

การควบคุมไฟจราจรอัตโนมัติ
ด้วยตัวตรวจจับการเคลื่อนที่ของรถยนต์

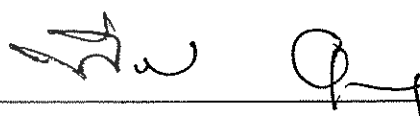
โดย

นายอภิสิทธิ์	พุทธศรี	รหัสนักศึกษา	B4907909
นางสาวอัครา	โสตก	รหัสนักศึกษา	B4908166
นางสาววิस्ता	สงเสริม	รหัสนักศึกษา	B5047192

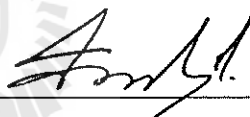
รายงานนี้เป็นส่วนหนึ่งของการศึกษาวิชา 427499 โครงการวิศวกรรมโทรคมนาคม
และวิชา 427494 โครงการศึกษาวิศวกรรมโทรคมนาคม
หลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมโทรคมนาคม หลักสูตรปรับปรุง พ.ศ. 2545
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี
ประจำภาคการศึกษาที่ 1 และ 2 ปีการศึกษา 2553

การควบคุมไฟจราจรอัตโนมัติด้วยตัวตรวจจับการเคลื่อนที่ของรถยนต์

คณะกรรมการสอบโครงการ



(ผู้ช่วยศาสตราจารย์ ดร. พีระพงษ์ อุฑารสกุล)
กรรมการ/อาจารย์ที่ปรึกษาโครงการ



(ผู้ช่วยศาสตราจารย์ ดร. มนต์ทิพย์ภา อุฑารสกุล)
กรรมการ



(ผู้ช่วยศาสตราจารย์ ดร. วิภาวี หัตถกรรม)
กรรมการ

มหาวิทยาลัยเทคโนโลยีสุรนารี อนุมัติให้นำรายงานโครงการฉบับนี้ เป็นส่วนหนึ่งของการศึกษาระดับปริญญาตรี สาขาวิชาวิศวกรรมโทรคมนาคม วิชา 427499 โครงการวิศวกรรมโทรคมนาคม และวิชา 427494 โครงการศึกษาวิศวกรรมโทรคมนาคม ประจำปีการศึกษา 2553

หัวข้อโครงการ	การควบคุมไฟจราจรอัตโนมัติด้วยตัวตรวจับการเคลื่อนที่ของรถยนต์		
โดย	นายอภิสิทธิ์ พุทธิศรี	รหัสนักศึกษา	B4907909
	นางสาวอัครา โสตก	รหัสนักศึกษา	B4908166
	นางสาววิστα ส่งเสริม	รหัสนักศึกษา	B5047192
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร.พิระพงษ์ อุฑารสกุล		
สาขาวิชา	วิศวกรรมโทรคมนาคม		
ภาคการศึกษาที่	3/2553		

บทคัดย่อ

โครงการการควบคุมไฟจราจรอัตโนมัติด้วยตัวตรวจับการเคลื่อนที่ของรถยนต์ เป็นโครงการที่นำ Sensor ตรวจับการเคลื่อนที่ของรถ โดยใช้ Microcontroller ประมวลผล ซึ่งระบบไฟจราจรที่พัฒนาขึ้น แบ่งออกเป็น 2 ส่วน คือ ส่วนแรกเป็นตัวตรวจับและประมวลผล ส่วนที่สอง คือ ส่วนที่ควบคุมไฟจราจร โดยใช้ Microcontroller เพื่อบันทึกเวลา เหมือนระบบไฟจราจรแบบเดิมอยู่แต่จะมีการเพิ่มการตรวจับรถยนต์ที่เข้ามาในระบบสี่แยกไฟแดง เพื่อนำผลตรวจับที่ได้นั้น ไปประมวลผลในการควบคุมทิศทางการให้รถหยุดอย่างเหมาะสมและมีประสิทธิภาพมากที่สุด

มหาวิทยาลัยเทคโนโลยีสุรนารี

กิตติกรรมประกาศ

โครงการเรื่อง การควบคุมไฟจราจรอัตโนมัติด้วยตัวตรวจจับการเคลื่อนที่ของรถยนต์ สามารถสำเร็จลุล่วงไปได้ด้วยดีเนื่องจากคณะผู้จัดทำโครงการได้รับความช่วยเหลือด้านต่างๆ จากอาจารย์ที่ปรึกษา ผู้ช่วยศาสตราจารย์ ดร.พีระพงษ์ อุทธารสกุล ผู้ที่เป็นแนวความคิดริเริ่มในโครงการเรื่อง การควบคุมไฟจราจรอัตโนมัติด้วยตัวตรวจจับการเคลื่อนที่ของรถยนต์ ซึ่งได้ให้ความช่วยเหลือและให้คำปรึกษาในทุกๆ ด้านแก่คณะผู้จัดทำมาโดยตลอด และขอบคุณเพื่อนๆ สาขาวิศวกรรมโทรคมนาคมทุกๆ คนที่คอยให้ความช่วยเหลือคณะผู้จัดทำโครงการในด้านต่างๆ และสุดท้ายนี้คณะผู้จัดทำโครงการขอขอบพระคุณบิดามารดาที่ท่านทั้งสองให้การดูแลเอาใจใส่คอยให้กำลังใจและอยู่เคียงข้างมาโดยตลอด จนกระทั่งโครงการนี้สำเร็จลุล่วงไปได้ด้วยดี

คณะผู้จัดทำโครงการใคร่ขอขอบพระคุณทุกๆ ท่านที่ได้กล่าวไปแล้วไว้ ณ ที่นี้ สำหรับส่วนดีของโครงการชิ้นนี้ ขออุทิศให้แก่คณาจารย์ทุกท่านที่ได้ประสิทธิ์ประสาทวิชาความรู้ให้แก่คณะผู้จัดทำโครงการ หากโครงการชิ้นนี้มีข้อผิดพลาดประการใดทางคณะผู้จัดทำโครงการใคร่ขอน้อมรับและขออภัยมา ณ ที่นี้

นายอภิสิทธิ์ พุทศรี

นางสาวอัศรา โสตก

นางสาววิस्ता ส่งเสริม

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญ

เรื่อง	หน้า
บทคัดย่อ	ก
กิตติกรรมประกาศ	ข
สารบัญ	ค
สารบัญรูป	ฉ
บทที่ 1 บทนำ	1
1.1 บทนำ	1
1.2 หลักการและเหตุผล	1
1.3 วัตถุประสงค์	1
1.4 ขอบเขตงาน	1
1.5 ผลที่คาดว่าจะได้รับ	1
บทที่ 2 หลักการและทฤษฎีที่เกี่ยวข้อง	3
2.1 บทนำ	3
2.2 คุณลักษณะเฉพาะของโฟโตไดโอด	4
2.3 หลักการทำงานของโฟโตไดโอด	4
2.4 คมลักษณะเฉพาะทางไฟฟ้า	5
2.4.1 Shunt Resistance, (R_{SH})	6
2.4.2 Series Resistance, (R_S)	6
2.4.3 Junction Capacitance, C_j	6
2.4.4. Rise/Fall Time และ การตอบสนองทางความถี่ τ_r/τ_{fb}	7
2.5 คุณลักษณะเฉพาะทางแสง	8
2.5.1 Quantum Efficiency, Q.E.	10
2.5.2 Non-Uniformity	10
2.5.3 Non-Linearity	10
2.6 คุณลักษณะเฉพาะของ I-V	11
2.7 Noise	12
2.7.1 Shot noise	12
2.7.2 Thermal หรือ Johnson noise	12
2.7.3 Total noise	13
2.7.4 Noise Equivalent Power (NEP)	13

สารบัญ(ต่อ)

เรื่อง	หน้า
2.8 ผลกระทบอันเนื่องมาจากอุณหภูมิ	13
2.8.1 Shunt Resistance and Dark Current	13
2.8.2 BreakDown Voltage	14
2.8.3 Responsivity	14
2.8.4 BIASING	14
2.8.5 Photoconductive mode(PC)	14
2.9 องค์ประกอบทั่วไปที่ถูกลำมาใช้ในการกำหนดค่า	15
2.10 Photovoltaic Mode(PV)	16
2.11 POSITION SENSING DETECTORS	18
2.12 อภิธานศัพท์	20
2.13 การคำนวณหาตำแหน่ง	21
2.14 การไบอัสทรานซิสเตอร์(Transistor Biasing)	25
2.14.1การไบอัสแบบคงที่	27
2.15 รูปแบบการทำงานของไฟจราจร	29
2.16 หลอด LED	29
2.16.1 ตัวแสดงผล 7 ส่วน	30
2.16.2 การต่อ LED ภายในตัว 7 Segment	31
2.16.3 การดูสัญลักษณ์การต่อภายใน 7 Segment	32
บทที่ 3 การออกแบบโปรแกรมและการทดสอบ โปรแกรม	34
3.1 บทนำ	34
3.2. การออกแบบ Sensor	34
3.3 ออกแบบการทำงานของโปรแกรม	36
3.4 Flow Chart การทำงานของ โปรแกรม	38
3.5 ไมโครคอนโทรลเลอร์ที่ใช้	47
3.6 บอร์ดแสดงผล 7-Segment	50
3.7 การเขียนโปรแกรม	51

สารบัญ(ต่อ)

เรื่อง	หน้า
บทที่ 4 ผลการทดสอบ โปรแกรม	73
4.1 บทนำ	73
4.2 การทดสอบการใช้งาน โปรแกรม“แสดงผลระบบสัญญาณไฟจราจรอัตโนมัติ”	73
บทที่ 5 สรุปและข้อเสนอแนะ	80
5.1 บทนำ	80
5.2 สรุปผลการทดสอบ	80
5.3 ปัญหาและอุปสรรค	80
5.4 ข้อเสนอแนะ	81
ประวัติผู้เขียน	82
บรรณานุกรม	83



สารบัญรูป

เรื่อง	หน้า
รูปที่ 2.1 แสดงการแพร่กระจายตัวของซิลิคอน โฟโตไดโอด	4
รูปที่ 2.2 กราฟแสดงการดูดกลืนแสงผ่านผิวซิลิคอนที่ความยาวคลื่นต่างๆ	5
รูปที่ 2.3 แสดงวงจรการทำงานของซิลิคอน โฟโตไดโอด	5
รูปที่ 2.4 กราฟแสดงความสัมพันธ์ระหว่างความจุไฟฟ้าเทียบกับแรงดันไบแอสย้อนกลับ	7
รูปที่ 2.5 กราฟแสดงความสัมพันธ์ระหว่างการตอบสนองทางสเปกตรัมกับความยาวคลื่น	9
รูปที่ 2.6 กราฟแสดงความสัมพันธ์ระหว่างอุณหภูมิกับความยาวคลื่น	9
รูปที่ 2.7 กราฟแสดงคุณลักษณะเฉพาะของ I-V	11
รูปที่ 2.8 แสดงวงจร Low Light Level / Wide Bandwidth	15
รูปที่ 2.9 แสดงวงจร High Light Level / High Speed Response	16
รูปที่ 2.10 แสดงวงจร Ultra low level light / Low Speed	17
รูปที่ 2.11 แสดง Block Diagram องค์ประกอบต่างๆ ของวงจร	18
รูปที่ 2.12 แสดงวงจรที่ถูกใช้งานร่วม UDS Sensor Segmented Photodiode	21
รูปที่ 2.13 แสดงวงจร one-dimensional lateral PSD's	22
รูปที่ 2.14 แสดงวงจร Two Dimensional Duo-lateral PSD's	23
รูปที่ 2.15 แสดงวงจร two dimensional tetra-lateral PSD's	24
รูปที่ 2.16 แสดงการทำงานของทรานซิสเตอร์แบบ NPN	25
รูปที่ 2.17 แสดงผลของอุณหภูมิที่มีผลต่อ แรงดัน V_{BE}	26
รูปที่ 2.18 แสดงผลของอุณหภูมิที่มีผลต่อค่า h_{FE}	26
รูปที่ 2.19 แสดงวงจร ไบอัสแบบคงที่	27
รูปที่ 2.20 แสดงวงจรทางด้านอินพุตของการไบอัสแบบคงที่	28
รูปที่ 2.21 แสดงวงจรทางด้านเอาต์พุตของการไบอัสแบบคงที่	29
รูปที่ 2.22 แสดงรูปแบบการปล่อยแบบที่ละเอียด	29
รูปที่ 2.23 แสดงลักษณะ ไดโอดเปล่งแสง	30
รูปที่ 2.24 แสดงวงจรการต่อหลอด LED เข้ากับ GPIO port ของไมโครคอนโทรลเลอร์	30
รูปที่ 2.25 แสดงรูปแบบต่างๆ และสัญลักษณ์	31
รูปที่ 2.26 แสดงตำแหน่งส่วนแสดงผล A- G	31
รูปที่ 2.27 แสดงการนำเอา LED มาต่อกัน แบบคอมมอน K	31

สารบัญรูป (ต่อ)

เรื่อง	หน้า
รูปที่ 2.28 แสดงการนำเอา LED มาต่อกัน แบบคอมมอน K	32
รูปที่ 2.29 แสดง 1 หลัก คอมมอน A ที่ขา 3 กับ 14 ส่วนขา 4 ,5,6,12 ไม่ได้ใช้	32
รูปที่ 2.30 แสดง 1 หลัก คอมมอน K ที่ขา 3 กับ 8	32
รูปที่ 2.31 แสดง 2 หลัก คอมมอน K ที่ขา 10(ตัวที่1) กับ 5(ตัวที่ 2)	33
รูปที่ 2.32 แสดง 2 หลัก คอมมอน A ที่ขา 10(ตัวที่1) กับ 5(ตัวที่ 2)	33
รูปที่ 2.33 แสดง 4 หลัก คอมมอน K ที่ขา 3(ตัวที่1) กับ 5(ตัวที่ 2) กับ 8(ตัวที่3) กับ 10(ตัวที่ 4)	33
รูปที่ 3.1 ลักษณะการต่อใช้งาน โฟโตไดโอด	34
รูปที่ 3.2 แสดงวงจรตัว Sensor	34
รูปที่ 3.3 แสดงเส้นทางสี่แยก	36
รูปที่ 3.4 แสดงการทำงาน โดยรวมของ โปรแกรม	38
รูปที่ 3.5 แสดงการทำงาน โปรแกรมย่อย Class A	39
รูปที่ 3.6 แสดงการทำงาน โปรแกรมย่อย Class B	40
รูปที่ 3.7 แสดงการทำงาน โปรแกรมย่อย Class C	41
รูปที่ 3.8 แสดงการทำงาน โปรแกรมย่อย Class D	42
รูปที่ 3.9 แสดงการทำงาน โปรแกรมย่อย Check A	43
รูปที่ 3.10 แสดงการทำงาน โปรแกรมย่อย Check B	44
รูปที่ 3.11 แสดงการทำงาน โปรแกรมย่อย Check C	45
รูปที่ 3.12 แสดงการทำงาน โปรแกรมย่อย Check D	46
รูปที่ 3.13 แสดงการ โครงสร้างบอร์ด ET-ARM7 LPC2214	48
รูปที่ 3.14 แสดง GPIO0 4 พอร์ต	49
รูปที่ 3.15 แสดง GPIO0 3 พอร์ต	49
รูปที่ 3.16 แสดง GPIO2 4 พอร์ต	49
รูปที่ 3.17 แสดง GPIO3 4 พอร์ต	50
รูปที่ 3.18 บอร์ดแสดงผล 7-Segment	50
รูปที่ 3.19 ผังวงจร PLL ภายในไมโครคอนโทรลเลอร์ ARM7	51
รูปที่ 3.20 แสดงวงจร RTC	52
รูปที่ 4.1 แสดงขณะที่มีรถจอด A, B, C และ D	73
รูปที่ 4.2 แสดงขณะที่มีรถจอด A และ B	74
รูปที่ 4.3 แสดงขณะที่มีรถจอด A, C และ D	74

สารบัญรูป(ต่อ)

เรื่อง	หน้า
รูปที่ 4.4 แสดงขณะที่มีรถจอด A และ D	75
รูปที่ 4.5 แสดงขณะที่มีรถจอด B และ C	75
รูปที่ 4.6 แสดงขณะที่มีรถจอด B และ D	76
รูปที่ 4.7 แสดงขณะที่มีรถจอด C และ D	76
รูปที่ 4.8 แสดงขณะที่มีรถจอด A , B และ C	77
รูปที่ 4.9 แสดงขณะที่มีรถจอด B และ D	77
รูปที่ 4.10 แสดงขณะที่มีรถจอด A และ D	78
รูปที่ 4.11 แสดงขณะที่มีรถจอด B , C และ D	78
รูปที่ 4.12 แสดงการปล่อยสัญญาณไฟเขียวเป็นสามเท่า	79



บทที่ 1

บทนำ

1.1 บทนำ

การควบคุมไฟจราจรอัตโนมัติด้วยตัวตรวจจับการเคลื่อนที่ของรถยนต์ เป็นโครงการเกี่ยวกับการพัฒนา Sensor เพื่อตรวจจับการเคลื่อนที่ของรถ โดยใช้ Microcontroller ประมวลผล ซึ่งระบบไฟจราจรที่พัฒนาขึ้นแบ่งออกเป็น 2 ส่วน คือ ส่วนแรกเป็นตัวตรวจจับและประมวลผล ส่วนที่สอง คือ ส่วนที่ควบคุมไฟจราจร โดยใช้ Microcontroller เพื่อนับเวลา เหมือนระบบไฟจราจรแบบเดิมอยู่แต่จะมีการเพิ่มการตรวจจับรถยนต์ที่เข้ามาในระบบสี่แยกไฟแดง เพื่อนำผลตรวจจับที่ได้นั้นไปประมวลผลในการควบคุมทิศทางทำให้รถหยุดอย่างเหมาะสมและมีประสิทธิภาพมากที่สุด

1.2 หลักการและเหตุผล

เนื่องจากในปัจจุบันนี้เกิดปัญหาจราจรติดขัดเพิ่มขึ้น โดยเฉพาะถนนสายหลักในการคมนาคมขนส่ง และปัญหาที่เกิดขึ้นบ่อยๆ เช่น ในกรณีถนนเส้นหลักเป็นไฟแดง ถนนเส้นรองเป็นไฟเขียวซึ่งไม่มีรถผ่าน ปัญหาในจุดนี้คือ ทำให้เกิดการสูญเสียพลังงานในการหยุดรถ ซึ่งเราจะสังเกตได้ว่า ส่วนมากจะเป็นรถขนส่งขนาดใหญ่และรถยนต์ส่วนบุคคล ซึ่งการสูญเสียดังกล่าวถ้าเกิดปัญหาในกรณีนี้เพียงจุดเดียวจะเป็นการสูญเสียพลังงานที่น้อย แต่ในความเป็นจริงแล้วปัญหาแบบนี้เกิดขึ้นในหลายๆจุดบนถนนทางหลวงสายหลักของประเทศ นับเป็นการสูญเสียพลังงานที่ไม่เกิดประโยชน์เป็นปริมาณมากพอสมควร

ดังนั้นจึงได้มีการพัฒนาระบบไฟจราจรเพื่อลดปัญหาการจราจรดังกล่าว โดยใช้ Sensor เข้ามาตรวจจับรถ เมื่อทางสายรองไม่มีรถเลยก็จะทำการปล่อยรถทางสายหลักผ่านไปโดยไม่ต้องหยุดรอ ซึ่งจะเป็นการเพิ่มประสิทธิภาพการขนส่งได้ดีขึ้น

1.3 วัตถุประสงค์

1. ช่วยลดปัญหาจราจรติดขัดบนท้องถนนได้
2. ช่วยประหยัดเวลา
3. ช่วยประหยัดพลังงานในการคมนาคมและขนส่ง

