ภาคผนวก ข

หน่วยความจำข้อมูลและหน่วยความจำโปรแกรมที่ใช้สำหรับคอนโทรลยูนิต

หน่วยความจำทั้งหมดที่ใช้สำหรับคอนโทรลยูนิตมีทั้งหมด 5 ชนิดคือ Internal Program Memory, Internal RAM, Internal EEPROM, External EEPROM และ External RAM โดยที่ Internal คือ หน่วยความจำที่อยู่ภายในตัวไมโครคอนโทรลเลอร์และ External คือหน่วยความจำที่นำมาต่อพ่วงภายนอก

1. Internal Program Memory สำหรับไมโครคอนโทรลเลอร์ที่นำมาใช้เบอร์ AT89S8252 ผลิตโดยบริษัท Atmel มี Internal Program Memory ขนาด 8 KB ซึ่งถูกบรรจุโปรแกรมไว้ 6,958 ไบต์ แต่หน่วยความจำส่วนนี้สามารถอ่านได้เท่านั้น ไม่สามารถเขียนบันทึกข้อมูลเพิ่มเติมได้ ถ้าต้องการแก้ไขโปรแกรมจะต้องใช้คอมพิวเตอร์หรือเครื่องมืออื่นๆ เป็นตัวช่วยในการส่งโปรแกรมให้กับตัววงจรรวม (Integrate Circuit : IC) นี้

2. ในส่วนของ Internal RAM มีทั้งหมด 256 ไบต์ เป็น Direct Addressing 128 ไบต์ คือ ตั้งแต่ตำแหน่ง 00H ถึง 7FH ที่เหลือเป็น Indirect Addressing 128 ไบต์ ที่ตำแหน่ง 80H ถึง FFH หน่วยความจำนี้มีการใช้งานดังนี้

2.1 ที่ตำแหน่ง 00H ถึง 07H ใช้สำหรับเป็นหน่วยความจำแบบ Register สำหรับใช้งานทั่วไป มีทั้งหมด 8 ไบต์ โดยกำหนดให้เป็น R0 – R7 ตามค่าปกติของตัวไมโครคอนโทรลเลอร์เอง

2.2 ตำแหน่งบิตที่ 7EH และ 7FH เป็นส่วนของ Internal RAM ที่เข้าถึงได้ในระดับบิตใช้เป็นตัวชี้ชนิดของข้อมูลและสถานะข้อมูลตามลำดับ

2.3 ตำแหน่ง A0H ถึง 70H กำหนดให้เป็นตัวแปรที่ทำหน้าที่เฉพาะงาน ตามที่กำหนดไว้ในส่วนของ Variable Declaration ของโปรแกรมภาษาแอสเซมบลี รายละเอียดเพิ่มเติมในภาคผนวก ช

2.4 ตำแหน่ง 80H ถึง 8AH ซึ่งอยู่ในส่วนของ Indirect Addressing ใช้เป็นที่พักข้อมูลที่รับมาจากเทอรามิเตอร์ แล้วทำการปรับค่าให้อยู่ในรูปแบบวิทยาศาสตร์ (เช่น -1.25E-3) หลังจากนั้นจึงค่อยถ่ายโอนไปไว้ในส่วนของ External RAM

2.5 ตำแหน่ง 9BH ถึง FFH กำหนดให้เป็นพื้นที่ของ Stack Pointer

3. Internal EEPROM ใช้บันทึกค่าต่างๆ ที่ถูกกำหนดจากเมนู Set...

4. External EEPROM มีขนาด 32 KB ใช้สำหรับบันทึกข้อมูลที่ได้จากการวัดในแต่ละชุดในลักษณะของแฟ้มข้อมูล มีการจัดการใช้งานดังนี้

