

บทที่ 3

วิธีการวิจัย

การวิจัยเรื่องมิเตอร์วัดรังสีโดยใช้ไมโครคอนโทรลเลอร์เป็นฐานสามารถแบ่งการออกแบบเป็นสองส่วนหลักๆ คือทางด้านฮาร์ดแวร์และซอฟต์แวร์ ทางด้านฮาร์ดแวร์เป็นการศึกษาและออกแบบวงจรในส่วนต่างๆทั้งวงจรจ่ายไฟฟ้าศักย์สูง ซึ่งมีหน้าที่สร้างศักย์ไฟฟ้าที่เหมาะสมสำหรับหัววัดไกเกอร์-มุลเลอร์เพื่อให้สามารถวัดรังสีได้อย่างถูกต้อง วงจรส่วนไมโครคอนโทรลเลอร์จะทำหน้าที่ในการควบคุมการทำงานของระบบ รวมทั้งการวัดสัญญาณที่ได้จากหลอดไกเกอร์-มุลเลอร์ วงจรส่วนนี้ประกอบด้วยสวิทช์รับข้อมูลการทำงานของระบบ และส่วนแสดงผลที่เป็นหน่วยแสดงผลแบบผลึกเหลวขนาด 8 ตัวอักษร 2 บรรทัดรวมทั้งการเชื่อมต่อกับคอมพิวเตอร์ส่วนบุคคลผ่านการเชื่อมต่อแบบ USB (Universal Serial Bus) ส่วนที่สองเป็นการศึกษาและออกแบบทางด้านซอฟต์แวร์ซึ่งเป็นซอฟต์แวร์ที่ใช้ในการทำงานของไมโครคอนโทรลเลอร์โดยเป็นส่วนควบคุมการทำงานของระบบทั้งหมด ซอฟต์แวร์จะต้องทำงานสัมพันธ์กันทุกส่วน ทั้งส่วนที่ใช้ในการรับข้อมูลอินพุตจากสวิทช์ การแสดงผลออกทางหน่วยแสดงผลแบบผลึกเหลว การเชื่อมต่อกับคอมพิวเตอร์ส่วนบุคคล รวมทั้งส่วนการวัดสัญญาณจากหลอดไกเกอร์-มุลเลอร์

3.1 การออกแบบส่วนฮาร์ดแวร์

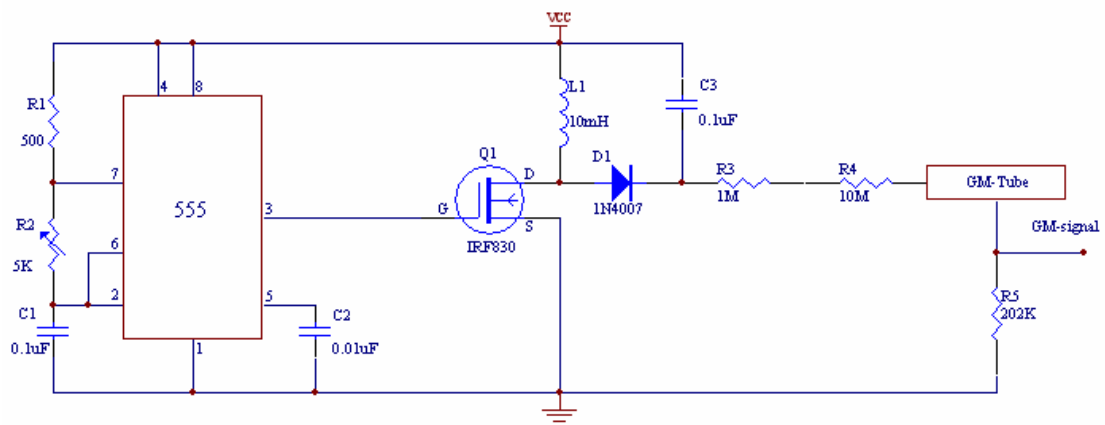
3.1.1 วงจรไฟฟ้าศักย์สูง

เนื่องจากหลอดไกเกอร์-มุลเลอร์ที่ใช้ต้องการศักย์ไฟฟ้าใช้งานอยู่ในช่วงระหว่าง 450 – 650 โวลต์ หรือค่าที่แนะนำโดยผู้ผลิตคือ 500 โวลต์ ดังนั้นจึงต้องทำการออกแบบวงจรที่สามารถสร้างศักย์ไฟฟ้าให้มีค่าดังที่ต้องการซึ่งใช้วงจรดังรูปที่ 3.1 ส่วนสำคัญส่วนแรกคือส่วนของวงจรไอซีเบอร์ 555 ที่ทำหน้าที่เป็นวงจรอะอสเตเบิลที่ใช้สำหรับสร้างสัญญาณความถี่ออกมาซึ่งกำหนดได้ด้วยตัวต้านทานและตัวเก็บประจุที่นำมาต่อประกอบกับตัวไอซีซึ่งสามารถคำนวณหาค่าความถี่ที่ผลิตออกมาได้ด้วยสมการ

$$f = \frac{1.44}{(R1 + 2R2)C1}$$

ในการทำงานเริ่มจากส่วนของวงจรอะอสเตเบิลที่เป็นวงจรกำเนิดสัญญาณความถี่ออกมาที่เอาต์พุตขาที่ 3 โดยในการกำหนดความถี่ใช้ค่า R1 มีค่า 500 โอห์ม R2 เป็นความ

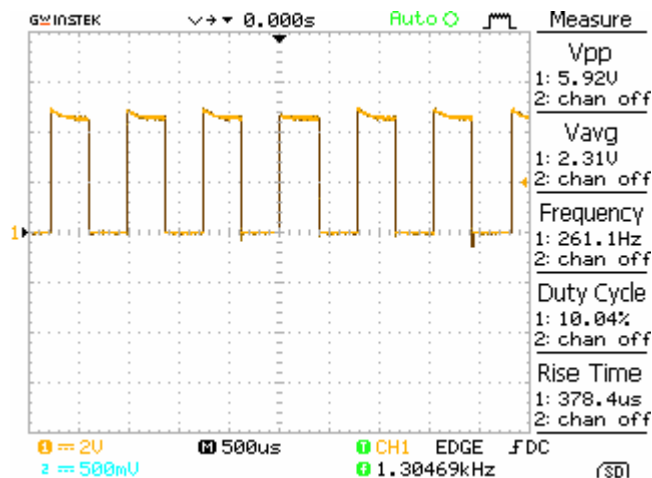
ต้านทานปรับค่าได้มีค่า 5 กิโลโอห์ม และใช้ C1 ค่า 0.1 ไมโครฟารัด ด้วยค่าต่างๆเหล่านี้จะทำให้ได้ความถี่จากการคำนวณด้วยสมการมีค่า 1371.43 เฮิรตซ์ สัญญาณนี้จะออกมาทางขาที่ 3 ของไอซีป้อนเข้าสู่ขาเกตของมอสเฟตเบอร์ IRF830 จะทำให้มอสเฟตเปิดและปิดคล้ายกับการทำงานของสวิตช์โดยขึ้นกับสัญญาณที่ออกมาจากไอซี 555 และที่ขาเดรนของมอสเฟตต่อกับตัวเหนี่ยวนำ ในขณะที่มอสเฟตเปิดและปิดตามจังหวะของสัญญาณจะทำให้ตัวเหนี่ยวนำถูกต่อเข้ากับกราวด์ของวงจรเป็นจังหวะเช่นกันทำให้ได้ศักย์ไฟฟ้ามีค่าสูงขึ้นเมื่อศักย์ไฟฟ้านี้ผ่านไดโอดเบอร์ 1N4007 จะทำให้เป็นไฟฟ้ากระแสตรงผ่านความต้านทาน 1 เมกะโอห์ม และ 10 เมกะโอห์ม เข้าสู่ขั้วแอโนดของหลอดไทเกอร์-มูลเลอร์ ในขณะที่ขั้วแคโทดต่อลงกราวด์ผ่านความต้านทาน 202 กิโลโอห์ม



รูปที่ 3.1

วงจรไฟฟ้าศักย์สูงสำหรับหลอดไทเกอร์-มูลเลอร์

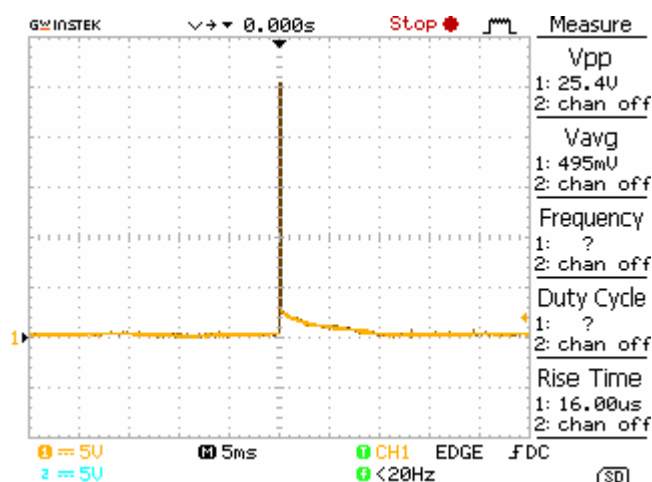
วงจรแรกนั้นเริ่มจากการออกแบบส่วนของวงจรกำเนิดสัญญาณเพื่อใช้ในการสร้างไฟฟ้าศักย์สูงซึ่งใช้ไอซีเบอร์ 555 มาออกแบบเป็นวงจรอะสเตเบิลได้ความถี่ที่สร้างออกมามีค่าประมาณ 1300 เฮิรตซ์ รูปสัญญาณที่ได้แสดงได้ดังรูปที่ 3.2 เป็นลักษณะสัญญาณที่ได้ออกมาจากเอาต์พุตขา 3 ของไอซีเบอร์ 555



รูปที่ 3.2

ลักษณะสัญญาณที่ได้จากวงจรอะอสเตเบิล

สัญญาณนี้จะป้อนให้กับมอสเฟททรานซิสเตอร์เพื่อให้ทำหน้าที่เป็นสวิตช์เปิดปิดให้ตัวเหนี่ยวนำต่อวงจรลงกราวด์ ทำให้เกิดเป็นไฟฟ้าศักย์สูงมีค่าประมาณ 469 โวลต์ป้อนให้กับหลอดไกเกอร์-มุลเลอร์ ทำให้หลอดไกเกอร์-มุลเลอร์พร้อมที่จะทำงาน เมื่อมีรังสีผ่านเข้าไปในหลอดทำให้เกิดการแตกตัวจะทำให้เกิดสัญญาณเอาต์พุตออกมามีค่าแรงดันประมาณ 25 โวลต์มีลักษณะดังรูปที่ 3.3 สัญญาณนี้ก่อนจะนำเข้าสู่ส่วนของวงจรไมโครคอนโทรลเลอร์จะผ่านวงจรปรับรูปสัญญาณเพื่อลดระดับแรงดันลงมาให้เป็น 5 โวลต์เพื่อให้เหมาะสมสำหรับไมโครคอนโทรลเลอร์และสัญญาณนี้จะถูกนำเข้าสู่อินพุตเพื่อนำไปประมวลผลต่อไป



