแกรมเป็นหลัก ดังนั้นวิธีการปรับปรุงเพื่อให้ได้ข้อมูลที่เหมาะสมนั้นจึงมุ่งมาทางด้านระบบของโปรแกรม และเพื่อให้สามารถนำกล้องถ่ายโทรทัศน์วงจรปิดเครื่องใดก็ตามมาต่อเข้ากับเครื่องไมโครคอมพิวเตอร์แล้ว เครื่องพิมพ์สามารถพิมพ์ภาพได้เลยโดยมิต้องมีการแก้ไขปรับปรุงระบบของกล้องถ่ายโทรทัศน์วงจรปิดก่อนนำมาเชื่อมต่อเข้ากับเครื่องไมโครคอมพิวเตอร์อีกด้วย

ต่อจากนี้จะเริ่มพิจารณาทางด้านระบบโปรแกรมต่อไป เริ่มด้วยพิจารณาการสะแกนของสัญญาณทางแนวนอนในแต่ละเส้นใช้เวลาทั้งหมด 63 ไมโครวินาที และเป็นเวลาที่ใช้ในการเกิดสัญญาณภาพจริง ๆ 52.5 ไมโครวินาที ซึ่งปัญหาที่เกิดขึ้นนั้นเนื่องมาจากการรับค่าของสัญญาณภาพแต่ละค่านั้นใช้เวลาห่างกันประมาณ 10 ไมโครวินาที และได้ค่ามาเพียง 5 จุดของการสะแกนของสัญญาณทางแนวนอน 1 เส้นเท่านั้น ดังนั้นจึงได้พยายามศึกษาและแก้ไขปัญหาข้อนี้ให้ได้เสียก่อน และได้พบว่าสิ่งที่เป็นไปได้คือพยายามเก็บค่าของสัญญาณภาพใน 1 เส้นนั้นหลาย ๆ ครั้ง นั่นคือการสร้างภาพหนึ่งภาพจากการเก็บค่าความเข้มของภาพนั้นจากหลาย ๆ ฟิลด์นั่นเอง ซึ่ง

I 18334957

การทำเช่นนี้จะทำให้สามารถเก็บค่าของสัญญาณภาพใน 1 เส้นของการสะแกนของสัญญาณทางแนวนอนได้มากตามที่ต้องการ



รูปที่ 4.1 แสดงถึงการสุ่มค่าความเข้มของภาพจากฟิลด์จำนวนหลาย ๆ ฟิลด์

ยกตัวอย่างตามรูปที่ 4.1 นั้น จะทำการสุ่มค่าความเข้มของภาพ ณ จุดที่ 1 จากฟิลด์ที่ 1 จุดที่ 2 จากฟิลด์ที่ 2 จุดที่ 3 จากฟิลด์ที่ 3 เช่นนี้เรื่อยไปจนกระทั่งได้จำนวนค่าความเข้มของภาพใน 1 เส้นของการสะแกนตามแนวนอนครบตามต้องการ ซึ่งเมื่อพิจารณาดูทั้งฟิลด์แล้วจะพบว่าในการสุ่มค่าความเข้มของภาพจาก 1 ฟิลด์นั้นก็จะได้ค่าความเข้มของภาพ ณ จุดที่ 1 ทั้งหมดของเส้นสะแกนตามแนวนอนที่ทำการสุ่มค่าความเข้มของภาพ เพราะเวลาในการสะแกนของจุดที่ 1 กับจุดที่ 1 ในเส้นต่อไปนั้นมีช่วงระยะเวลาห่างพอที่โปรแกรมสามารถสุ่มค่าได้ เพราะฉะนั้นการสุ่มค่าความเข้มของภาพจึงต้องทำการสุ่มทั้งหมด n ฟิลด์ด้วยกัน ซึ่งแต่ละฟิลด์จะใช้เวลาในการสะแกนทั้งหมด $\frac{1}{50}$ วินาที ดังนั้นเวลาที่ใช้ทั้งหมดในการสุ่มค่าความเข้มของภาพด้วยวิธีนี้จะเป็น $\frac{n}{50}$ วินาที

คราวนี้จึงมาพิจารณาค่าของ n ว่าควรมีค่าเป็นเท่าไรจึงจะเหมาะสม ซึ่งการที่จะพิมพ์ภาพให้ได้ดีที่สุดนั้น ค่าของระดับความเข้มของภาพหรือจุดที่แทนความเข้มในระดับต่าง ๆ กันของภาพควรจะละเอียดที่สุด ดังนั้นจึงสรุปได้ในขั้นแรกว่าค่า n ควรมีค่ามากที่สุด ดังนั้นจึงต้องคำนวณหาค่า n ที่สามารถจะเป็นไปได้ เนื่องจากการสุ่มค่าความเข้มจากสัญญาณภาพนี้ได้ใช้

ระบบโปรแกรมเป็นหลัก ดังนั้นจากการทำงานให้ได้ละเอียดที่สุดคือ 1 ทีสเตท นั่นก็คือการสุ่มค่าความเข้มจากสัญญานภาพในเส้นสะแกนตามแนวนอนเส้นเดียวกันนั้นข้อมูลที่อยู่ติดกันจะใช้เวลาน้อยที่สุด 0.5 ไมโครวินาที ซึ่งไม่สามารถทำให้ละเอียดมากกว่านี้ได้เพราะถูกจำกัดโดยเวลา 1 ทีสเตทของหน่วยควบคุม ซึ่งมีค่าเท่ากับ 0.5 ไมโครวินาที ดังนั้นข้อมูลซึ่งก็คือค่าความเข้มของภาพที่สุ่มจากสัญญานภาพซึ่งสามารถสุ่มได้มากที่สุดใน 1 เส้นของการสะแกนตามแนวนอนจะมีค่าเท่ากับ $\frac{52.5}{0.5}$ นั่นคือ 105 จุด ซึ่งก็คือค่าที่มากที่สุดที่จะเป็นไปได้ของ n

จากที่กล่าวมาแล้วสรุปได้ว่าระบบโปรแกรมที่ต้องสร้างขึ้นนั้นจะต้องมีความละเอียดอย่างมากในการใช้คำสั่งแต่ละคำสั่งในโปรแกรมการสุ่มค่าความเข้มของภาพเพราะจะต้องใช้ระบบโปรแกรมที่คำนึงถึงเวลาในการเอคเซคคิวท์คำสั่งแต่ละคำสั่งอย่างละเอียดอีกด้วยจึงจะสามารถสุ่มค่าความเข้มของภาพได้เป็นจำนวนตามที่กล่าวข้างต้นแล้วนั้น นอกจากนี้แล้วยังมีปัญหาที่ต้องแก้ไขเพื่อให้ระบบโปรแกรมนี้เป็นไปได้อย่างสมบูรณ์อีกด้วย นั่นคือการที่จะสุ่มค่าความเข้มของภาพจากหลาย ๆ ฟิลด์นั้นก่อนอื่นจะต้องทราบว่าขณะนี้การสะแกนของภาพอยู่ ณ ที่ใดของฟิลด์แล้วหรือกล่าวง่าย ๆ ว่าจะต้องพยายามหาวิธีการเพื่อให้ทราบว่าการสะแกนมาถึงจุดเริ่มต้นของฟิลด์หรือจุดเริ่มต้นของเส้นสะแกนตามแนวตั้ง และได้สะแกนมาถึงจุดเริ่มต้นของเส้นสะแกนตามแนวนอนหรือยัง ซึ่งถ้าแก้ไขปัญหานี้ได้แล้วอย่างอื่นก็จะเป็นปัญหาทางด้านระบบโปรแกรมแต่เพียงอย่างเดียว

<u>วิธีการหาจุดเริ่มต้นของฟิลด์และจุดเริ่มต้นของเส้นสะแกนตามแนวนอน</u>

ด้วยจุดมุ่งหมายก็คือต้องการทราบว่าเมื่อไรสัญญานภาพจึงจะมีการสะแกนไปถึงจุดเริ่มต้นของฟิลด์ ดังนั้นจะต้องหาทางเอาสัญญานจากกล้องถ่ายโทรทัศน์วงจรปิดมาเพื่อบอกหน่วยควบคุมว่าขณะนี้ได้สะแกนมาถึงจุดเริ่มต้นของฟิลด์แล้วซึ่งสัญญานที่รับเข้ามาคือสัญญานทางแนวตั้ง ในทำนองเดียวกันก็ต้องการทราบว่าขณะนี้สัญญานภาพรวมมีการสะแกนมาถึงรุดเริ่มต้นของเส้นสะแกนตามแนวนอนหรือมาอยู่ทางซ้ายสุดของฟิลด์หรือยัง ก็ต้องทำการรับสัญญานจากกล้องถ่ายโทรทัศน์วงจรปิดคือสัญญานทางแนวนอนเข้ามาเพื่อบอกหน่วยควบคุมเช่นกัน

เนื่องจากสัญญานที่จะเข้ามาควบคุมมีถึงสองสัญญานด้วยกันคือสัญญานทางแนวตั้งและ

สัญญาณทางแนวนอน ดังนั้นจึงต้องพิจารณาว่าสัญญาณทั้งสองจะเข้าไปมีบทบาทในลักษณะเช่นใดดี เริ่มต้นด้วยการพิจารณาคุณสมบัติของหน่วยควบคุมเกี่ยวกับการทำอินเทอรัปและรีเซท

1. คุณสมบัติของการอินเทอรัป

การอินเทอรัปนั้นมีจุดประสงค์เพื่อให้อุปกรณ์จากภายนอกที่เชื่อมต่อกับเครื่องไมโครคอมพิวเตอร์สามารถป้อนสัญญาณเข้าไปยังหน่วยควบคุมเพื่อให้หน่วยควบคุมหยุดการทำงานที่ทำอยู่นั้นไว้ชั่วขณะแล้วไปเริ่มทำงานในส่วนของโปรแกรมย่อย ๆ ซึ่งมักเป็นโปรแกรมที่เกี่ยวกับการรับหรือเปลี่ยนแปลงข้อมูลที่อยู่ในการควบคุมของอุปกรณ์ภายนอกนั้น ๆ และเมื่อการทำงานของโปรแกรมย่อยนั้นเสร็จเรียบร้อยแล้ว หน่วยควบคุมจึงจะกลับไปเริ่มทำงานที่ค้างไว้ต่อไป

การยอมให้เกิดอินเทอรัปและการไม่ยอมให้เกิดอินเทอรัปของหน่วยควบคุม

หน่วยควบคุมของ Z-80 นั้นมีลักษณะของการอินเทอรัปอยู่ 2 ลักษณะ คือมาค์สเอเบิลอินเทอรัป (maskable interrupt) และนอนมาค์สเอเบิลอินเทอรัป (non maskable interrupt)

ก. นอนมาค์สเอเบิลอินเทอรัป นั้นผู้เขียนโปรแกรมไม่สามารถควบคุมการเกิดอินเทอรัปหรือไม่ให้เกิดอินเทอรัปได้ นั่นคือถ้าหากมีสัญญาณให้มีการอินเทอรัปเกิดขึ้นโดยส่งจากอุปกรณ์ภายนอกเข้ามายังหน่วยควบคุมแล้วจะต้องเกิดการอินเทอรัปได้เลย ซึ่งการอินเทอรัปกรณีนี้มักเตรียมไว้ใช้เมื่อมีเหตุจำเป็นเกิดขึ้นเช่น ไฟฟ้าดับ