4.1 ที่ตำแหน่ง 01H ใช้สำหรับบันทึกจำนวนแฟ้มข้อมูลที่มีอยู่ทั้งหมด ถ้าที่ตำแหน่งนี้ไม่เท่ากับศูนย์ที่ตำแหน่งอื่นๆ จะแปรตามตำแหน่งนี้ โดยข้อมูลที่เป็นส่วนอธิบายแฟ้มข้อมูลของไฟล์ใดๆ (X) จะเริ่มที่ตำแหน่ง 16X และในแต่ละแฟ้มจะมีส่วนอธิบาย 16 ไบต์ เช่น แฟ้มข้อมูลที่ 1 ก็จะมีส่วนอธิบายเริ่มที่ตำแหน่ง 10H (16) ถึงตำแหน่ง 20H (32) มีการใช้งานตามรายละเอียดในหัวข้อ 4.2

4.2 ส่วนอธิบายแฟ้มข้อมูล บันทึกข้อมูลในรูปแบบชนิด Byte โดยตรง แต่มีบางตำแหน่งจะนำมาต่อกันเป็นข้อมูลชุดเดียวมี 2 ไบต์ ดังนี้

4.2.1 ตำแหน่ง 16X บันทึกค่า High Byte ของตำแหน่งเริ่มต้นของชุดข้อมูลของแฟ้มที่ตัวชี้นี้อยู่

4.2.2 ตำแหน่ง 16X+1 บันทึกค่าตำแหน่งเริ่มต้นของชุดข้อมูล (Low Byte) ในการใช้หน่วยความจำที่ตำแหน่ง 16X และ 16X+1 จะนำมาต่อกันได้ 16 บิตเป็นข้อมูลชนิด Word จึงทำให้สามารถชี้ตำแหน่งข้อมูลได้สูงสุดที่ FFFFH (65,535) ซึ่งเพียงพอสำหรับการชี้หน่วยความจำที่ทุกๆ ตำแหน่งของ External EEPROM ในกรณีพิเศษถ้าแฟ้มที่กำลังบันทึกนี้เป็นแฟ้มแรกคือที่ตำแหน่ง 01H (ตามหัวข้อ 4.1) มีค่าเท่ากับ 1 ค่าของ Word ในที่นี้จะกำหนดให้ด้วยค่าปกติ (Default) เท่ากับ 640H (1,600) นั่นคือที่ตำแหน่ง 16X เท่ากับ 06H และที่ตำแหน่ง 16X+1 เท่ากับ 40H ส่วนในกรณีอื่นๆ ถ้าแฟ้มที่กำลังจะบันทึกไม่ใช่แฟ้มแรกบนหน่วยความจำ ค่าของ Word ที่นี้จะเท่ากับค่าของ Word ในตำแหน่งสุดท้ายของชุดข้อมูลของแฟ้มก่อนหน้านี้ (ตามหัวข้อ 4.2.5) บวกด้วยหนึ่ง ถ้าสมมติว่าแฟ้มปัจจุบันที่กำลังจะเขียนเป็นแฟ้มที่ X สามารถหาตำแหน่งของชุดข้อมูลในหน่วยความจำเป็นสัญญลักษณ์ได้คือ

$$\{ @[16X], @[16X+1] \} = \{ @[16(X-1)+3], @[16(X-1)+4] \} + 1$$

4.2.3 ตำแหน่ง 16X+2 บันทึกจำนวนข้อมูลทั้งหมดที่มีในแฟ้มนี้ เช่น สมมติในแฟ้มนี้มีข้อมูล 3 จุดดังนี้ 1.23 mV, 1.25 mV และ 1.26 mV จะทำให้ค่าที่ตำแหน่ง 16X+2 เก็บค่าเท่ากับ 3 ไว้

4.2.4 ตำแหน่ง 16X+3 บันทึกค่าตำแหน่งสุดท้ายของชุดข้อมูล (High Byte)