รูปที่ 3.3

ลักษณะสัญญาณเอาต์พุตทางแคโทดของหลอดไกเกอร์-มูลเลอร์

3.1.2 ไมโครคอนโทรลเลอร์

ในการทดลองใช้ไมโครคอนโทรลเลอร์เบอร์ PIC18F2550 ของบริษัทไมโครชิพ สำหรับการใช้งานมีเพียงการต่ออุปกรณ์เพิ่มอีกบางส่วนก็จะสามารถทำให้ไมโครคอนโทรลเลอร์ เบอร์ PIC18F2550 สามารถทำงานได้ส่วนแรกเป็นไฟเลี้ยงของวงจร ไฟบวกต่อเข้ากับขา 20 ส่วน กราวด์ต่อเข้ากับขา 8 และ 19 ส่วนถัดมาเป็นส่วนของวงจรรีเซ็ตโดยถ้าต้องการให้ ไมโครคอนโทรลเลอร์ทำงานในสถานะปกติต้องทำให้ขา MCLR มีสัญญาณไฟเป็นบวกแต่ถ้า ต้องการให้รีเซ็ตเพื่อเริ่มต้นการทำงานงานใหม่ต้องทำให้สัญญาณที่ขานี้เป็นกราวด์ ในการ ออกแบบใช้ความต้านทานต่อเข้ากับไฟเลี้ยงของวงจรและต่อเข้ากับขา MCLR เพื่อให้ขานี้มีลอจิก ‘1’ เพื่อให้ไมโครคอนโทรลเลอร์ทำงานในขณะเวลาปกติ แต่ถ้าต้องการรีเซ็ตทำได้โดยการกด สวิตช์ที่ต่ออยู่ระหว่างขา MCLR และกราวด์ เมื่อกดสวิตช์จะเป็นการเชื่อมต่อขา MCLR เข้ากับ กราวด์โดยตรงทำให้ขานี้มีลอจิกเป็น ‘0’ เป็นการรีเซ็ตไมโครคอนโทรลเลอร์ สำหรับขาเรีเซตนี้ สามารถจะกำหนดได้ด้วยโปรแกรมให้สามารถรีเซ็ตได้ด้วยการโปรแกรมโดยกำหนดได้ที่รีจิสเตอร์ คอนฟิก 3 ไบต์สูง หรือ CONFIG3H

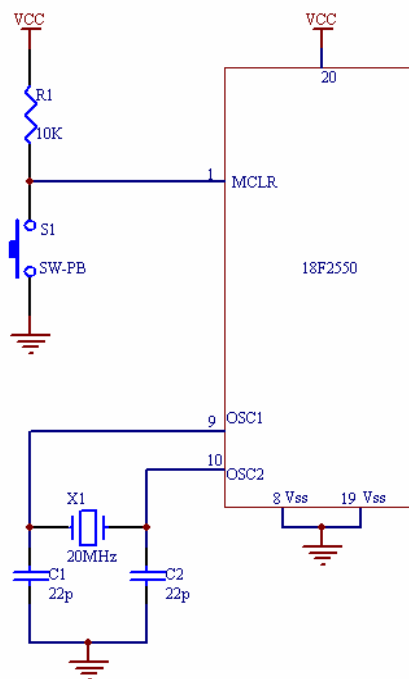
R/P-1	U-0	U-0	U-0	U-0	R/P-0	R/P-1	R/P-1
MCLRE	—	—	—	—	LPT1OSC	PBADEN	CCP2MX
bit 7							bit 0

รูปที่ 3.4

รีจิสเตอร์คอนฟิก 3 ไบต์สูง

จากรูปที่ 3.4 บิตที่ 7 ของรีจิสเตอร์หรือบิต MCLRE เป็นตัวที่จะกำหนด สถานะของขาเรีเซต โดยถ้ากำหนดให้บิตนี้มีลอจิกเป็น 0 จะกำหนดให้ขา MCLR ไม่ทำงานแต่จะ ทำให้ขานี้กลายเป็นขาอินพุตเอาต์พุตของ RE3 แทนหรือทำหน้าที่เป็นพอร์ตอินพุตและเอาต์พุต แทนที่จะเป็นขาเรีเซต แต่ถ้ากำหนดให้บิตนี้มีลอจิกเป็น 1 จะกำหนดให้ขานี้เป็นขาเรีเซตซึ่งก็คือทำ ให้ขา MCLR ทำงานโดยจะปิดไม่ให้กลายเป็นขาอินพุตและเอาต์พุตของ RE3 ด้วยลักษณะนี้จะ ทำให้สามารถรีเซ็ตไมโครคอนโทรลเลอร์ด้วยการทำให้ขา MCLR เป็นลอจิก 0 ได้ตลอดเวลาที่ ต้องการดังนั้นในการออกแบบจึงกำหนดให้ขานี้เป็นลอจิก 1 ด้วยเหตุผลดังที่ได้กล่าวไว้

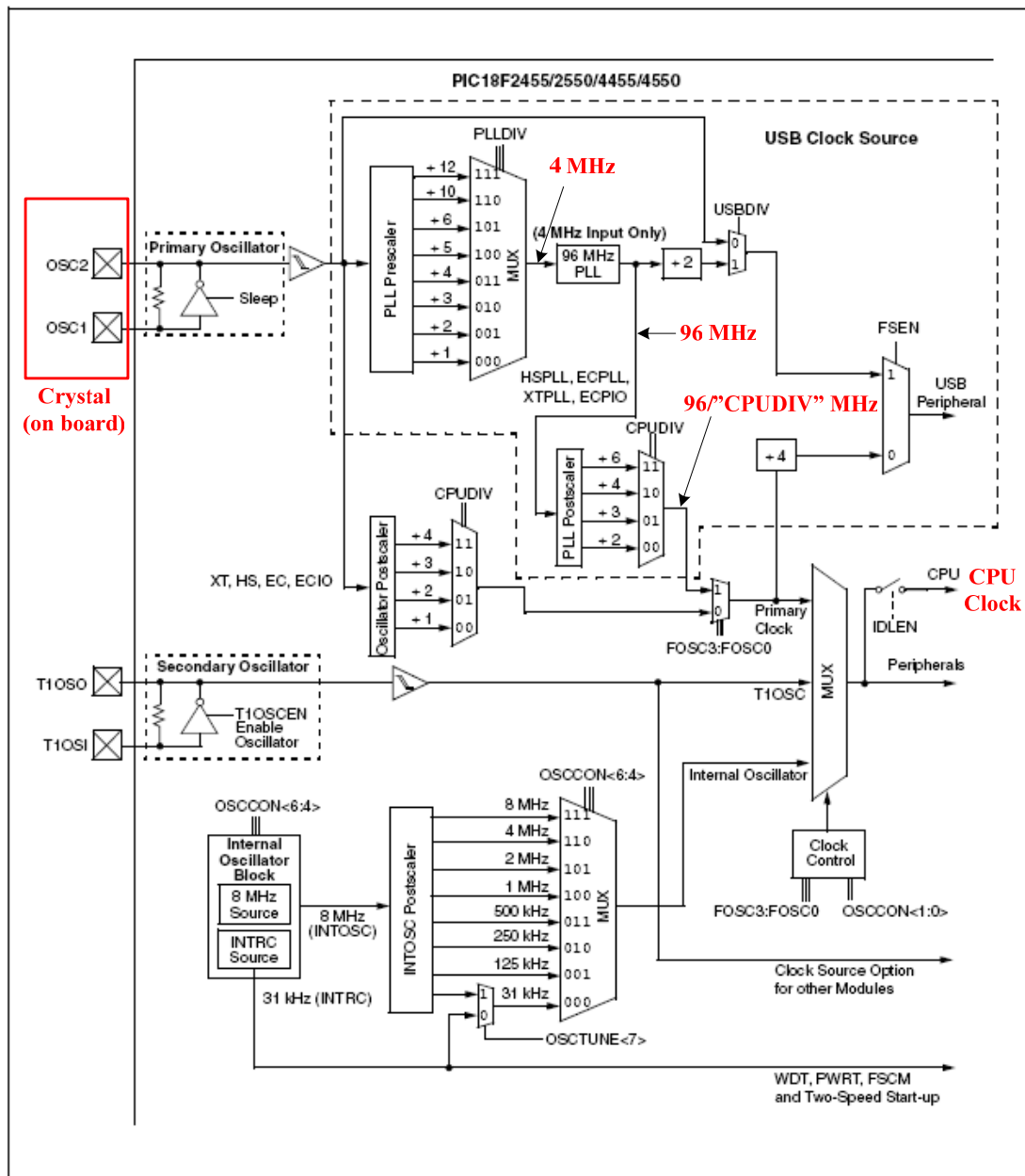
ส่วนวงจรอีกส่วนที่สำคัญคือส่วนของวงจรกำเนิดสัญญาณนาฬิกาเพื่อกำหนดจังหวะในการประมวลผลของไมโครคอนโทรลเลอร์สำหรับเบอร์นี้จะทำได้สองทางคือสามารถใช้แหล่งของสัญญาณนาฬิกาจากภายในของตัวไมโครคอนโทรลเลอร์ได้ และอีกทางคือการต่อสัญญาณนาฬิกาจากภายนอกป้อนเข้าสู่ตัวไอซี ในการทดลองนี้ใช้การป้อนสัญญาณนาฬิกาจากภายนอกโดยการต่ออุปกรณ์เพิ่มอีกเพียงเล็กน้อยโดยใช้คริสตัลความถี่ 20 เมกะเฮิร์ตซ์ ต่อเข้าที่ขา OSC1 ขาที่ 9 และ OSC2 ขาที่ 10 ตามลำดับจากนั้นต่อกับตัวเก็บประจุค่า 22 พิโคฟารัดลงกราวด์ทั้งสองขาดังรูปที่ 3.5



รูปที่ 3.5

แสดงวงจรส่วนรีเซ็ตและวงจรมกำเนิดสัญญาณนาฬิกา

การกำหนดสัญญาณนาฬิกาของไมโครคอนโทรลเลอร์สามารถกำหนดได้จากการตั้งค่าหรือการเซตค่าให้กับรีจิสเตอร์คอนฟิกที่เกี่ยวข้องกับสัญญาณนาฬิกาแสดงดังรูป 3.6



รูปที่ 3.6

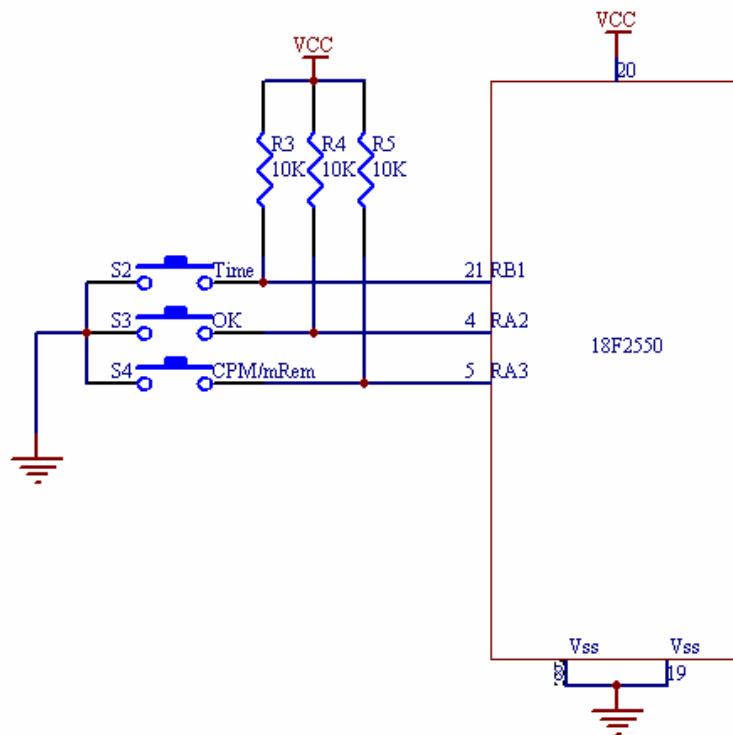
การกำหนดความถี่ของสัญญาณนาฬิกาของไมโครคอนโทรลเลอร์