ข. มาค์สเอเบิลอินเทอรัป การอินเทอรัปแบบนี้ผู้เขียนโปรแกรมสามารถควบคุมการเกิดอินเทอรัปของหน่วยควบคุมได้ โดยใช้คำสั่ง DI (disable interrupt) เมื่อไม่ต้องการให้เกิดการอินเทอรัป และใช้คำสั่ง EI (enable interrupt) เมื่อยอมให้สัญญาณอินเทอรัปมีผลต่อหน่วยควบคุม โดยภายในหน่วยควบคุมของ Z-80 นี้มีเอนเอเบิลฟลิบฟลอบ (enable flip-flop) อยู่อันหนึ่งเรียกว่า IFF ซึ่งฟลิบฟลอบนี้จะถูกกำหนดค่าได้โดยผู้เขียนโปรแกรมว่าจะให้มีค่าเป็นศูนย์หรือหนึ่ง ถ้าใช้คำสั่ง DI แล้ว IFF จะมีค่าเป็นศูนย์ แสดงว่าถึงแม้จะมีสัญญาณอินเทอรัปจากภายนอกเข้ามาก็ไม่สามารถทำให้หน่วยควบคุมเกิดการอินเทอรัปได้ แต่ถ้าใช้คำสั่ง EI แล้ว IFF มีค่าเป็นหนึ่ง ดังนั้นจะสามารถเกิดการอินเทอรัปได้ทุกเมื่อถ้ามีสัญญาณอินเทอรัปจากภายนอก

ลักษณะการทำงานของเอนเอเบิลฟลิบฟลอบ

จากการใช้งานจริง ๆ นั้น IFF จะมีอยู่ 2 ตัวคือ IFF_1 และ IFF_2 โดย IFF_1 นี้เป็นฟลิบฟลอบสำหรับควบคุมการอินเทอรัป และ IFF_2 เป็นเพียงที่เก็บชั่วคราวของค่าใน IFF_1 ในกรณีของนอนมาค์สเอเบิลอินเทอรัป ซึ่งการที่ต้องเก็บค่าของ IFF_1 ไว้นั้นจะได้อธิบายต่อไป

โดยปกติแล้ว IFF_1 และ IFF_2 จะมีค่าเป็นศูนย์ นั่นคือไม่อาจเกิดการอินเทอรัปได้ถึงแม้จะมีสัญญานอินเทอรัปจากภายนอกเข้ามาก็ตาม ตราบจนกระทั่งมีคำสั่ง EI เกิดขึ้นสัญญานนั้นจึงจะมีผลต่อหน่วยควบคุม ซึ่งเมื่อมีคำสั่ง EI ถูกเอคเซคคิวท์แล้วนั้น หน่วยควบคุมก็ยังไม่สามารถรับการอินเทอรัปได้จนกว่าคำสั่งที่อยู่ต่อจากคำสั่ง EI หนึ่งคำสั่งนั้นจะถูกเอคเซคคิวท์เรียบร้อยแล้วด้วย เหตุที่ต้องรอให้คำสั่งต่อมาถูกเอคเซคคิวท์ก่อนเนื่องจากถ้ากรณีที่คำสั่งต่อมานั้นเป็นคำสั่ง RETURN แล้ว จะต้องทำตามคำสั่งนี้ให้เรียบร้อยก่อน หน่วยควบคุมจึงจะยอมรับสัญญานอินเทอรัปได้ คำสั่ง EI นี้จะมีผลให้ IFF_1 และ IFF_2 มีค่าเป็นหนึ่ง ต่อเมื่อหน่วยควบคุมรับสัญญานอินเทอรัปเข้ามาเรียบร้อยแล้ว ทั้ง IFF_1 และ IFF_2 จะมีค่าเป็นศูนย์โดยอัตโนมัติเพื่อป้องกันการเกิดอินเทอรัปขึ้นมาอีก จนกว่าจะมีคำสั่ง EI เข้ามาอีกครั้งหนึ่ง IFF_1 และ IFF_2 จึงจะเตรียมพร้อมเพื่อรับการอินเทอรัป ซึ่งในกรณีของมาค์สเอเบิลอินเทอรัปนี้ IFF_1 และ IFF_2 จะมีค่าเท่ากันตลอด

จะกล่าวถึงการเก็บค่าของ IFF_1 ไว้ใน IFF_2 เมื่อเกิดการนอนมาค์สเอเบิลอินเทอรัป กล่าวคือเมื่อมีนอนมาค์สเอเบิลอินเทอรัปเกิดขึ้นนั้น IFF_1 จะต้องมีค่าเป็นศูนย์ในทันที ตราบจนกระทั่งมีคำสั่ง EI เกิดขึ้นเมื่ออยู่ในกรณีของมาค์สเอเบิลอินเทอรัป IFF_1 จึงจะมีค่าเป็นหนึ่งอีกครั้งหนึ่ง ดังนั้นกล่าวได้ว่าหลังจากที่หน่วยควบคุมยอมรับการเกิดนอนมาค์สเอเบิลอินเทอรัปแล้ว มาค์สเอเบิลอินเทอรัปจะอยู่ในสภาพของ IFF_1 มีค่าเป็นศูนย์คือไม่ยอมรับการเกิดอินเทอรัปนั่นเอง แต่ถ้าต้องการจะทราบค่าของ IFF_1 ก่อนที่จะมีการนอนมาค์สเอเบิลอินเทอรัปเกิดขึ้นนั้นก็สามารถทำได้ ด้วยเหตุที่ได้เก็บค่าของ IFF_1 นั้นไว้ใน IFF_2 ก่อนแล้ว ดังนั้นจึงสามารถเรียกค่าเดิมของ IFF_1 มาดูได้ โดยการใช้คำสั่ง LD A, I (นำค่าในรีจิสเตอร์ I มาใส่ในรีจิสเตอร์ A) หรือคำสั่ง LD A, R (นำค่าในรีจิสเตอร์ R มาใส่ในรีจิสเตอร์ A) เมื่อคำสั่งที่กล่าวนี้ถูกเอคเซคคิวท์ ค่าของ IFF_2 ก็จะถูกนำมาใส่ใน

แพริทีแฟลก (parity flag) เพื่อตรวจสอบค่าของ IFF_1 ได้ ซึ่งวิธีการเก็บค่าของ IFF_1 อีกวิธีหนึ่งก็โดยการใช้คำสั่ง RETURN ในโปรแกรมย่อยของการทำนอนมาค์สเอเบิลอินเทอรัป โดยให้คำสั่งนี้เป็นคำสั่งสุดท้ายของโปรแกรมเพื่อแสดงว่าการทำนอนมาค์สเอเบิลอินเทอรัปนี้เสร็จสิ้นลงแล้ว ซึ่งด้วยคำสั่งนี้ค่าใน IFF_2 จะถูกนำกลับมาใส่ลงใน IFF_1 อีกครั้งหนึ่งซึ่งคือค่าของ IFF_1 ก่อนที่จะเกิดการนอนมาค์สเอเบิลอินเทอรัปนั่นเอง ตารางที่ 4.2 จะสรุปค่าของ IFF_1 และ IFF_2 ในขณะที่มีคำสั่งแต่ละคำสั่งแตกต่างกันไป

ลักษณะที่เกิด	IFF_1	IFF_2	
DI	0	0	
EI	1	1	
LD A, I	•	•	เอาค่า IFF_2 ไปไว้ที่แพริทีแฟลก
LD A, R	•	•	เอาค่า IFF_2 ไปไว้ที่แพริทีแฟลก
non maskable interrupt	0	•	
RETURN	IFF_2	•	เอาค่า IFF_2 ไปใส่ใน IFF_1

"•" หมายถึงไม่มีการเปลี่ยนแปลงค่า

ตารางที่ 4.2 ค่าในเอนเอเบิลฟลิบฟลอปในกรณีต่าง ๆ

<u>ลักษณะการยอมรับการเกิดอินเทอรัปของหน่วยควบคุม</u>

ก. <u>นอนมาค์สเอเบิลอินเทอรัป</u>

หน่วยควบคุมยอมรับการเกิดนอนมาค์เอเบิลอินเทอรัปตลอดเวลา โดยที่เมื่อมีอินเทอรัปกรณีนี้เกิดขึ้น หน่วยควบคุมจะเอคเซคคิวท์คำสั่งที่กำลังทำค้างอยู่นั้นจนเสร็จแล้วจะไปเริ่มเอคเซคคิวท์คำสั่งของโปรแกรมย่อยที่เริ่มจากตำแหน่ง 0066 (เลขฐาน 16) ของหน่วยความจำในทันที

ข. มาค์สเอเบิลอินเทอรัป

ผู้เขียนโปรแกรมสามารถควบคุมการยอมรับการเกิดอินเทอรัปของหน่วยควบคุมได้โดยใช้คำสั่ง EI หรือ DI ดังกล่าวแล้ว ซึ่งในกรณีของมาค์สเอเบิลอินเทอรัปนี้สามารถเลือกกระทำได้ 3 โมด (modes) ด้วยกัน คือโมด 0 โมด 1 และโมด 2 ซึ่งในการทำวิทยานิพนธ์นี้ได้เลือกใช้โมด 1 ด้วยเหตุที่มีลักษณะการใช้ได้ง่ายเช่นเดียวกับนอนมาค์สเอเบิลอินเทอรัป นั่นคือเมื่อได้เลือกใช้โมดนี้แล้ว หน่วยควบคุมจะยอมรับการอินเทอรัปโดยการไปเอคเซคคิวท์คำสั่งซึ่งอยู่ที่ตำแหน่ง 0038 (เลขฐาน 16) ของหน่วยความจำในทันทีที่มีสัญญานอินเทอรัปมาจากภายนอก ซึ่งลักษณะที่แตกต่างกับนอนมาค์สเอเบิลอินเทอรัปก็ตรงที่ไปเริ่มเอคเซคคิวท์ที่ตำแหน่ง 0038 แทนที่จะเป็น 0066 ดังที่กล่าวแล้วข้างต้น