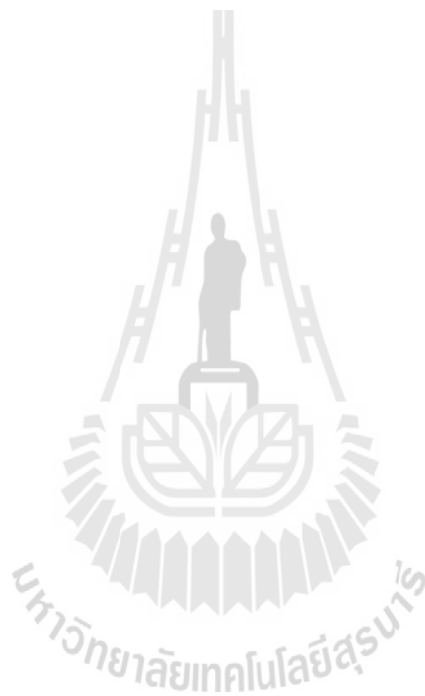
1.4 ขอบเขตงาน

1. ออกแบบ Sensor ตรวจจับรถบนท้องถนน
2. สร้างโปรแกรมประมวลผลควบคุมไฟจราจร โดยใช้ Microcontroller
3. สร้างแบบจำลองไฟจราจร

1.5 ผลที่คาดว่าจะได้รับ

1. สามารถลดปัญหาจราจรติดขัดบนท้องถนน
2. สามารถช่วยประหยัดเวลาในการเดินทาง
3. ช่วยประหยัดพลังงานในการคมนาคมขนส่ง
4. มีความรู้ความเข้าใจในการใช้โปรแกรม Microcontroller ในการประมวลผล และควบคุมไฟจราจร

5. สามารถพัฒนา Sensor เพื่อตรวจจับการเคลื่อนที่ของรถ เพื่อนำผลตรวจจับที่ได้ขึ้นไปประมวลผลในการควบคุมทิศทางการ ให้รถหยุดอย่างเหมาะสมและมีประสิทธิภาพมากที่สุด



บทที่ 2

ทฤษฎีและเนื้อหาที่เกี่ยวข้อง

2.1 บทนำ

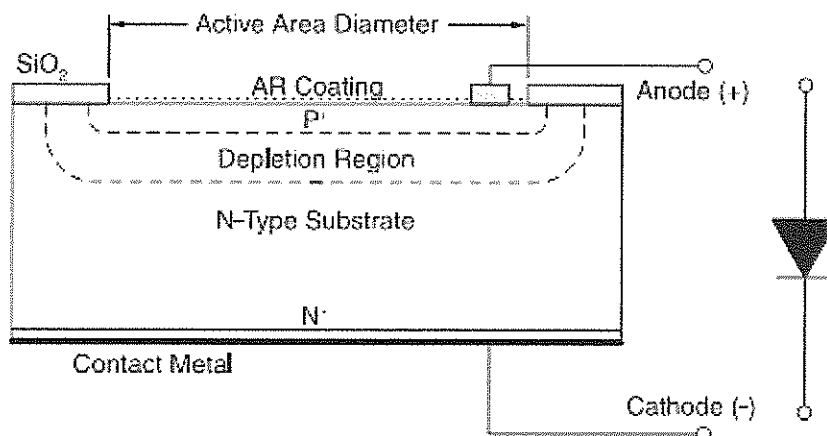
เนื้อหาในบทนี้จะกล่าวถึงทฤษฎีที่เกี่ยวข้องกับ หลักการทำงาน คุณลักษณะเฉพาะของโฟโตไดโอด และ โครงสร้างของซิลิคอนไดโอด แบบพลาเนาติฟิฟวส์

2.2 คุณลักษณะเฉพาะของ โฟโตไดโอด

ซิลิคอนโฟโตไดโอด คืออุปกรณ์ประเภทเซมิคอนดักเตอร์ ซึ่งตอบสนองต่ออนุภาคลังงานสูง และโฟตอนโฟโตไดโอดทำงานโดยการดูดซับโฟตอนหรืออนุภาคซึ่งถูกประจุพลังงาน แล้วจะสร้างกระแสไฟไหลไปยังวงจรภายนอก ตามสัดส่วนของพลังงานที่เกิดขึ้น โฟโตไดโอดยังสามารถใช้เพื่อตรวจหาปริมาณของแสง และสามารถใช้ในการปรับตั้งค่า(calibrate) สำหรับการวัดค่าซึ่งมีความละเอียดสูง ตั้งแต่ความหนาแน่น 1 pW/cm^2 ไปจนถึงความหนาแน่น 100 mW/cm^2 ซิลิคอนโฟโตไดโอดถูกนำมาประยุกต์ใช้ประโยชน์อย่างหลากหลายใน spectroscopy, photography, analytical instrumentation, optical position sensor, beam alignment, surface characterization, Laser Range Finder, optical communication และ medical imaging instrument

โครงสร้างของซิลิคอนไดโอด แบบพลาเนาติฟิฟวส์ ซิลิคอนไดโอด แบบพลาเนาติฟิฟวส์ จัดเป็นไดโอดประเภท จุดต่อ P-N(P-N junction) โดยไดโอดประเภทนี้จะถูกสร้างขึ้น โดยการแพร่อนุภาคอะตอมของสาร impurity ชนิด P-type(anode) ยกตัวอย่างเช่น โบรอน(Boron) เข้าไปยังเนื้อซิลิคอนเวเฟอร์ประเภท P-type หรือโดยการแพร่อนุภาคอะตอมของสาร impurity ชนิด N-typeยกตัวอย่างเช่น ฟอสฟอรัส (Phosphorous) เข้าไปยังเนื้อซิลิคอนเวเฟอร์ประเภท P-type พื้นที่ซึ่งมีอนุภาคอะตอมแพร่กระจายเข้าไปจะกลายเป็นพื้นที่การทำงานของโฟโตไดโอด เพื่อจะทำให้เกิด โอห์มิกคอนแทค (ohmic contact) การแพร่อนุภาคอะตอมของสาร impurity ไปยังด้านหลังของเวเฟอร์เป็นสิ่งจำเป็น ในพื้นที่การทำงานของ P-type สาร impurity จะเป็น N-type และเช่นเดียวกัน ในพื้นที่การทำงานของ N-type สาร impurity จะเป็น P-type แผ่นคอนแทคจะอยู่บริเวณด้านหน้าของพื้นที่การทำงาน และด้านหลังเพื่อให้สามารถห่อหุ้มอุปกรณ์โดยสมบูรณ์ พื้นที่การทำงานจะถูกวางไว้ด้วย anti-reflection coating เพื่อลดการสะท้อนกลับของแสง เพื่อให้สามารถเจาะจงความยาวคลื่นได้ พื้นที่ซึ่งไม่ถูกใช้งานด้านบนสุดจะถูกห่อหุ้มด้วยชั้นของซิลิคอนออกไซด์ การควบคุมความหนาของพื้นผิวจะทำให้สามารถควบคุมความเร็วและอัตราการตอบสนองของโฟโตไดโอดได้

ข้อควรจำ : การไบอัส จะต้องทำในโหมด ไบอัสกลับ(reverse bias) นั่นคือจ่ายไฟบวกให้ขั้วแคโทด และจ่ายไปลบให้ขั้วแอโนด

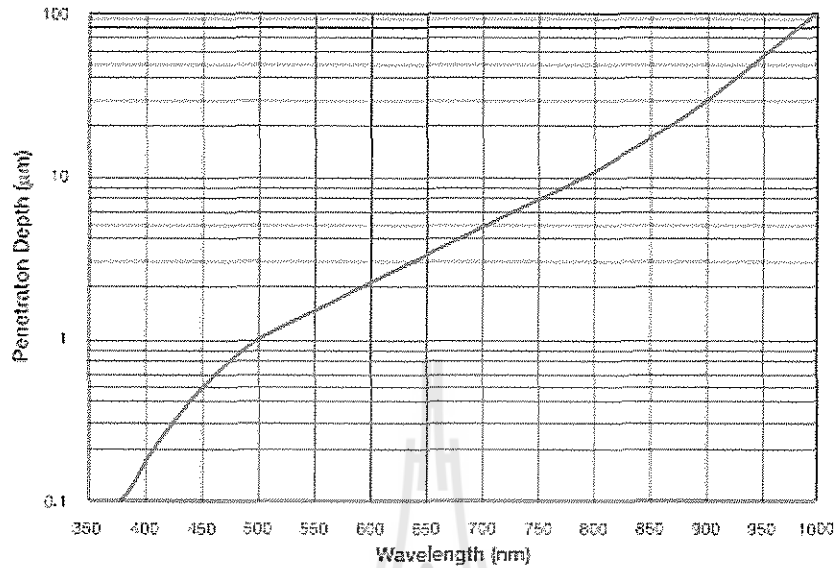


รูปที่ 2.1 แสดงการแพร่กระจายตัวของซิลิคอนโฟโตไดโอด

2.3 หลักการทำงาน

ซิลิคอนเป็นสารกึ่งตัวนำซึ่งมี band gap energy อยู่ที่ 1.12 eV ที่อุณหภูมิห้อง ซึ่งก็คือช่องว่างระหว่าง valence band และ conduction band ที่อุณหภูมิศูนย์องศาสัมบูรณ์ valence band จะถูกเติมเต็มอย่างสมบูรณ์และ conduction band จะหายไป เมื่ออุณหภูมิเพิ่มขึ้น อิเล็กตรอนจะถูกกระตุ้นและกระโดดจาก valence band ไปยัง conduction band โดยอาศัยพลังงานความร้อน เช่นเดียวกัน อิเล็กตรอนสามารถกระโดดไปยัง conduction band โดยอนุภาคอะตอม หรือ โฟตอนซึ่งมีพลังงานมากกว่า 1.12 eV ซึ่งตรงกับความยาวคลื่นที่สั้นกว่า 1100 nm ซึ่งส่งผลให้อิเล็กตรอนใน conduction band เป็นอิสระ และสามารถนำกระแสได้ เนื่องจากผลของการไล่ระดับความเข้ม, การแพร่ของอิเล็กตรอนจากพื้นที่ N-type ไปยังพื้นที่ P-type และการแพร่ของโฮล (hole) จากพื้นที่ P-type ไปยังพื้นที่ N-type จะก่อให้เกิด “built-in voltage” ข้ามจุดต่อ (junction) การแพร่กระจายภายใน (inner-diffusion) ของอิเล็กตรอน และโฮล ระหว่างพื้นที่ P และ N ข้ามจุดต่อจะส่งผลให้ไม่มีพาหะอิสระ (free carrier) ซึ่งพื้นที่ดังกล่าวจะถูกเรียกว่า “depletion region” การที่ built-in voltage ข้าม depletion region จะส่งผลให้เกิดสนามไฟฟ้าสูงสุดที่จุดต่อ และไม่มีสนามไฟฟ้าภายนอก depletion region เมื่อทำการไบอัสกลับให้แก่ built-in voltage จะส่งผลให้ depletion region ขยายกว้างขึ้น คู่อิเล็กตรอน โฮลที่ถูกสร้างขึ้นโดยแสงจะถูกกวาดไปไว้ใน depletion region และจะถูกรวบรวมกลับเข้ามา โดยการแพร่กระจายจาก undepleted region กระแสจะถูกสร้างขึ้นจะแปรผันตามอิทธิพลของแสง หรือพลังงานที่แผ่ออกมา แสงที่ถูกดูดกลืน จะขึ้นอยู่กับระยะทาง และประสิทธิภาพในการดูดกลืน โดยประสิทธิภาพในการดูดกลืนจะสูงมากในความยาวคลื่นที่สั้นกว่าย่าน UV และต่ำมากในความยาวคลื่นที่ยาวกว่า (รูปที่ 2.2) ด้วยเหตุนี้ ความยาวคลื่นช่วงสั้นเช่น ช่วง UV จะถูกดูดกลืนในบริเวณพื้นผิวด้านบนสุด

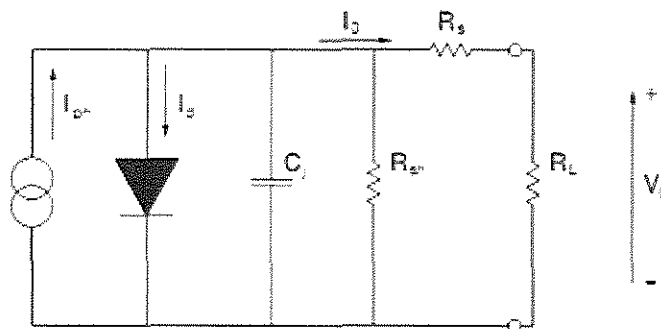
ขณะที่ซิลิคอนจะยอมให้แสงที่มีความยาวคลื่นมากกว่า 1200 nm. ผ่านไปได้ส่วนอิเล็กตรอนซึ่งมีพลังงานน้อยกว่าค่า band gab จะไม่ถูกดูดกลืน



รูปที่ 2.2 กราฟแสดงการดูดกลืนแสงผ่านผิวซิลิคอนที่ความยาวคลื่นต่างๆ

2.4 คุณสมบัติเฉพาะทางไฟฟ้า

เราสามารถแสดงวงจรการทำงานของซิลิคอน โฟโตไดโอดได้ดังรูปที่ 3 โดยแหล่งจ่ายกระแสจะแสดงถึงกระแสซึ่งถูกสร้างขึ้นปรากฏการณ์ทางแสงที่เกิดขึ้นในโฟโตไดโอด ไดโอดจะแสดงถึงการทำงานของขั้วต่อ P-N และสามารถอธิบายการทำงานที่เกิดขึ้นในโฟโตไดโอดทั้งหมดโดยการอธิบายด้วยส่วนของ junction capacitance (C_j) shunt resistance (R_{sh}) และ resistance (R_s) ดังรูปที่ 2.3



รูปที่ 2.3 แสดงวงจรการทำงานของซิลิคอน โฟโตไดโอด

2.4.1 Shunt Resistance, (R_{SH})

Shunt Resistance คือค่าความลาดชันของกราฟกระแสแรงดันของโฟโตไดโอดในช่วงเริ่มต้นเช่นที่ $V=0$ ในทางอุดมคติ ค่า Shunt Resistance จะมีค่าเป็นอนันต์ แต่ในความเป็นจริงนั้น ค่าที่เกิดขึ้นจะอยู่ระหว่าง 10s ถึง 1000s ของเมกาโอห์ม โดยจะคำนวณได้จากการจ่ายพลังงานให้แก่โฟโตไดโอดเท่ากับ +10-(-10) mV จากนั้นทำการวัดกระแส และคำนวณหาค่าความต้านทานออกมา Shunt resistance จะใช้เพื่ออธิบายถึงกระแสรบกวน(noise current) ในโฟโตไดโอดในขณะยังไม่ทำการไบอัส (photovoltaic mode) โฟโตไดโอดที่ดีจึงควรมี Shunt resistance สูงๆ

2.4.2 Series Resistance, (R_s)

Series resistance ของโฟโตไดโอดเกิดขึ้นจากความต้านทานของหน้าสัมผัส และความต้านทานของ undepleted silicon (Figure 1) โดยสามารถคำนวณได้ดังนี้:

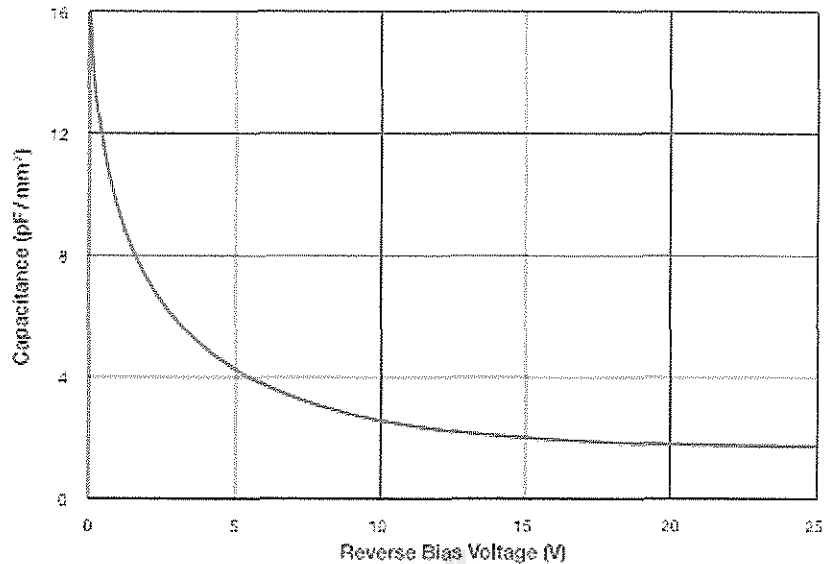
$$R_s = \frac{(W_s - W_d)}{A} + R_c \quad (1)$$

เมื่อ W_s คือความหนาของพื้นผิว(substrate) W_d คือความกว้างของ depleted region A คือพื้นที่การแพร่กระจายของจังก์ชัน(junction) p คือค่าความต้านทานของพื้นผิว และ R_c คือความต้านทานของหน้าสัมผัส Series-resistance จะใช้เพื่ออธิบายถึงความเป็นเชิงเส้นของโฟโตไดโอดในขณะยังไม่ทำการไบอัส (photovoltaic mode)($V=0$) ถึงแม้ว่าโฟโตไดโอดในอุดมคติไม่ควรจะมีค่า Series resistance แต่โดยทั่วไปแล้วในโฟโตไดโอดทั่วไป ค่าดังกล่าวจะอยู่ที่ 10 ถึง 1000 โอห์ม

2.4.3 Junction Capacitance, C_j

ขอบของ depletion region จะมีพฤติกรรมเสมือนเป็น parallel plate capacitor (รูปที่ 2.1) ค่า junction capacitance จะแปรผันตามพื้นที่แพร่กระจาย และ แปรผกผันกับความกว้างของ depletion region ดังนั้น พื้นผิวที่มีค่าความต้านทานสูง ย่อมมีค่า junction capacitance ต่ำ และค่าความเก็บประจุจะขึ้นอยู่กับ การไบอัสกลับดังนี้ :

$$C_j = \frac{\epsilon_{si}\epsilon_0 A}{\sqrt{2\mu\rho(V_A + V_{bi})}} \quad (2)$$



รูปที่ 2.4 กราฟแสดงความสัมพันธ์ระหว่างความจุไฟฟ้าเทียบกับแรงดันไบแอสย้อนกลับ

เมื่อความสามารถดูดซับในสุญญากาศ $\epsilon_0 = 8.854 \times 10^{-14} \text{ F/cm}$, ค่าคงที่อิเล็กทริกของซิลิคอน $\epsilon_{Si} = 11.9$, ค่าการเคลื่อนที่ของอิเล็กตรอนที่ 300°K $\mu = 1400 \text{ cm}^2/\text{Vs}$, p คือค่าความต้านทานของซิลิคอน, V_{bi} คือความต่างศักย์ภายในของซิลิคอน (built-in voltage) และ V_A คือความต่างศักย์ที่ใช้ในการไบแอส รูปที่ 2.4 แสดงให้เห็นถึงความเกี่ยวข้องกันระหว่างความเก็บประจุ และความต่างศักย์ที่ใช้ไบแอส ความเก็บประจุของขั้วต่อจะชี้ให้เห็นถึงความเร็วในการตอบสนองของโฟโตไดโอดนั้นๆ

2.4.4. Rise/Fall Time และ การตอบสนองทางความถี่ $t_r/t_f/f_{3dB}$

Rise time และ Fall time ในโฟโตไดโอดจะถูกนิยามให้หมายถึงเวลาที่สัญญาณเปลี่ยนแปลงเพิ่มขึ้นจาก 10%-90% หรือลดลงจาก 90%-10% โดยอาจสามารถหาพารามิเตอร์ดังกล่าวได้โดยใช้ค่าการตอบสนองทางความถี่ซึ่งก็คือความถี่ที่ทำให้เอาต์พุตของโฟโตไดโอดลดลง 3dB ซึ่งสามารถหาค่าโดยประมาณได้จาก:

$$t_r \cong \frac{0.35}{f_{3dB}} \quad (3)$$

มี 3 องค์ประกอบที่ใช้ในการนิยามถึงค่าการตอบสนองทางเวลาของโฟโตไดโอด:

1. t_{DRIFT} , คือค่าการเก็บสะสมทางเวลาของพาหะ(carriers) ใน depleted region ของโฟโตไดโอด
2. $t_{DIFFUSED}$, คือค่าการเก็บสะสมทางเวลาของพาหะใน undepleted region ของโฟโตไดโอด
3. t_{RC} , คือค่าคงที่ทางเวลาของวงจรไดโอด

$t_{RC} = 2.2RC$ เมื่อ R คือผลรวมของการอนุกรมความต้านทานของไดโอด และความต้านทานของโหลด