4.2.5 ตำแหน่ง 16X+4 บันทึกค่าตำแหน่งสุดท้ายของชุดข้อมูล (Low Byte) ที่ตำแหน่ง 16X+3 และ 16X+4 นั้นจะใช้เช่นเดียวกับตำแหน่ง 16X+1 และ 16X+2 แต่เป็นตำแหน่งสุดท้ายของชุดข้อมูล ซึ่งจะมีความสำคัญมาก เมื่อจะมีการบันทึกแฟ้มใหม่ต่อจากแฟ้มนี้ จะต้องอ่านค่าจากทั้ง 2 ตำแหน่งนี้เพื่อทำการบันทึกข้อมูลในตำแหน่งถัดไปบนหน่วยความจำ

4.2.6 ตำแหน่ง 16X+5 บันทึกระยะห่างของขั้วไฟฟ้า (Electrode Interval) แต่จะไม่ได้อยู่ในรูปแบบของจุดทศนิยม เช่น ถ้าระยะห่างมีขนาด 5.0 เมตร จะบันทึกเป็น 50 ดังนั้นจะต้องหารค่าที่อ่านได้ด้วย 10 เสมอ

4.2.7 ตำแหน่ง 16X+6 บันทึกชนิดของข้อมูล เนื่องจากระบบที่สร้างขึ้นนี้สามารถทำการวัดได้ทั้งเอสพีและสภาพต้านทานไฟฟ้าแต่ในการนำข้อมูลไปแปลความจะมีการจัดระเบียบของชุดข้อมูลไม่เหมือนกัน จึงมีความจำเป็นที่จะต้องบันทึกไว้ว่าข้อมูลในแฟ้มนี้เป็นชนิดใด ซึ่งถ้าข้อมูลเป็นเอสพีที่ตำแหน่งนี้จะเท่ากับ 0 และถ้าเป็นข้อมูลของสภาพต้านทานไฟฟ้าที่ตำแหน่งนี้จะเท่ากับ 1 โดยที่ข้อมูลของสภาพต้านทานไฟฟ้านี้จะมีการจัดขบวนขั้วไฟฟ้า (Electrode Array) แบบ Double Dipole แต่ในอนาคตจะมีการพัฒนาต่อให้สามารถวัดได้หลายรูปแบบ ดังนั้นที่ตำแหน่งนี้ก็จะไว้ใช้สำหรับบันทึกชนิดของข้อมูลที่จะพัฒนาต่อด้วย

4.2.8 ที่ตำแหน่ง 16X+7 จะมีการใช้งานเมื่อที่ตำแหน่ง 16X+6 (ตามหัวข้อ 4.2.7)มีค่าเท่ากับหนึ่ง ตำแหน่งนี้จะบันทึกค่าของ n ในรูปแบบของระเบียบการวัดสภาพต้านทานไฟฟ้าแบบ Double Dipole โดยที่ n สามารถปรับได้ตามที่ผู้ทำการสำรวจกำหนดไว้ ซึ่งจะไม่เกิน 6 แต่ทั้งนี้ก็จะขึ้นอยู่กับจำนวนขั้วไฟฟ้าทั้งหมด (ตามหัวข้อ 4.2.10) ด้วย ค่าของ n จะมีความสำคัญในการคำนวณหาจำนวนของ Station ทั้งหมด ซึ่งโปรแกรมไม่อนุญาตให้เปลี่ยนค่า แต่จะคำนวณให้เอง สมมติกำหนดตัวแปรให้ขั้วไฟฟ้าทั้งหมดเป็น E และจำนวน Station เป็น S จะได้ความสัมพันธ์ระหว่าง E, n และ S ดังนี้

- ถ้า E=4 จะได้ n=1 และ S=1
- ถ้า 4<E<10 สามารถปรับค่า 0<n<(E-2) ทำให้ได้ S=E-N-2
- ถ้า E>9 สามารถปรับค่า 0<n<7 ทำให้ได้ S=E-N-2