การออกแบบในงานวิจัยนี้ใช้แหล่งกำเนิดสัญญาณนาฬิกาจากภายนอกโดยใช้คริสตัลความถี่ 20 เมกะเฮิร์ตซ์ จากนั้นจะใช้การหารความถี่โดยใช้ PLL Prescaler โดยต้องทำให้ความถี่มีค่า 4 เมกะเฮิร์ตซ์ จึงต้องหารด้วย 5 ดังนั้นในรีจิสเตอร์ CONFIG1L ต้องทำการตั้งค่าที่บิตของ PLLDIV2 PLLDIV1 และ PLLDIV0 ซึ่งอยู่ที่บิต 2 1 และ 0 ในรีจิสเตอร์ตามลำดับ โดยต้อง

กำหนดให้มีค่าเป็นลอจิก 1 0 0 ในบิตที่กำหนด จะได้ความถี่ 4 เมกะเฮิร์ตซ์ ออกมาจาก PLL Prescaler เข้าสู่ส่วนกำเนิดสัญญาณ 96 เมกะเฮิร์ตซ์ จากนั้นความถี่ที่ได้นี้จะเข้าสู่ส่วน PLL Postscaler ซึ่งสามารถจะเลือกตั้งความถี่จาก 96 เมกะเฮิร์ตซ์ ให้เป็นความถี่ค่าอื่นได้ด้วยการหาร โดยในงานวิจัยได้ใช้การหารด้วย 2 ซึ่งเลือกได้ที่บิต CPUDIV1 และ CPUDIV0 อยู่ที่บิต 4 และ 3 ในรีจิสเตอร์ CONFIG1L กำหนดให้มีค่าลอจิกเป็น 0 และ 0 ตามลำดับ และต้องกำหนดค่าในบิต FOSC3 FOSC2 FOSC1 และ FOSC0 ในรีจิสเตอร์ CONFIG1H เป็น 1 1 1 x ตามลำดับ เพื่อเป็นการเลือกชนิดของคริสตัลและเลือกใช้ PLL จากการตั้งค่าดังได้กล่าวมาทั้งหมดจะได้ความถี่ เป็น 48 เมกะเฮิร์ตซ์ ป้อนเข้าสู่ระบบสัญญาณนาฬิกาของไมโครคอนโทรลเลอร์

3.1.2.1 สวิตช์ใช้งาน

ในส่วนของการรับข้อมูลเพื่อป้อนข้อมูลที่ต้องการหรือเป็นการ กำหนดค่าให้กับระบบได้ใช้สวิตช์แบบกดติดปล่อยดับ โดยขาด้านหนึ่งต่อเข้าสู่พอร์ตของ ไมโครคอนโทรลเลอร์โดยตรงและใช้ความต้านทานค่า 10 กิโลโอห์ม ต่อกับไฟเลี้ยงของวงจรและ ต่อเข้าพอร์ตของไมโครคอนโทรลเลอร์เพื่อยกระดับแรงดันของขาสัญญาณให้อยู่ที่ระดับแรงดัน ลอจิก '1' ตลอดเวลาเพื่อป้องกันไม่ให้พอร์ตของไมโครคอนโทรลเลอร์มีสัญญาณไฟฟ้าตกลงมา เองในขณะที่ไม่ได้กดสวิตช์ดังนั้นที่พอร์ตนี้จะมีระดับแรงดันดังกล่าวจนกว่าจะมีการกดสวิตช์ เมื่อ มีการกดสวิตช์จะทำให้พอร์ตของไมโครคอนโทรลเลอร์ต้องกราวด์โดยตรงทำให้สัญญาณไฟฟ้า เปลี่ยนเป็นลอจิก '0' เป็นการบอกว่าสวิตช์ได้ถูกกดและสามารถตรวจสอบได้ด้วยโปรแกรม วงจร แสดงได้ดังรูปที่ 3.7



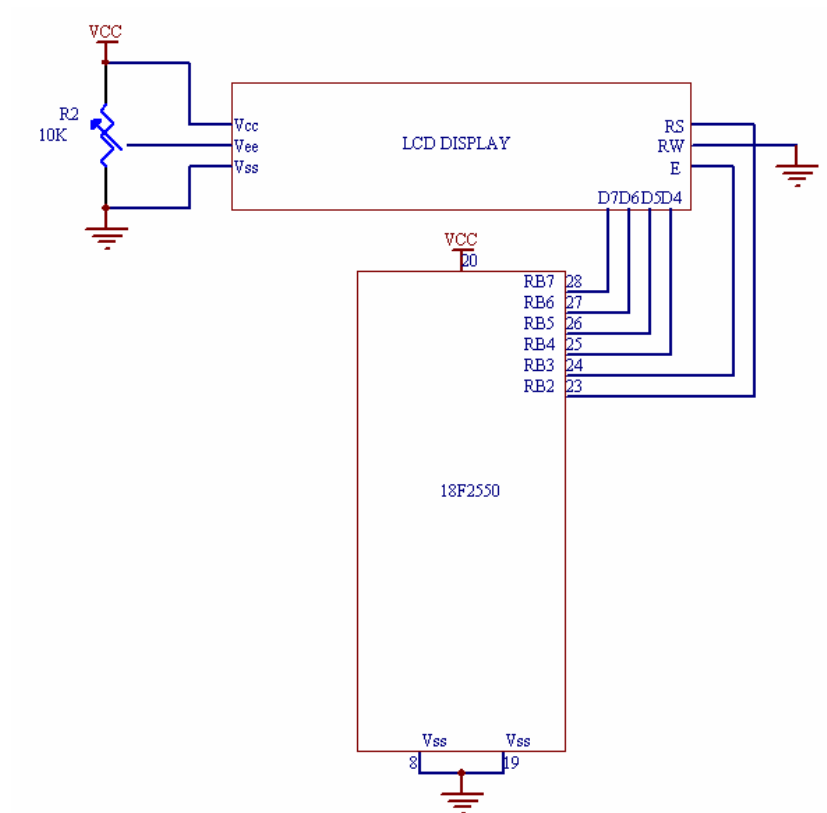
รูปที่ 3.7

แสดงการต่อสวิทช์

สวิทช์ประกอบด้วย 3 สวิทช์ เริ่มด้วยสวิทช์สำหรับเลือกหน่วยในการวัดรังสีโดยสามารถเลือกได้ระหว่าง อัตรานับต่อนาทีกับมิลลิเมตรต่อชั่วโมง สวิทช์ที่สองเป็นสวิทช์เมนูพร้อมกับการทำหน้าที่ตกกลงในสวิทช์เดียวกัน ส่วนสวิทช์สุดท้ายเป็นสวิทช์ที่ใช้สำหรับเปลี่ยนช่วงเวลานับว่าจะให้วัดอัตรานับต่อ 1 นาที อัตรานับต่อ 5 นาที และวัดแบบไม่จำกัดเวลา

3.1.2.2 ส่วนแสดงผล

ในการแสดงผลใช้อุปกรณ์แสดงผลที่เป็นหน่วยแสดงผลตัวอักษรแบบ ผลึกเหลวขนาด 8 ตัวอักษร 2 บรรทัด ในการต่อใช้การเชื่อมต่อแบบ 4 บิต เพื่อเป็นการลดพอร์ตที่จะต้องใช้นั้นน้อยลงไปได้ระดับหนึ่งดังรูปที่ 3.8 โดยในการต่อมีขาสัญญาณดังนี้ ขา R/W ต่อลงกราวด์เพื่อเป็นการเขียนข้อมูลสู่หน่วยแสดงผล ขา RS ต่อกับขา RB2 เพื่อควบคุมรีจิสเตอร์ในการเขียนข้อมูลหน่วยแสดงผล ขา E ต่อกับ RB3 เพื่อเป็นสัญญาณควบคุมการแสดงผล และขา RB4 ถึง RB7 ต่อกับ D4, D5, D6 และ D7 ตามลำดับ เพื่อเป็นการส่งผ่านข้อมูลระหว่างไมโครคอนโทรลเลอร์และหน่วยแสดงผล ส่วนขาสัญญาณข้อมูล D0 ถึง D3 ต่อลงกับกราวด์ของวงจรป้องกันสัญญาณรบกวนที่จะทำให้การส่งข้อมูลผิดพลาดได้