$(RS+RL)$ และ C คือ ผลรวมของขั้วต่อไดโอดและ stray capacitance (C_j+C_s) โดย C_j จะขึ้นอยู่กับพื้นที่การแพร่กระจายของโฟโตไดโอด และการทำไบแอสกลับ(Equation 2) Rise time ที่เร็วกว่าย่อมมาจากพื้นที่การ-

แพร่กระจายที่เล็กกว่า และการทำไบอัสกลับด้วยพลังงานที่มากกว่า นอกจากนั้น C_s สามารถทำให้ลดลงได้ โดยการออกแบบทางอิเล็กทรอนิกส์ของโฟโตไดโอด

$$t_R = \sqrt{t_{\text{DRIFT}}^2 + t_{\text{DIFFUED}}^2 + t_{\text{RC}}^2} \quad (4)$$

โดยทั่วไป ใน photovoltaic mode นั้น rise time จะขึ้นอยู่กับเวลาที่ใช้ในการแพร่กระจายในพื้นที่แพร่กระจายที่มีขนาดน้อยกว่า 5mm^2 และ RC time constant ในพื้นที่แพร่กระจายที่มีขนาดใหญ่กว่าในทุกย่านความยาวคลื่น ใน photoconductive mode (ทำการไบอัสกลับให้แก่วงจร) หากโฟโตไดโอดเป็นแบบ fully depleted ยกตัวอย่างเช่น high speed series ตัวแปรสำคัญที่ส่งผลต่อโฟโตไดโอดคือ drift time หากโฟโตไดโอดเป็นแบบ non-fully depleted ทั้งสามตัวแปรข้างต้นจะส่งผลต่อเวลาในการตอบสนองของโฟโตไดโอด

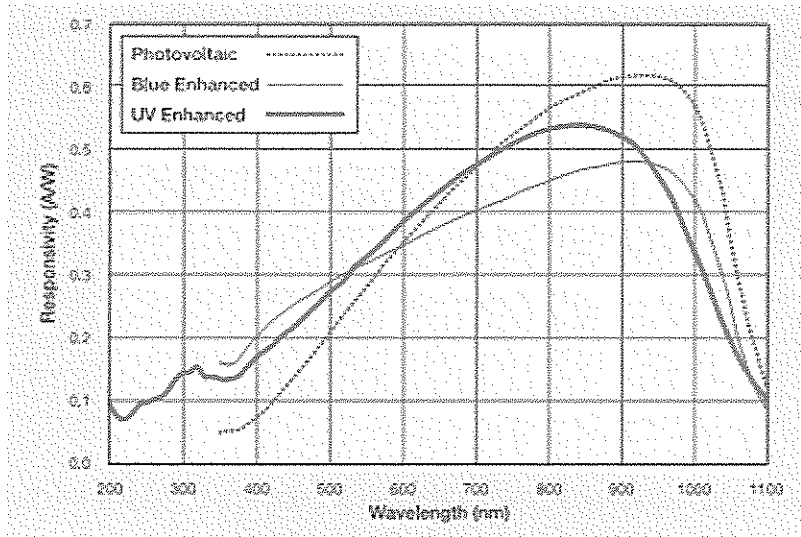
2.5 คุณสมบัติเฉพาะทางแสง

Responsivity, R_λ

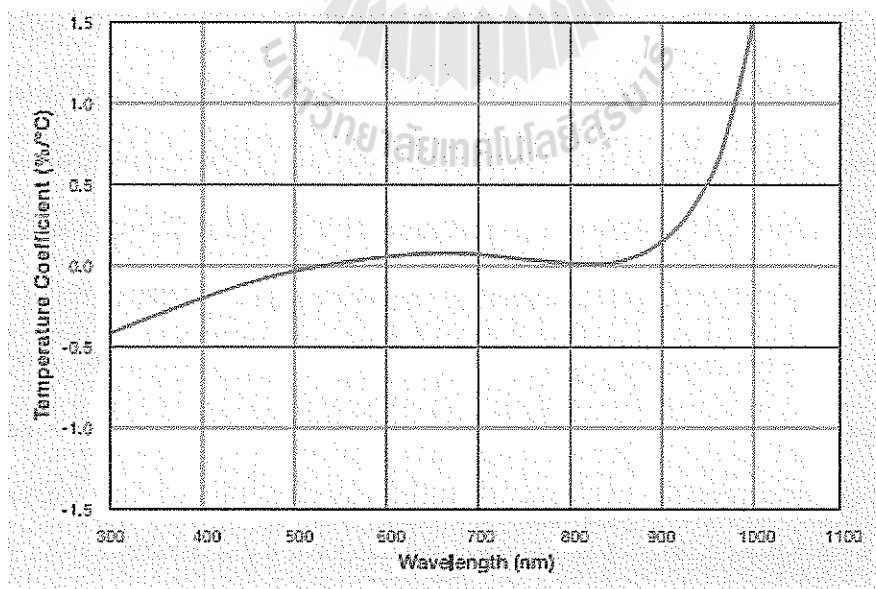
ค่าการตอบสนอง (Responsivity) ของซิลิคอนโฟโตไดโอดจะคำนวณได้จากความไวต่อการตอบสนองทางแสงซึ่งจะถูกนิยามให้เป็นสัดส่วนของ I_p ต่อ พลังงานของแสง P :

$$R_\lambda = \frac{I_p}{P} \quad (5)$$

หากจะอธิบายอีกทางหนึ่ง ค่าการตอบสนอง (Responsivity) คือค่าที่บ่งบอกประสิทธิภาพของการแปลงพลังงานแสงให้กลายเป็นกระแสไฟฟ้านั่นเอง โดยค่าดังกล่าวจะขึ้นอยู่กับความยาวคลื่น (รูปที่ 2.5), การไบอัสกลับ และอุณหภูมิ



รูปที่ 2.5 กราฟแสดงความสัมพันธ์ระหว่างการตอบสนองทางสเปกตรัมกับความยาวคลื่น ค่าการตอบสนองจะค่อยๆเพิ่มขึ้นด้วยการเพิ่มการไบอัสกลับเนื่องจากการเพิ่มขึ้นของประสิทธิภาพการทำงานในโฟโตไดโอด การเปลี่ยนแปลงของค่าการตอบสนองยังขึ้นอยู่กับอุณหภูมิอีกด้วยดังแสดงให้เห็นในรูปที่ 2.6 ซึ่งแสดงให้เห็นถึงการเพิ่มขึ้น และลดลงของ band gap อันเนื่องมาจาก การเพิ่มขึ้น และลดลงของอุณหภูมิ การตอบสนองทางสเปกตรัมจะเปลี่ยนแปลงตามความยาวคลื่นอีกด้วย อย่างไรก็ตาม ความเปลี่ยนแปลงที่เกี่ยวข้องกับค่าการตอบสนอง สามารถถูกทำให้ลดลงได้อย่างน้อย 1% ตามกฎดังกล่าว



รูปที่ 2.6 กราฟแสดงความสัมพันธ์ระหว่างอุณหภูมิกับความยาวคลื่น

2.5.1 Quantum Efficiency, Q.E.

Quantum efficiency คือปริมาณโฟตอนส่วนหนึ่งที่ทำให้เกิดกระแสโฟโต(photo current) :

$$\begin{aligned} \text{Q. E.} &= \frac{R_{\lambda \text{ observed}}}{R_{\lambda \text{ Ideal}}} \quad (6) \\ &= R_{\lambda} \frac{hc}{\lambda q} \\ &= 1240 \frac{R_{\lambda}}{\lambda} \end{aligned}$$

เมื่อ Planck constant $h=6.63 \times 10^{-34}$ J-s , ความเร็วแสง $c=3 \times 10^8$ m/s , electron charge $q=1.6 \times 10^{-19}$ C , R_{λ} ค่าการตอบสนองใน A/W และ λ คือ ความยาวคลื่นในหน่วยของ nm

2.5.2 Non-Uniformity

ความไม่เป็นรูปแบบเดียวกัน(Non-Uniformity) ของการตอบสนองคือการเปลี่ยนแปลงของค่าการตอบสนองที่เกิดขึ้นบนพื้นผิวของพื้นที่ทำงาน(active area) บนโฟโตไดโอดที่เกิดจากจุดแสงเพียงเล็กน้อย โดยความไม่เป็นรูปแบบเดียวกันจะแปรผกผันกับขนาดของจุดแสงเช่น ความไม่เป็นรูปแบบเดียวกันที่มากกว่าจะเกิดจากจุดแสงที่เล็กกว่า

2.5.3 Non-Linearity

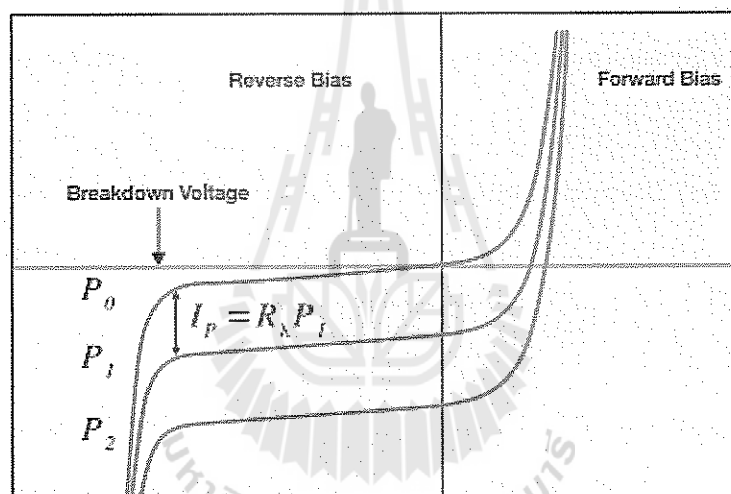
ซิลิคอนโฟโตไดโอดจะถูกพิจารณาให้มีความเป็นเชิงเส้นหาก photocurrent ที่ถูกสร้างขึ้นมา มีการเพิ่มขึ้นอย่างเป็นเชิงเส้นกับพลังงานแสง photocurrent Linearity จะหาได้จากการหาค่าการเปลี่ยนแปลงเพียงเล็กน้อยของ photocurrent อันเนื่องมาจากการเปลี่ยนแปลงเพียงเล็กน้อยของพลังงานแสง Non-Linearity คือการเปลี่ยนแปลงที่เป็นไปตามสัดส่วนของการเปลี่ยนแปลงของ photocurrent ไปเป็นพลังงานแสง เช่น $\Delta I / \Delta P$ ซึ่งอาจสามารถอธิบายได้ว่า Linearity คือความสัมพันธ์กันของค่าการตอบสนองต่อช่วงพลังงานของแสง ขอบเขตที่ต่ำกว่าของ photocurrent linearity จะถูกนิยามให้เป็น กระแสรบกวน(noise current) และ ขอบเขตที่สูงกว่าจะนิยามให้เป็น series resistance และ load resistance เมื่อ photocurrent เพิ่มขึ้น เริ่มแรก non-linearity จะค่อยๆเพิ่มขึ้นตามการเพิ่มขึ้นของ photocurrent และในท้ายที่สุด ที่ระดับอิ่มตัว photocurrent จะมีค่าคงที่เมื่อเพิ่มพลังงานแสงขึ้น โดยทั่วไป การเปลี่ยนแปลงระดับของการสร้าง photocurrent จะเป็นเช่นเดียวกับการเปลี่ยนแปลงของพลังงานแสง คือ เปลี่ยนแปลงน้อยกว่าที่ระดับกระแสสูงกว่า เมื่อตัวตรวจจับแสงมีความเป็น non-linearity ช่วงของ linearity สามารถขยายขึ้นเล็กน้อยโดยการเพิ่มการไบอัสกลับให้แก่โฟโตไดโอด

2.6 คุณลักษณะเฉพาะของ I-V

คุณลักษณะเฉพาะของ I-V ของโฟโตไดโอดในขณะไม่มีแสงจะมีลักษณะเหมือนกับ rectifying diode เมื่อทำการไบอัสตรงให้กับโฟโตไดโอดจะมีการเพิ่มขึ้นของกระแสอย่างรวดเร็ว เมื่อทำการไบอัสกลับจะทำให้กระแสอิ่มตัวย้อนกลับ(reverse saturation current)บางส่วนหายไปซึ่งจะขึ้นอยู่กับ dark current ดังนี้:

$$I_D = I_{SAT} \left(e^{\frac{qV_A}{k_B T}} - 1 \right) \quad (7)$$

เมื่อ I_D คือ dark current ของโฟโตไดโอด, I_{SAT} คือกระแสอิ่มตัวย้อนกลับ, q คือ electron charge, V_A คือความต่างศักย์ที่ใช้ไบอัส, $k_B = 1.38 \times 10^{-23}$ J/K คือ Boltzmann Constant และ T คือ absolute temperature ($273K = 0^\circ C$)



รูปที่ 2.7 กราฟแสดงคุณลักษณะเฉพาะของ I-V

ความสัมพันธ์ดังกล่าวสามารถแสดงให้เห็นดังรูปที่ 2.7 โดยสามารถนิยามสถานะต่างๆ ได้ดังนี้

- $V=0$ กระแสที่เกิดขึ้นจะเปลี่ยนเป็นกระแสอิ่มตัวย้อนกลับ
- $V=+V$ กระแสจะเพิ่มขึ้นอย่างรวดเร็ว ซึ่งอาจเรียกสถานะนี้ว่า forward bias mode
- $V=-V$ เมื่อทำการไบอัสกลับให้แก่โฟโตไดโอด จะทำให้กระแสมีคุณสมบัติดังรูปที่ 2.8

การอธิบายโฟโตไดโอดด้วยการแพร่กระจายของแสง การเคลื่อนที่ของ I-V Curve เนื่องจาก photocurrent (I_p) โดย:

$$I_{TOTAL} = I_{SAT} \left(e^{\frac{qV_A}{k_B T}} - 1 \right) - I_p \quad (8)$$

เมื่อ I_p คือ photocurrent ตามสมการ (5)

เมื่อเพิ่มการไบอัสกลับให้กับโฟโตไดโอด จะมีการเพิ่มขึ้นของ photocurrent ตามไปด้วย การทำไบอัสกลับในจุดที่เรียกว่า breakdown Voltage จะทำให้เกิดการไบอัสกลับสูงสุด หากไบอัสกลับต่ำกว่าจุดดังกล่าว โฟโตไดโอดจะทำงานตามปกติ(อาจเรียกว่า maximum reverse voltage) Breakdown voltage ในโฟโตไดโอดแต่ละตัวจะไม่เหมือนกัน แต่สามารถหาค่าได้จากการป้อน photocurrent $10 \mu A$ เข้าไป

2.7 Noise

ในโฟโตไดโอด มีสัญญาณรบกวน(Noise)อยู่สองประเภทคือ Shot noise และ Johnson noise

2.7.1 Shot noise

Shot noise จะเกิดขึ้นจากความผันผวนที่เกิดขึ้นภายใน photocurrent และ dark current โดยสามารถหาค่าขนาดของมันได้จาก:

$$I_{sn} = \sqrt{2q(I_p + I_D)\Delta f} \quad (9)$$

เมื่อ electron charge $q=1.6 \times 10^{-19} C$, I_p คือ photogenerated current, I_D คือ photodetector dark current และ Δf คือ แบนด์วิดธ์ของสัญญาณรบกวน โดย Shot noise จะมีผลกระทบต่อแหล่งกำเนิดเมื่อทำงานใน photoconductive mode

2.7.2 Thermal หรือ Johnson noise

ใน Shunt resistance ของ photodetector จะมี Johnson noise เป็นส่วนประกอบอยู่ด้วยซึ่งเกิดขึ้นจากการกำเนิดความร้อนของพาหะ โดยสามารถหาค่าขนาดของมันได้จาก:

$$I_{jn} = \sqrt{\frac{4k_B T \Delta f}{R_{SH}}} \quad (10)$$

เมื่อ Boltzmann Constant $k_B=1.38 \times 10^{-29} J/K$, T คือ absolute temperature (273 K), Δf คือ แบนด์วิดธ์ของสัญญาณรบกวน, R_{SH} คือ shunt resistance ของโฟโตไดโอด โดย Johnson noise จะมีผลกระทบต่อ current noise ใน photovoltaic mode

ข้อควรจำ: ตัวต้านทานทั้งหลายจะมี Johnson noise เป็นองค์ประกอบอยู่ด้วย Johnson noise ที่เกิดขึ้นจาก Shunt resistance จะทำให้เกิด noise current ขนาดใหญ่ขึ้น

2.7.3 Total noise

Noise current ที่เกิดขึ้นทั้งหมดในโฟโตไดโอดสามารถหาค่าได้จาก:

$$I_{tn} = \sqrt{I_{sn}^2 + I_{jn}^2} \quad (11)$$

2.7.4 Noise Equivalent Power (NEP)

Noise Equivalent Power คือกลุ่มของพลังงานแสงบน photodetector ซึ่งให้กำเนิด photocurrent ปริมาณเท่ากับ noise current โดยหาได้จาก :

$$NEP = \frac{I_{tn}}{R_\lambda} \quad (12)$$

เมื่อ R_λ คือค่าการตอบสนองใน A/W, I_{tn} คือ ผลรวมสัญญาณรบกวนของ photodetector ค่าของ NEP สามารถเปลี่ยนแปลงได้จาก 10^{-11} W/√Hz ในโฟโตไดโอดที่มีพื้นที่ทำงานขนาดใหญ่ ไปจนถึง 10-15 W/√Hz ในโฟโตไดโอดที่มีพื้นที่ทำงานขนาดเล็ก

2.8 ผลกระทบอันเนื่องมาจากอุณหภูมิ

การเปลี่ยนแปลงของอุณหภูมิจะสร้างผลกระทบต่อค่าคุณลักษณะเฉพาะทั้งหมดของโฟโตไดโอด ไม่ว่าจะเป็น Shunt resistance, dark current, breakdown voltage, responsivity และ junction capacitance เป็นต้น

2.8.1 Shunt Resistance and Dark Current:

โฟโตไดโอดมีกระแสสำคัญอยู่สองชนิดที่ทำให้เกิด dark current และ shunt resistance กระแสแรก ได้แก่ Diffusion current ซึ่งมีอิทธิพลต่อ photovoltaic mode โดยพิจารณาตาม Shunt resistance ซึ่งจะเปลี่ยนแปลงตามการเปลี่ยนแปลงของอุณหภูมิ ใน photoconductive mode นั้น drift current จะเปลี่ยนเป็น dark current และแปรผันตรงกับอุณหภูมิ ดังนั้น การเปลี่ยนแปลงของอุณหภูมิจะส่งผลกระทบต่อโฟโตไดโอดใน photovoltaic mode มากกว่าใน photoconductive mode ใน photoconductive mode นั้นจะสามารถประมาณค่าคร่าวๆของ dark current ได้โดยการเพิ่มค่าขึ้นเป็นสองเท่าในการเพิ่มอุณหภูมิขึ้นทุกๆ 10°C และใน photovoltaic mode นั้น shunt resistance จะเพิ่มขึ้นสองเท่าโดยประมาณในการลดอุณหภูมิลงทุกๆ 6°C ส่วนค่าที่แน่นอนนั้นจะขึ้นอยู่กับปัจจัยอื่นๆที่เข้ามาเกี่ยวข้องด้วยยกตัวอย่างเช่น การไบอัสกลับ ,ค่าความต้านทานของพื้นผิว เป็นต้น

2.8.2 BreakDown Voltage:

สำหรับอุปกรณ์โฟโตไดโอดซึ่งมีพื้นที่ทำงานเล็ก ๆ นั้น สามารถนิยามให้ breakdown voltage เป็นแรงดันไฟฟ้าที่ทำให้ dark current มีค่า 10 A ได้ เมื่อ dark current เพิ่มขึ้นตามอุณหภูมิ ดังนั้นจึงทำให้ breakdown voltage ลดลงตามการเพิ่มขึ้นของอุณหภูมิ