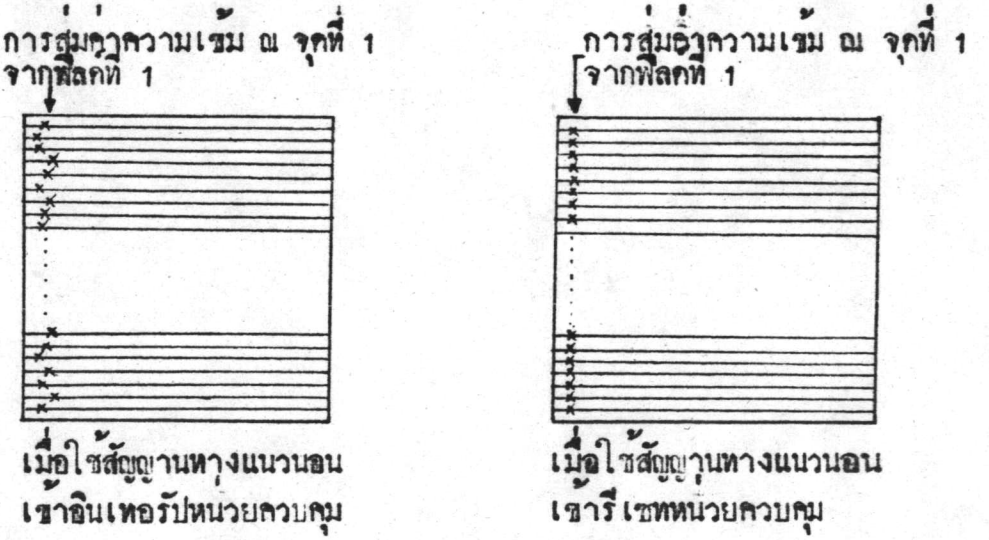
2. คุณสมบัติของการรีเซท

เมื่อมีสัญญานรีเซทจากภายนอกมายังหน่วยควบคุม จะมีผลทำให้โปรแกรมเคาท์เตอร์มีค่าเป็นศูนย์ นั่นคือเมื่อมีสัญญานรีเซทจากภายนอกเข้ามานั้นหน่วยควบคุมจะเลิกทำคำสั่งที่กำลังทำอยู่นั้นแล้วไปเอคเซคคิวท์คำสั่งที่อยู่ที่ตำแหน่ง 0000 ของหน่วยความจำซึ่งโปรแกรมเคาท์เตอร์ชี้อยู่นั้นในทันที

3. การนำคุณสมบัติของการอินเทอรัปและรีเซทมาใช้ในการหาจุดเริ่มต้นของฟิลด์และเส้นสะแกนตามแนวนอน

จากคุณสมบัติของการอินเทอรัปและรีเซทที่กล่าวมานั้นสามารถนำมาพิจารณาได้ว่าสัญญานทั้งสองสัญญานจากกล้องถ่ายโทรทัศน์วงจรปิดคือ สัญญานทางแนวตั้งและสัญญานทางแนวนอนนั้น สัญญานใดที่เหมาะสมกับการอินเทอรัปหน่วยควบคุมและสัญญานใดควรจะเข้าไปรีเซทหน่วยควบคุม ด้วยเหตุที่เมื่อมีสัญญานจากภายนอกเข้ามาอินเทอรัปหน่วยควบคุมนั้น หน่วยควบคุมต้องทำการเอคเซคคิวท์คำสั่งที่ทำค้างอยู่ขณะนั้นให้จบเสียก่อนจึงจะยอมรับการอินเทอรัปนั้นและไปเริ่มเอคเซคคิวท์ยังโปรแกรมย่อยของการอินเทอรัป ส่วนการรีเซทนั้นสัญญานจะมีผลต่อหน่วยควบคุมทันทีไม่ว่าขณะนั้นหน่วยควบคุมจะทำสิ่งใดค้างอยู่ก็ตาม หน่วยควบคุมจะไปเริ่มต้นเอคเซคคิวท์คำสั่งที่ตำแหน่ง 0000 (เลขฐาน 16) ทันที ซึ่งเมื่อพิจารณาดูจะพบว่าการสุ่มค่าความเข้มจากสัญญานภาพทางแนวนอนนั้นจะต้องเริ่มทันทีที่มีสัญญานทางแนวนอนเริ่มผ่านเข้ามาทางซ้ายสุด

ของจอภาพ จะกอยเวลาเนื่องจากให้หน่วยควบคุมเอกเซคคิวท์คำสั่งที่ค้างอยู่นั้นไม่ได้ เพราะมีคำสั่งที่ใช้เวลาน้อยที่สุดในการเอกเซคคิวท์ก็คือ 4 ทีสเตทแล้ว[1] ดังนั้นถ้าจะใช้การอินเทอรัปสำหรับสัญญานทางแนวนอนแล้ว ภาพที่เกิดจากการสุ่มค่าความเข้มเหล่านั้นก็จะโยไปโยมา อันเนื่องมาจากจุดเริ่มต้นของการสุ่มค่าความเข้มของภาพในแต่ละเส้นไม่ได้เริ่มจากจุดเดียวกันนั่นเอง และวิธีการสุ่มค่าความเข้มของภาพนั้นจะทำโดยการสุ่มจากฟิลด์หลาย ๆ ฟิลด์ ซึ่งถ้าใช้การอินเทอรัปสัญญานทางแนวนอนนี้แล้วอาจทำให้ภาพที่ได้ดูแล้วไม่รู้เรื่องก็ได้ ดังนั้นสัญญานทางแนวนอนนี้จึงต้องเข้าไปรีเซทหน่วยควบคุมได้อย่างเดียวเท่านั้นดังรูปที่ 4.2 และสัญญานทางแนวตั้งจะต้องเข้าไปอินเทอรัปแทน เพราะสัญญานทางแนวตั้งนั้นมีผลเพื่อบอกให้ทราบว่าขณะนี้การสะแกนของภาพได้มาถึงจุดเริ่มต้นของฟิลด์ หรืออยู่ที่จุดเริ่มต้นของจอภาพแล้วเท่านั้น ไม่มีผลต่อการที่จะให้เริ่มสุ่มค่าความเข้มจากสัญญานภาพเหมือนกับสัญญานทางแนวนอน และการกอยเวลาในการเอกเซคคิวท์คำสั่งที่ค้างอยู่สำหรับการยอมรับสัญญานที่เข้ามาอินเทอรัปของหน่วยควบคุมนั้นก็แทบจะไม่มีผลต่อการสร้างภาพเลย เพราะดังกล่าวแล้วว่าสัญญานทางแนวตั้งนี้มีเพื่อบอกให้ทราบถึงจุดเริ่มต้นของฟิลด์เท่านั้น ดังนั้นจึงสรุปได้ว่าสัญญานทางแนวนอนนั้นได้เลือกให้เข้าทางรีเซท ส่วนสัญญานทางแนวตั้งจะเข้าอินเทอรัป



รูปที่ 4.2 แสดงให้เห็นว่าสัญญานทางแนวนอนต้องเข้ารีเซทหน่วยควบคุม

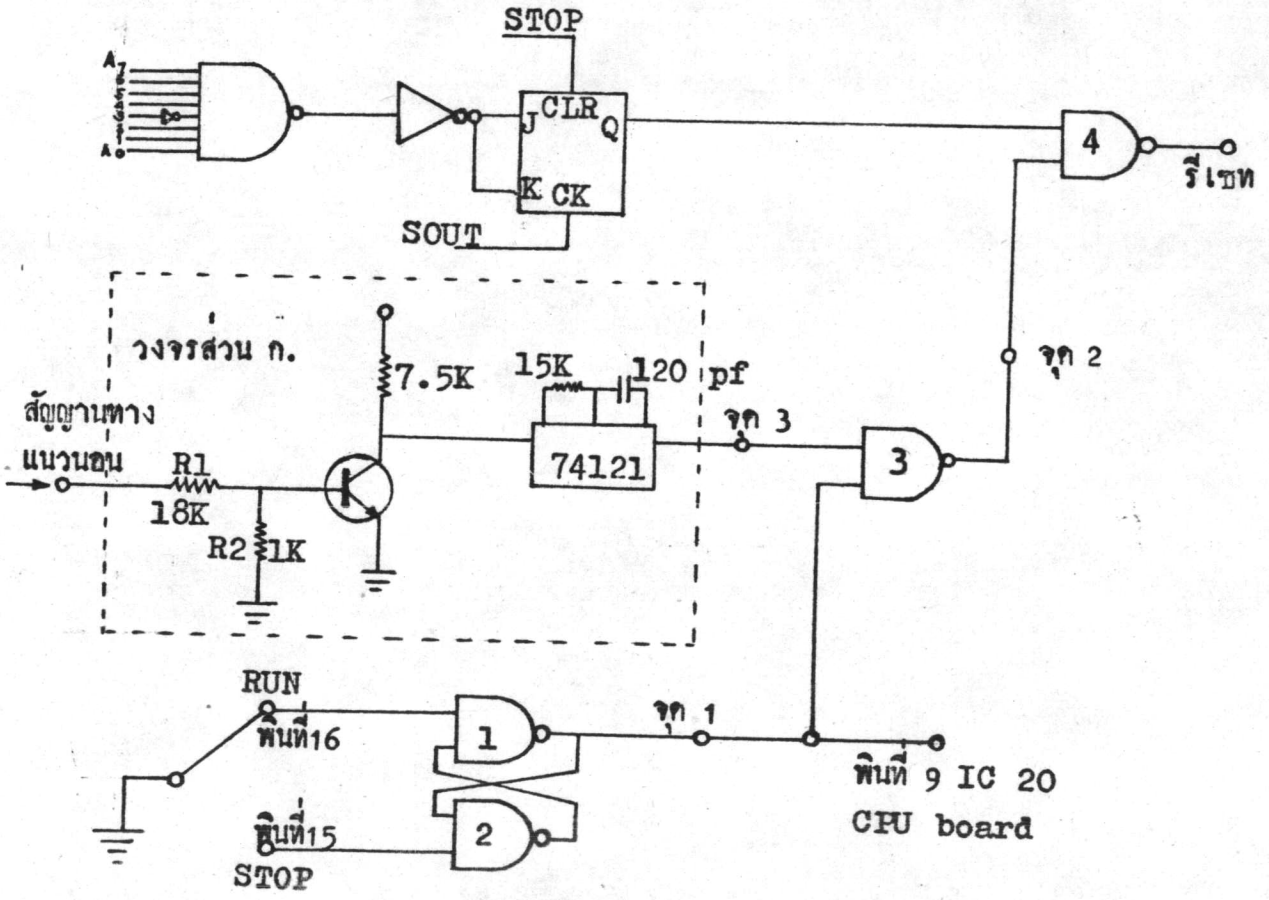
---

[1] พิจารณาเวลาที่ใช้ในการเอกเซคคิวท์คำสั่งต่าง ๆ จากภาคผนวก ก.