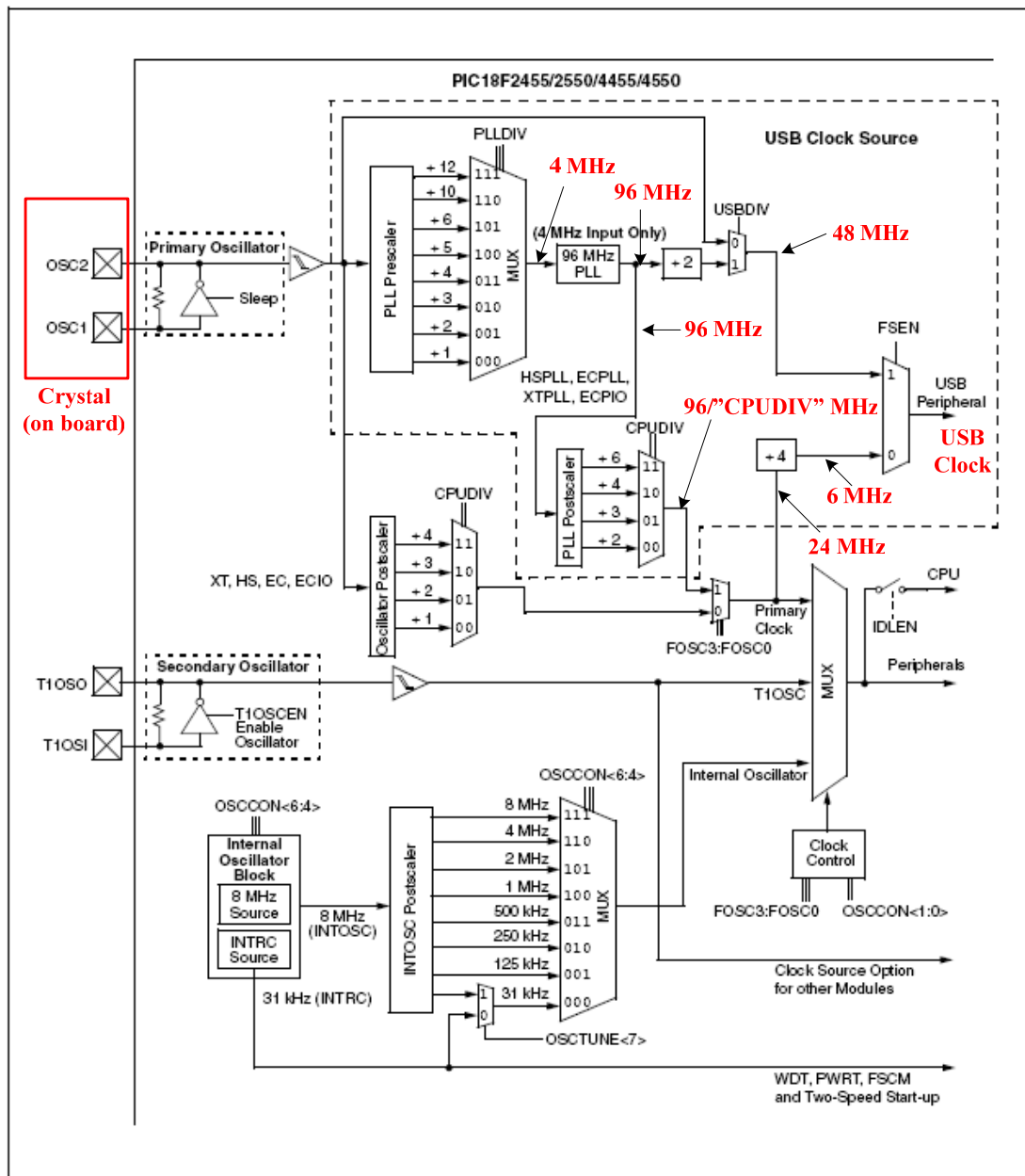


รูปที่ 3.8

แสดงส่วนของวงจรแสดงผลโดยหน่วยแสดงผลแบบผลึกเหลว

3.1.3 การเชื่อมต่อแบบ USB

สำหรับคุณสมบัติของ PIC18F2550 นั้นผู้ผลิตได้ออกแบบมาเพื่อให้สามารถเชื่อมต่อกับพอร์ต USB ของเครื่องคอมพิวเตอร์ได้ โดยมีการกำหนดขาสัญญาณไว้ โดยให้พอร์ต RC4 เป็น D- และ RC5 เป็น D+ การกำหนดความเร็วของการรับส่งข้อมูลสามารถทำได้ด้วยการกำหนดค่าในส่วนของรีจิสเตอร์คอนฟิกของไมโครคอนโทรลเลอร์ PIC18F2550



รูปที่ 3.9

ลักษณะการตั้งค่าความถี่สัญญาณนาฬิกาส่วน USB

ในส่วนของความถี่สัญญาณนาฬิกาของ USB จะมีอยู่ 2 โหมด คือ โหมดการทำงานแบบความเร็วต่ำ (Low Speed) และในโหมดการทำงานที่ความเร็วสูง (High Speed) โดยสามารถเลือกการใช้งานได้ดังนี้

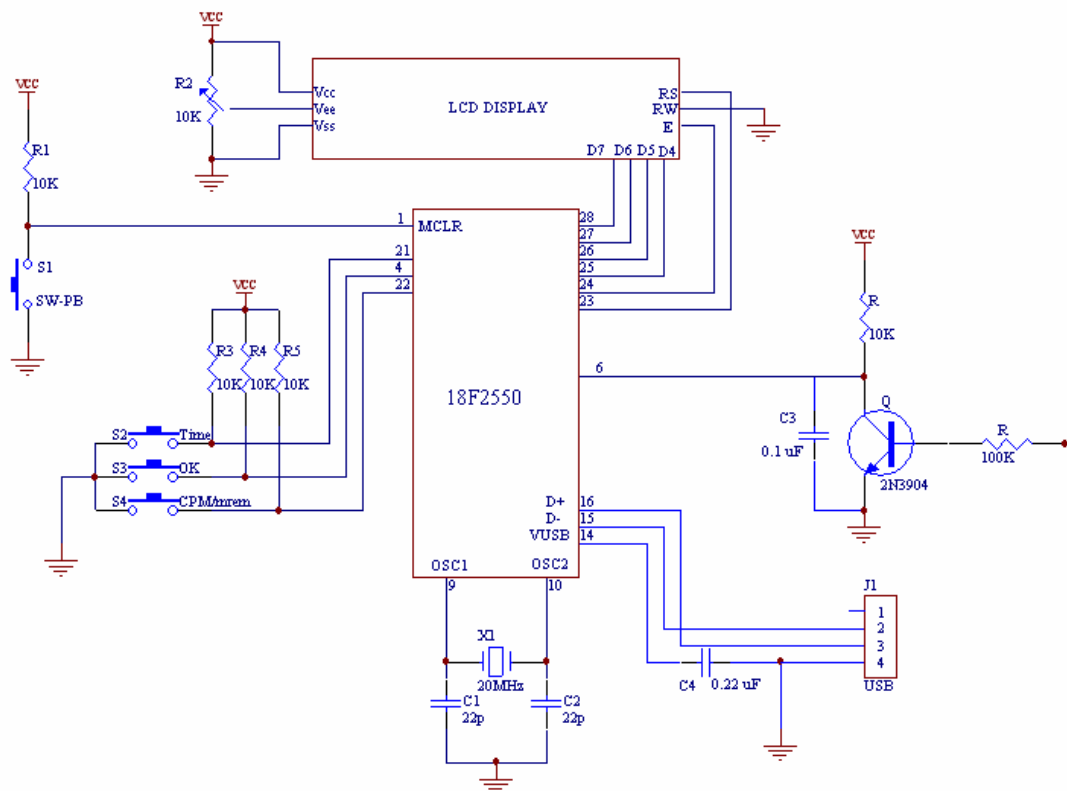
3.1.3.1. โหมดการทำงานแบบความเร็วต่ำ (Low Speed)

สำหรับการทำงานที่ความเร็วต่ำเป็นการตั้งค่าความถี่สัญญาณนาฬิกาที่ 6 เมกะเฮิร์ตซ์ จะเริ่มด้วยคริสตอลที่ความถี่ 20 เมกะเฮิร์ตซ์ จะถูกหารด้วยค่า PALL Prescaler เพื่อให้ได้ค่าความถี่ 4 เมกะเฮิร์ตซ์ จึงจำเป็นต้องหารด้วย 5 ด้วยการกำหนดในค่าของ PALLDIV เมื่อได้ความถี่ 4 เมกะเฮิร์ตซ์ จะถูกป้อนเข้าสู่ส่วนสร้างความถี่ 96 เมกะเฮิร์ตซ์ ความถี่นี้จะถูกหารด้วยค่าของ CPUDIV โดยเลือกที่ CPUDIV ที่มีค่าเท่ากับ 4 จะได้ความถี่ออกมา 24 เมกะเฮิร์ตซ์ และจะถูกหารด้วย 4 ก่อนที่จะออกสู่ส่วนของ USB ด้วยการเลือกที่ FSEN ให้มีค่าเป็น 0 จะได้ความถี่ 6 เมกะเฮิร์ตซ์ตามต้องการ ดังแสดงในรูปที่ 3.9

3.1.3.2. โหมดการทำงานแบบความเร็วสูง (High Speed)

ในโหมดความเร็วสูงสัญญาณนาฬิกาจากความถี่คริสตอล 20 เมกะเฮิร์ตซ์ถูกหารด้วย PALL Prescaler ได้ความถี่ 4 เมกะเฮิร์ตซ์ ป้อนเข้าสู่ส่วนสร้างความถี่ 96 เมกะเฮิร์ตซ์ จากนั้นจะถูกหารด้วย 2 ได้ความถี่ 48 เมกะเฮิร์ตซ์ป้อนเข้าสู่ส่วน USBDIV เมื่อให้เป็น 1 ก็จะได้ความถี่ 48 เมกะเฮิร์ตซ์ และเมื่อเลือกให้ FSEN เป็น 1 ความถี่นี้จะเข้าสู่ส่วนของ USB เป็นความถี่ที่ใช้สำหรับโหมดการทำงานแบบความเร็วสูง ดังแสดงในรูปที่ 3.9

สำหรับในงานวิจัยนี้เลือกโหมดการทำงานของการเชื่อมต่อแบบ USB เป็นแบบความเร็วสูงเพื่อทำให้การรับส่งข้อมูลมีความรวดเร็ว และวงจรรวมทั้งหมดเมื่อนำแต่ละส่วนมารวมกันได้ดังรูปที่ 3.10



รูปที่ 3.10

วงจรรวมของมิเตอร์วัดรังสีโดยใช้ไมโครคอนโทรลเลอร์เป็นฐาน

3.2 โปรแกรมควบคุมระบบ

ส่วนสำคัญอีกส่วนที่ต้องมีประกอบกับวงจรส่วนไมโครคอนโทรลเลอร์ก็คือโปรแกรมที่เขียนขึ้นเพื่อให้ไมโครคอนโทรลเลอร์ทำงานได้ดังที่ต้องการ สำหรับโปรแกรมประกอบด้วยสองส่วนหลักๆคือโปรแกรมส่วนของไมโครคอนโทรลเลอร์ที่เป็นส่วนของมิเตอร์วัดรังสีโดยใช้ไมโครคอนโทรลเลอร์เป็นฐานและส่วนโปรแกรมทางด้านของคอมพิวเตอร์ซึ่งจะเป็นส่วนใช้ในการเชื่อมต่อกับตัวเครื่องวัดรังสีโดยผ่านทางโปรแกรกดังกล่าว

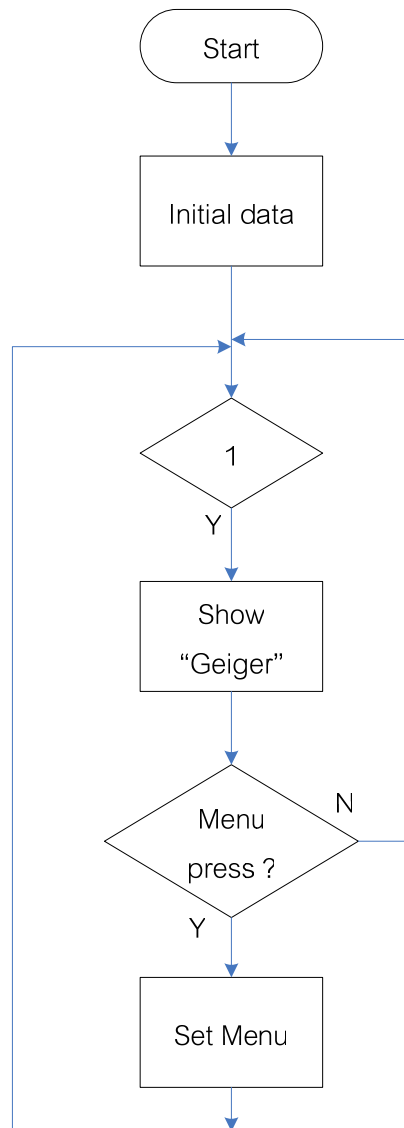
3.2.1 โปรแกรมไมโครคอนโทรลเลอร์ PIC

3.2.1.1 โปรแกรมแสดงการติดต่อกับผู้ใช้งาน

ทางด้านของโปรแกรมไมโครคอนโทรลเลอร์ เป็นส่วนสำคัญที่ใช้ควบคุมการทำงานของเครื่องมือวัดรังสีโดยรวมทั้งหมดซึ่งจะทำให้สามารถวัดรังสีได้ด้วยตัวของเครื่องมือเองโดยไม่ต้องมีอุปกรณ์เพิ่มเติมแต่อย่างใด ในงานวิจัยนี้ได้ใช้โปรแกรมแปรภาษาซี