2.8.3 Responsivity

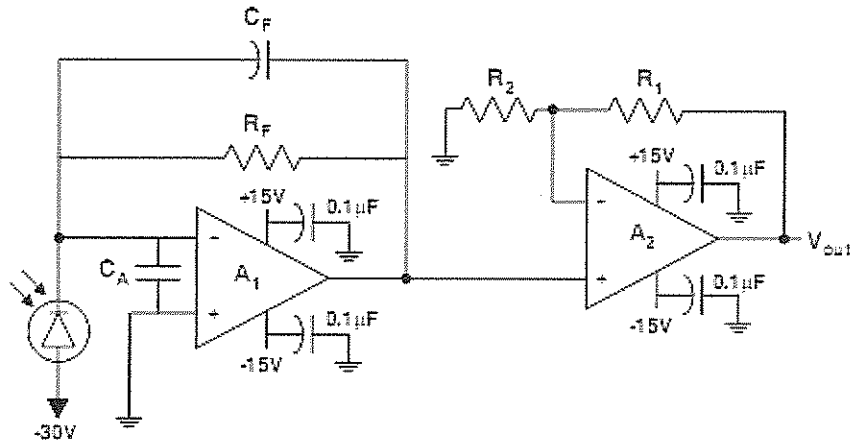
ผลการทบทของอุณหภูมิต่อค่าการตอบสนองนั้น ได้อธิบายไว้แล้วในส่วนของ Responsivity

2.8.4 BIASING

สัญญาณที่เกิดขึ้นในโฟโตไดโอดสามารถหาค่าได้จากแรงดันไฟฟ้า หรือกระแสไฟฟ้าที่เกิดขึ้น การหาค่าเหล่านี้กระแสไฟฟ้าที่ไหลจะแสดงให้เห็นถึงความเป็นเชิงเส้น , การชดเชย และประสิทธิภาพของแบนด์-วิทซ์ photocurrent ที่ถูกสร้างขึ้นจะเป็นไปตามสัดส่วนของพลังงานแสง และมันจะถูกแปลงเป็นแรงดันไฟฟ้าโดยใช้การกำหนดค่า transimpedance โฟโตไดโอดสามารถทำงานได้ในขณะที่ทำการไบอัสกลับ หรือไม่ไบอัสกลับก็ได้ขึ้นอยู่กับความต้องการในการใช้งาน โดยจะเรียกว่าเป็น Photoconductive mode และ Photovoltaic mode

2.8.5 Photoconductive mode(PC)

การใช้ประโยชน์การไบอัสกลับเพื่อเพิ่มความเร็วของการตอบสนอง และความเป็นเชิงเส้นของตัวอุปกรณ์ อันเนื่องมาจากการเพิ่มขึ้นของความกว้างของ depletion region และผลจากการลดลงของ junction capacitance แต่อย่างไรก็ตาม ในการทำไบอัสกลับก็จะก่อให้เกิดการเพิ่มขึ้นของ dark current และ noise current ยกตัวอย่างเช่น แสงระดับต่ำจะมีอัตราการตอบสนองที่รวดเร็วเมื่อทำงานใน photoconductive mode ดังรูป figure 8 ในลักษณะดังกล่าว detector จะถูกไบอัสเพื่อลด junction capacitance ลงเพื่อที่จะทำให้ noise และ rise time ลดลง การขยายสัญญาณเป็นสองช่วงจะถูกนำมาใช้ในตัวอย่างนี้เมื่อต้องการอัตราการขยายที่สูงในช่วงแบนด์วิทซ์กว้าง การขยายสองช่วงนี้จะทำได้โดยการเพิ่ม transimpedance pre-amp เพื่อแปลงกระแสให้เป็นความต่างศักย์ และเพิ่ม non-inverting amplifier เพื่อใช้ขยายความต่างศักย์ อัตราการขยายสัญญาณ และแบนด์วิทซ์ (f_{3dBMAX}) จะสามารถหาได้จาก R_f ในสมการที่ (13) และ (14) อัตราการขยายของช่วงที่สองจะหาค่าโดยประมาณได้จาก $1+R1/R2$ feedback capacitor (C_f) จะเป็นตัวจำกัดการตอบสนองทางความถี่ และหลีกเลี่ยงค่าการขยายสูงสุด (gain peaking)



รูปที่ 2.8 แสดงวงจร Low Light Level / Wide Bandwidth

$$f_{3dBMax} [Hz] = \sqrt{\frac{GBP}{2\pi R_F (C_j + C_F + C_A)}} \quad (13)$$

เมื่อ GBP คือ Gain Bandwidth Product ของภาคขยายสัญญาณ (A_1) และ C_A คือ amplifier input capacitance

$$\text{Gain (V/W)} = \frac{V_{OUT}}{P} = R_F \left(1 + \frac{R_1}{R_2} \right) R_\lambda \quad (14)$$

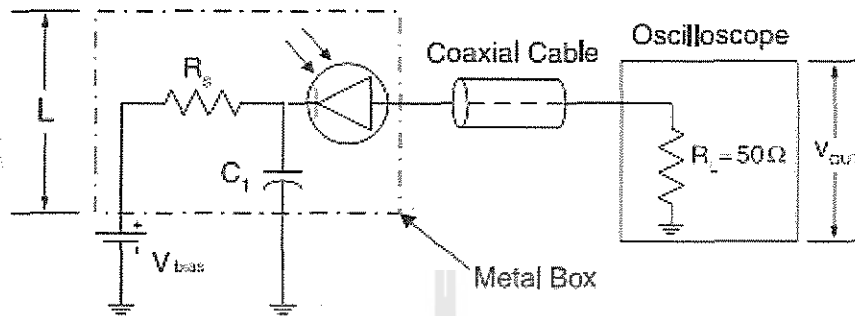
ในการใช้งานที่ความเร็วต่ำแต่ต้องการอัตราขยายที่สูงเช่น มากกว่า 10MΩ สามารถทำได้โดยการเพิ่มค่า R_F โดยไม่ต้องเข้าสู่การขยายช่วงที่สอง

2.9 องค์ประกอบทั่วไปที่ถูกลำเอียงใช้ในการกำหนดค่ามีดังนี้

Amplifier:	OPA-637, OPA-686, OPA687, or similar
R_F :	1 to 10 kΩ Typical, depending on C_j
R_1 :	10 to 50 kΩ
R_2 :	0.5 to 10 kΩ
R_F :	0.2 to 2 pF

ในการใช้งานที่ความเร็วสูงนั้น การคำนวณค่าของแสงที่มีระดับสูงนั้นจะใช้วิธีการที่แตกต่างออกไป ตัวอย่างพื้นฐานที่สุดคือการหาค่าความกว้าง pulse ใน short pulse gas laser, solid state laser diode หรือกรณีอื่นๆที่เป็น short pulse light source เหมือนกัน เอาท์พุทของโฟโตไดโอดสามารถต่อตรงไปยัง oscilloscope

(รูปที่ 2.9) หรือนำไปจ่ายให้กับ response amplifier ได้ เมื่อต่อเข้ากับ oscilloscope จะสามารถตั้งค่าแบนด์วิธของสโคปให้วัดค่าความกว้าง pulse ของแหล่งกำเนิดแสง เพื่อให้ได้ค่า S/N (signal to noise ratio) สูงสุด ในการนี้ต้องใช้การไบอัสด้วยความต่างศักย์ขนาดใหญ่ ควรต่อ protection diode ที่อินพุทของ oscilloscope ขวางอินพุท และกราวด์

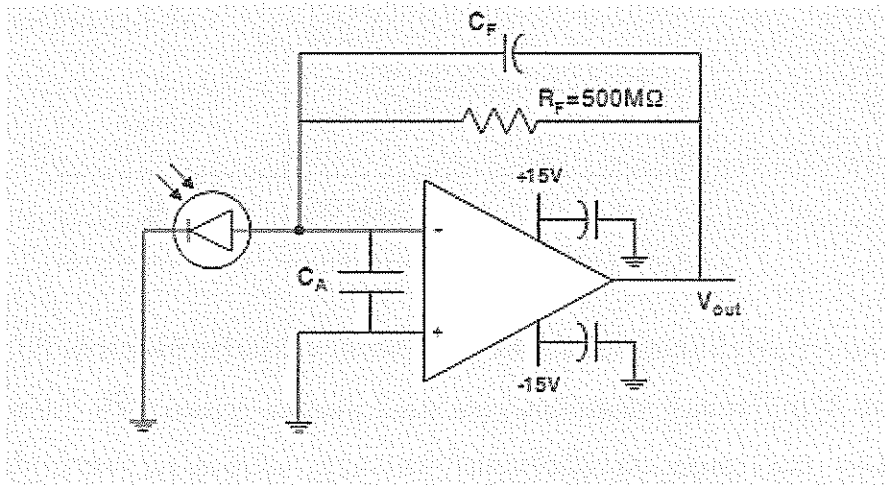


รูปที่ 2.9 แสดงวงจร High Light Level / High Speed Response

เพื่อหลีกเลี่ยง ringing ในสัญญาณเอาต์พุท สายสัญญาณระหว่าง detector และ oscilloscope ควรจะสั้น (น้อยกว่า 20 cm.) และมี load resistance (R_L) 50 โอห์ม โฟโตไดโอดควรติดตั้งอยู่ในกล่องโลหะ เพื่อให้ได้ระยะทางที่สั้นระหว่าง detector และ capacitor และระหว่าง detector และสายโคแอกเซียล (coaxial cable) กล่องโลหะควรจะยึดติดกับ capacitor (C_1) ด้วย lead length (L) น้อยกว่า 2 cm. เมื่อ $RLC_1 > 10T$ (T คือ ความกว้าง pulse ใน 1 วินาที) R_S จะอยู่ในค่า $R_S < V_{BIAS} / 10I_{PDC}$ เมื่อ I_{PDC} คือ DC photocurrent แบนด์วิธถูกนิยามให้เป็น $0.35 / T$ และต้องทำการไบอัสกลับด้วยความต่างศักย์อย่างน้อย 10V จะสังเกตว่าไม่สามารถไบอัสด้วยความต่างศักย์มากกว่า reverse voltage สูงสุดของโฟโตไดโอดได้

2.10 Photovoltaic Mode (PV)

Photovoltaic mode จะถูกนำมาใช้เมื่อโฟโตไดโอดถูกนำมาใช้ในงานที่มีความถี่ต่ำ (น้อยกว่า 350 kHz) ซึ่งรู้จักกันดีในช่วง ultra low light level นอกจากนั้น ในการใช้งานโดยทั่วไป โฟโตไดโอดในโหมดนี้ จะมีการเปลี่ยนแปลงค่าการตอบสนองน้อยมากเมื่ออุณหภูมิเปลี่ยนแปลง ตัวอย่าง ultra low light level/low-speed ดังแสดงในรูปที่ 2.10



รูปที่ 2.10 แสดงวงจร Ultra low level light / Low Speed

ในตัวอย่างนี้ จะพิจารณาค่าความต้านทานย้อนกลับ(resistance feedback)ขนาดใหญ่ของรีซิสเตอร์(R_F) ของการใช้งานอินพุทของ amplifier ชนิด FET โดย detector จะไม่ได้รับการไบอัสเพื่อกำจัด noise current ที่จะเกิดขึ้น เอาท์พุททั้งหมดจะพิจารณาจากสมการ (15) noise current ของ op-amp จะหาค่าได้จาก R_F ในสมการ (16)

$$V_{OUT} = I_P \times R_F \quad (15)$$

$$I_N = \left[\frac{A_{rms}}{\sqrt{Hz}} \right] = \sqrt{\frac{4kT}{R_F}} \quad (16)$$

เมื่อ $k=1.38 \times 10^{-23}$ J/K และ T คืออุณหภูมิในหน่วย °K

เพื่อให้ระบบมีความเสถียร ต้องเลือกค่า C_F ที่ทำให้

$$\sqrt{\frac{GBP}{2\pi R_F (C_j + C_F + C_A)}} > \frac{1}{2\pi R_F C_f} \quad (17)$$

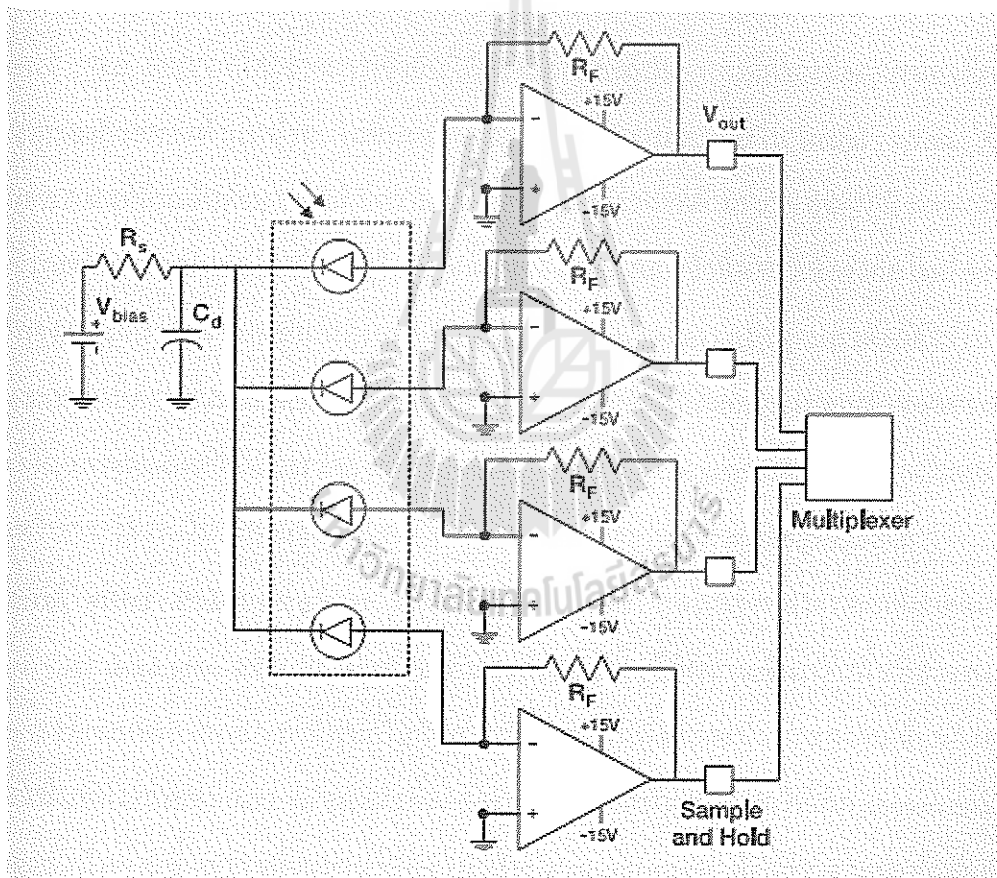
ถ้า operating bandwidth หลังจากทำให้ภาคขยายมีค่าสูงสุดคือ

$$f_{OP} [Hz] = \frac{1}{2\pi R_F C_f} \quad (18)$$

ข้อเสนอแนะในการกำหนดค่าคือ

Amplifier:	OPA-117 or similar
R_F :	500 M Ω

ตัวอย่างเหล่านี้เป็นตัวอย่างสำหรับโฟโตไดโอดเพียงตัวเดียว และสามารถนำไปประยุกต์ใช้งานกับ UDT sensors' monolithic หรือ common substrate linear array photodiode ประเภทอื่นๆ ได้ เอาท์พุทที่เกิดจากการขยายช่วงที่หนึ่งของ pre-amplifier สามารถนำไปต่อเข้ากับวงจร sample and hold circuit และ multiplexer ได้ รูปที่ 2.11 แสดงถึง block diagram องค์ประกอบต่างๆของวงจร



รูปที่ 2.11 แสดง Block Diagram องค์ประกอบต่างๆ ของวงจร

2.11 POSITION SENSING DETECTORS

ซิลิคอนโฟโตไดโอดถูกนำมาใช้งานอย่างแพร่หลายในการวัดพลังงานแสงเช่น bar-code reader, laser printers, medical imaging, spectroscopy และอื่นๆ ยังมีลักษณะการใช้งานแบบอื่นๆที่สามารถใช้

ประโยชน์จาก photodetector เช่น optical position sensor มีการใช้งานในลักษณะของ Position Sensing Detector หรือ PSD อย่างกว้างขวางเช่น การตรวจสอบการเคลื่อนไหวของดวงตา, การจับความเคลื่อนไหวแบบสามมิติ, แหล่งกำเนิดของแสง และการปรับตั้งแ่งมุมของกระจก หรือใช้ประโยชน์ในการช่วยปรับโฟกัสภาพอย่างรวดเร็ว และแม่นยำ ในระบบ optical เช่น microscope, เครื่องมือตั้งค่าแ่งมุมเครื่องจักร, การวิเคราะห์ความสั่นไหว และอื่นๆ สามารถหาตำแหน่งที่แน่นอนของบีมขนาดเล็กเป็นเศษเสี้ยวของไมครอนได้โดยใช้ PSD สามารถแบ่ง PSD ออกเป็นสองประเภทคือ segmented PSD's และ lateral effect PSD's

Segmented PSD's คือการแบ่งพื้นผิวของโฟโตไดโอดทั่วไปออกเป็นสองหรือสี่ส่วน(segment) (คำนวณค่าเป็นหนึ่งหรือสองมิติ,ตามลำดับ)โดยใช้ gab หรือ dead region ในการแบ่งแยก บีมแสงสมมาตรจะให้กำเนิด photocurrent ที่เท่ากันในทุกๆส่วนหากวัดที่จุดศูนย์กลาง ตำแหน่งสัมพัทธ์จะหาได้จากการหาค่ากระแสเอาท์พุทของแต่ละส่วน โดยตำแหน่งที่หาได้จะมีความละเอียดสูงกว่า $1\mu\text{m}$ และความแม่นยำมากกว่า lateral effect PSD's เนื่องจากการเข้ากันได้ของแต่ละองค์ประกอบมีค่าการตอบสนองที่เหนือกว่าเนื่องจากความละเอียดของตำแหน่งจะไม่ขึ้นอยู่กับ S/N ของระบบเหมือนใน lateral effect PSD's จึงสามารถตรวจหาแสงที่มีระดับต่ำมากๆ ได้ มีความขอดีเยี่ยมในเสถียรภาพด้านเวลา, อุณหภูมิ, และเวลาในการตอบสนองที่รวดเร็วซึ่งจำเป็นต่อการใช้งานในลักษณะของ pulse อย่างไรก็ตาม มีข้อจำกัดในเรื่องขีดจำกัดบางอย่างเช่นจุดของแสงต้องซ้อนทับกันบนทุกส่วน(segment)ตลอดเวลา และจุดแสงดังกล่าวไม่สามารถเล็กกว่า gab ระหว่างแต่ละส่วน segment) ได้ ซึ่งเป็นสิ่งสำคัญที่จะทำให้เกิดการกระจายตัวของความเข้มที่มีลักษณะเดียวกันของจุดแสงเพื่อให้การวัดค่าเป็นไปอย่างถูกต้องดังนั้น Segmented PSD's จึงเป็นอุปกรณ์ที่มีประสิทธิภาพในการใช้งานในลักษณะของแสงที่มีพลังงานน้อยมากๆและลำแสงอยู่ตรงกลาง

Lateral effect PSD's คือ planar diffused photodiode แบบมีองค์ประกอบเดียวต่อเนื่องโดยปราศจาก gab และ dead area PSD ประเภทนี้จะทำการอ่านจุดแสงที่ตัดผ่านเข้ามาในพื้นที่ทำงานโดยตรงทำงานด้วยการจัดการกับอนุภาคเอาท์พุทเป็นสัดส่วน โดยตรงกับทั้งตำแหน่ง และความเข้มของจุดแสงที่ตกกระทบบนพื้นที่ทำงานของ detector จุดแสงที่ตกกระทบบนพื้นที่ทำงานจะก่อให้เกิด photocurrent ขึ้นซึ่งจะไหลจากจุดกำเนิดผ่านชั้นความต้านทาน ไปยังหน้าสัมผัส photocurrent ดังกล่าวจะแปรผกผันกับความต้านทานระหว่างจุดกำเนิดแสง และหน้าสัมผัส เมื่ออินพุทของจุดแสงตกลงตรงกลางของอุปกรณ์อย่างเที่ยงตรงสัญญาณของกระแสที่เท่ากันจะถูกสร้างขึ้น หากเคลื่อนจุดแสงไปยังจุดต่างๆบนพื้นที่ทำงาน กลุ่มของกระแสที่ถูกสร้างขึ้นที่หน้าสัมผัสจะสามารถพิจารณาได้จากตำแหน่งแน่นอนของจุดแสง ณ เวลานั้นๆ สัญญาณไฟฟ้าเหล่านี้จะเป็นไปตามความสัมพันธ์ของตำแหน่งของจุดแสงจากจุดกึ่งกลาง ประโยชน์หลักของ lateral effect diode คือมันมี dynamic range ที่กว้างมาก มันสามารถใช้หาตำแหน่งของจุดแสงได้จนถึง