4.2.9 ตำแหน่ง 16X+8 บันทึกจำนวนแฟ้มข้อมูลที่ทำการวัดแบบต่อเนื่องกันของข้อมูลเอสพีซึ่งจะใช้งานเมื่อตำแหน่ง 16X+6 (ตามหัวข้อ 4.2.7) มีค่าเท่ากับศูนย์ ทั้งนี้เนื่องมาจากในการวัดเอสพีนั้นสามารถวัดได้ 2 แบบคือ แบบธรรมดาและแบบอัตโนมัติ โดยถ้าทำการวัดแบบธรรมดาจากการใช้เมนู Run SP จะสามารถกำหนดให้วัดซ้ำที่ตำแหน่งเดิมได้ ซึ่งที่ตำแหน่งนี้จะทำการบันทึกไว้ว่าในแฟ้มข้อมูลนี้มีการวัดซ้ำกี่ครั้ง เช่น สมมติว่ามีจำนวนขั้วไฟฟ้าทั้งหมด (ตามหัว

ข้อ 4.2.10) 10 ขั้วและมีการวัดซ้ำ 4 ครั้ง จะทำให้ที่ตำแหน่งนี้เท่ากับ 4 และส่งผลให้ตำแหน่ง 16X+2 มีการเปลี่ยนค่าตามไปด้วยคือ จะเปลี่ยนจาก 0AH (10) เป็น 50H (80)

4.2.10 ตำแหน่ง 16X+9 บันทึกจำนวนขั้วไฟฟ้า (Electrode) ทั้งหมด หรือเท่ากับจำนวนไคลเอนท์ทั้งหมดด้วย เพราะจะต้องใช้ไคลเอนท์ 1 เครื่องต่อขั้วไฟฟ้า 1 ขั้ว

4.2.11 ตำแหน่ง 16X+10 บันทึกเวลาที่ทำการวัดข้อมูล ในตำแหน่งของชั่วโมง

4.2.12 ตำแหน่ง 16X+11 บันทึกเวลาที่ทำการวัดข้อมูล ในตำแหน่งของนาที

4.2.13 ตำแหน่ง 16X+12 บันทึกเวลาที่ทำการวัดข้อมูล ในตำแหน่งของวินาที

4.2.14 ตำแหน่ง 16X+13 บันทึกวันที่ทำการวัดข้อมูล

4.2.15 ตำแหน่ง 16X+14 บันทึกเดือนที่ทำการวัดข้อมูล

4.2.16 ตำแหน่ง 16X+15 บันทึกปีที่ทำการวัดข้อมูล

4.3 ส่วนของข้อมูลดิบที่ได้จากการวัด จะบันทึกข้อมูลในรูปแบบของรหัสแอสสกี (ASSCII Code) มีการเริ่มต้นบันทึกของแฟ้มแรกที่ตำแหน่ง 640H (1,600) ส่วนแฟ้มอื่นๆ ก็จะถูกบันทึกต่อไป โดยมีตำแหน่งเริ่มต้นและตำแหน่งสิ้นสุดของแฟ้มข้อมูลตามหัวข้อ 4.2.2 และ 4.2.5 ตามลำดับ สำหรับข้อมูล 1 จุดจะใช้หน่วยความจำ 10 ไบต์ เช่น ถ้าเครื่องเทอรามิเตอร์อ่านค่าได้ –123.5 mV ก็จะบันทึกเป็น –123.5E-3 โดยมีลำดับของรหัสแอสสกีที่บันทึกไว้ในหน่วยความจำตามตาราง 6.1

ตาราง 6.1ตัวอย่างการบันทึกข้อมูล

ลำดับที่	1	2	3	4	5	6	7	8	9	10
รหัสแอสสกี	2DH	31H	32H	33H	2EH	35H	45H	2DH	33H	20H
ค่าประจำตำแหน่ง	-	1	2	3	.	5	E	-	3	