4. การจัดให้การรีเซทและการอินเทอรัปกระทำได้อย่างมีประสิทธิภาพสำหรับโปรแกรมการสุ่มค่าความเข้มของภาพ

การต่อสัญญาณทางแนวนอนเข้าไปรีเซทหน่วยควบคุมโดยตรงนั้นจะทำให้มีการรีเซทตลอดเวลาทุกครั้งที่สัญญาณทางแนวนอนได้เริ่มสะแกนเข้ามาทางซ้ายสุดของภาพนั้น ดังนั้นจึงควรทำให้หน่วยควบคุมสามารถยอมรับการรีเซทหรือไม่รับการรีเซทได้เมื่อต้องการ โดยการใช้คำสั่งทางโปรแกรมเช่นเดียวกับลักษณะของการอินเทอรัปนั่นเอง ดังนั้นจึงได้ต่อแผงวงจรส่วนหนึ่งเข้าไปเพื่อควบคุมการรีเซทอันเกิดจากสัญญาณทางแนวนอนนั้น ซึ่งลักษณะและส่วนประกอบของแผงวงจรมีลักษณะดังรูปที่ 4.3 และรูปที่ 4.4 ซึ่งแผงวงจรนี้จะรับทั้งสัญญาณทางแนวนอนและสัญญาณทางแนวตั้งจากกล้องถ่ายโทรทัศน์วงจรปิดมาเพื่อผ่านวงจรดังกล่าวแล้วและส่งสัญญาณไปเข้ายังขาอินเทอรัปและขารีเซทของหน่วยควบคุมต่อไป

ก. การทำงานของวงจรเพื่อรับสัญญาณทางแนวนอนเข้ารีเซท



รูปที่ 4.3 วงจรซึ่งต่อขึ้นเพิ่มเติมในส่วนของการรับสัญญาณทางแนวนอน

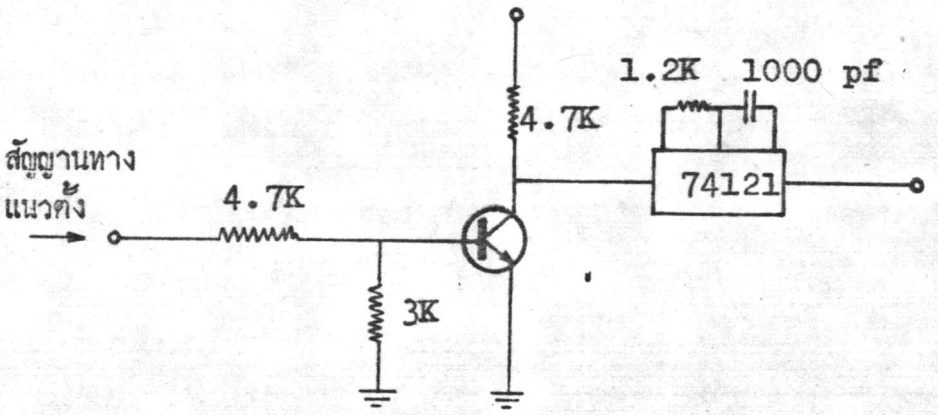
ลักษณะการทำงานของวงจรที่ต่อเพิ่มเติม

ได้ใช้สัญญาน STOP และ RUN จากสวิทช์บนหน้าปัดของเครื่องไมโครคอมพิวเตอร์มาใช้ในการควบคุมสัญญานทางแนวนอนเพื่อที่จะเอาไปรีเซท และได้เลือกใช้พินสองพินที่ว่างอยู่ของซ็อกเก็ตเพื่อต่อเอาสัญญาน STOP และRUN ออกมาใช้ พินที่ใช้คือพินที่ 15 เป็นสัญญาน STOP และพินที่ 16 เป็นสัญญาน RUN สัญญานทั้งสองนี้จะนำมาต่อเข้ากับ RS ฟลิบฟลอบ (RS flip-flop) คือแนนเกท(nand gate) ที่ 1 และ 2 ซึ่งถ้ามีสัญญาน RUN มาจะทำให้จุด 1 มีค่าเป็น 1 และถ้ามีสัญญาน STOP มาจะทำให้จุด 1 มีค่าเป็นศูนย์ สังเกตุดูจะพบว่าสัญญานที่จุด 1 นี้จะถูกต่อเข้ากับแนนเกทที่ 3 ซึ่งแนนเกทที่ 3 นี้จะมีสัญญานเข้ามาอีกสัญญานหนึ่งเป็นสัญญานทางแนวนอนที่ถูกเปลี่ยนให้มีขนาดเหมาะสมแล้วโดยผ่านวงจรส่วน ก. ถ้าสัญญานที่จุด 1 มีค่าเป็นศูนย์แล้วสัญญานที่ผ่านออกจากแนนเกทที่ 3 คือ ณ จุด 2 จะมีค่าเป็น 1 ตลอด นั่นแสดงถึงว่าถ้ามีสัญญาน STOP มาแล้วสัญญานนี้จะไปบังคับไม่ให้สัญญานทางแนวนอนผ่านมาได้ แต่ถ้าสัญญานที่จุด 1 มีค่าเป็น 1 สัญญาน ณ จุด 2 จะมีค่าตรงข้ามกับสัญญานที่มาจากจุด 3 แสดงว่าถ้ามีสัญญาน RUN มาแล้วสัญญานทางแนวนอนนั้นจะสามารถผ่านไปได้ แต่จะไปรีเซทหน่วยควบคุมยังไม่ได้ จะมีสัญญานอีกสัญญานหนึ่งคือสัญญานที่เข้าไปแนนเกทที่ 4 ด้วยกัน ซึ่งสัญญานที่กล่าวนี้สามารถควบคุมได้โดยคำสั่งในโปรแกรมคือ OUT F7 ซึ่งคำสั่งนี้จะใช้ควบคุมการเกิดรีเซทโดยถ้ามีคำสั่ง OUT F7 จะมีสัญญานออกมาถ้าผ่านเข้า JK ฟลิบฟลอบ (JK flip-flop)แล้วได้สัญญานออกจากจุด Q มีค่าเป็น 1 สัญญานที่ผ่านออกจากแนนเกทที่ 4 จะมีค่าตรงกันข้ามกับสัญญานที่มาจากจุด 2 ดังนั้นสัญญานที่ผ่านออกจากแนนเกทที่ 4 นี้จะมีค่าเท่ากับสัญญานทางแนวนอนที่ผ่านเข้ามายังจุด 3 ซึ่งจะทำให้หน่วยควบคุมยอมรับการรีเซทได้ แต่ถ้าไม่ต้องการให้มีสัญญานออกไปรีเซทหน่วยควบคุมก็ทำได้โดยใช้คำสั่ง OUT F7 นี้อีกครั้งหนึ่ง JK ฟลิบฟลอบจะให้สัญญานออกมามีค่าตรงข้ามกับค่าเดิม นั่นคือสัญญานที่มาจากจุด Q มีค่าเป็นศูนย์ ซึ่งจะทำให้สัญญานที่เข้าไปรีเซทมีค่าเป็น 1 เสมอ

การทำให้สัญญานทางแนวนอนมีขนาดของสัญญานที่พอเหมาะทำได้โดยการนำสัญญานทางแนวนอนจากกล้องถ่ายโทรทัศน์วงจรปิดผ่านเข้าความต้านทาน (resistor) ซึ่งมีค่า 18 กิโลโอห์ม และ 1 กิโลโอห์ม ดังวงจรส่วน ก. ของรูปที่ 4.3 เพื่อให้สัญญานลดลงโดยมีค่าสูงสุดไม่ควรเกิน 1 โวลท์ และไม่ต่ำกว่า 0.75 โวลท์ และช่วงต่ำไม่ควรสูงกว่า 0.65

โวลท์ ทั้งนี้เพราะจะนำสัญญาณนี้ไปเข้าขาเบส (base) ของทรานซิสเตอร์(transistor) ซึ่งทรานซิสเตอร์นี้จะตัดค่าต่ำสุดของความต่างศักย์ เมื่อความต่างศักย์มีค่าน้อยกว่า 0.65 โวลท์ และค่าสูงสุดคือมีค่ามากกว่า 0.75 โวลท์ ทรานซิสเตอร์ตัวนี้จะทำหน้าที่เป็นอินเวอร์เตอร์ (Invertor) และจะให้ค่าความต่างศักย์ที่ขาคอลเลคเตอร์(Collector) มีค่าระหว่าง 0 โวลท์ และ 5 โวลท์เท่านั้น ซึ่งสัญญาณนี้จะมีคุณสมบัติให้ใช้ได้กับ TTL ได้เลย ต่อจากนั้นจึงนำสัญญาณนี้ไปผ่าน IC 74121 ซึ่งจะให้สัญญาณออกมามีช่วงกว้างของเวลาที่พอเหมาะซึ่งสามารถควบคุมโดยต่อตัวความต้านทานและตัวจุไฟฟ้า (Capacitor) ให้กับ IC 74121 นี้

ข. การทำงานของวงจรเพื่อรับสัญญาณทางแนวตั้ง



รูปที่ 4.4 วงจรส่วนที่รับสัญญาณทางแนวตั้งเพื่อการอินเทอรัป

การอินเทอรัปเนื่องจากสัญญาณทางแนวตั้งนั้นมีการต่อวงจรเล็กน้อย เพื่อเปลี่ยนขนาดของสัญญาณทางแนวตั้งที่มาจากกล้องถ่ายโทรทัศน์วงจรปิดให้มีขนาดและช่วงกว้างของเวลาที่เหมาะสมก่อน ลักษณะของวงจรเหมือนกับวงจรส่วน ก. ของรูปที่ 4.3 ซึ่งเป็นวงจรการเปลี่ยนสัญญาณทางแนวนอนนั้น แต่ช่วงกว้างของเวลามีความสำคัญมากคือจะต้องทราบว่าจะทำการอินเทอรัปเมื่อไร คือในโปรแกรมจะใช้คำสั่งอะไรอยู่ ซึ่งเวลานี้จะต้องมากกว่าเวลาที่ใช้ในการเอคเซคคิวท์คำสั่งนั้นและต้องไม่มากเกินไปด้วย ถ้าเวลานี้มากเกินไปแล้วเมื่อเกิดการอินเทอรัปครั้งหนึ่งแล้วทำโปรแกรมในช่วงของการอินเทอรัปเสร็จแล้ว และมีคำสั่งให้ทำการอินเทอรัปอีกก็จะเกิดขึ้นในขณะที่ยังมีสัญญาณอินเทอรัปเดิมอยู่ ซึ่งที่จริงแล้วต้องการให้สัญญาณที่จะมาอินเทอรัปนั้นเป็นอีกช่วงหนึ่งต่างหาก ดังนั้นจึงต้องดูลักษณะของโปรแกรมด้วย คือการอินเทอรัปแต่ละครั้งนั้น

จะให้โปรแกรมหยุดไว้ก่อนโดยให้เอคเซคคิวท์คำสั่ง NOP ซึ่งคำสั่งนี้ใช้เวลาในการเอคเซคคิวท์ 4 ทีสเตท หรือ 2 ไมโครวินาที ดังนั้นจึงจัดให้ช่วงกว้างของเวลาเกินกว่า 2 ไมโครวินาทีเล็กน้อย และสัญญานนี้จะเข้าไปอินเทอรัปยังหน่วยควบคุมได้โดยตรงเลยโดยไม่ต้องมีการต่อวงจรเพิ่มเติมเหมือนกับสัญญานทางแนวนอน เพราะมีคำสั่งที่สามารถให้หน่วยควบคุมยอมรับสัญญานอินเทอรัปหรือไม่ยอมรับได้ คือคำสั่ง EI และ DI และมีฟลิบฟลอบอยู่ในหน่วยควบคุมแล้ว

ระบบโปรแกรมการสุ่มค่าความเข้มของภาพตามสมมติฐานที่ตั้งไว้

ก. ทำการสุ่มค่าความเข้มของภาพจากฟิลด์หลาย ๆ ฟิลด์ และจากการวิเคราะห์แล้วปรากฏว่าจำนวนฟิลด์ที่สุ่มทั้งหมดมีได้มากที่สุดคือ 105 ฟิลด์ ซึ่งก็คือจำนวนค่าความเข้มของภาพที่สามารถสุ่มได้มากที่สุดในการสะแกนของสัญญานทางแนวนอน 1 เส้นนั่นเอง