ขอบนอกสุดของเซนเซอร์ ไม่ขึ้นกับสถานะของจุดแสง และการกระจายตัวของความเข้มซึ่งเป็นผลกระทบต่อการอ่านตำแหน่งของ segmented diode อินพุตของบีมแสงอาจมีรูปร่างและขนาดที่หลากหลาย เนื่องจากตำแหน่งของจุดแสงจากจุดกลางจะเป็นตัวบอกถึงสัญญาณเอาต์พุตทางไฟฟ้า ซึ่งจะแปรผันตามความคาดเคลื่อนจากตำแหน่งกลาง ตัวอุปกรณ์สามารถแยกแยะตำแหน่งได้ละเอียดมากกว่า $5 \mu\text{m}$ โดยความละเอียดในการวัดจะขึ้นอยู่กับ S/N ของ detector/circuit อีกด้วย UDT Sensor สร้างขึ้นมาเป็น lateral effect PSD's สองชนิดคือ โครงสร้างแบบ Duo-Lateral และ Tetra-Lateral ซึ่งทั้งสองโครงสร้างนี้สามารถกำหนดองค์ประกอบได้หนึ่ง และสองมิติ ใน Duo-Lateral PSD's จะมีชั้นความต้านทานอยู่สองชั้น โดยจะอยู่ที่ด้านบนสุด และล่างสุดของโฟโตไดโอด photocurrent จะถูกแบ่งเป็นสองส่วนในแต่ละชั้น โครงสร้างประเภทนี้สามารถแยกแยะการเคลื่อนที่ของจุดแสงระยะน้อยกว่า $0.5 \mu\text{m}$ ได้และมีความคาดเคลื่อนในการวัดน้อยมาก สามารถตรวจจับจุดแสงได้จนถึงขอบของพื้นที่ทำงาน จึงสามารถนำมาใช้งานได้อย่างดีเยี่ยมในการหาตำแหน่งที่เป็นเชิงเส้นในพื้นที่ทำงาน ใน Tetra-Lateral PSD's มีชั้นความต้านทานเพียงชั้นเดียวซึ่ง photocurrent จะถูกแบ่งออกเป็นสองหรือสี่ส่วนเมื่อใช้องค์ประกอบหนึ่งหรือสองมิติตามลำดับ อุปกรณ์เหล่านี้จะเหมาะสมกับการใช้หาตำแหน่งที่ไม่เป็นเชิงเส้นที่ระยะทางที่ห่างออกมาจากจุดกึ่งกลางมีความคาดเคลื่อนในการวัดสูงกว่าชนิด Duo-Lateral

2.12 อภิธานศัพท์

Position Detection Error(PDE) หรือ Position non-linearity คือ ความผันแปรทางเรขาคณิตระหว่างตำแหน่งของจุดแสงที่แท้จริงกับตำแหน่งที่หาค่าได้ มันถูกหาได้จากมากกว่า 80%ของระยะการตรวจจับสำหรับ PSD's แบบหนึ่งมิติ และมากกว่า 64%ของพื้นที่ตรวจจับสำหรับ PSD's แบบสองมิติ สำหรับทุกการคำนวณ จุดศูนย์(zero point) คือจุดกึ่งกลางทางไฟฟ้าซึ่งเป็นจุดที่ทำให้ $I_1=I_2$ ค่าความคาดเคลื่อนสามารถคำนวณได้จากสมการนี้:

$$PDE[\mu\text{m}] = \left(\frac{I_2 - I_1}{I_2 + I_1} \right) L - X \quad (19)$$

เมื่อ I_1 และ I_2 คือ photocurrent ที่ขอบของ PSD, L คือพื้นที่ตรวจจับครึ่งระยะทางในหน่วย μm และ X คือจุดตกกระทบที่แท้จริงของจุดแสงที่คาดเคลื่อนไปจากจุดกึ่งกลางทางไฟฟ้าในหน่วยของ μm Percentage Position Non-linearity จะหาได้จากการแยกค่าความคาดเคลื่อนของตำแหน่งที่ตรวจหาได้จากระยะทั้งหมดของพื้นที่ตรวจจับ

Interelectrode Resistance คือค่าความต้านทานระหว่างขอบของหน้าสัมผัสทั้งสองขอบในหนึ่งแกน

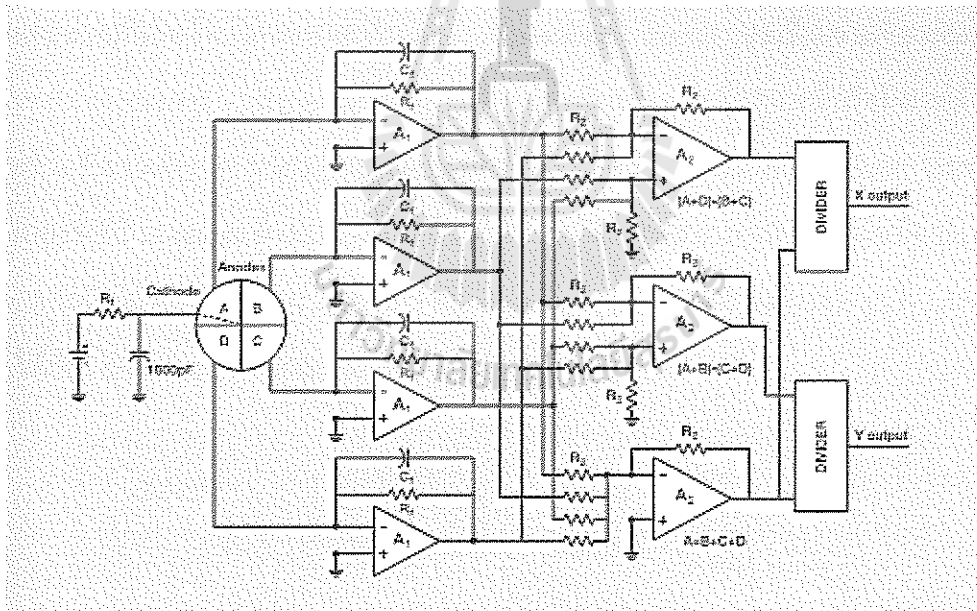
Position Detection Thermal Drift คือตำแหน่งที่เลื่อนไปเมื่ออุณหภูมิเปลี่ยนไป มันจะเปลี่ยนไปยังตำแหน่งที่ถูกแยกออกจากกระยะทั้งหมด มันถูกนิยามให้อยู่ภายใน 80%ของระยะหรือ64%ของพื้นที่สำหรับ PSD's แบบสองมิติ

Position Resolution ถูกนิยามให้เป็นการเคลื่อนที่ของจุดแสงด้วยระยะที่เล็กที่สุดในพื้นที่ทำงานของ detector ที่สามารถตรวจจับได้ ความละเอียดดังกล่าวถูกจำกัดโดย S/N ของระบบ มันขึ้นอยู่กับความเข้มของแสง,detector noiseและ แบนด์วิธที่ทางอิเล็กทรอนิกส์ โดย UDT Sensor lateral effect PSD's จะมีความละเอียดทางตำแหน่งที่มากกว่า 1 ใน 10 ล้าน

2.13 การคำนวณหาตำแหน่ง

Segmented PSD's

จากรูปที่ 2.12 แสดงถึงวงจรทั่วไปซึ่งถูกใช้งานร่วม UDS sensors segmented photodiode



รูปที่ 2.12 แสดงวงจรที่ถูกใช้งานร่วม UDS Sensor Segmented Photodiode

ตำแหน่ง X และ Y ของจุดแสงจะถูกพิจารณาร่วมกับจุดกึ่งกลางบน quadrant photodiode โดย:

$$X = \frac{(A+D)-(B+C)}{A+B+C+D} \quad (20)$$

$$X = \frac{(A+D)-(C+D)}{A+B+C+D} \quad (21)$$

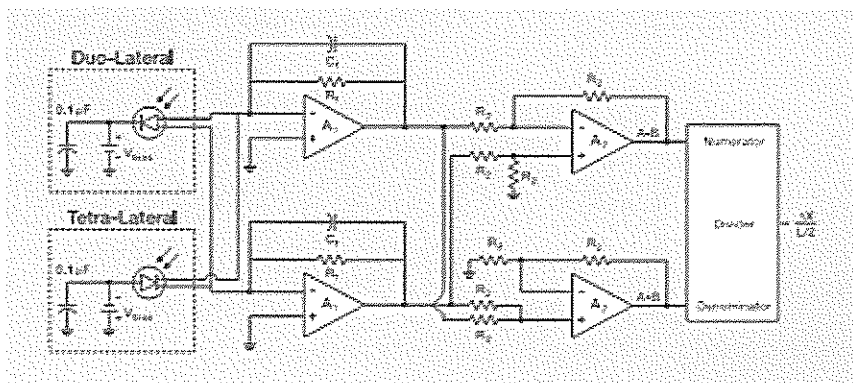
เมื่อ A,B,C และ D คือ photocurrent ที่ทำได้จากแต่ละส่วน องค์ประกอบของวงจรที่แนะนำนี้จะต้องมีความเฉพาะเจาะจง แต่อย่างไรก็ตาม องค์ประกอบที่ถูกใช้งานอย่างกว้างขวางคือ

Amplifiers A_1 and A_2 :	OPA-37 or similar
Divider:	AD-534 or similar
R_F and R_2 :	10 k Ω to 10 M Ω
C_F :	$1/(2\pi R_F f)$

วงจรเดียวกันนี้สามารถใช้งานกับการหาค่าในหนึ่งมิติ(bi-cell) ได้อีกด้วย

Lateral Effect PSD's

ผลกระทบที่เกิดขึ้นในการหาค่าใน one dimensional lateral จะเป็นเช่นเดียวกับที่เกิดขึ้นในโครงสร้างของ duo-lateral และ tetra-lateral เนื่องจากทั้งสองชนิดมีสองหน้าสัมผัสอยู่ด้านบน และหน้าสัมผัสร่วมที่ส่วนล่างสุด อุปกรณ์ชนิด tetra-lateral จะมีหน้าสัมผัสร่วมเป็น anode และหน้าสัมผัสด้านบนทั้งสองจะเป็น cathode ซึ่งจะทำให้มันเป็นตัวให้กำเนิดกระแสบวก ในอุปกรณ์ชนิด duo-lateral นั้นจะมี anode สองตัวที่ด้านบนสุด และ cathode ร่วม(common cathode)อยู่ที่ส่วนล่างสุด รูปที่ 2.13 แสดงวงจรที่สร้างขึ้นให้เป็น one-dimensional lateral PSD's



รูปที่ 2.13 แสดงวงจร one-dimensional lateral PSD's

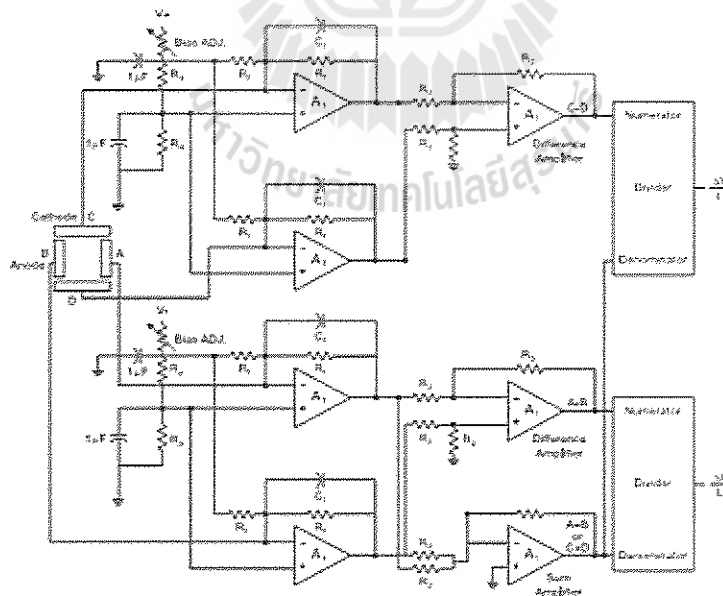
ในวงจรดังกล่าว เอาท์พุทจากช่วงแรกจะถูกรวมและหักล้างไปในช่วงที่สอง และจะถูกแยกออกมาโดย divider ในช่วงสุดท้าย ผลรวม, ผลการหักล้าง และผลการแยกสามารถแสดงออกมาได้โดยซอฟต์แวร์ดังนี้

$$X = \frac{A-B}{A+B} \quad (22)$$

องค์ประกอบเช่นกันนี้ถูกนำไปใช้กับ segmented diode ได้โดยสามารถปรับค่า R_2 จาก $1k\Omega$ ถึง $100k\Omega$ สำหรับการใช้งานที่ต้องการความเร็วสูงเราสามารถทำไบอัสกลับให้กับจุดเชื่อมต่อ(junction) ด้วยกำลังขยายเล็กๆ(R_p)ได้ สำหรับการใช้งานที่ความถี่ต่ำนั้น แม้ว่าโฟโตไดโอดสามารถไม่ต้องทำการไบอัส และอัตราขยาย R_f สามารถสูงได้ถึง $100M\Omega$ ค่าประจวบย้อนกลับ(feedback capacitor)ที่สร้างเสถียรภาพให้กับความถี่จะขึ้นอยู่กับอัตราขยายและสามารถเปลี่ยนแปลงได้ตั้งแต่ $1pF$ ถึง $10 \mu F$ อัตราขยายของ amplifier ช่วงแรกคือ $I_p R_f$ และอัตราขยายในช่วงที่สองก็เป็นเช่นเดียวกัน

Two Dimensional Duo-lateral PSD's

Two Dimensional Duo-lateral PSD's จะประกอบด้วย anode สองตัวที่ด้านบนสุด และ cathode สองตัวที่ด้านหลังพื้นผิวของ โฟโตไดโอดเพื่อใช้หาตำแหน่งในทิศทางที่ต่างกันสองทิศทาง มันจะทำการอ่านตำแหน่งอย่างต่อเนื่องบนพื้นที่ทำงานซึ่งจะทำให้มันมีความเที่ยงตรงมากกว่า tetra-lateral PSD's รูปที่ 2.14 แสดงให้เห็นถึงวงจรของ Two Dimensional Duo-lateral PSD's



รูปที่ 2.14 แสดงวงจร Two Dimensional Duo-lateral PSD's

สำหรับการใช้งานที่ต้องการความเร็วสูงนั้น จะต้องทำการไบอัสตรงให้กับขั้ว cathode ขณะที่ต้องทำการไบอัสกลับให้กับขั้ว anode ซึ่งจะทำให้ระยะของการไบอัสขยายออกแต่จะถูกจำกัดด้วยค่า reverse voltage สูง อุปกรณ์ที่เลือกใช้ในการต่อวงจรจะเป็นเช่นเดียวกับวงจร one-dimensional PSD's ค่าเอาต์พุตหาได้จาก

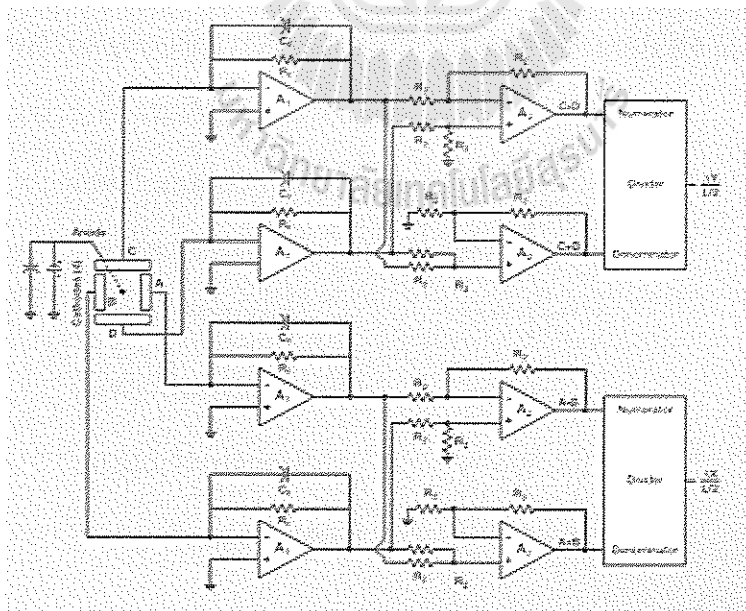
$$X = \frac{A-B}{A+B} \quad (23)$$

$$Y = \frac{C-D}{C+D} \quad (24)$$

Tetra-Lateral PSD's

ใน two dimensional tetra-lateral PSD จะมีขั้ว cathode 4 ตัว และ anode รวม 1 ตัว เช่นเดียวกับ PSD ชนิดอื่น สัญญาณจาก detector จะถูกเปลี่ยนเป็นความต่างศักย์ไฟฟ้าในช่วงแรก และถูกรวมและหักล้างในช่วงที่สองจากนั้นจึงถูกแยกออกมาในช่วงสุดท้ายดังแสดงในรูปที่ 2.15

สำหรับการใช้งานที่ต้องการความเร็วสูงนั้น จะต้องไบอัสกลับให้แก่ anode และ feedback resistor (R_f) ต้องมีค่าน้อยๆการเพิ่มอัตราการขยายสามารถทำได้โดยการเพิ่มช่วงการทำงาน (stage) อุปกรณ์ที่เลือกใช้ในการต่อวงจรจะเป็นเช่นเดียวกับวงจร two-dimensional duo-lateral



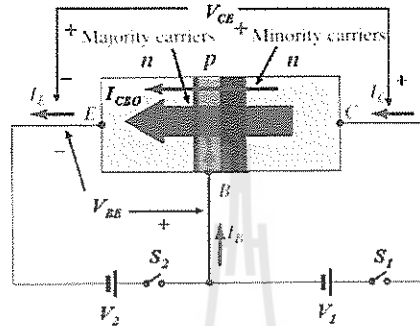
รูปที่ 2.15 แสดงวงจร two dimensional tetra-lateral PSD's

2.14 การไบอัสทรานซิสเตอร์ (Transistor Biasing)

การไบอัสทรานซิสเตอร์

การไบอัส (Biasing) หมายถึง การกำหนดค่าแรงดันและกระแสทางไฟฟ้ากระแสตรงให้กับอุปกรณ์ทางอิเล็กทรอนิกส์เพื่อให้ทำงานตามต้องการ

จากคุณสมบัติของทรานซิสเตอร์ที่ทำมาจากสารเจอร์เมเนียม มีค่าแรงดันตกคร่อม $V_{BE} = 0.3 \text{ V}$ และทำมาจากสารซิลิกอนมีค่า $V_{BE} = 0.7 \text{ V}$ (ที่ 25 องศาเซลเซียส)



รูปที่ 2.16 แสดงการทำงานของทรานซิสเตอร์แบบ NPN

จากกระแสของทรานซิสเตอร์

$$I_E = I_C + I_B \quad (25)$$

ซึ่งอัตราส่วนของกระแส I_C ต่อ I_B จะแทนด้วยสัญลักษณ์ β (เบต้า) หรือ h_{FE} ส่วนอัตราส่วนของกระแส I_C ต่อ I_E แทนด้วยสัญลักษณ์ α (อัลฟา) และอัตราส่วนของกระแส I_E ต่อ I_B แทนด้วยสัญลักษณ์ γ (แกมมา) ซึ่งแต่ละตัวจะมีความสัมพันธ์กัน ดังนี้คือ

$$\text{เบต้าในเทอมอัลฟา} \quad \beta = \frac{\alpha}{1-\alpha} \quad (26)$$

$$\text{เบต้าในเทอมแกมมา} \quad \beta = \gamma - 1 \quad (27)$$

$$\text{อัลฟาในเทอมเบต้า} \quad \alpha = \frac{\beta}{\beta+1} \quad (28)$$

$$\text{อัลฟาในเทอมแกมมา} \quad \alpha = \frac{\gamma-1}{\gamma} \quad (29)$$

$$\text{แกมมาในเทอมเบต้า} \quad \gamma = \beta + 1 \quad (30)$$

$$\text{แกมมาในเทอมอัลฟา} \quad \gamma = \frac{1}{1-\alpha} \quad (31)$$

อัตราส่วนของของกระแสในตัวทรานซิสเตอร์ I_B และ I_C จากสมการ

$$I_C = h_{FE} I_B \quad (32)$$

หรือ

$$I_C = \frac{I_C}{h_{FE}}$$

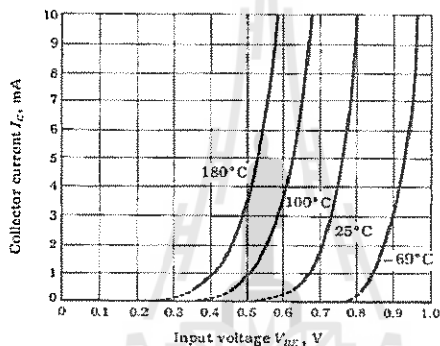
ดังนั้นจะได้

$$\begin{aligned} I_E &= I_B + h_{FE} I_B \\ &= (1 + h_{FE}) I_B \\ &\cong I_C \end{aligned}$$