ไมโครคอนโทรลเลอร์ของบริษัท MikroElektronika ชื่อโปรแกรม mikroC for PIC version 8.1 เป็นโปรแกรมในการคอมไพล์ภาษาซีของไมโครคอนโทรลเลอร์ PIC เป็นโปรแกรมภาษาเครื่องเพื่อทำการเขียนเข้าสู่หน่วยความจำโปรแกรมของไมโครคอนโทรลเลอร์ต่อไป ส่วนสำคัญคือเมื่อเริ่มสร้างโปรแกรมจำเป็นจะต้องทำการตั้งค่าคอนฟิกต่างๆของไมโครคอนโทรลเลอร์ PIC เสียก่อนโดยทำการเซตค่าในส่วนที่เรียกว่า Edit Project ในการตั้งค่าเป็นการกำหนดความถี่ของสัญญาณนาฬิกาที่ได้กล่าวมาแล้ว คือเซตที่บล็อกล็อก PALLDIV_5_1L เพื่อหารสัญญาณนาฬิกาจากคริสตัล 20 เมกะเฮิร์ตซ์ ให้เหลือ 4 เมกะเฮิร์ตซ์ เซตที่บล็อกล็อก CPUDIV_OSC1_PALL2_1L เพื่อทำการหารสัญญาณนาฬิกาจาก 96 เมกะเฮิร์ตซ์ ให้เป็น 48 เมกะเฮิร์ตซ์ เซตที่บล็อกล็อก FOSC_HSPALL_HS_1H เพื่อเป็นการเลือกลักษณะการกำเนิดสัญญาณนาฬิกาจากตัวกำเนิด 96 เมกะเฮิร์ตซ์ จากการเซตค่าทั้งหมดก็จะได้สัญญาณนาฬิกาเป็น 48 เมกะเฮิร์ตซ์ ป้อนให้กับหน่วยประมวลผลของไมโครคอนโทรลเลอร์

ในการออกแบบโปรแกรมควบคุมการทำงานของมิเตอร์วัดรังสีจะกล่าวเป็นส่วนๆไป โดยเริ่มจากในส่วนแรกของโปรแกรมก่อนเป็นอันดับแรกแสดงได้ดังรูป 3.11 เป็นส่วนฟลัวร์ชาร์ตการทำงานของโปรแกรมส่วนแสดงหน้าแรก เป็นส่วนที่เมื่อเปิดสวิทช์ขึ้นมาป้อนกระแสไฟฟ้าเข้าสู่วงจรไมโครคอนโทรลเลอร์โปรแกรมจะแสดงหน้าจอขึ้นมาทำงานเป็นอันดับแรกอธิบายได้ดังนี้คือโปรแกรมจะแสดงคำว่า Geiger ที่หน้าจอหน่วยแสดงผลและจะวนรอกการกดสวิทช์เมนูถ้าไม่มีการกดสวิทช์ก็จะแสดงหน้านี้นี้ตลอดไปแต่เมื่อใดก็ตามที่มีการกดสวิทช์เมนูก็จะเข้าสู่โปรแกรมส่วนการเลือกเมนูต่อไป

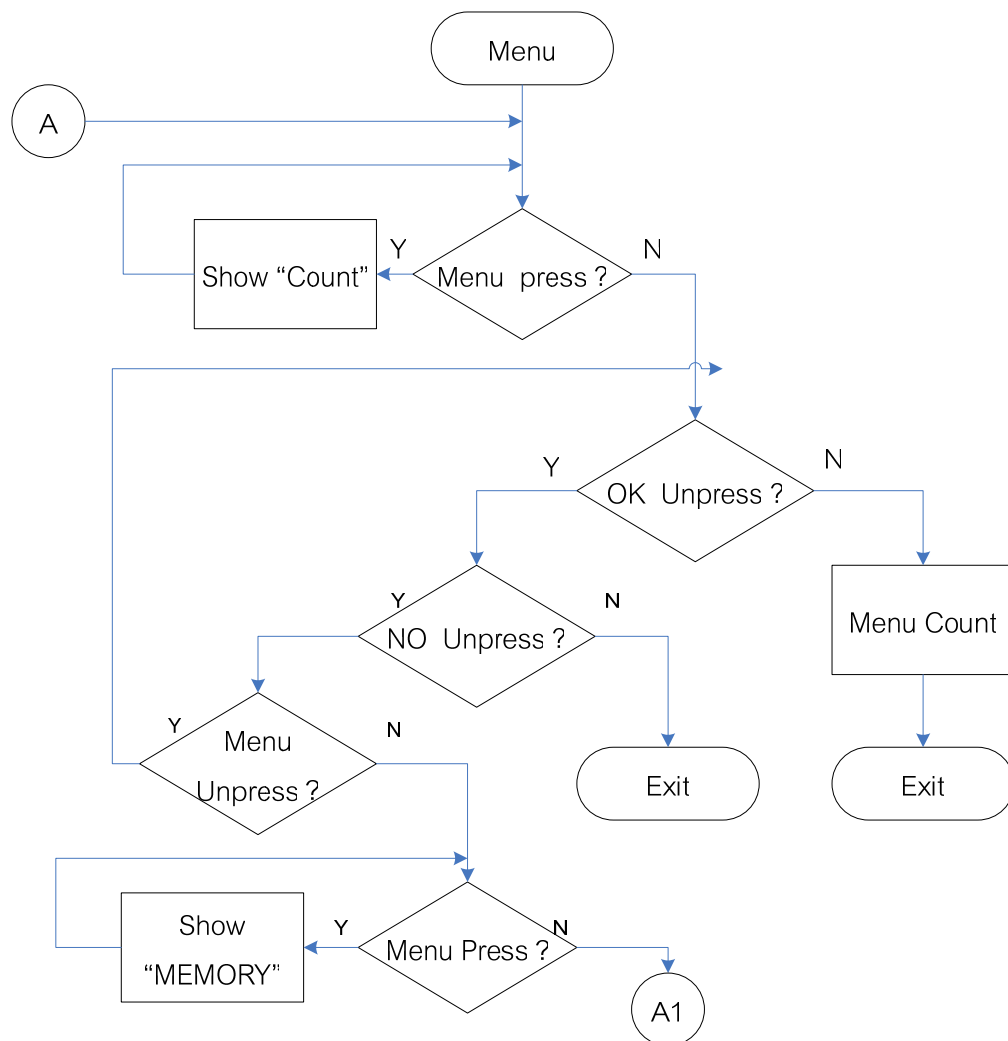


รูปที่ 3.11

โฟลว์ชาร์ตแสดงหน้าเริ่มต้นของมิเตอร์วัดรังสี

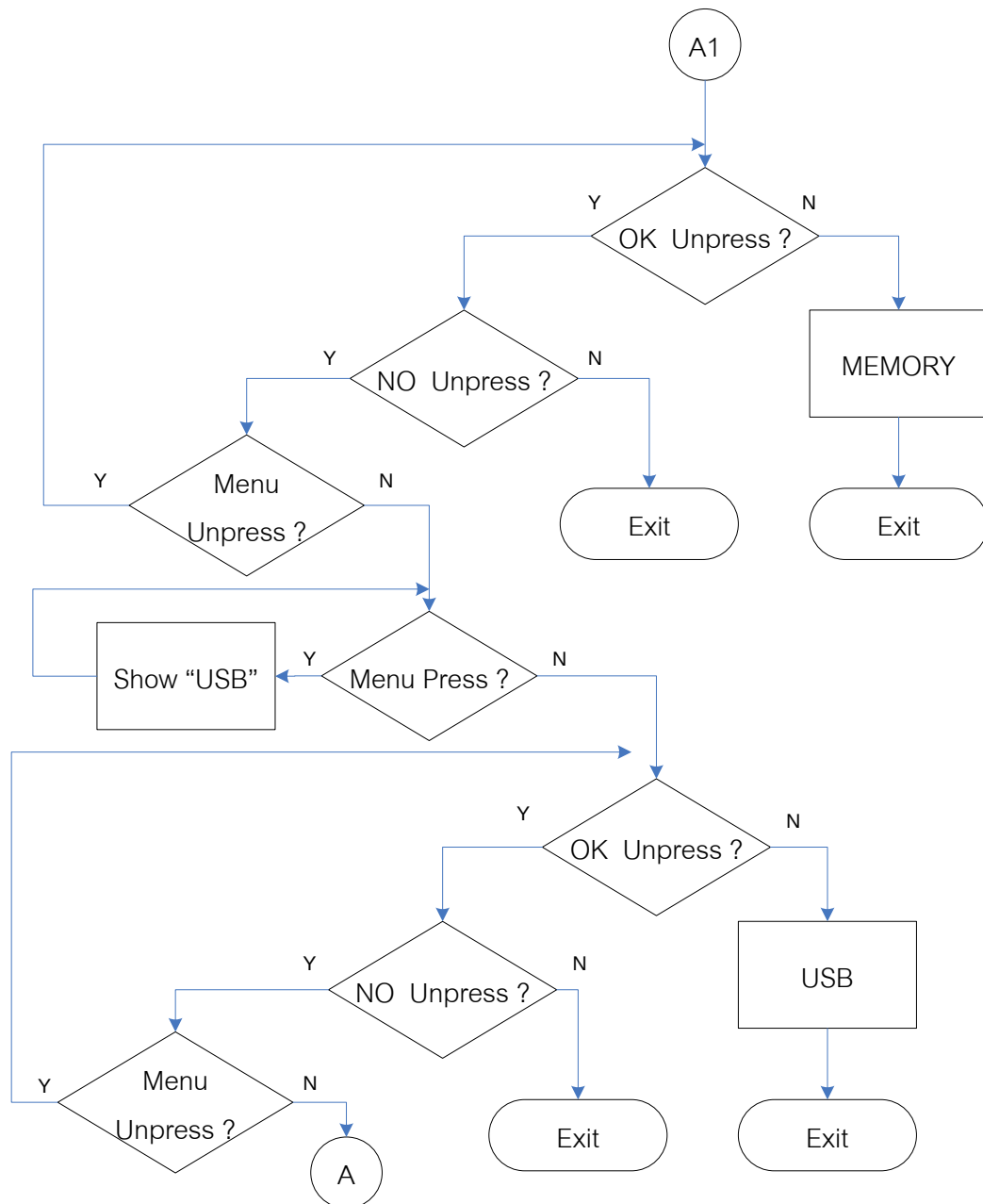
โฟลว์ชาร์ตในรูปที่ 3.12 จะเป็นการเลือกเมนูในการทำงานโดยจะมีการตรวจสอบสวิตช์เมนูหรือสวิตช์ตกลง ว่าถูกกดค้างอยู่หรือไม่ถ้าใช่ก็จะแสดงหน้าจอคำว่า COUNT ถ้าปล่อยสวิตช์เมนูจะไปสู่ขั้นตอนการตรวจสอบสวิตช์ที่ตรงกับคำว่า OK ที่หน้าจอหน่วยแสดงผล เป็นการให้ตัดสินใจว่าจะตอบตกลงเลือกเมนู Count หรือไม่ ถ้าไม่ถูกกดจะไปตรวจสอบสวิตช์ต่อไปคือตรวจสอบสวิตช์ที่ตรงกับคำว่า NO ที่แสดงที่จอแสดงผล ถ้าไม่มีการกดสวิตช์นี้ก็จะตรวจสอบสวิตช์ต่อไปคือสวิตช์เมนูถ้าไม่กดสวิตช์ใดๆโปรแกรมจะทำการวนตรวจสอบสวิตช์อยู่