ข. ใช้คุณสมบัติของมาร์สเอเบิลอินเทอรัปในการจับสัญญานภาพที่จุดเริ่มต้นของฟิลด์ เนื่องจากมาร์สเอเบิลอินเทอรัปนี้โปรแกรมสามารถควบคุมการเกิดอินเทอรัปเมื่อมีสัญญานทางแนวตั้งเข้ามายังหน่วยควบคุม โดยใช้คำสั่ง EI เมื่อต้องการการอินเทอรัป และใช้คำสั่ง DI เมื่อไม่ต้องการ โปรแกรมย่อยของการอินเทอรัปนี้ใช้ในการควบคุมจำนวนฟิลด์ที่ใช้ในการสุ่มค่าความเข้มของภาพ

ค. ใช้คุณสมบัติของการรีเซทในการควบคุมการสุ่มค่าความเข้มของภาพของเส้นสะแกนตามแนวนอน โดยโปรแกรมใช้คำสั่ง OUT F7 ในการควบคุมการยอมรับและไม่ยอมรับการรีเซทของหน่วยควบคุม และโปรแกรมย่อยของการรีเซทนี้ใช้สุ่มค่าความเข้มของภาพ ณ จุดใดจุดหนึ่งที่กำหนดไว้ในเส้นสะแกนตามแนวนอนแต่ละเส้นให้ครบตามจำนวนเส้นที่ต้องการในหนึ่งภาพ ซึ่งจะได้กล่าวถึงในรายละเอียดของโปรแกรมต่อไป

ง. ข้อสำคัญที่สุดของระบบโปรแกรมนี้ก็คือต้องคำนึงถึงเวลาที่ใช้ในการเอคเซคคิวท์คำสั่งทุก ๆ คำสั่งที่ใช้ในโปรแกรม นั่นคือการจะเลือกใช้คำสั่งใดในโปรแกรมนี้จะต้องพิจารณาเวลาที่ใช้ในการเอคเซคคิวท์คำสั่งเป็นกรณีแรก และต้องคำนวณระยะเวลาในการเอคเซคคิวท์โปรแกรมนี้ตั้งแต่เริ่มทำการอินเทอรัปเป็นต้นมาจนกระทั่งสุ่มค่าความเข้มของภาพจากสัญญานภาพได้ครบหนึ่งภาพตามต้องการ ซึ่งจะเห็นว่าในโปรแกรมนี้คำสั่งบางคำสั่งที่อาจนำมาใช้ได้เนื่องจากลักษณะของการทำงานของคำสั่งเป็นไปตามต้องการ แต่คำสั่งเหล่านั้นได้ถูกละเลยไปก็เนื่อง

มาจากเวลาที่ใช้ในการเอคเซคคิวท์คำสั่งนั้นไม่เหมาะสมกับเวลาที่ถูกคำนวณไว้นั้นเอง

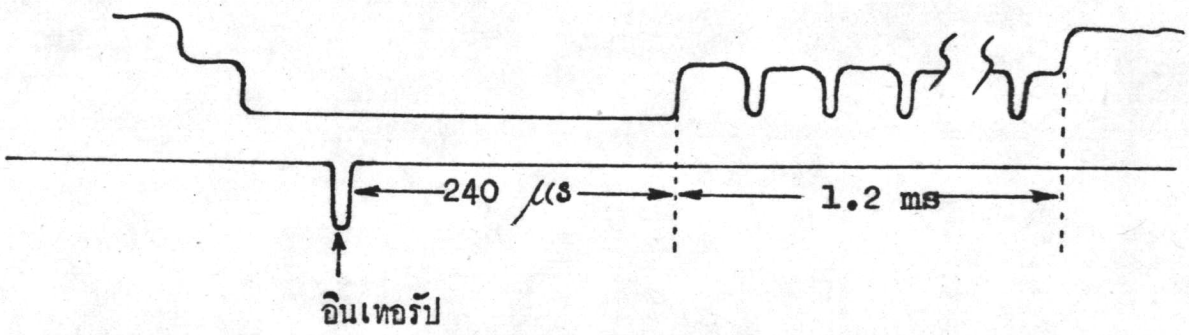
ดังนั้นจะเห็นได้ว่าระบบโปรแกรมการสุ่มค่าความเข้มของภาพนี้เป็นระบบโปรแกรมที่ต้องสัมพันธ์กับเวลา เนื่องจากการสุ่มค่าความเข้มของภาพนั้นต้องสุ่มมาจากสัญญาณภาพที่มีการสะแกนในเวลาที่แน่นอนและรวดเร็วมากนั่นเอง

โปรแกรมการสุ่มค่าความเข้มของภาพ

ก. สัญญาณภาพรวมจากกล้องถ่ายโทรทัศน์วงจรปิด

ก่อนที่จะเขียนโปรแกรมการสุ่มค่าความเข้มของภาพที่สมบูรณ์ได้นั้น จำเป็นต้องทราบเวลาในการสะแกนของสัญญาณภาพรวมในช่วงที่ต้องใช้ในการสร้างโปรแกรมการสุ่มค่าความเข้มของภาพ ซึ่งสามารถวัดค่าได้โดยใช้เครื่องออสซิลโลสโคปปรากฏผลดังนี้

1. สัญญาณแบลงค์กิงและสัญญาณซิงค์ตามแนวตั้ง ซึ่งเป็นผลต่อการเกิดอินเทอรัป ดังรูปที่ 4.5



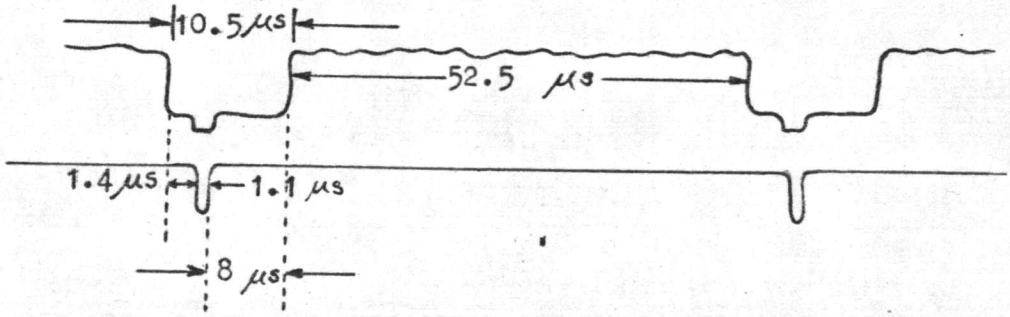
รูปที่ 4.5 ขนาดของสัญญาณแบลงค์กิงและสัญญาณซิงค์ตามแนวตั้ง

สัญญาณแบลงค์กิงและสัญญาณซิงค์ตามแนวตั้งทั้งหมดนับจากหน่วยควบคุมเริ่มยอมรับการอินเทอรัปเป็นต้นไปใช้เวลา 1440 ไมโครวินาที ซึ่งในช่วง 240 ไมโครวินาทีแรกนั้นยังไม่ปรากฏสัญญาณภาพปนอยู่ แต่จะปรากฏขึ้นในช่วง 1200 ไมโครวินาทีต่อไป ซึ่งสัญญาณภาพนี้จะเกิดขึ้นพร้อม ๆ กับสัญญาณแบลงค์กิงและสัญญาณซิงค์ตามแนวตั้ง คิดเป็นจำนวนเส้นสะแกน

ตามแนวนอนประมาณ 20 เส้น เมื่อหมดสัญญานแบลงค์กิงและสัญญานซิงค์ตามแนวตั้งนี้แล้ว จึงเริ่มรีเซทเพื่อทำการสุ่มค่าความเข้มของภาพต่อไป

2. สัญญานภาพกับสัญญานแบลงค์กิงและสัญญานซิงค์ตามแนวนอน

การสะแกนของสัญญานภาพทางแนวนอน 1 เส้นนั้นจะประกอบไปด้วยสัญญานแบลงค์กิงและสัญญานซิงค์ตามแนวนอนเป็นเวลา 10.5 ไมโครวินาที และสัญญานภาพที่แท้จริง 52.5 ไมโครวินาที ดังรูปที่ 4.6



รูปที่ 4.6 ขนาดของสัญญานภาพกับสัญญานแบลงค์กิงและสัญญานซิงค์ตามแนวนอน

จะเห็นได้ว่าเมื่อหน่วยควบคุมยอมรับการรีเซทแล้วจะยังคงมีสัญญานแบลงค์กิงตามแนวนอนอีก 8 ไมโครวินาที จึงจะเริ่มเกิดสัญญานภาพที่แท้จริงซึ่งจะได้เริ่มสุ่มค่าความเข้มของภาพตั้งแต่จุดนี้เป็นต้นไป

ข. การกำหนดค่าต่าง ๆ ในโปรแกรมการสุ่มค่าความเข้มของภาพ

1. จำนวนพิลล์ที่ทำการสุ่ม

เนื่องจากจำนวนค่าความเข้มของภาพที่สามารถสุ่มมาได้ในการสะแกนของสัญญานภาพตามแนวนอน 1 เส้นนั้นมีได้มากที่สุดเป็นจำนวน 105 จุด เพื่อความสะดวกในการสร้างโปรแกรมการสุ่มค่าความเข้มของภาพจึงกำหนดให้ทำการสุ่มค่าความเข้มของภาพในเส้นสะแกนตามแนวนอนแต่ละเส้นเป็นจำนวน 100 จุด ซึ่งเมื่อใช้เครื่องพิมพ์พิมพ์ตัวอักษร 100 ตัว

นี้ต่อกันไปจะได้เป็นตัวอักษรต่อ ๆ กันยาว 10 นิ้ว ซึ่งเป็นความกว้างของภาพ จากที่กล่าวแล้วว่าสายตาของคนเราส่วนมากนิยมมองดูภาพที่มีขนาดอัตราส่วนความกว้างต่อความสูงของภาพเป็นสี่ต่อสามนั้น พบว่าถ้าภาพที่จะสร้างขึ้นมานี้มีความกว้างของภาพเป็น 10 นิ้วแล้ว ความสูงของภาพควรเป็น 7.5 นิ้ว จึงจะมีอัตราส่วนเป็นสี่ต่อสามพอดี และจากมาตรฐานของเครื่องพิมพ์ซึ่งพิมพ์ได้ 6 บรรทัดต่อ 1 นิ้วนั้น ถ้าต้องการให้ความสูงของภาพเป็น 7.5 นิ้วแล้ว จำนวนบรรทัดของตัวอักษรที่พิมพ์ควรเป็น 45 บรรทัด

จากการที่จะทำการสุ่มข้อมูลของภาพเป็นฟิลด์ ๆ นั้น ซึ่งหนึ่งฟิลด์จะมีจำนวนเส้นสะแกนตามแนวนอนประมาณ 312 เส้น และในจำนวนนี้ 20 เส้นได้ปะปนอยู่ในส่วนของสัญญาณแบลงค์กิงและซิงค์ตามแนวตั้งไปแล้ว ดังนั้นจึงเหลือเส้นสะแกนตามแนวนอนที่จะสามารถใช้สุ่มข้อมูลอีกเพียง 292 เส้น ซึ่งจะต้องทำการคำนวณดูว่าในจำนวนเส้นสะแกนตามแนวนอน 292 เส้นนี้ควรทำการสุ่มในอัตราเท่าใดจึงจะเหมาะสมให้ได้ภาพที่พิมพ์ออกมามีความสูง 7.5 นิ้ว หรือคิดเป็น 45 บรรทัดพอดี