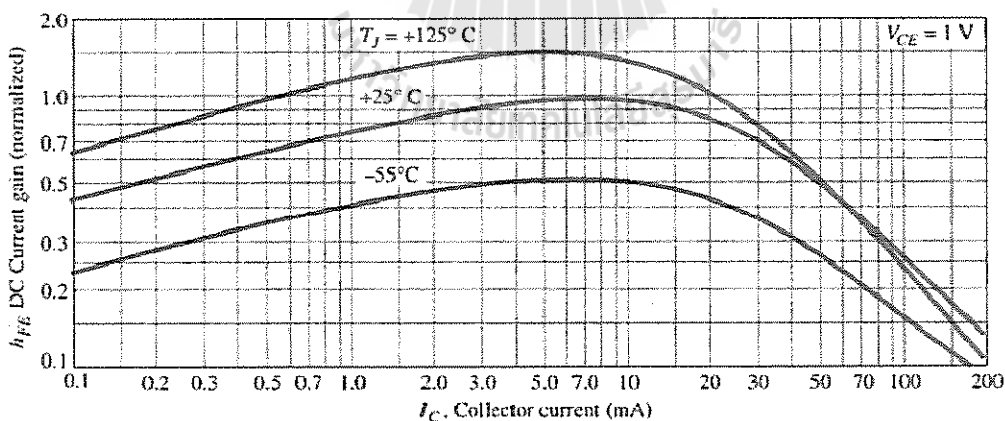
และ

$$I_C = I_{C(\text{majority})} + I_{CBO(\text{minority})} \tag{33}$$

เมื่อทรานซิสเตอร์ทำงานที่อุณหภูมิที่ต่างจากอุณหภูมิห้องก็จะทำให้ค่าของ V_{BE} และ h_{FE} เปลี่ยนแปลงไปดังรูป ตามลำดับ



รูปที่ 2.17 แสดงผลของอุณหภูมิที่มีผลต่อ แรงดัน V_{BE}



รูปที่ 2.18 แสดงผลของอุณหภูมิที่มีผลต่อค่า h_{FE}

เมื่อนำเอาทรานซิสเตอร์ไปใช้งานจริง จะพบว่ามิมีปัญหาเกิดขึ้นคือ ทรานซิสเตอร์แต่ละตัวจะมี อุณหภูมิและจุดการทำงานที่แตกต่างกันออกไป ถ้าให้กระแสไหลในวงจรมากเกินไปจะทำให้ ทรานซิสเตอร์อยู่ในสภาวะอิ่มตัวได้ และถ้าลดกระแสลงมากเกินไปก็อาจจะทำให้ถึงจุดคัทออฟได้

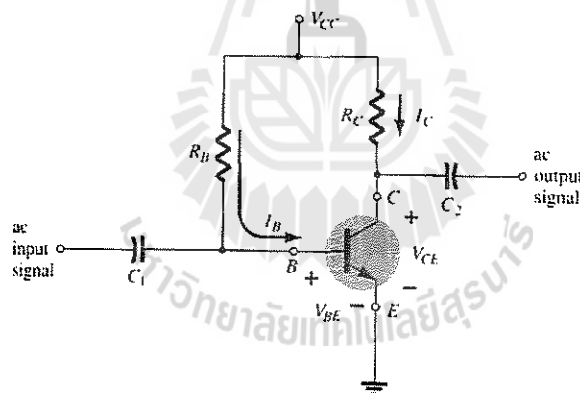
ปัญหาอีกอย่างหนึ่งคือ เวลาใช้งานเมื่อทรานซิสเตอร์แต่ละตัวต้องการกระแสไม่เท่ากันหรือจุดทำงานแตกต่างกันไปแล้วแต่เบอร์ของทรานซิสเตอร์ จึงจำเป็นที่แหล่งจ่ายจะต้องเปลี่ยนแปลงตามไปด้วย แต่เวลาใช้งานจริงจะใช้แหล่งจ่ายหลาย ๆ ภาค ก็เป็นการสิ้นเปลืองโดยใช่เหตุ จึงจำเป็นต้องมีการจัดการทำงานให้กับทรานซิสเตอร์หรือที่เรียกว่า การจัดไบอัสทรานซิสเตอร์

การไบอัสทรานซิสเตอร์แบ่งออกเป็น 3 แบบคือ

- การไบอัสแบบคงที่ (Fixed Biasing)
- การไบอัสแบบป้อนกลับ (Self Biasing)
- การไบอัสแบบป้อนกลับ (Stabilize Biasing)

2.14.1 การไบอัสแบบคงที่

เป็นการไบอัสแบบที่ง่ายที่สุด ซึ่งเป็นการพิจารณาวงจรไบอัสแบบคงที่ ในขณะที่ไม่มีสัญญาณเข้ามาทางอินพุต ซึ่ง C_1 เป็นตัวเก็บประจุทำหน้าที่ ักปลั้งสัญญาณอินพุต เข้ามายังขาเบสของทรานซิสเตอร์ทำให้กระแสเบสแปรไปตามอินพุต และกระแสคอลเลกเตอร์ก็จะแปรผันตามกระแสเบส ทำให้สัญญาณเอาต์พุตมีขนาดใหญ่กว่าอินพุต และจะถูกนำออกไปยังวงจรอื่นโดย C_2



รูปที่ 2.19 แสดงวงจรไบอัสแบบคงที่

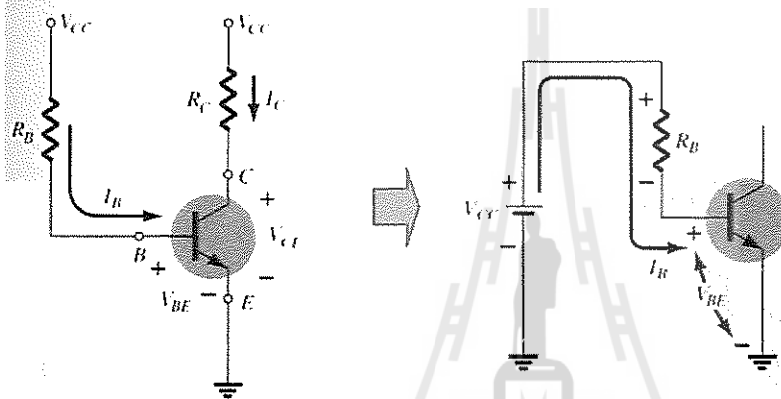
จากวงจรไบอัสแบบคงที่ ความต้านทาน R_B เป็นความต้านทานที่กำหนดค่ากระแส I_B โดยมีแรงดัน V_{CC} จ่ายไบอัสตรงให้กับขา B เทียบกับขา E เมื่อกระแส I_B ไหลจะทำให้กระแส I_C ไหลด้วยทรานซิสเตอร์เมื่อนำกระแสจะเกิดความร้อน ทำให้ความต้านทานรอยต่อระหว่างขา C และขา E ลดลง กระแส I_C จะไหลมากขึ้น ซึ่งถ้ายังจ่าย V_{BE} เท่าเดิม กระแส I_B จะไหลมากขึ้น นั่นหมายถึงไบอัสที่จ่ายให้ขา B ของทรานซิสเตอร์มากเกินไป R_B ซึ่งถูกกำหนดไบอัสให้ขา B ไม่สามารถเปลี่ยนแปลงได้ ทำให้ไบอัสถูกจ่ายให้ขา B ของทรานซิสเตอร์มากขึ้นตลอดเวลา กระแส I_C ก็จะไหลมากขึ้นตลอดเวลา ทรานซิสเตอร์จะร้อนมากขึ้นทุกขณะจนชำรุดเสียหายได้

นอกจากนี้ I_{CBO} ยังมีความไวต่ออุณหภูมิ เช่น ทรานซิสเตอร์ที่ทำจากเยอรมันเนียม ค่า I_{CBO} จะเพิ่มขึ้นประมาณหนึ่งเท่าตัวทุก ๆ อุณหภูมิ 10 องศาเซลเซียส ที่เพิ่มขึ้นสูงกว่าอุณหภูมิห้อง เป็นผลให้ค่าของกระแส I_C มีค่าสูงขึ้นตามไปด้วย

ข้อดี ใช้อุปกรณ์น้อย

ข้อเสีย ไม่คงที่อุณหภูมิ ไม่สามารถนำไปใช้งานแบบต่อเนื่องได้

จากวงจรไบอัสแบบคงที่ สามารถแบ่งออกเป็น 2 วงจร คือวงจรการไบอัสทางอินพุต และเอาต์พุต



รูปที่ 2.20 แสดงวงจรทางด้านอินพุตของการ ไบอัสแบบคงที่

จากวงจรทางด้านอินพุตของการ ไบอัสแบบคงที่ เมื่อพิจารณา โดยใช้กฎแรงดันของเคอร์ชอฟฟ์ (KVL) จะได้

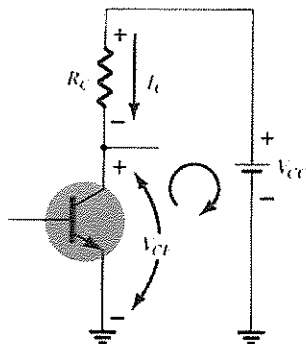
$$\text{สมการ } V_{CC} - I_B R_B - V_{BE} = 0$$

$$V_{CC} = I_B R_B + V_{BE} \quad (34)$$

ดังนั้น I_B จึงถูกจำกัด ด้วย R_B ดังนี้

$$I_B = \frac{V_{CC} - V_{BE}}{R_B} \quad (35)$$

$$R_B = \frac{V_{CC} - V_{BE}}{I_B} \quad (36)$$



รูปที่ 2.21 แสดงวงจรทางด้านเอาต์พุตของการไบอัสแบบคงที่

จากวงจรทางด้านเอาต์พุตของการไบอัสแบบคงที่ เขียนสมการเพื่อหาค่า I_C และ V_{CE} จากความสัมพันธ์ของ I_C และ I_B จะได้

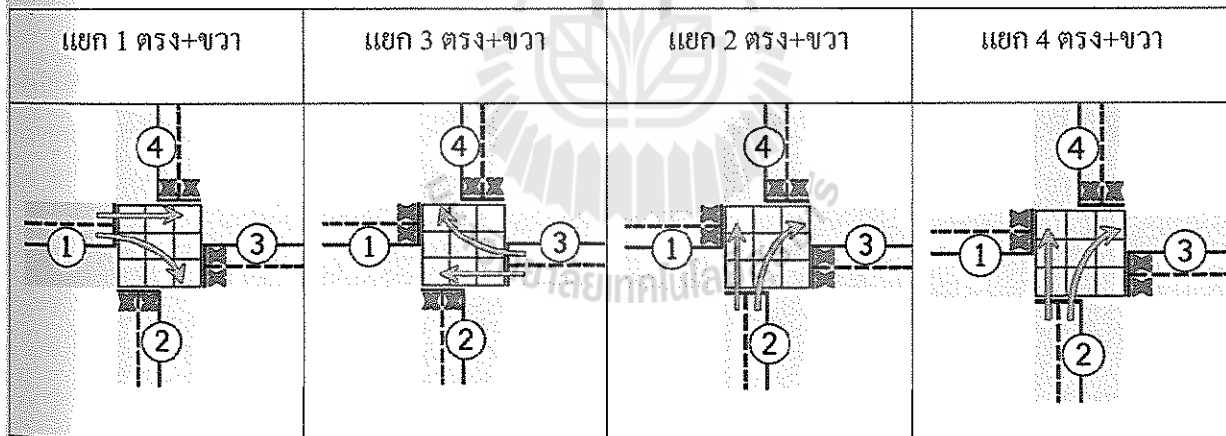
$$V_{CC} = (I_C R_C) + V_{CE}$$

$$V_{CE} = V_{CC} - (I_C R_C) \tag{37}$$

$$R_C = \frac{V_{CC} - V_{CE}}{I_C} \tag{38}$$

2.15 รูปแบบการทำงานของไฟจรรยา

การทำงานของไฟจรรยาสามารถจำแนกรูปแบบของการปล่อยยานพาหนะได้หลายรูปแบบ ซึ่งสามารถจำแนกลำดับการทำงานของรูปแบบได้ดังรูป ซึ่งในที่นี้ จะไม่การเลี้ยวซ้ายมาพิจารณา



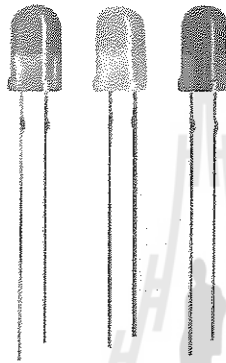
รูปที่ 2.22 แสดงรูปแบบการปล่อยแบบทีละแยก

รูปแบบการปล่อยแบบทีละแยก เหมาะสำหรับการจราจรที่มีการจราจรติดขัด เนื่องจากปริมาณยานพาหนะมีความหนาแน่นในทุกเส้นทางการจราจร จึงทำให้จำเป็นต้องมีการปล่อยแบบทีละแยกเพื่อเป็นการระบายยานพาหนะให้ได้มากที่สุด

2.16 หลอด LED

หลอด LED หรือไดโอดเปล่งแสง เป็นอุปกรณ์สารกึ่งตัวนำโดยจัดอยู่ในพวกไดโอด โดยการเปล่งแสงจะอยู่ในรูปของปรากฏการณ์ electroluminescence สีของแสงที่เปล่งออกมาขึ้นอยู่กับองค์ประกอบทางเคมีของวัสดุสารกึ่งตัวนำ

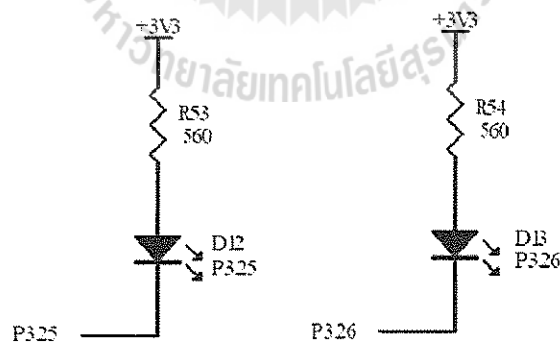
ในปัจจุบันการใช้หลอด LED มาใช้แทนหลอดไส้เนื่องจากหลอด LED มีอายุการใช้งานที่นานกว่า เนื่องจากหลอด LED ไม่ต้องใช้การเผาไหม้ของไส้หลอด และก่อให้เกิดความร้อนน้อยมากเมื่อเทียบกับหลอดไส้เพราะพลังงานส่วนใหญ่ของหลอด LED เปลี่ยนไปเป็นแสงทั้งหมด และในปัจจุบันราคาหลอด LED ราคาถูก ทำให้สามารถลดต้นทุนการผลิตและนำไปใช้งานได้อย่างหลากหลาย ตัวอย่างหลอด LED ดังรูป



รูปที่ 2.23 แสดงลักษณะไดโอดเปล่งแสง

การควบคุมสัญญาณไฟจราจรนี้ ได้นำเอาหลอด LED มาจำลองเป็นไฟสัญญาณจราจรโดยการต่อหลอด LED เข้ากับ GPIO port ของไมโครคอนโทรลเลอร์ ส่วนของหลอด LED อีกขาจะต่อกับ VCC +3V

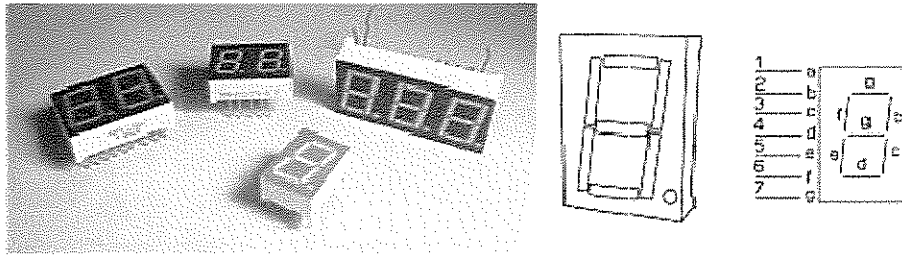
ดังรูป



รูปที่ 2.24 แสดงวงจรการต่อหลอด LED เข้ากับ GPIO port ของไมโครคอนโทรลเลอร์

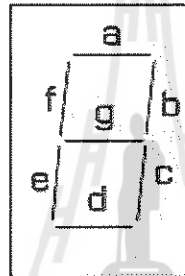
2.17 ตัวแสดงผล 7 ส่วน

ตัวแสดงผล 7 ส่วน หรือ 7 Segment เป็นอุปกรณ์อิเล็กทรอนิกส์ประเภท Display เช่นเดียวกับ ไดโอดเปล่งแสงหรือ LED ตัว 7 Segment ภายในคือ LED 7 ตัว หรือมากกว่า 7 ตัว มาต่อกันเป็นรูปตัวเลข 8 ดังนั้นการใช้งาน 7 Segment จึงเหมือนกับการใช้งาน LED



รูปที่ 2.25 แสดงรูปแบบต่างๆ และสัญลักษณ์

จอแสดงผลแบบ 7 Segment นั้นประกอบไปด้วย 7 ส่วนซึ่งจะถูกทำการเปิดหรือปิดเพื่อแสดงรูปแบบของตัวเลขฐานสิบ 7 Segment จะถูกจัดเรียงเป็นรูปสี่เหลี่ยม โดยในแนวตั้งมีอยู่ด้านละ 2 แห่ง และแนวนอนมีอยู่ 1 แห่ง 3 ชั้น บน กลาง และล่าง แต่ละแห่งของ 7 Segment จะถูกอ้างอิงจากตัวอักษรตั้งแต่ A ถึง G โดยที่ DP จะเป็นแท่งที่ 8 ใช้สำหรับตัวเลขที่ไม่ใช่จำนวนเต็ม