ในลำดับที่ 10 จะมีรหัสแอสสกีเป็น 20H ซึ่งเป็นช่องว่าง (Space) มีค่าเท่ากับการกดปุ่ม Space Bar บนคีย์บอร์ดของเครื่องคอมพิวเตอร์ ถ้าหากว่าในข้อมูลจุดใดใช้ตัวอักษรไม่ครบ 10 ตัวจะทำให้ที่เหลือเป็นช่องว่างต่ออยู่จนครบ ดังนั้นถ้าสมมติว่าใน 1 แฟ้มมีข้อมูลอยู่ 40 จุด ก็จะใช้หน่วยความจำในการบันทึก 400 ไบต์ เมื่อรวมกับส่วนอธิบายแฟ้มข้อมูลอีก 16 ไบต์ จะทำให้ใช้หน่วยความจำ 416 ไบต์ ซึ่งจาก EEPROM ที่ใช้มีขนาด 32 KB หรือเท่ากับ 32,768 ไบต์ โดยที่ข้อมูลดิบในแฟ้มเริ่มใช้หน่วยความจำตั้งแต่ตำแหน่งที่ 1,600 จึงเหลืออยู่ 31,168 ไบต์ อนุมาณได้ว่าสามารถบันทึกแฟ้มที่มีขนาดเดียวกันนี้ได้ประมาณ 77 แฟ้ม ส่วนสาเหตุที่เริ่มบันทึกตั้งแต่ตำแหน่ง 1,600 เพราะในตำแหน่ง 16 ถึง 1,599 ใช้สำหรับเก็บส่วนอธิบายแฟ้มข้อมูล (ตามราย

ละเอียดในหัวข้อ 4.2) ซึ่งสามารถใช้อธิบายแฟ้มได้ถึง 99 แฟ้ม ค่านี้จะเป็นจำนวนสูงสุดของแฟ้มข้อมูลที่จะทำการบันทึกได้ เมื่อพิจารณาร่วมกับพื้นที่เก็บข้อมูลดิบที่มีขนาด 31,168 ไบต์ แล้วพบว่าแต่ละแฟ้มจะมีข้อมูลได้ 31 จุดจึงจะทำให้สามารถบันทึกได้ครบ 99 แฟ้ม แต่ทั้งนี้ในการสำรวจทั่วไปสามารถกำหนดได้อิสระ สามารถให้ 1 แฟ้มมีข้อมูลกี่จุดก็ได้ และสำหรับพื้นที่หน่วยความจำที่เหลือคือ ที่ตำแหน่ง 0 และ 2 ถึง 15 จะเว้นไว้สำหรับใช้ในการพัฒนาต่อไปในอนาคต (ตำแหน่งที่ 1 ใช้เก็บจำนวนแฟ้มทั้งหมด ตามรายละเอียดในหัวข้อ 4.1)

5. External RAM เป็นหน่วยความจำภายนอกที่นำมาต่อพ่วงเบอร์ M48T08 ผลิตโดยบริษัท STMicroelectronics มีขนาด 8 KB และมีความพิเศษกว่าหน่วยความจำ RAM ทั่วไปเพราะมีนาฬิกา (Real Time Clock) อยู่ภายในด้วย แม้จะไม่มีไฟฟ้าเลี้ยงวงจร นาฬิกานี้ก็จะยังสามารถเดินต่อไปได้เพราะมีแบตเตอรี่อยู่ภายในแล้ว การนำมาต่อเข้ากับไมโครคอนโทรลเลอร์มีการจัดตำแหน่งไม่ให้ซ้อนทับกับตำแหน่งของ Internal Program Memory ที่มีขนาด 8 KB สาเหตุที่ต่อหน่วยความจำแบบนี้ก็เพื่อไว้ใช้ในกรณีที่ Internal Program Memory ไม่พอใช้ก็จะใช้ External RAM เป็น Program Memory ด้วย จึงทำให้ตำแหน่งของหน่วยความจำ External RAM นี้เริ่มตั้งแต่ 2000H (8,192) ถึง 3FFFH (16,383) ในการอธิบายถึงการจัดการกับหน่วยความจำชนิดนี้จึงใช้ตำแหน่งที่ตัวไมโครคอนโทรลเลอร์อ้างถึง ซึ่งจะไม่ใช่ตำแหน่งจริงบนตัว External RAM เอง นั่นคือที่ตำแหน่ง 2000H ถึง 3FFFH จะเท่ากับตำแหน่งภายใน External RAM คือ 0000H ถึงตำแหน่ง 2000H หน่วยความจำนี้มีการใช้งานดังนี้