ตลอดเวลาจนกว่าจะมีการกดสวิตช์ใดสวิตช์หนึ่ง โดยถ้ากดสวิตช์ตรงกับคำว่า OK ก็เข้าสู่ส่วนของโปรแกรมเมนู Count ต่อไป แต่ถ้ากดสวิตช์ที่ตรงกับคำว่า NO ก็จะเป็นการออกจากส่วนของเมนูสู่หน้าจอเริ่มต้นของโปรแกรม และถ้ากดสวิตช์เมนูโปรแกรมจะเปลี่ยนหน้าจอเป็นเมนู MEMORY และเมื่อปล่อยสวิตช์ โปรแกรมก็จะทำการตรวจสอบสวิตช์ OK สวิตช์ NO และสวิตช์เมนู โดยถ้ากดสวิตช์เลือก OK จะไปสู่ส่วนของโปรแกรม MEMORY ถ้าเลือกสวิตช์ NO จะออกจากการเลือกเมนูไปสู่หน้าจอเริ่มต้นของโปรแกรม แต่ถ้ากดสวิตช์เมนูจะเปลี่ยนเมนูไปสู่ USB จากนั้นจะตรวจสอบสวิตช์ที่ตรงกับคำว่า OK ถ้าถูกกดก็จะเข้าสู่ส่วนของโปรแกรม USB ถ้าไม่ถูกกดจะไปตรวจสอบสวิตช์ที่ตรงกับคำว่า NO ถ้าที่สวิตช์นี้ถูกกดจะออกจากเมนูไปสู่หน้าจอหลักของโปรแกรม ถ้าสวิตช์นี้ไม่ถูกกดจะไปตรวจสอบสวิตช์เมนูว่าถูกกดหรือไม่ถ้ากดโปรแกรมจะวนกลับไปแสดงเมนู Count อีกครั้ง



รูปที่ 3.12

โฟลว์ชาร์ตการทำงานของโปรแกรมส่วนแสดงเมนู

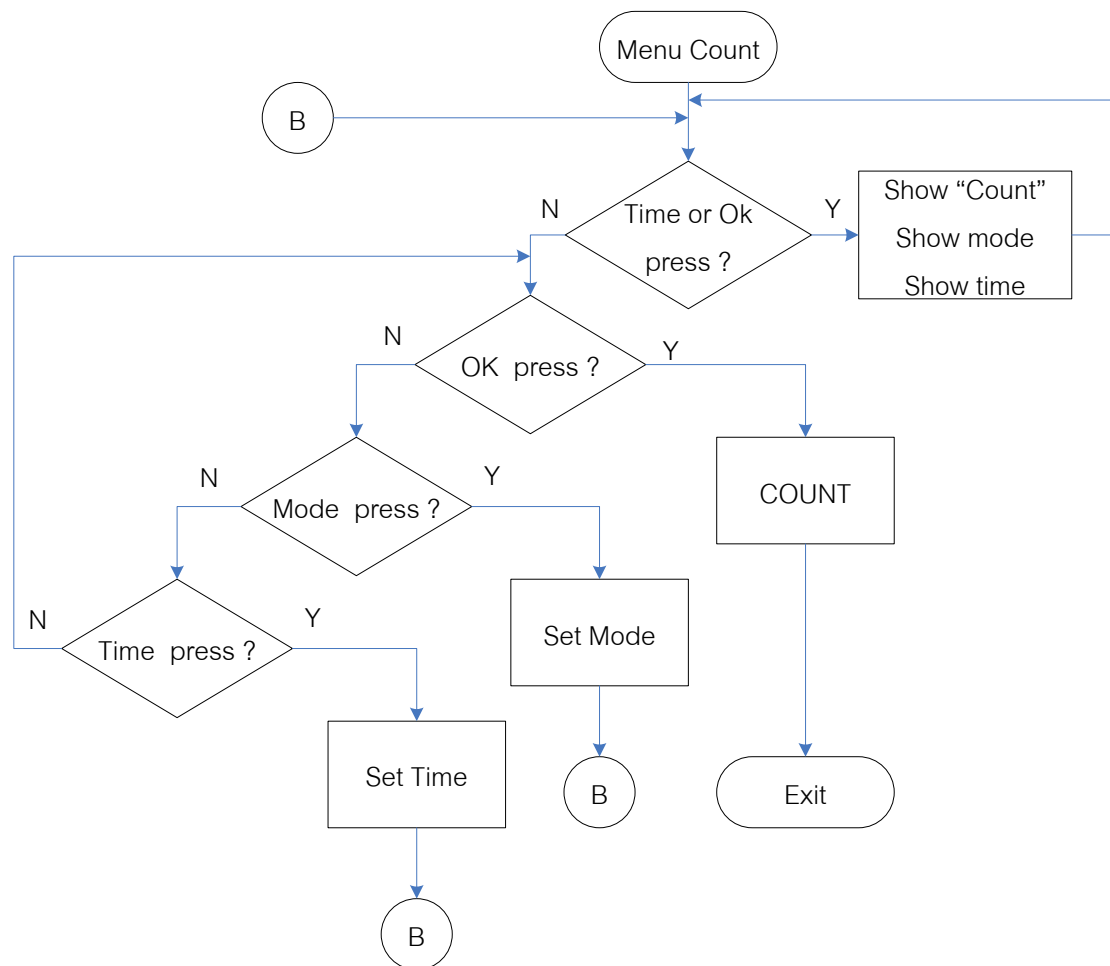


รูปที่ 3.12 (ต่อ)

ฟลิวชาร์ตการทำงานของโปรแกรมส่วนแสดงเมนู

จากโปรแกรมส่วนที่ผ่านมาเมื่อมีการกดสวิทช์ส่วนของเมนูการนับหรือเมนู Count จะเข้าสู่ส่วนหน้าหลักของการเลือกการนับดังรูปที่ 3.13 โดยเริ่มต้นเข้ามาจะเป็นการตรวจสอบการกดสวิทช์ว่า สวิทช์ Time หรือสวิทช์ OK ถูกกดหรือไม่ ถ้าใช่จะแสดงคำว่า Count และแสดงโหมดว่าเป็นโหมดอะไรและแสดงเวลาที่ใช้สำหรับการนับ แต่ถ้าปล่อยสวิทช์โปรแกรมจะมานตรวจสอบสวิทช์เมนู สวิทช์โหมดและสวิทช์เวลาถ้าไม่มีสวิทช์ใดถูกกดก็จะวนตรวจสอบไปตลอด แต่ถ้ามี

สวิทช์เมนูถูกกดโปรแกรมจะไปสู่โปรแกรม Count ต่อไป ถ้าสวิทช์โหมดถูกกดจะไปสู่ส่วนของการเลือกโหมดเช่นเดียวกันกับถ้าสวิทช์เวลาถูกกดก็จะไปสู่ส่วนของการเลือกเวลาในการนับ

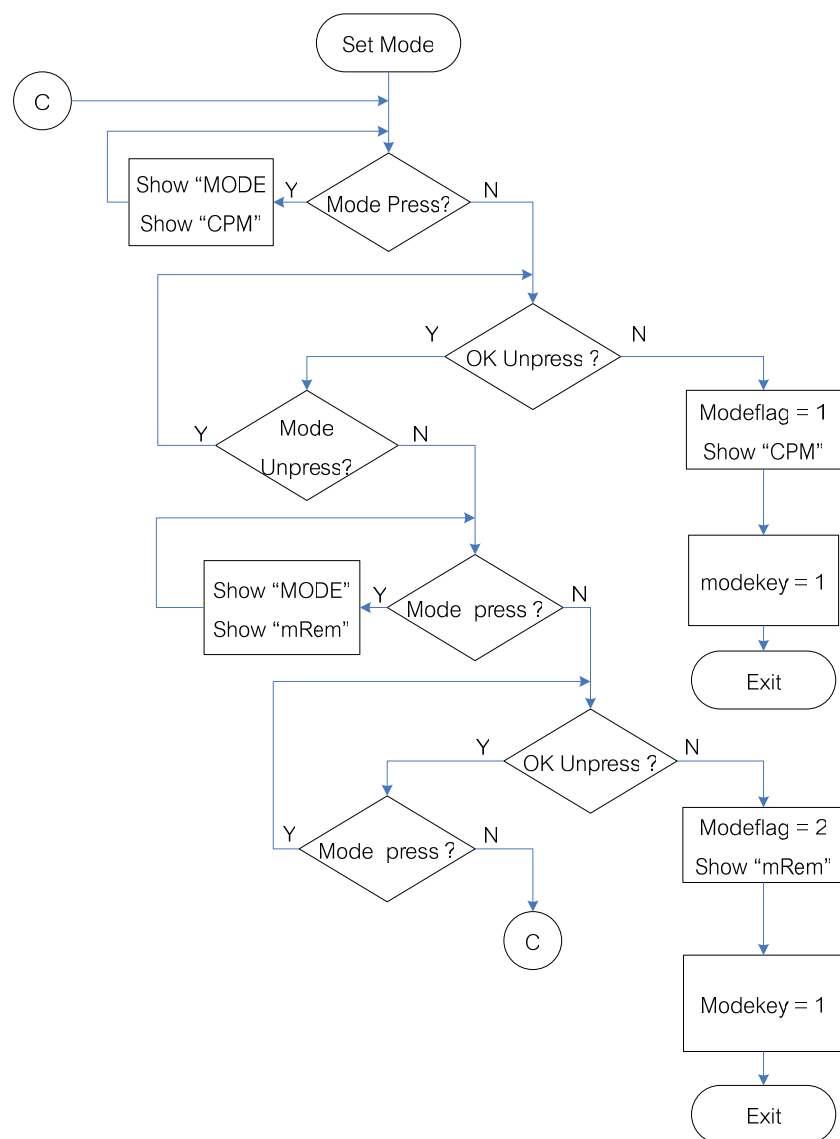


รูปที่ 3.13

โฟลว์ชาร์ตแสดงการทำงานของโปรแกรมส่วนของเมนูการนับ

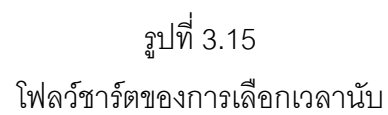
ส่วนของโปรแกรมเลือกโหมดดังรูปที่ 3.14 เป็นส่วนถัดมาที่เกิดจากการกดสวิทช์โหมดในส่วนของเมนู Count เมื่อเข้ามาจุดเริ่มต้นของโปรแกรมจะทำการตรวจสอบการกดสวิทช์ค้างของสวิทช์โหมดถ้าใช่จะแสดงผลเป็นข้อความ MODE และ CPM แต่เมื่อปล่อยสวิทช์โปรแกรมจะไปสู่การตรวจสอบสวิทช์ OK ถ้าไม่ถูกกดจะไปตรวจสอบสวิทช์โหมดถ้าไม่ถูกกดอีกก็จะวนตรวจสอบทั้งสองสวิทช์ไปตลอดแต่ถ้าสวิทช์ OK ถูกกดจะทำการใส่ค่าให้กับตัวแปรที่ใช้สำหรับตรวจสอบโหมดให้เป็น 1 และให้ตัวแปรที่ใช้ตรวจสอบโหมดเป็น CPM และตั้งค่าให้พอร์ท