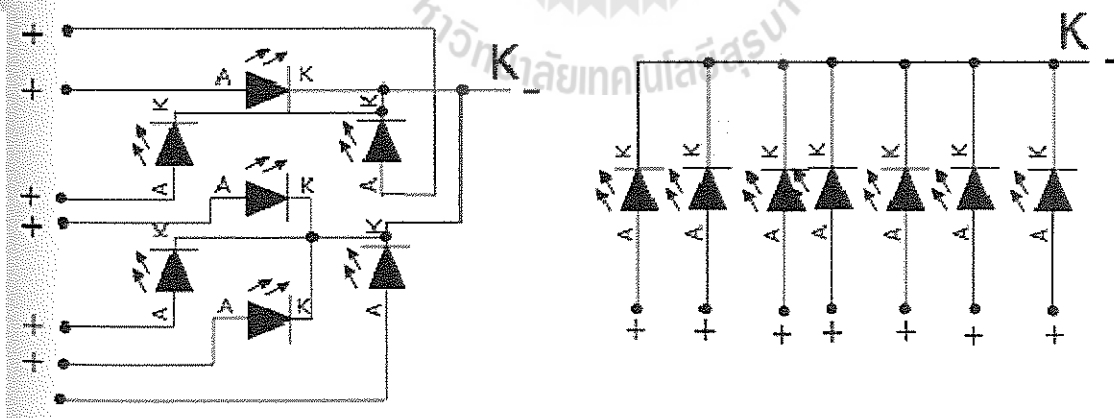


รูปที่ 2.26 แสดงตำแหน่งส่วนแสดงผล A- G

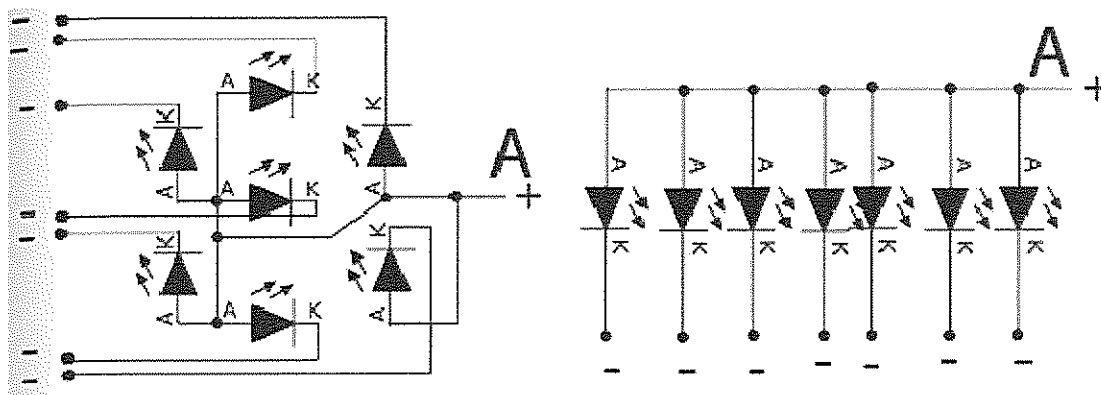
2.17.1 การต่อ LED ภายในตัว 7 Segment

7 Segment นั้นจะมีอยู่ 2 คอมมอนหลักๆ คือ แบบคอมมอน A (อาโนด) และแบบคอมมอน K (คา

โทด)



รูปที่ 2.27 แสดงการนำเอา LED มาต่อกัน แบบคอมมอน K



รูปที่ 2.28 แสดงการนำเอา LED มาต่อกัน แบบคอมมอน K

การต่อแบบคอมมอน A เราจะใช้ขั้วลบ (-) ป้อนให้ที่ขา A - G ส่วนไฟบวก (+) จะมาป้อนที่จุดรวม

ของขา A

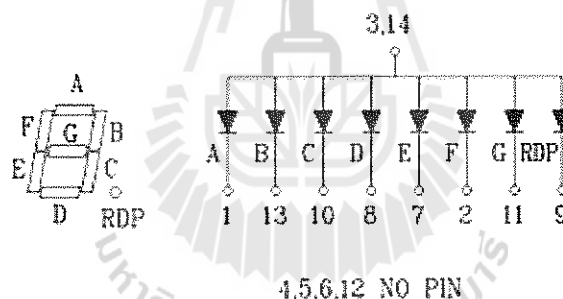
การต่อแบบคอมมอน K เราจะใช้ขั้วบวก (+) ป้อนให้ที่ขา A - G ส่วนไฟลบ (-) จะมาป้อนที่จุดรวม

ของขา K

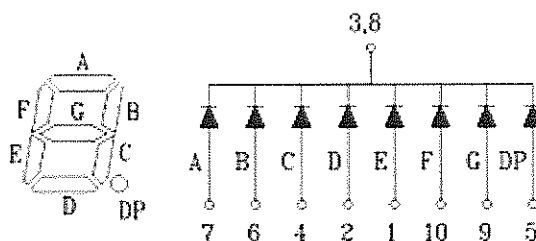
2.17.2 การดูสัญลักษณ์การต่อภายใน 7 Segment

รูปดังกล่าวต่อไปนี้จะแสดงการต่อ LED ไว้ภายใน ซึ่งจะมีทั้งคอมมอน A และ K และแบบรวม

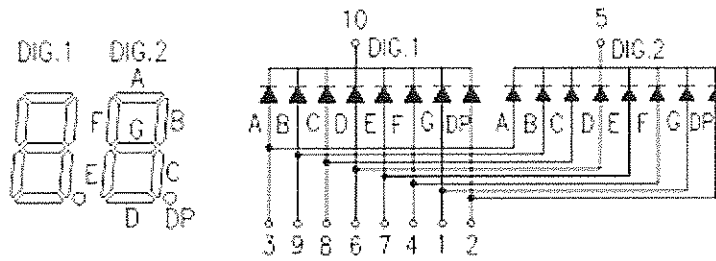
โดยที่สัญลักษณ์จะแสดงตำแหน่งของขา LED ไว้ให้ด้วย



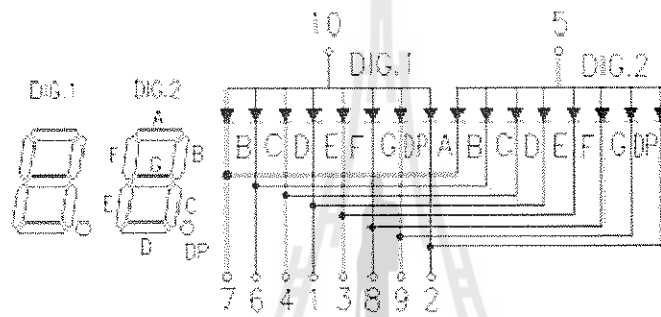
รูปที่ 2.29 แสดง 1 หลัค คอมมอน A ที่ขา 3 กับ 14 ส่วนขา 4,5,6,12 ไม่ได้ใช้



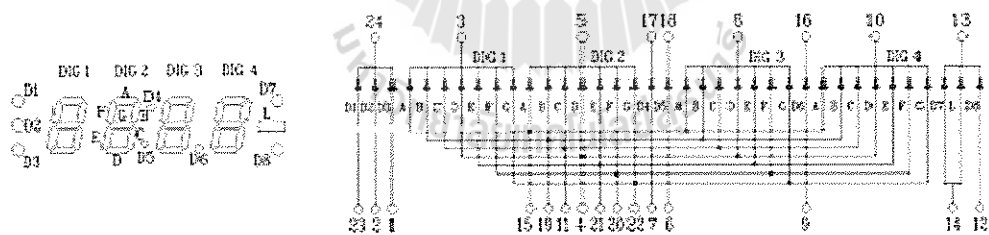
รูปที่ 2.30 แสดง 1 หลัค คอมมอน K ที่ขา 3 กับ 8



รูปที่ 2.31 แสดง 2 หลัก คอมมอน K ที่ขา 10(ตัวที่1) กับ 5(ตัวที่ 2)



รูปที่ 2.32 แสดง 2 หลัก คอมมอน A ที่ขา 10(ตัวที่1) กับ 5(ตัวที่ 2)



รูปที่ 2.33 แสดง 4 หลัก คอมมอน K ที่ขา 3(ตัวที่1) กับ 5(ตัวที่ 2) กับ 8(ตัวที่3) กับ 10(ตัวที่ 4)

บทที่ 3

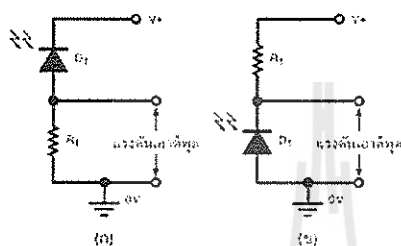
การออกแบบโปรแกรมและการทดสอบโปรแกรม

3.1 บทนำ

ในบทนี้จะกล่าวถึงวิธีการออกแบบวงจร และเขียน โปรแกรมเพื่อให้สามารถใช้ควบคุมระบบ สัญญาณไฟจราจร แบบอัตโนมัติโดยมีตัว Sensor เป็นตัวแปรในการประมวลผล

3.2. การออกแบบ Sensor

จากการทดลอง Photo Diode มีคุณสมบัติดังนี้

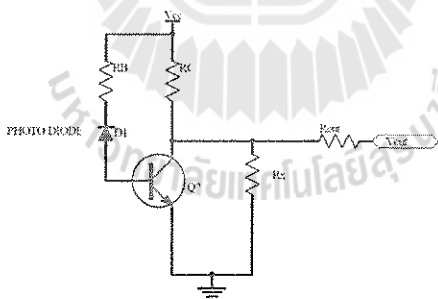


รูปที่ 3.1 ลักษณะการต่อใช้งานโฟโตไดโอด

เมื่อ Photo Diode รับแสงจะเกิดแรงดัน $V_d \approx 0.5 \text{ V}$ (Open circuit)

$I_{so} \approx 50 \text{ nV}$ มีค่าน้อยมาก (data sheet)

วงจรที่ใช้ใน Sensor



รูปที่ 3.2 แสดงวงจรตัว Sensor

คำนวณ Input

เมื่อมีแสงกระทบ Photo Diode แรงดัน $V_d \approx 0.5 \text{ V}$ สามารถคำนวณหากระแส I_b ดังนี้

$$V_{cc} = R_B I_b + V_{be} - V_d$$

$$I_b = \frac{V_{cc} - V_{be} + V_d}{R_B}$$

กำหนดให้ $R_B = 47 \text{ k}\Omega$, $V_{cc} = 5\text{V}$, $V_{be} = 0.6 \text{ V}$, $V_d = 0.5 \text{ V}$, $\beta = 5$

$$I_b = \frac{5 - 0.6 + 0.5}{47 \text{ k}\Omega} \text{ V}$$

$$\approx 100 \mu\text{A}$$

ดังนั้น $I_c = 5\text{mA}$

ด้าน Output

กรณีที่ 1 เมื่อมีแสงกระทบ Photo Diode $I_c = 5\text{mA}$ ต้องการ $V_{\text{out}} \approx 0 \text{ V}$ แรงดันที่ตกคร่อม R_c ต้องมีค่าประมาณ V_{cc} จึงสามารถคำนวณหา R_c ได้ดังนี้

$$R_c = \frac{V_{cc}}{I_c}$$

$$= \frac{5\text{V}}{5\text{mA}}$$

$$= 1000 \Omega$$

เลือกใช้ $R_c = 1.2 \text{ k}\Omega$

กรณีที่ 2 เมื่อไม่มีแสงตกกระทบ Photo Diode I_b มีค่าน้อยมากจึงทำให้ทรานซิสเตอร์ไม่ทำงาน เสมือนเปิดวงจรทำให้กระแสไหลผ่าน R_x เราสามารถคำนวณได้ดังนี้

$$\frac{R_x}{R_c + R_x} \times V_{cc} = V_{\text{out}}$$

$$\frac{R_x}{R_c + R_x} \times 5\text{V} = 3.3 \text{ V}$$

$$5R_x = 3.3(R_c + R_x)$$

$$5R_x - 3.3R_x = 3.3R_c$$

$$1.7R_x = 3.3R_c$$

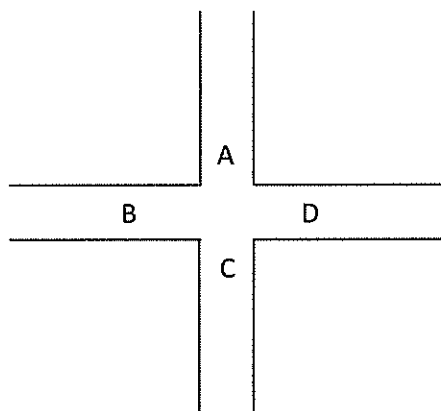
$$R_x = \frac{3.3}{1.7} R_c$$

$$= \frac{3.3}{1.7} \times 1200$$

$$= 2.3 \text{ K}\Omega$$

เลือก $R_x = 2.4 \text{ K}\Omega$

3.3 ออกแบบการทำงานของโปรแกรม



รูปที่ 3.3 แสดงเส้นทางสี่แยก

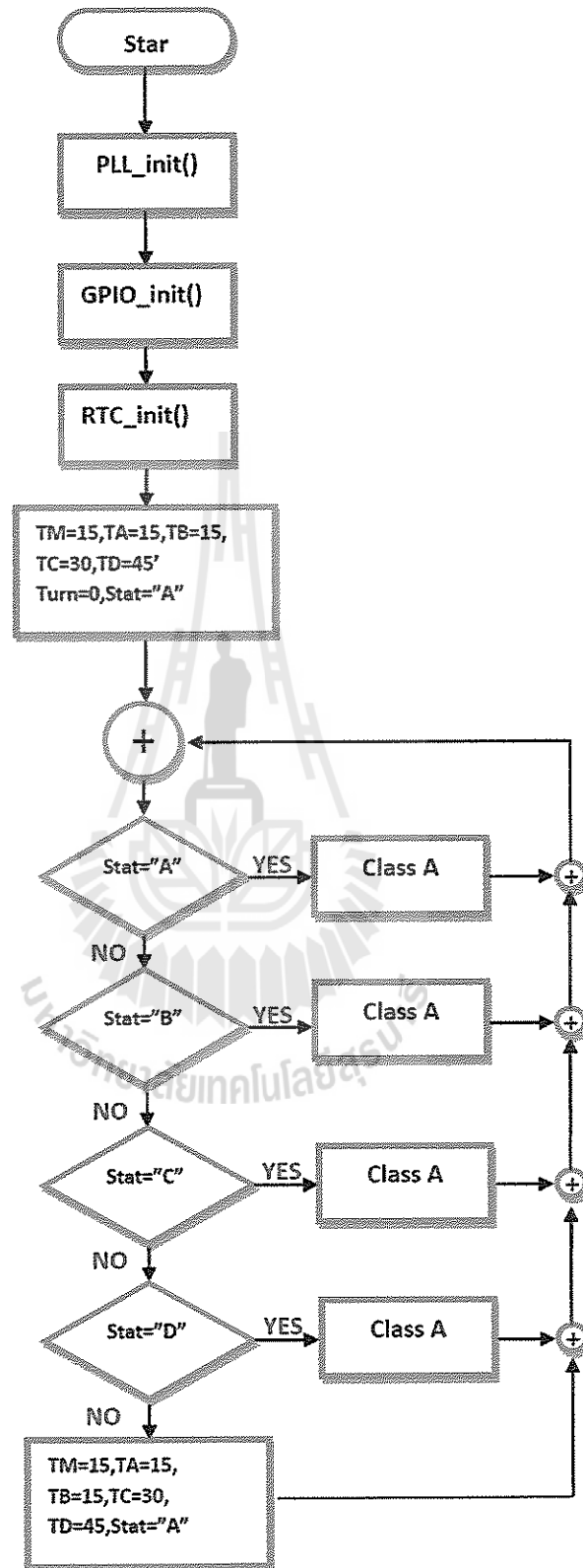
จากรูปประกอบไปด้วยเส้นทาง A , B , C และ D ซึ่งโดยปกติระบบสัญญาณไฟจราจรจะทำการปล่อยรถ (สัญญาณไฟเขียว) ตามลำดับขั้นต้น ซึ่งในบางกรณีทำให้เกิดสัญญาณไฟเขียวโดยไม่มีรถ จึงทำการปรับปรุงให้มีลักษณะการทำงานดังตารางต่อไปนี้

A	B	C	D	การทำงาน
0	0	0	0	ปล่อยที่ละเส้นทางโดยเพิ่มเวลาสัญญาณไฟเขียวเป็นสามเท่า
0	0	0	1	ปล่อยเส้นทาง D เป็นเวลา 15 วินาที และ 30 วินาทีในรอบที่สอง และ 45 วินาที ในรอบที่สาม
0	0	1	0	ปล่อยเส้นทาง C เป็นเวลา 15 วินาที และ 30 วินาทีในรอบที่สอง และ 45 วินาที ในรอบที่สาม
0	0	1	1	ปล่อยเส้นทาง C เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง D เป็นเวลา 15 วินาที
0	1	0	0	ปล่อยเส้นทาง B เป็นเวลา 15 วินาที และ 30 วินาทีในรอบที่สอง และ 45 วินาที ในรอบที่สาม
0	1	0	1	ปล่อยเส้นทาง B เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง D เป็นเวลา 15 วินาที
0	1	1	0	ปล่อยเส้นทาง B เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง C เป็นเวลา 15 วินาที
0	1	1	1	ปล่อยเส้นทาง B เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง C เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง D เป็นเวลา 15 วินาที

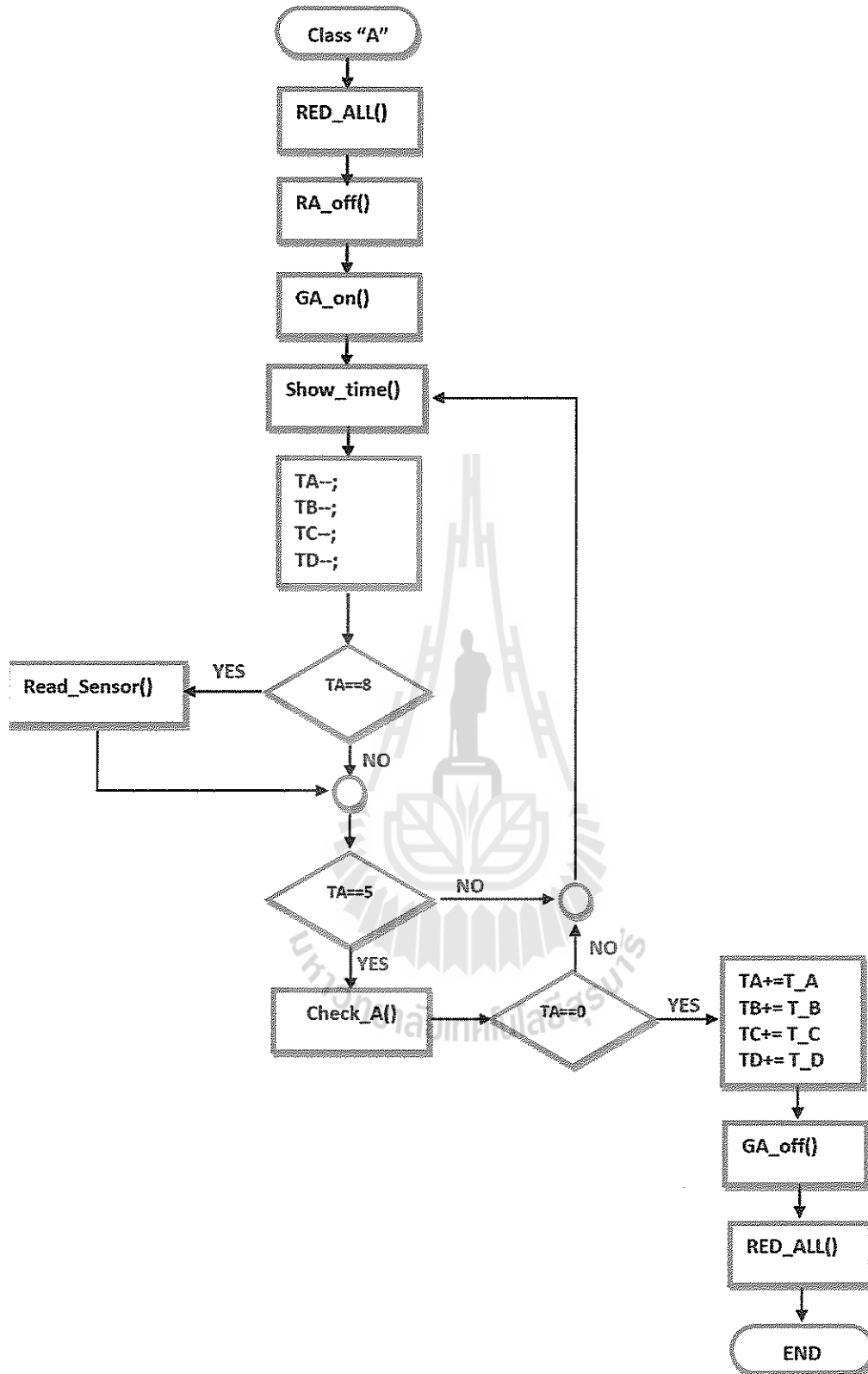
A	B	C	D	การทำงาน
1	0	0	0	ปล่อยเส้นทาง A เป็นเวลา 15 วินาที และ 30 วินาทีในรอบที่สอง และ 45 วินาที ในรอบที่สาม
1	0	0	1	ปล่อยเส้นทาง A เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง D เป็นเวลา 15 วินาที
1	0	1	0	ปล่อยเส้นทาง A เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง C เป็นเวลา 15 วินาที
1	0	1	1	ปล่อยเส้นทาง A เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง C เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง D เป็นเวลา 15 วินาที
1	1	0	0	ปล่อยเส้นทาง A เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง B เป็นเวลา 15 วินาที
1	1	0	1	ปล่อยเส้นทาง A เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง B เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง D เป็นเวลา 15 วินาที
1	1	1	0	ปล่อยเส้นทาง A เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง B เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง C เป็นเวลา 15 วินาที
1	1	1	1	ปล่อยเส้นทาง A เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง B เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง C เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง D เป็นเวลา 15 วินาที