5.1 ตั้งแต่ตำแหน่ง 0000H เป็นต้นไปจะใช้สำหรับเก็บข้อมูลที่ได้จากการวัดไว้ชั่วคราว โดยรับมาจากหน่วยความจำ Internal RAM (ตามหัวข้อ 2.4) ซึ่ง Internal RAM จะทำหน้าที่รับข้อมูลจากการส่งมาจาก Terrameter โดยตรง ซึ่งในการวัด 1 ครั้งนอกจากมีข้อมูลตัวเลขแล้วจะมีรหัสแอสสกีอื่นๆ อีกด้วย ในการโอนย้ายข้อมูลจาก Internal RAM มาไว้ที่ External RAM จะตัดเอาเฉพาะส่วนที่เป็นตัวเลขเท่านั้นและแปลงให้ 1 จุดวัดมีการเก็บข้อมูล 10 ตัวอักษร ดังนั้นถ้าในการวัด 1 ชุดมีข้อมูล 40 จุด ก็จะใช้หน่วยความจำ 400 ไบต์ โดยเริ่มที่ตำแหน่ง 0000H (0) ถึงตำแหน่ง 0190H (400)

5.2 ในส่วนของนาฬิกา มีรูปแบบการเก็บข้อมูลแบบ BCD มีการใช้งานดังนี้

5.2.1 ที่ตำแหน่ง 3FF8H (16,376) เป็นส่วนควบคุมการเข้าถึงหน่วยความจำในส่วนที่เป็นนาฬิกา เมื่อต้องการอ่านเวลาหรือตั้งเวลาจะต้องกระทำที่ตำแหน่งนี้ก่อน โดยจะกำหนดให้บิตที่ 7 เป็น Logic 1 เมื่อต้องการตั้งเวลา และกำหนดให้บิตที่ 6 เป็น Logic 1 เมื่อต้องการอ่านเวลา หลังจากอ่านหรือตั้งเวลาเสร็จก็ต้องกำหนดให้ทั้งสองบิตเป็น Logic 0 เพื่อความพร้อมในการ

ที่จะอ่านหรือตั้งเวลาในครั้งต่อไป ทั้งนี้เนื่องจากในการอ่านเวลาจากตำแหน่งหน่วยความจำที่เก็บค่าเวลาไว้นั้น ไม่ใช่ตำแหน่งของนาฬิกาจริงๆ แต่จะถูกเปลี่ยนให้ตรงกับค่าเวลาของนาฬิกาเมื่อมีการกำหนดให้บิตที่ 6 เป็น Logic 1 เท่านั้น และค่านี้จะคงที่ต่อไปแม้ว่าจะอ่านซ้ำกี่ครั้งค่าเวลาก็ยังคงเวลาเดียว ดังนั้นเมื่อต้องการอ่านเวลาที่เป็นปัจจุบันของนาฬิกาจึงต้องกำหนดให้บิตที่ 6 เปลี่ยนจาก Logic 0 ไปเป็น Logic 1 เท่านั้น ส่วนบิตที่ 5 ถ้ากำหนดเป็น Logic 1 จะเป็นการกำหนดให้นาฬิกาหยุดเดิน