สวิทช์เป็นลอจิก 1 ก่อนออกจากโปรแกรมไป แต่ถ้าเป็นการกดสวิทช์ใหม่จะเป็นการเปลี่ยนโหมดจาก CPM ไปสู่ mrem/hr และถ้ากดสวิทช์ค้างก็จะวนแสดงค่าไปจนกว่าจะปล่อยสวิทช์ ทันทีที่ปล่อยสวิทช์โปรแกรมจะไปสู่กระบวนการในการตรวจสอบสวิทช์ OK และสวิทช์ใหม่ถ้าไม่มีการกดก็จะวนตรวจสอบไปตลอดเวลาแต่ถ้ากดสวิทช์ OK ก็จะเป็นการให้ค่ากับตัวแปรตรวจสอบโหมดให้เป็น 2 และกำหนดตัวแปรที่ใช้ตรวจสอบโหมดเป็น mrem/hr และเช่นกันก่อนออกจากโปรแกรมได้ทำการเซตค่าของสวิทช์ใหม่ให้เป็น 1 ด้วย และสุดท้ายแล้วถ้ากดสวิทช์ใหม่โปรแกรมก็จะวนกลับไป CPM ดังเดิม



รูปที่ 3.14

แสดงฟลอร์ชาร์ตของโปรแกรมส่วนของการเลือกโหมดในการแสดงค่าการนับ



ไฟล์ชาร์ตของการเลือกเวลานับ

รูป 3.15 เป็นโปรแกรมส่วนของการตั้งเวลาให้กับการนับเริ่มจากส่วนหัวของโปรแกรมถ้าสวิทช์เวลาถูกกดค้างไว้ก็จะแสดง Time และ 1 M ที่หน้าจอแสดงผล เมื่อปล่อยสวิทช์จะไปทำการตรวจสอบสวิทช์ OK และสวิทช์ Time ถ้าไม่มีสวิทช์ใดถูกกดจะวนตรวจสอบไปตลอดเวลา แต่เมื่อสวิทช์ OK ถูกกดจะใส่ข้อมูลให้กับตัวแปรเวลาเพื่อบอกว่าเป็นการเลือกเวลา 1 นาทีในการนับและออกสู่โปรแกรมส่วนนี้ไปและถ้าสวิทช์เวลาถูกกดจะเปลี่ยนแสดงหน้าจอเป็น 5 M โดยถ้ายังกดค้างก็จะแสดงไปเรื่อยๆเมื่อปล่อยสวิทช์ก็จะไปสู่การตรวจสอบสวิทช์ OK และสวิทช์ เวลาถ้าไม่มีสวิทช์ใดถูกกดก็จะวนตรวจสอบไปตลอดเช่นกันแต่ถ้าสวิทช์ OK ถูกกดก็จะเป็นการใส่ข้อมูลให้ตัวแปรเวลาเป็น 5 M หรือเป็นการเลือกเวลา 5 นาที แต่ถ้าสวิทช์เวลาถูกกดก็จะไปแสดงเวลาเป็น inf หรือเป็นตัวย่อของการเลือกเวลาแบบไม่มีจำกัดเวลาถ้ากดสวิทช์ค้างไว้ก็จะแสดงหน้าจอตลอดเวลาแต่ถ้าปล่อยสวิทช์จะไปสู่ขั้นตอนการตรวจสอบสวิทช์ OK และสวิทช์เวลา เช่นเดิมถ้าไม่มีสวิทช์ใดถูกกดก็จะวนตรวจสอบไปจะกว่าจะมีสวิทช์ใดเกิดการเปลี่ยนแปลงโดยถ้าสวิทช์ OK ถูกกดจะเป็นการป้อนข้อมูลให้ตัวแปรเวลาเป็น inf หรือเป็นการบอกว่าเป็นการเลือกเวลาแบบไม่จำกัดและออกจากโปรแกรมแต่ถ้ายังกดสวิทช์เวลาอีกโปรแกรมจะวนกลับไปเพื่อแสดงเวลา 1 M อีกครั้ง

3.2.1.2 การนับสัญญาณไฟฟ้า

ส่วนสำคัญของมิเตอร์วัดรังสีคือการวัดสัญญาณไฟฟ้าที่มาจากหลอดไกเกอร์-มูลเลอร์เพื่อนำเข้าสู่ตัวไมโครคอนโทรลเลอร์ผ่านทางพอร์ตอินพุต เนื่องจากสัญญาณต้องมีขนาดไม่เกิน 5 โวลต์ดังนั้นสัญญาณที่มาจากหลอดไกเกอร์-มูลเลอร์จึงต้องผ่านทรานซิสเตอร์เพื่อทำให้สัญญาณลดระดับแรงดันลงมาประมาณ 5 โวลต์เพื่อจะได้เหมาะสมนำเข้าสู่พอร์ตอินพุตต่อไป สัญญาณที่เข้าไปนี้เพื่อความรวดเร็วและแม่นยำในการวัดจึงได้ใช้การโปรแกรมส่วนของไทเมอร์และเคาน์เตอร์โดยโปรแกรมให้ทำหน้าที่เป็นเคาน์เตอร์วัดการเปลี่ยนแปลงสัญญาณจากภายนอกตัวไมโครคอนโทรลเลอร์ผ่านทางขาสัญญาณโดยเลือกใช้ไทเมอร์ 0 ซึ่งจะต้องนำสัญญาณเข้ามาทางพอร์ตอินพุตขาที่ 6 หรือ T0CKI ของไมโครคอนโทรลเลอร์เมื่อมีการเปลี่ยนแปลงของสัญญาณโดยกำหนดให้พิจารณาที่ขาขึ้นของสัญญาณรีจิสเตอร์ที่อยู่ภายในมีขนาด 16 บิตก็จะนับขึ้นไปตามสัญญาณที่ป้อนเข้าสู่ขา T0CKI การนับจะเริ่มนับตั้งแต่ค่า 0000H ไปจนถึงค่า FFFFH เป็นจำนวน 65535 ครั้ง

ในการกำหนดเวลาสำหรับการนับ ช่วงเวลาที่กำหนดว่าจะให้มิเตอร์วัดรังสีวัดนานเท่าใดกำหนดได้ด้วยโปรแกรมเช่นกันและเพื่อความแม่นยำของเวลาจึงได้เลือกใช้ไท

เมอร์เป็นส่วนในการทำงานโดยใช้ไมเมอร์ 1 เป็นตัวนับเวลาร่วมกับอินเทอร์รัพท์ หลักการของการนับคือจะกำหนดให้ไทเมอร์นับจากค่าที่กำหนดไปจนถึงค่า FFFFH ในการออกแบบได้กำหนดให้ทำการนับ 5 มิลลิวินาที จะต้องทำการคำนวณเพื่อหาค่าเริ่มต้นให้กับไทเมอร์ซึ่งคำนวณได้จากสมการ

$$TMR1 = 65536 - \left(\frac{Td}{Pr} \right) \times \left(\frac{fosc}{4} \right)$$

เมื่อ TMR1 เป็นค่าเริ่มต้นที่ต้องป้อนให้กับรีจิสเตอร์ของไทเมอร์ 1

Td เป็นคาบเวลาทั้งหมดที่ต้องการให้นับ

Pr เป็นค่าของ Prescaler ซึ่งอยู่ภายในไมโครคอนโทรลเลอร์

fosc เป็นความถี่ของสัญญาณนาฬิกาของหน่วยประมวลผลกลาง

จากการออกแบบแทนค่าต่างๆดังนี้ Td คาบเวลากำหนดเป็น 10 มิลลิวินาที Pr กำหนดให้เป็น 1 และ fosc ความถี่ของคริสตัลเป็น 20 เมกะเฮิร์ตซ์ แทนลงในสมการ

$$TMR1 = 65536 - \left(\frac{5 \times 10^{-3}}{1} \right) \times \left(\frac{48 \times 10^6}{4} \right)$$

ได้ TMR1 = 5536 ในฐานสิบ หรือ 15A0H ในฐานสิบหก ซึ่งเป็นค่าที่ต้องป้อนให้กับรีจิสเตอร์ TMR1 โดยแบ่งเป็นการป้อนค่า 15H ให้กับรีจิสเตอร์ TMR0H และค่า A0H ให้กับรีจิสเตอร์ TMR0L เมื่อไทเมอร์ทำงานก็จะนับค่าขึ้นไปจนถึงค่า FFFFH ใช้เวลาทั้งสิ้น 5 ms

จากฐานเวลาที่ได้จากที่กล่าวมาข้างต้นเมื่อต้องการจะนำไปเพื่อให้สามารถวัดเวลาได้มากขึ้นทำได้ด้วยการกำหนดจำนวนรอบของการนับเช่นถ้าต้องการให้นับได้ 1 วินาที ก็จะกำหนดจำนวนรอบทั้งหมด 200 รอบ รอบละ 5 มิลลิวินาที ก็จะได้ทั้งหมด 1 วินาที โดยถ้าต้องการเวลาเพิ่มขึ้นก็เพียงเพิ่มจำนวนรอบเข้าไปอีก ในการออกแบบของมิเตอร์วัดรังสีกำหนดให้สามารถทำการกำหนดเวลาได้ 3 กรณีคือ 1 นาที 5 นาที และ นับไม่จำกัดเวลา หรือสรุปก็คือให้นับเป็นจำนวนนับต่อนาที(count per minute ,cpm) นับเป็นจำนวนนับต่อ 5 นาที และ นับเป็นจำนวนนับแบบต่อเนื่อง

3.2.1.3 หน่วยความจำ

หน่วยความจำที่ใช้สำหรับเก็บข้อมูลการนับโดยเลือกใช้หน่วยความจำแบบอีอีพรอม(EEPROM) โดยมีคุณสมบัติเด่นตรงที่ถึงแม้ว่าจะไม่มีกระแสไฟฟ้าป้อนเข้าสู่วงจรด้วยเหตุจากการปิดเครื่อง การที่ไฟฟ้าดับ หรือใดๆก็ตามข้อมูลของหน่วยความจำนี้ก็จะไม่สูญหายไป สำหรับหน่วยความจำแบบอีอีพรอมของไมโครคอนโทรลเลอร์เบอร์ PIC18F2550 มีขนาด 256 ไบต์หรือสามารถเก็บข้อมูลขนาด 8 บิตได้ 256 ค่าข้อมูล สำหรับการออกแบบการเก็บข้อมูลได้กำหนดดังนี้คือ หน่วยความจำตำแหน่งแรกที่ Address 00 กำหนดให้ใช้สำหรับเก็บค่าตำแหน่งสุดท้ายที่ได้เก็บข้อมูลไปแล้ว ตำแหน่งต่อไป Address 01 เป็นการเก็บเวลาในการนับและอีกสองตำแหน่งเป็นอัตรานับที่ได้คือ Address 02 เก็บข้อมูลอัตรานับที่เป็นไบต์ต่ำ และ Address 03 เก็บข้อมูลอัตรานับที่เป็นไบต์สูง ดังนั้นในการนับ 1 ครั้งจะใช้เนื้อที่ในหน่วยความจำอีอีพรอม 3 ไบต์ต่อ 1 ชุดข้อมูล เมื่อเก็บข้อมูลอัตรานับชุดที่สองก็จะเริ่มด้วยการเก็บเวลาที่ตำแหน่ง Address 04 อัตรานับไบต์ต่ำที่ตำแหน่ง Address 05 อัตรานับไบต์สูงตำแหน่ง Address 06 ต่อเนื่องไปจนหมด ความจุของหน่วยความจำ ดังนั้นหน่วยความจำอีอีพรอมมีขนาด 256 ไบต์จะสามารถเก็บข้อมูลได้สูงสุด 85 ชุดข้อมูล ลักษณะการเก็บแสดงได้ดังรูปที่ 3.18