" กำหนดให้อัตราการสุ่มเส้นสะแกนตามแนวนอนหมายถึง จำนวนเส้นสะแกนตามแนวนอนที่ทำการสุ่มค่าความเข้มของภาพต่อจำนวนเส้นสะแกนตามแนวนอนที่เกิดขึ้นจริง "

ซึ่งคำนวณได้ว่าถ้าต้องการจำนวนบรรทัดของภาพที่พิมพ์ได้ 45 บรรทัดแล้ว จะต้องสุ่มเส้นสะแกนตามแนวนอน 1 เส้นในจำนวนเส้นสะแกนจริง ๆ $\frac{292}{45}$ ซึ่งได้ค่าประมาณ 6 – 7 เส้น ซึ่งถ้าหากสุ่มในอัตรา 1 ใน 6 เส้นแล้ว จะได้ภาพที่มีจำนวนบรรทัดที่พิมพ์ตัวอักษรเป็น 48 บรรทัด หรือคิดเป็นความสูงของภาพ 8 นิ้วพอดี ดังนั้นจึงกำหนดได้ว่าจะสร้างภาพโดยให้มีจำนวนข้อมูลใน 1 เส้นเป็น 100 จุด และมีจำนวนเส้นเป็น 48 เส้น หรือคิดเป็นความกว้างของภาพ 10 นิ้ว และความสูงของภาพ 8 นิ้วนั่นเอง

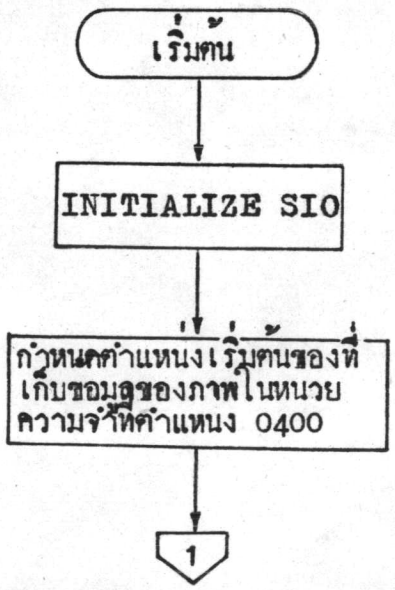
จากนั้นจึงนำเอาค่าทั้งสองที่หาได้นี้มาใช้สร้างโปรแกรมการสุ่มค่าความเข้มของภาพต่อไป โดยจะทำการสุ่มค่าความเข้มของภาพจากฟิลด์ทั้งหมด 100 ฟิลด์ ตามจำนวนค่าความเข้มของภาพที่จะสุ่มทั้งหมดของเส้นสะแกนตามแนวนอนแต่ละเส้น และฟิลด์แต่ละฟิลด์ที่สุ่มมาก็จะทำการสุ่มเส้นสะแกนตามแนวนอนมาทั้งหมด 48 เส้น โดยสุ่มข้อมูลหรือค่าความเข้มของภาพเส้นละ 1 จุด ซึ่งค่าความเข้มของภาพ ณ จุดที่สุ่มค่าความเข้มของทุก ๆ เส้นในฟิลด์หนึ่งนั้นจะมี

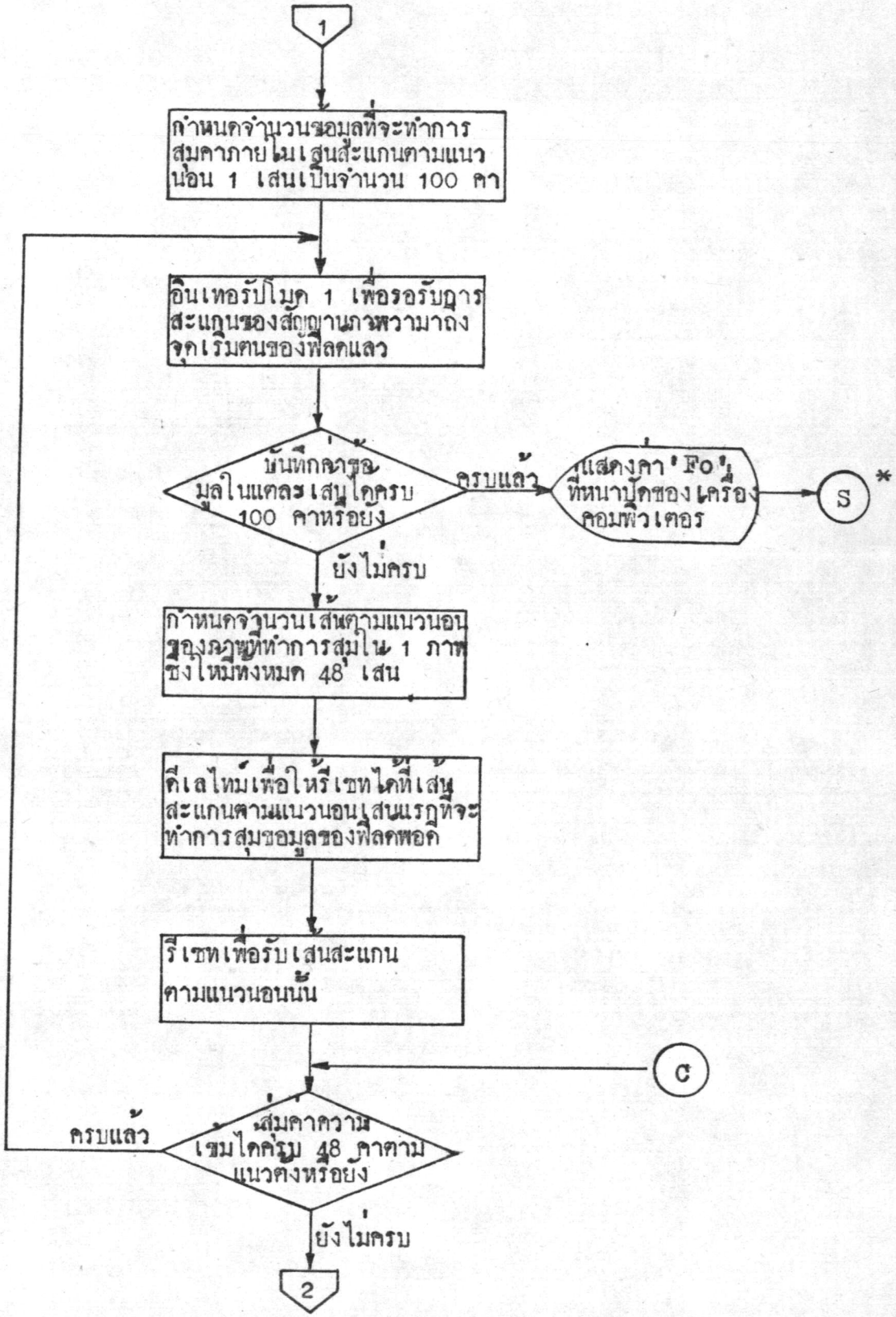
ตำแหน่งเป็นแถวเดียวกันตลอด นั่นคือเมื่อสุ่มค่าความเข้มได้ครบ 100 ฟิลด์ ก็จะได้ข้อมูลซึ่งแทนความเข้มของภาพทั้งหมด 4800 ค่า ซึ่งนับเป็นข้อมูลครบหนึ่งภาพพอดี

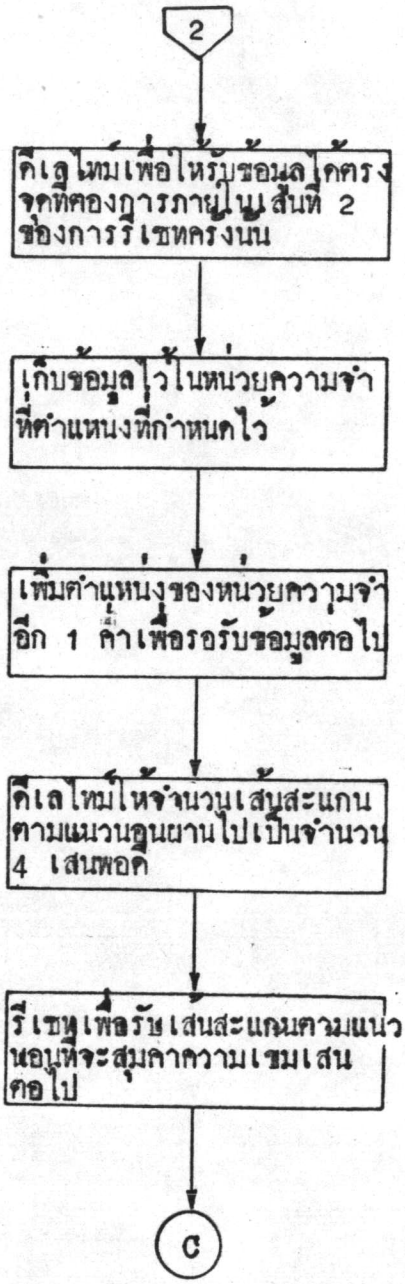
2. การใช้รีจิสเตอร์ในโปรแกรม

รีจิสเตอร์	ลักษณะการใช้
D	ควบคุมข้อมูลที่ทำการสุ่มในเส้นสะแกนตามแนวนอน 1 เส้น
E	ควบคุมจำนวนเส้นสะแกนตามแนวนอนที่ทำการสุ่มใน 1 ฟิลด์
HL	แทนตำแหน่งในหน่วยความจำซึ่งเป็นที่เก็บค่าความเข้มของภาพที่สุ่มมาได้
IX	แทนตำแหน่งในหน่วยความจำซึ่งเป็นจุดเริ่มต้นของการใช้โปรแกรมดีเลไทม์

ก. แผนผังของโปรแกรม (Flow chart) และตัวโปรแกรม




1
กำหนดจำนวนข้อมูลที่จะทำการ
สุ่มค่าภายในเส้นสะแกนตามแนว
นอน 1 เส้นเป็นจำนวน 100 ค่า
อินเทอรัปโมด 1 เพื่อรอรับการ
สะแกนของสัญญาณภาพว่ามาถึง
จุดเริ่มต้นของฟิลด์แล้ว
ขั้นตอนการสุ่ม
ข้อมูลในแต่ละเส้นได้ครบ
100 ค่าหรือยัง
ครบแล้ว
แสดงค่า 'Fo'
ที่หน้าปัดของเครื่อง
คอมพิวเตอร์
S
*
ยังไม่ครบ
กำหนดจำนวนเส้นตามแนวนอน
ของภาพที่ทำการสุ่มใน 1 ภาพ
ซึ่งในที่นี้มีทั้งหมด 48 เส้น
ดีเลไทม์เพื่อให้รีเซทได้ที่เส้น
สะแกนตามแนวนอนเส้นแรกที่จะ
ทำการสุ่มข้อมูลของฟิลด์พอดี
รีเซทเพื่อรับเส้นสะแกน
ตามแนวนอนนั้น
C
สุ่มค่าความ
เข้มได้ครบ 48 ค่าตาม
แนวตั้งหรือยัง
ครบแล้ว
ยังไม่ครบ
2