5.2.2 ตำแหน่ง 3FF9H (16,377) เก็บเวลาในส่วนของวินาที โดยบิตที่ 0 ถึง 3 เก็บค่าหลักหน่วย มีค่าอยู่ระหว่าง 0 ถึง 9 ส่วนบิตที่ 4 ถึง 6 เก็บค่าหลักสิบ มีค่าอยู่ระหว่าง 0 ถึง 5 สำหรับบิตที่ 7 ไม่ได้ใช้งานและต้องกำหนดเป็น Logic 0 เสมอ

5.2.3 ตำแหน่ง 3FFAH (16,378) เก็บเวลาในส่วนของนาที โดยบิตที่ 0 ถึง 3 เก็บค่าหลักหน่วย มีค่าอยู่ระหว่าง 0 ถึง 9 ส่วนบิตที่ 4 ถึง 6 เก็บค่าหลักสิบ มีค่าอยู่ระหว่าง 0 ถึง 5 สำหรับบิตที่ 7 ไม่ได้ใช้งานและต้องกำหนดเป็น Logic 0 เสมอ

5.2.4 ตำแหน่ง 3FFBH (16,379) เก็บเวลาในส่วนของชั่วโมง โดยบิตที่ 0 ถึง 3 เก็บค่าหลักหน่วย มีค่าอยู่ระหว่าง 0 ถึง 9 ส่วนบิตที่ 4 และ 5 เก็บค่าหลักสิบ มีค่าอยู่ระหว่าง 0 ถึง 2 สำหรับบิตที่ 6 และ 7 ไม่ได้ใช้งานและต้องกำหนดเป็น Logic 0 เสมอ

5.2.5 ตำแหน่ง 3FFCH (16,380) เก็บค่าของวันด้วยบิตที่ 0 ถึง 2 มีค่าอยู่ระหว่าง 0 ถึง 7 ส่วนบิตอื่นๆ ที่เหลือไม่ได้ใช้งานและต้องกำหนดเป็น Logic 0 เสมอ

5.2.6 ตำแหน่ง 3FFDH (16,381) เก็บวันที่ โดยบิตที่ 0 ถึง 3 เก็บค่าหลักหน่วย มีค่าอยู่ระหว่าง 0 ถึง 9 ส่วนบิตที่ 4 ถึง 5 เก็บค่าหลักสิบ มีค่าอยู่ระหว่าง 0 ถึง 3 ส่วนบิตที่ 6 และ 7 ไม่ได้ใช้งานและต้องกำหนดเป็น Logic 0 เสมอ

5.2.7 ตำแหน่ง 3FFEH (16,382) เก็บค่าของเดือน โดยบิตที่ 0 ถึง 3 เก็บค่าหลักหน่วย มีค่าอยู่ระหว่าง 0 ถึง 9 ส่วนบิตที่ 4 เก็บค่าหลักสิบ มีค่าอยู่ระหว่าง 0 และ 1 ส่วนบิตอื่นๆ ที่เหลือไม่ได้ใช้งานและต้องกำหนดเป็น Logic 0 เสมอ

5.2.8 ตำแหน่ง 3FFFH (16,383) เก็บค่าของปี โดยบิตที่ 0 ถึง 3 เก็บค่าหลักหน่วย มีค่าอยู่ระหว่าง 0 ถึง 9 ส่วนบิตที่ 4 ถึง 7 เก็บค่าหลักสิบ มีค่าอยู่ระหว่าง 0 ถึง 9

ในส่วนของ External EEPROM ที่มีขนาด 32 KB นั้น ถ้าในอนาคตมีการพัฒนาเครื่องมือนี้เพิ่มแล้วทำให้หน่วยความจำส่วนนี้ไม่พอใช้ ก็จะสามารถนำมาเพิ่มได้อีก 7 ตัว รวมทั้งหมดจะได้ 256 กิโลไบต์