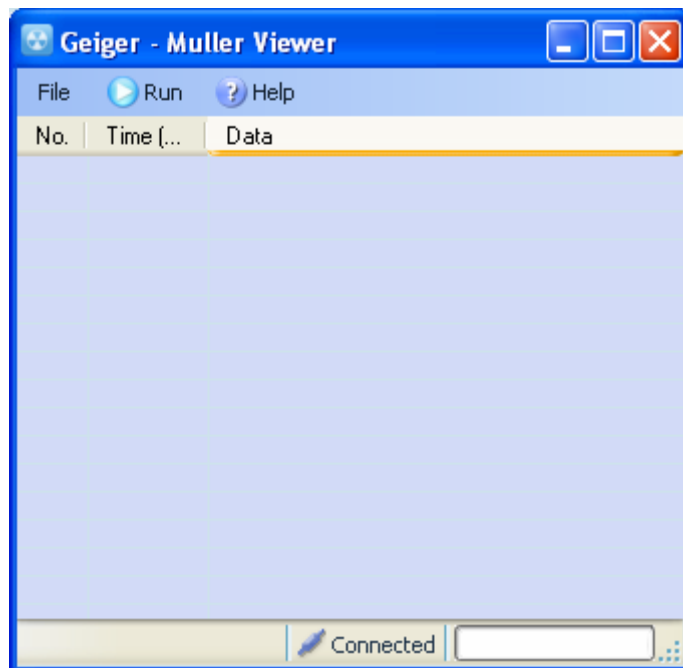


รูปที่ 3.16

แสดงรูปแบบการเก็บข้อมูลในอีอีพรอม

3.2.2 โปรแกรมคอมพิวเตอร์

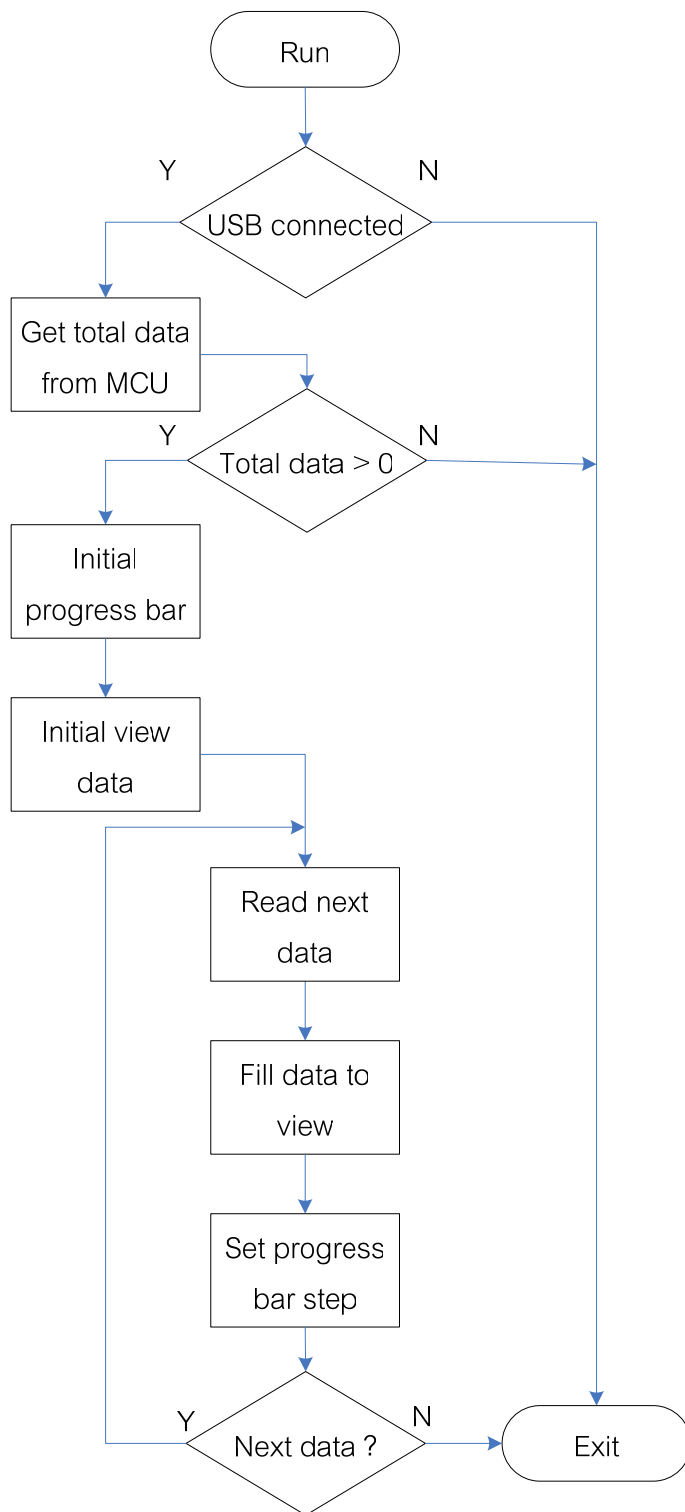
ส่วนของโปรแกรมคอมพิวเตอร์เป็นส่วนที่ใช้สำหรับการติดต่อระหว่างไมโครคอนโทรลเลอร์กับผู้ใช้งาน โดยผ่านการเชื่อมต่อข้อมูลแบบ USB เป็นโปรแกรมที่ใช้ในการนำข้อมูลของการนับวัดรังสีที่เก็บไว้ที่หน่วยความจำอีอีพรอมมาแสดงที่คอมพิวเตอร์และสามารถจะบันทึกเก็บไว้เป็นไฟล์เพื่อนำไปใช้งานต่อได้ ในการเขียนโปรแกรมใช้ภาษา C# ในการเขียน หน้าของโปรแกรมแสดงได้ดังรูปที่ 3.17



รูปที่ 3.17

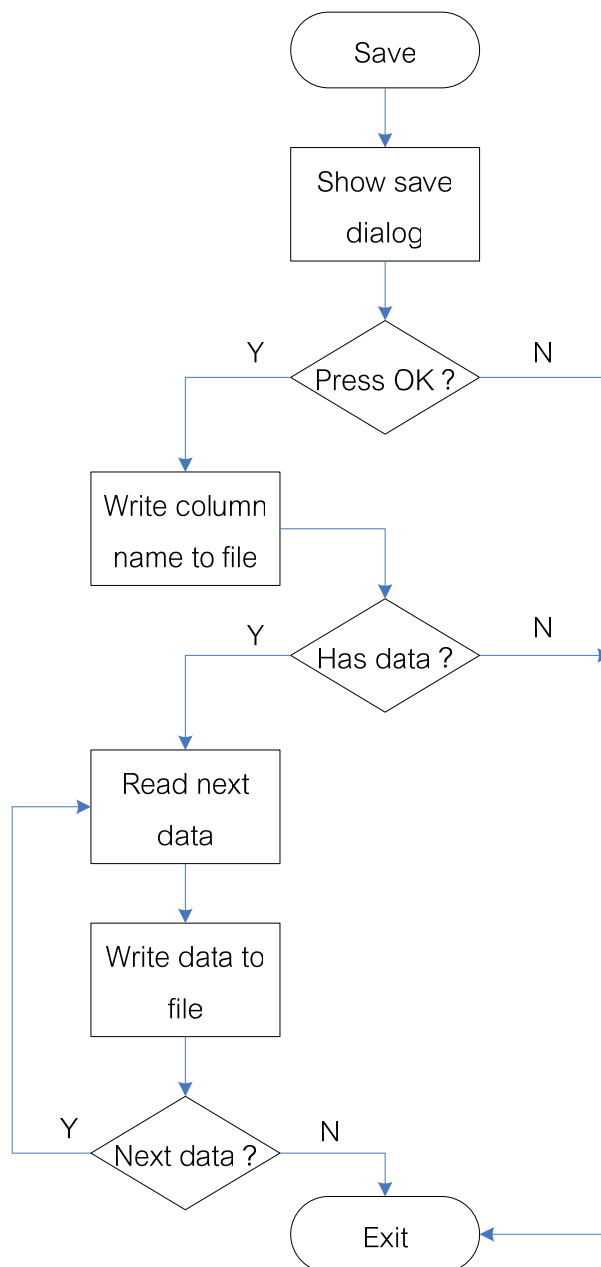
โปรแกรม Geiger – Muller Viewer

โปรแกรมส่วนบนสุดประกอบด้วยเมนูหลักคือ File Run และ Help โดยเมนู File ใช้สำหรับเปิดเมนูเพื่อเข้าการบันทึกข้อมูลเป็นไฟล์ หรือการเข้าสู่เมนู Save ส่วนเมนูถัดมาคือ Run เมื่อกด โปรแกรมจะทำการดึงข้อมูลที่เก็บไว้ในอีพ롬ของไมโครคอนโทรลเลอร์ที่อยู่ภายในมิเตอร์วัดรังสี ข้อมูลที่ได้มาจะนำมาแสดงในส่วนพื้นที่ตรงกลางจะมีลำดับที่ของข้อมูล เวลาที่ใช้ในการเก็บ และข้อมูลที่เก็บจากค่าอัตรานับที่นับได้ ส่วนด้านล่างตรงกลางเป็นที่บอกสถานะของการเชื่อมต่อระหว่างมิเตอร์วัดรังสีกับคอมพิวเตอร์ผ่าน USB ว่าสามารถเชื่อมต่อได้หรือไม่ได้ โดยถ้าสามารถเชื่อมต่อกับมิเตอร์วัดรังสีได้จะขึ้นว่า Connected แต่ถ้าไม่ได้จะขึ้น Not Connected เป็นการบอกถึงสถานะการณ้เชื่อมต่อ



รูปที่ 3.18
ฟลอร์ชาร์ตของขั้นตอนการแสดงผลการนับ

ในการบันทึกผลการนับนั้นเมื่อโปรแกรมแสดงผลการนับร้องขอและแสดงผลการนับจากอุปกรณ์ไมโครคอนโทรลเลอร์แล้ว โปรแกรมสามารถบันทึกผลการนับด้วยการคลิกที่ปุ่ม File จากนั้นคลิกปุ่ม Save เพื่อบันทึกผลการนับในรูปแบบเท็กซ์ไฟล์



รูปที่ 3.19
ไฟล์ชาร์ตของขั้นตอนการบันทึกผลการนับ