* S คือโปรแกรมการจัดตัวอักษรและการพิมพ์ภาพ

STARTING PROGRAM

ตำแหน่งของคำสั่งในหน่วยความจำ	รหัส		คำสั่ง
01 00	3E CA	INITIAL:	LD A, CA
01 02	D3 03		OUT 03
01 04	3E 27		LD A, 27
01 06	D3 03		OUT 03
01 08	21 00 04	SETVAL:	LD HL, (0400)
01 0B	16 65		LD D, 65
01 0D	DD 21 F9 00		LD IX, (00F9)
01 11	ED 56		IM 1
01 13	FB		EI
01 14	76		HALT

หมายเหตุ

HL = ตำแหน่งเริ่มต้นในหน่วยความจำซึ่งใช้เก็บข้อมูลอยู่ที่ 0400 ถึง 16C0

D = จำนวนข้อมูลที่บันทึกใน 1 เส้นสะแกนตามแนวนอนเท่ากับ $(65-1)_{16}$

Status : /* Enable interrupt */

/* Disable reset */

INTERRUPT SERVICE ROUTINE

ตำแหน่งของคำสั่งในหน่วยความจำ	รหัส		คำสั่ง	จำนวนของทีสเตท
00 38	31 FF 1F		LD SP, (1FFF)	10
00 3B	F3		DI	4
00 3C	15		DEC D	4
00 3D	C2 47 00		JNZ NEND	10
00 40	3E F0	END:	LD A, F0	
00 42	D3 FF		OUT FF	
00 44	C3 50 01		JMP SETDT	
00 47	1E 31	NEND:	LD E, 31	7
00 49	DD 22 9E 00		LD (009E), IX	20
00 4D	3A 9E 00		LD A, (009E)	13
00 50	C6 20		ADD 20	7
00 52	F6 80		OR 80	7
00 54	32 9E 00		LD (009E), A	13
00 57	DD 2A 9E 00		LD IX, (009E)	20
00 5B	3E C1		LD A, C1	7
00 5D	3D	TIM1:	DEC A	4
00 5E	C2 5D 00		JNZ TIM1	10
00 61	D3 F7		OUT F7	11
00 63	76		HALT	

E = จำนวนเส้นสะแกนตามแนวนอนใน 1 ฟิลด์เท่ากับ $(31-1)_{16}$

Status : /* Disable interrupt */

/* Enable reset */

RESET ROUTINE

ตำแหน่งของคำสั่งในหน่วยความจำ	รหัส		คำสั่ง	จำนวนของทีสเตท
00 00	D3 F7		OUT F7	11
00 02	1D		DEC E	4
00 03	C2 08 00		JNZ NINTR	10
00 06	FB	INTR:	EI	
00 07	76		HALT	
00 08	DD E9	NINTR:	JMP IX	8
00 0A	DB 1F	KPDT:	IN 1F	10
00 0C	77		LD (HL), A	4
00 0D	23		INC HL	6
00 0E	3E 21		LD A, 21	7
00 10	3D	TM1:	DEC A	4
00 11	C2 10 00		JNZ TM1	10
00 14	D3 F7		OUT F7	11
00 16	76		HALT	

DELAY TIME ROUTINE

ตำแหน่งของคำสั่งในหน่วยความจำ	รหัส	คำสั่ง	จำนวนของที่สเตท
00 80	00	NOP	4
•	•	•	•
•	•	•	•
•	•	•	•
•	•	•	•
•	25 ครั้ง	25 ครั้ง	•
•	•	•	•
•	•	•	•
•	•	•	•
•	•	•	•
00 98	00	NOP	4
00 99	3E 00	LD A, 00	7
00 9B	C3 20 00	JMP DELAY1	10
00 A0	00	NOP	4
•	•	•	•
•	•	•	•
•	•	•	•
•	•	•	•
•	25 ครั้ง	25 ครั้ง	•
•	•	•	•
•	•	•	•
•	•	•	•
•	•	•	•
00 B8	00	NOP	4
00 B9	CB 7F	BIT 7, A	8
00 BB	C3 20 00	JMP DELAY1	10
00 C0	00	NOP	4
•	•	•	•
•	•	•	•
•	•	•	•
•	•	•	•
•	25 ครั้ง	25 ครั้ง	•
•	•	•	•
•	•	•	•
•	•	•	•
•	•	•	•

ตำแหน่งของคำสั่งในหน่วยความจำ	รหัส	คำสั่ง	จำนวนของทีสเตท
00 D8	00	NOP	4
00 D9	ED 47	LD I, A	9
00 DB	C3 20 00	JMP DELAY1	10
00 E0	00	NOP	4
⋮	⋮	⋮	⋮
⋮	25 ครั้ง	25 ครั้ง	⋮
⋮	⋮	⋮	⋮
00 F8	00	NOP	4
00 F9	C3 20 01	JMP DELAY2	10

DELAY TIME SUBROUTINE

ตำแหน่งของคำสั่งในหน่วยความจำ	รหัส		คำสั่ง	จำนวนของทีสเตท
00 20	3E 04	DELAY1:	LD A, 04	7
00 22	3D	TM2:	DEC A	4
00 23	C2 22 00		JNZ TM2	10
00 26	00		NOP	4
00 27	00		NOP	4
00 28	00		NOP	4
00 29	C3 0A 00		JMP KPDT	10

ตำแหน่งของคำสั่งในหน่วยความจำ	รหัส		คำสั่ง	จำนวนของทีสเตท
01 20	7B	DELAY2:	LD A, E	4
01 21	FE 01		CP 01	7
01 23	C2 2B 01		JNZ NOTE	10
01 26	DD 2B		DEC IX	10
01 28	C3 30 01		JMP TIME	10
01 2B	00	NOTE:	NOP	4
01 2C	00		NOP	4
01 2D	00		NOP	4
01 2E	00		NOP	4
01 2F	00		NOP	4
01 30	3E 02	TIME:	LD A, 02	7
01 32	3D	TM3:	DEC A	4
01 33	C2 32 01		JNZ TM3	10
01 36	ED 47		LD I, A	9
01 38	C3 0A 00		JMP KPDT	10

ง. อธิบายการทำงานของโปรแกรมโดยสังเขป

ลักษณะคำสั่งของโปรแกรมเริ่มด้วยกำหนดค่าของรีจิสเตอร์ต่าง ๆ ดังกล่าวแล้ว จึงใช้คำสั่งอินเทอรัปโมด 1 จากนั้นเอนเอเบิลอินเทอรัปเพื่อให้หน่วยควบคุมยอมรับการอินเทอรัปนั้น เมื่อมีสัญญานแบลงค์กิงและสัญญานซิงค์ทางแนวตั้งผ่านเข้ามาแสดงว่าถึงจุดเริ่มต้นของฟิลด์แล้ว หน่วยควบคุมจะไปเริ่มเอกเซคคิวท์คำสั่งที่ตำแหน่ง 0038 ซึ่งเป็นคำสั่งแรกของโปรแกรมอินเทอรัป

เมื่อเริ่มเอกเซคคิวท์โปรแกรมอินเทอรัปก็ต้องเริ่มคำนวณเวลาในการเอกเซคคิวท์แต่ละคำสั่งทันที เพราะจะทำการสุ่มค่าความเข้มของภาพแล้ว เวลาในการเอกเซคคิวท์ตั้งแต่คำสั่งแรกของโปรแกรมจนถึงคำสั่งที่กำหนดจุดเริ่มต้นของการเอกเซคคิวท์โปรแกรมดีเลไทม์ (Delay time routine) ใช้เวลาไป 57.5 ไมโครวินาที หรือ 115 ทีสเตท (ทั้งนี้จะไม่คำนึงถึงคำสั่งที่จะถูกเอกเซคคิวท์ต่อเมื่อสุ่มค่าความเข้มของภาพได้ครบแล้ว) จากที่กล่าวแล้วว่าช่วงเวลานับจากเริ่มอินเทอรัปจนกระทั่งหมดสัญญานแบลงค์กิงและสัญญานซิงค์ทางแนวตั้งนั้นเป็นเวลาถึง 1440 ไมโครวินาที หรือ 2880 ทีสเตท ดังนั้นการที่จะเริ่มสุ่มข้อมูลคือค่าความเข้มของภาพของแต่ละฟิลด์ได้นั้น ต้องทำการรีเซทภายหลังจากที่หน่วยควบคุมยอมรับการอินเทอรัปแล้วเป็นเวลาประมาณ 1400 ถึง 1430 ไมโครวินาที หรือ 2800 ถึง 2860 ทีสเตท จะเห็นได้ว่าจะต้องมีคำสั่งส่วนหนึ่งภายในโปรแกรมอินเทอรัปนี้ซึ่งคอยควบคุมให้เวลาผ่านไปถึงจุดที่จะเริ่มทำการรีเซท ส่วนของโปรแกรมที่ใช้ควบคุมเวลานี้จะประกอบด้วยคำสั่ง 3 คำสั่งดังนี้

```
        LD A, n          ใช้เวลาในการเอกเซคคิวท์ 7  ทีสเตท
TIM1:   DEC A            ใช้เวลาในการเอกเซคคิวท์ 4  ทีสเตท
        JNZ TIM1         ใช้เวลาในการเอกเซคคิวท์ 10 ทีสเตท
```

ค่าของรีจิสเตอร์ A จะได้จากการคำนวณว่าเวลาที่ต้องการให้ผ่านไปนั้นเป็นเวลาเท่าใด

เนื่องจากได้ใช้เวลาในการเอกเซคคิวท์คำสั่งของโปรแกรมอินเทอรัปนี้ไป 115 ทีสเตท และรวมกับเวลาในการเอกเซคคิวท์คำสั่งที่กำหนดค่าของรีจิสเตอร์ A กับคำสั่งรับการรีเซทจะเป็นเวลาทั้งหมด 133 ทีสเตท ดังนั้นจึงคำนวณหาค่าของรีจิสเตอร์ A โดยตั้งเป็นสมการได้ว่า

$$133 + 14n = 2800 \longrightarrow 2860$$

เมื่อ n คือค่าที่เป็นไปได้ของรีจิสเตอร์ A ซึ่งจากสมการ n ควรมีค่าประมาณ 193 หรือ C1 (เลขฐาน 16)

เมื่อหน่วยควบคุมยอมรับการรีเซทแล้วจะไปเริ่มเอคเซคคิวท์ยังตำแหน่งที่ 0000 ซึ่งเป็นจุดเริ่มต้นของโปรแกรมรีเซท(Reset routine) การทำงานของโปรแกรมนี้เริ่มด้วยการดิสเอเบิลรีเซท (disable reset) เพื่อมิให้เกิดการรีเซทขึ้นอีกในขณะที่ทำการสุ่มข้อมูล จากนั้นทำการตรวจสอบจำนวนเส้นสะแกนตามแนวนอนทั้งหมดที่สุ่มมาของฟิลด์นี้ว่าได้ครบ 48 เส้น หรือได้ข้อมูลครบ 48 ค่าหรือยัง ถ้าครบแล้วจึงใช้คำสั่งเอนเอเบิลอินเทอรัปเพื่อรับฟิลด์ใหม่ต่อไป ถ้ายังได้ข้อมูลไม่ครบต้องทำการสุ่มเส้นต่อไปโดยไปเริ่มเอคเซคคิวท์ที่โปรแกรมคีเลไทม์ ณ ตำแหน่งซึ่งกำหนดไว้ในโปรแกรมอินเทอรัปเพื่อให้ได้ข้อมูลตามจุดที่ต้องการ

ยกตัวอย่างการสุ่มข้อมูลจุดแรกของฟิลด์ที่ 1 เมื่อเริ่มเอคเซคคิวท์คำสั่งแรกของโปรแกรมรีเซทจนถึงคำสั่งที่สั่งให้ไปยังโปรแกรมคีเลไทม์รวมเป็นเวลาในการเอคเซคคิวท์ 33 ทีสเตท และเนื่องจากอัตราการสุ่มเส้นสะแกนตามแนวนอนเป็น 1 ใน 6 เส้น และเริ่มรับข้อมูลจริง ๆ ที่เส้นที่ 2 ของแต่ละกลุ่ม ดังนั้นจุดแรกของแต่ละเส้นที่จะรับข้อมูลเข้ามาเก็บไว้นั้นต้องรับหลังจากที่เริ่มรีเซทแล้วเป็นเวลา 71 ไมโครวินาที หรือ 142 ทีสเตท แต่เนื่องจากในเส้นสะแกนตามแนวนอน 1 เส้นนั้นจะรับข้อมูล 100 ค่า จากที่รับได้มากที่สุด 105 ค่า ดังนั้นจึงสุ่มรับข้อมูลค่าแรกของเส้นสะแกนแต่ละเส้นหลังจากที่เริ่มรีเซทแล้วเป็นเวลา 72.5 ไมโครวินาที หรือ 145 ทีสเตท เพื่อให้ข้อมูลที่สุ่มทางด้านซ้ายและขวาของภาพสมดุลย์กัน จากที่กล่าวมาแล้วจะเห็นได้ว่าจำเป็นต้องมีโปรแกรมคีเลไทม์อีกหนึ่งโปรแกรม เพื่อเป็นโปรแกรมที่ควบคุมเวลาให้รับข้อมูลได้ตรงจุดที่ต้องการพอดี และควบคุมให้การสุ่มข้อมูลของฟิลด์ต่อ ๆ กันนั้นรับข้อมูลได้ห่างกันเป็นเวลา 0.5 ไมโครวินาทีหรือ 1 ทีสเตท ของข้อมูลในเส้นสะแกนตามแนวนอนด้วย ซึ่งลักษณะของโปรแกรมคีเลไทม์นี้จะกล่าวถึงต่อไป

ย้อนกลับมาถึงตัวอย่างการสุ่มข้อมูลจุดแรกของฟิลด์ที่ 1 นั้น เมื่อได้ใช้เวลาในการเอคเซคคิวท์คำสั่งไป 33 ทีสเตท ก็ไปเอคเซคคิวท์คำสั่งยังโปรแกรมคีเลไทม์ตามจุดเริ่มต้นสำหรับฟิลด์นั้น ๆ ซึ่งกำหนดไว้แล้วด้วยรีจิสเตอร์ IX สำหรับฟิลด์แรกนี้จะเริ่มเอคเซคคิวท์ที่ตำแหน่ง 0099 ในโปรแกรมคีเลไทม์เป็นต้นไป จนกระทั่งถึงคำสั่งที่ย้อนกลับไปยังโปรแกรม

รีเซทรวมเวลาในการเอคเซคคิวท์ได้ 102 ทีสเตท ดังนั้นเมื่อรวมเวลาเอคเซคคิวท์คำสั่งทั้งหมดตั้งแต่เริ่มรีเซทจนถึงคำสั่งให้รับข้อมูลเข้ามานี้เป็นเวลา 72.5 ไมโครวินาที หรือ 145 ทีสเตทพอดี

ในขณะที่รับข้อมูลเข้าเก็บในหน่วยความจำและเตรียมที่สำหรับเก็บข้อมูลของค่าต่อไปแล้ว ก็ต้องคำนวณเวลาไปด้วยเพื่อให้เวลาผ่านไปให้ครบช่วงของการสุ่มเส้นสะแกนตามแนวนอนกลุ่มนั้น นั่นคือต้องใช้คำสั่ง 3 คำสั่งที่กล่าวแล้วข้างต้นคอยควบคุมเวลาเพื่อให้เส้นสะแกนตามแนวนอน 4 เส้นหลังของกลุ่มนั้นผ่านไปก่อน จึงเริ่มเอนเอเบิลรีเซท (enable reset) เพื่อสุ่มเส้นต่อไป

ลักษณะของโปรแกรมดีเลไทม์

โปรแกรมดีเลไทม์นี้เป็นส่วนโปรแกรมย่อยของโปรแกรมการสุ่มค่าความเข้มของภาพดังกล่าวแล้วข้างต้น ซึ่งลักษณะของโปรแกรมดีเลไทม์นี้จะแบ่งเป็น 4 ตอนด้วยกัน โดยจะบรรจุคำสั่งของทุก ๆ ตอนไว้ในหน่วยความจำมีตำแหน่งต่อเนื่องกันไปดังที่ปรากฏอยู่ในตัวโปรแกรมนั้น ตามตารางที่ 4.3 นี้ได้จัดโปรแกรมดีเลไทม์ให้มีลักษณะเป็นตอน ๆ 4 ตอน เพื่อให้สะดวกแก่การศึกษาถึงการทำงานของโปรแกรมนี้

ตอนที่ 1			ตอนที่ 2			ตอนที่ 3			ตอนที่ 4		
ตำแหน่ง	คำสั่ง	ทีสเตท	ตำแหน่ง	คำสั่ง	ทีสเตท	ตำแหน่ง	คำสั่ง	ทีสเตท	ตำแหน่ง	คำสั่ง	ทีสเตท
0080	NOP	4	00A0	NOP	4	00C0	NOP	4	00E0	NOP	4
	25 ครั้ง			25 ครั้ง			25 ครั้ง			25 ครั้ง	
0098	NOP	4	00B8	NOP	4	00D8	NOP	4	00F8	NOP	4
0099	LD A, 00	7	00B9	BIT 7, A	8	00D9	LD I, A	9	00F9	JMP DELAY 2	10
009B	JMP DELAY 1	10	00BB	JMP DELAY 1	10	00DB	JMP DELAY 1	10			

ตารางที่ 4.3 แสดงโปรแกรมดีเลไทม์ทั้ง 4 ตอน

เนื่องจากการสุ่มค่าความเข้มของภาพในเส้นสะแกนตามแนวนอนแต่ละเส้นนั้น ได้กล่าวแล้วว่าจะสุ่มให้ได้ละเอียดที่สุดโดยถือเอาเวลาที่น้อยที่สุดที่ใช้ในการเอคเซคคิวท์คำสั่งเป็นหลัก ซึ่งเวลาที่น้อยที่สุดก็คือ 1 ทีสเตท หรือ 0.5 ไมโครวินาทีนั่นเอง และคำสั่งที่ใช้เวลาในการเอคเซคคิวท์น้อยที่สุดคือ NOP ก็ใช้เวลาถึง 4 ทีสเตทแล้ว ดังนั้นลักษณะของโปรแกรมดีเลไทม์จึงต้องจัดเป็น 4 ตอน เพื่อให้สัมพันธ์กับเวลาที่ใช้ในการเอคเซคคิวท์คำสั่ง NOP นั้น โดยให้ลักษณะการใช้โปรแกรมนี้มีจุดเริ่มต้นของการเอคเซคคิวท์คำสั่งต่างกันสำหรับฟิลด์แต่ละฟิลด์ที่ใช้สุ่มข้อมูลซึ่งมีผลให้ข้อมูลแต่ละจุดที่สุ่มมาได้ใน 1 เส้นสะแกนตามแนวนอนนั้นมีระยะห่างกันคิดเป็นเวลา 1 ทีสเตท หรือ 0.5 ไมโครวินาทีนั่นเอง

ตัวอย่างเช่น พิจารณาจากตารางที่ 4.3 คำสั่งที่ตำแหน่ง 0099, 00B9, 00D9 และ 00F9 ของแต่ละตอนนั้นใช้เวลาในการเอคเซคคิวท์เป็น 7, 8, 9 และ 10 ทีสเตท เรียงกันตามลำดับ จะเห็นว่าเวลาของการเอคเซคคิวท์คำสั่งต่างกัน 1 ทีสเตทตามลำดับไป ซึ่งจุดเริ่มต้นของการใช้โปรแกรมดีเลไทม์นี้จะเริ่มที่ตำแหน่ง 0099 สำหรับฟิลด์แรกที่ทำการสุ่ม ฟิลด์ที่สองจะเริ่มที่ 00B9 จนถึงฟิลด์ที่สี่เริ่มที่ 00F9 พอถึงฟิลด์ที่ห้าจะมาเริ่มที่ 0098 ซึ่งจะเห็นว่าเวลาที่ใช้ในการเอคเซคคิวท์คำสั่งที่ตำแหน่ง 0098 และ 0099 รวมเป็น 11 ทีสเตท ซึ่งต่างจากฟิลด์ที่สี่ 1 ทีสเตท และฟิลด์ที่หกเริ่มที่ตำแหน่ง 00B8 คิดเป็นเวลาในการเอคเซคคิวท์ 12 ทีสเตทเมื่อรวมเข้ากับคำสั่งที่ตำแหน่ง 00B9 ดังนั้นเวลาที่ใช้ในการเอคเซคคิวท์ฟิลด์ที่หกก็จะต่างจากฟิลด์ที่ห้า 1 ทีสเตทเช่นกัน เมื่อทำการสุ่มข้อมูลครบ 100 ฟิลด์ ก็จะได้ข้อมูลครบ 4800 ค่า คือ 1 ฟิลด์สามารถสุ่มข้อมูลได้ 48 ค่า จากเส้นสะแกนตามแนวนอน 48 เส้นดังกล่าวแล้วข้างต้น ดังนั้นเมื่อสุ่มข้อมูลครบ 100 ฟิลด์จึงได้ข้อมูลครบ 4800 ค่า ตามต้องการ แต่ลักษณะของข้อมูลที่สุ่มมาได้และเก็บอยู่ในหน่วยความจำที่ตำแหน่ง 0400 ถึง 1600 นั้น จะเก็บในลักษณะของข้อมูลตามแนวตั้งซึ่งสุ่มจากฟิลด์แต่ละฟิลด์เรียงต่อกัน 100 ฟิลด์ และข้อมูลเหล่านั้นคือค่าความเข้มของภาพ ณ จุดที่ทำการสุ่มค่านั่นเอง

ดังนั้นการที่จะพิมพ์ภาพนั้นให้ปรากฏในลักษณะของตัวอักษรซึ่งแทนความเข้มของภาพ และจัดลำดับของตัวอักษรเหล่านั้นให้เรียงต่อกันในลักษณะของเส้นสะแกนตามแนวนอน 48 เส้นนั้นต้องใช้ระบบโปรแกรมการจัดตัวอักษรและการพิมพ์ภาพซึ่งจะกล่าวถึงในบทต่อไป