หมายเหตุ 0 คือ ไม่มีรถ
1 คือ มีรถ

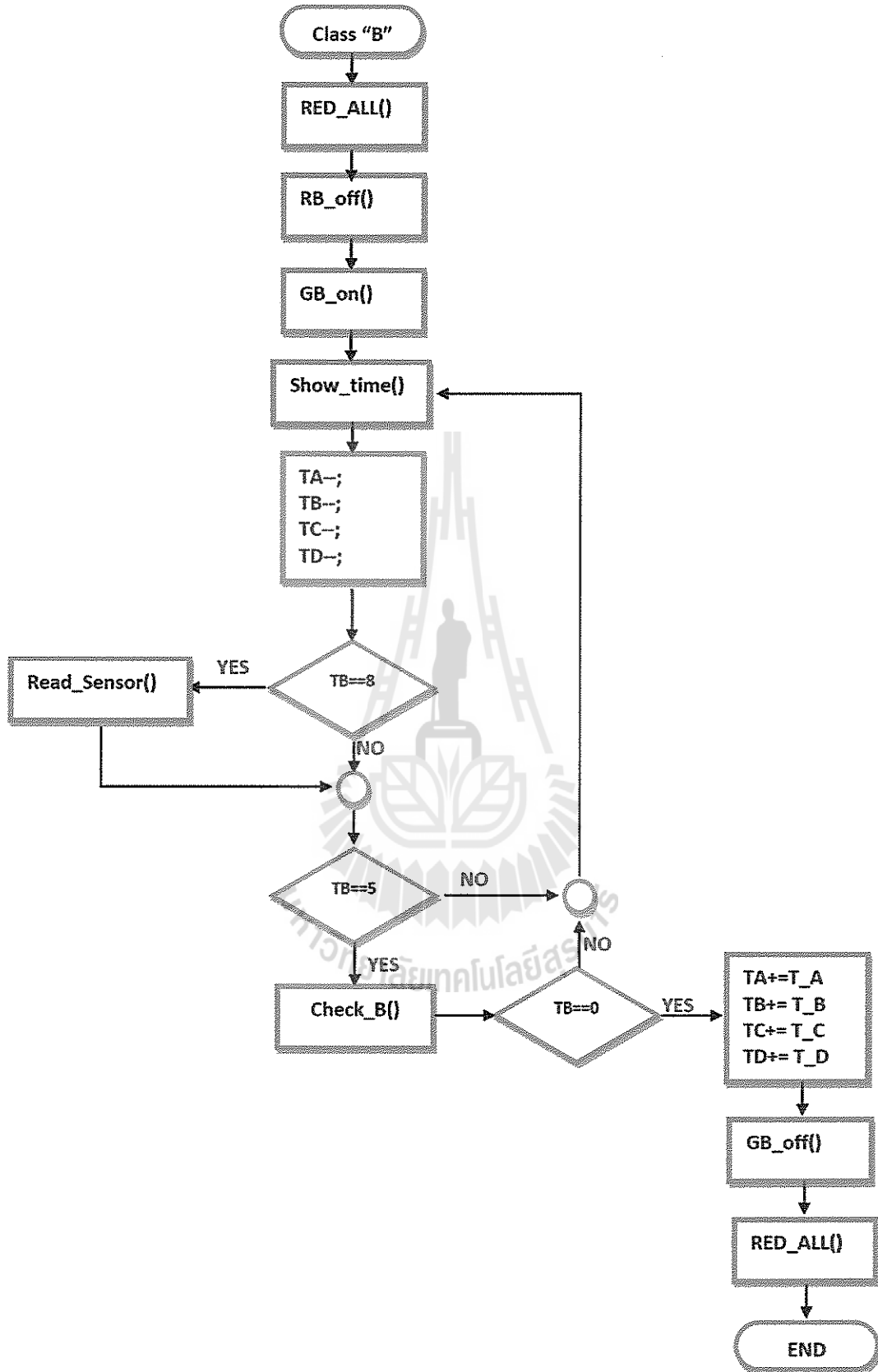
3.4 Flow Chart การทำงานของโปรแกรม



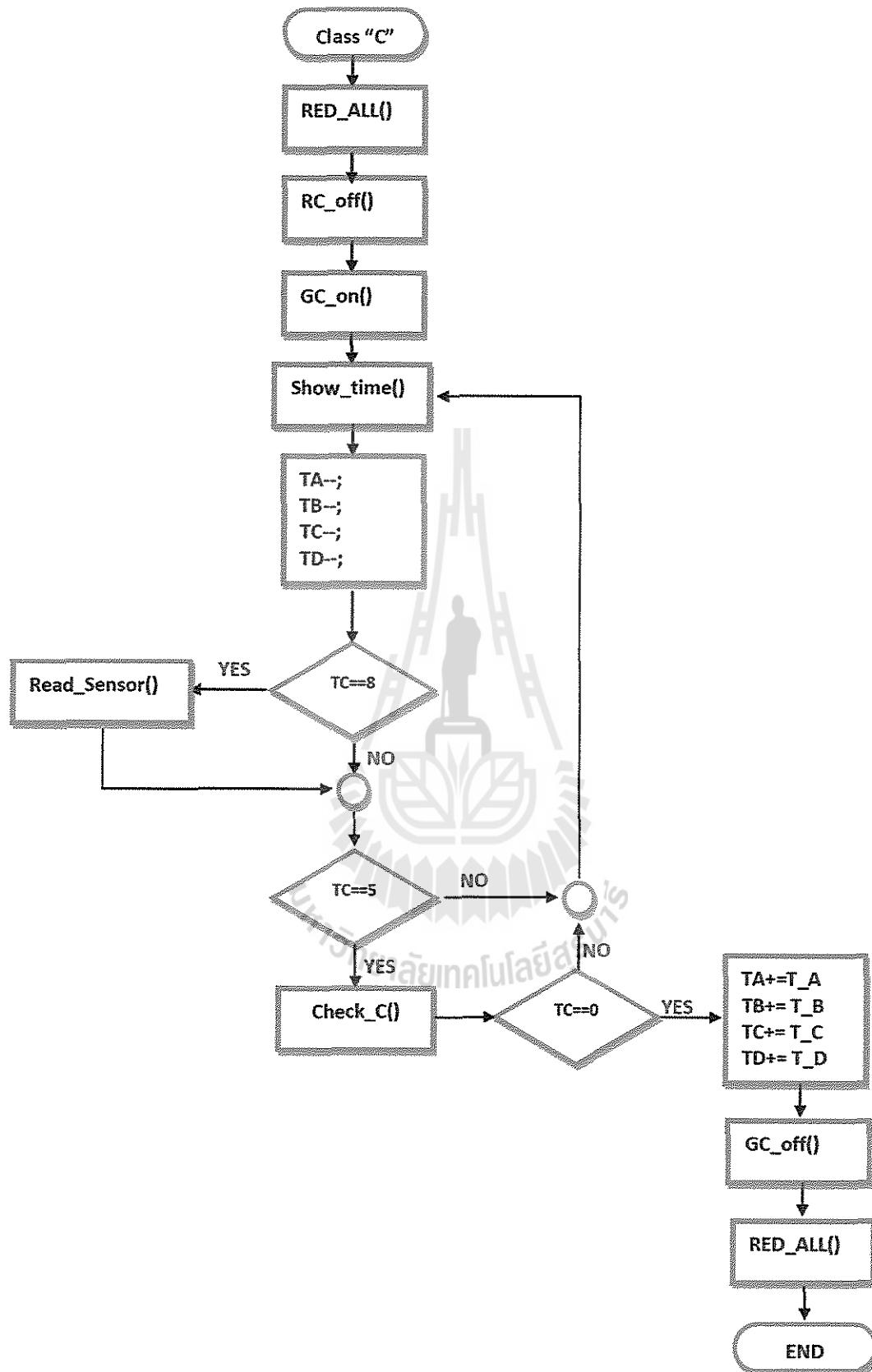
รูปที่ 3.4 แสดงการทำงานโดยรวมของโปรแกรม



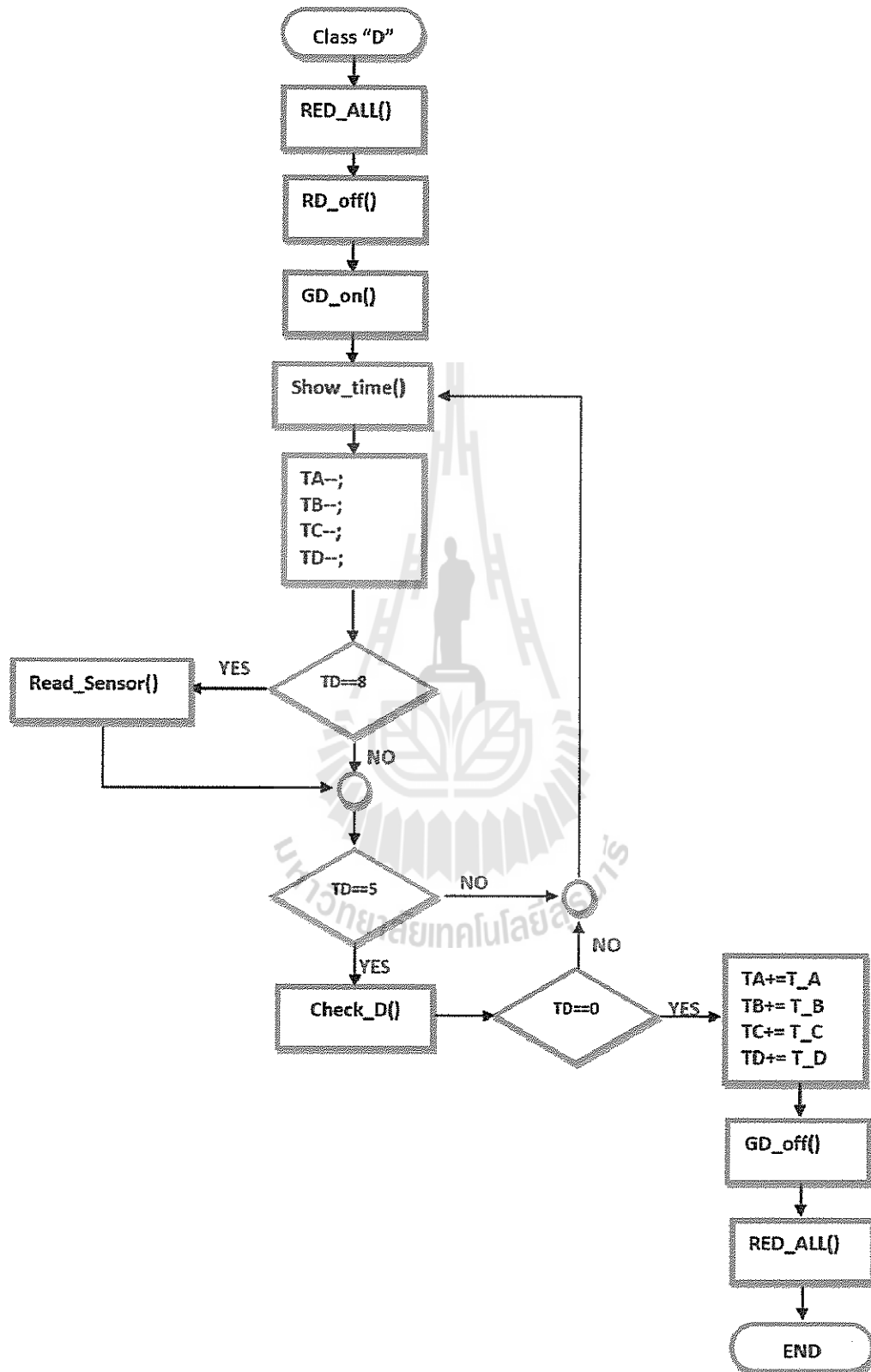
รูปที่ 3.5 แสดงการทำงาน โปรแกรม Class A



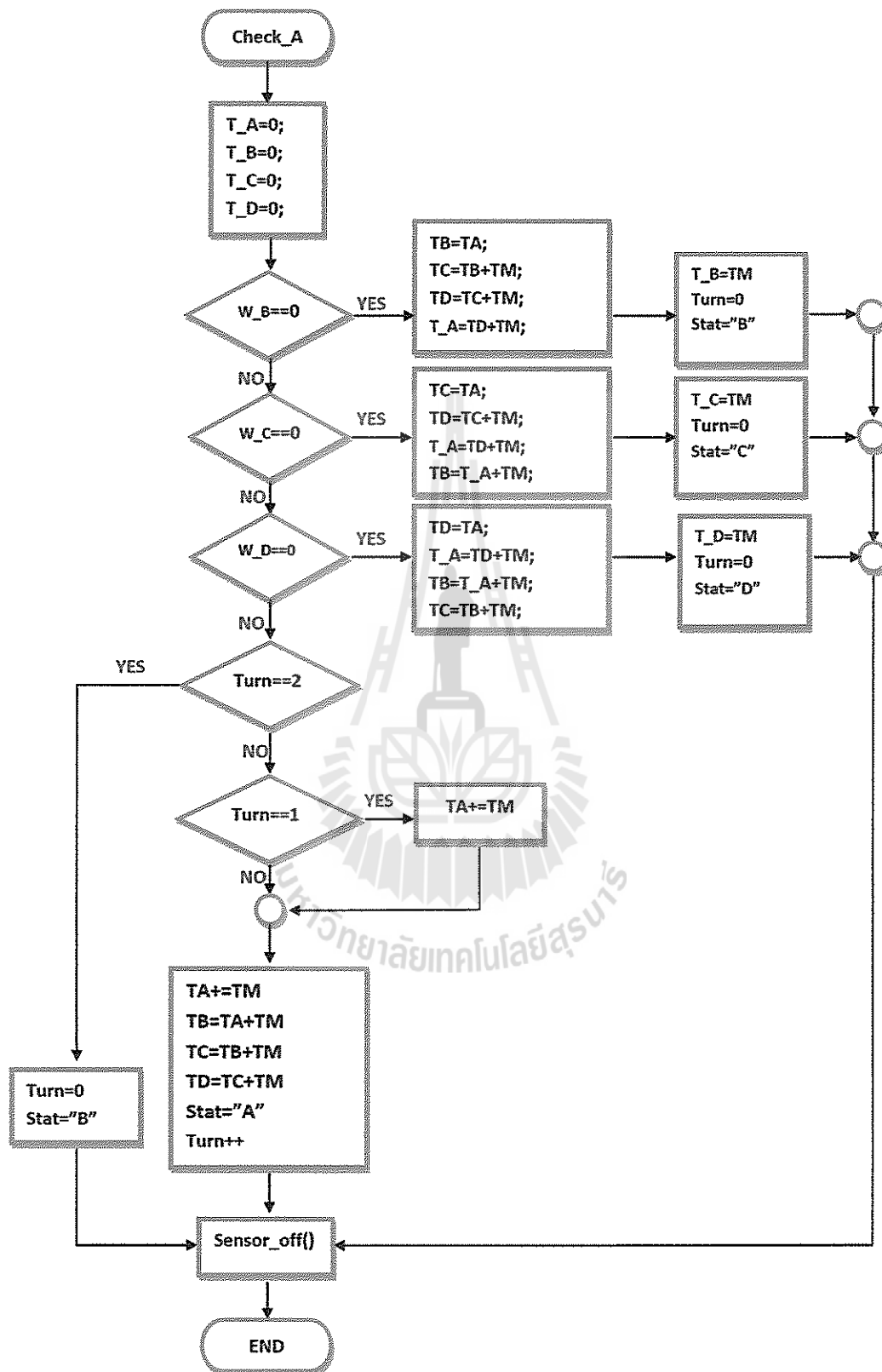
รูปที่ 3.6 แสดงการทำงานของโปรแกรมย่อย Class B



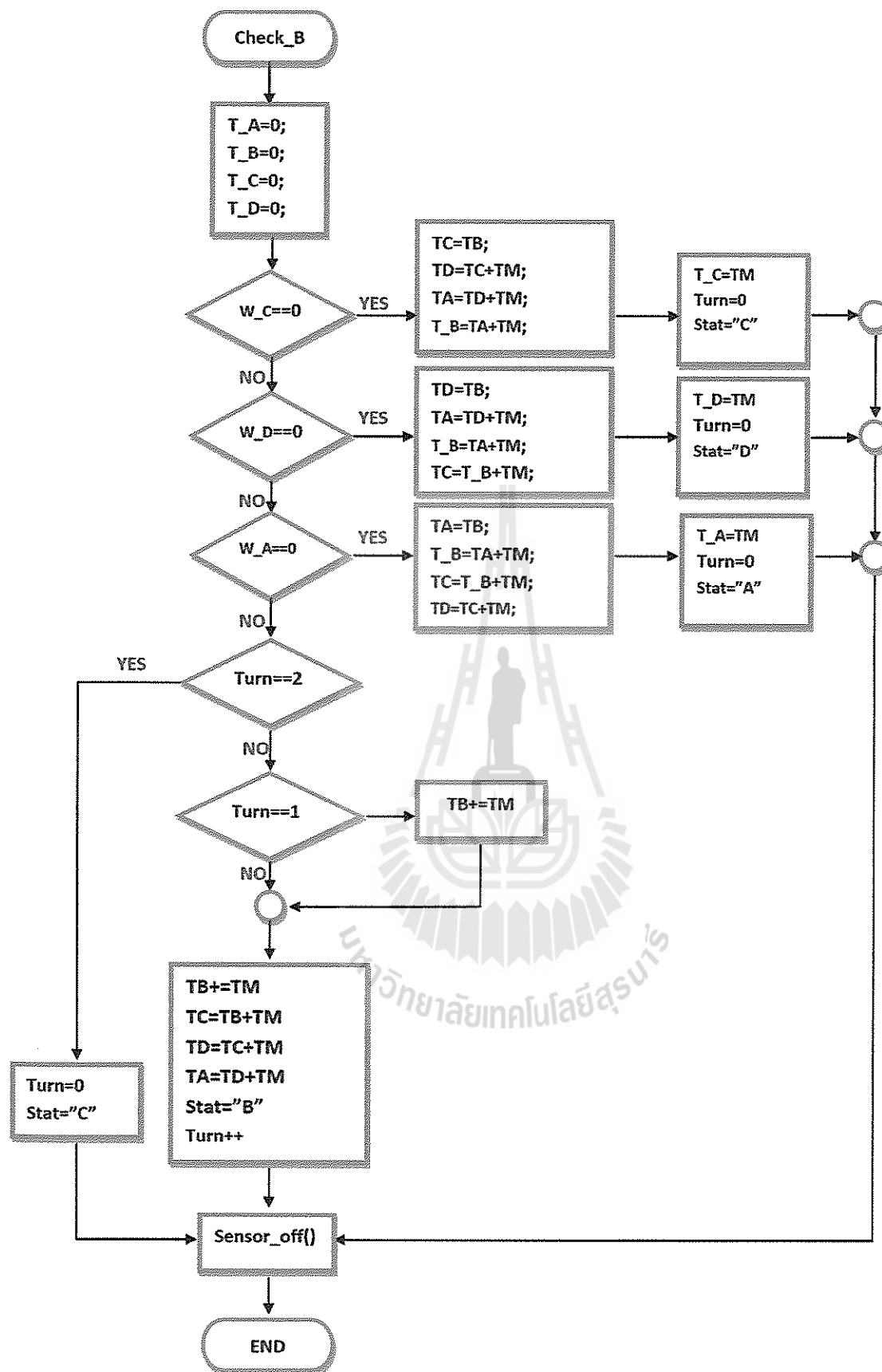
รูปที่ 3.7 แสดงการทำงาน โปรแกรมย่อย Class C



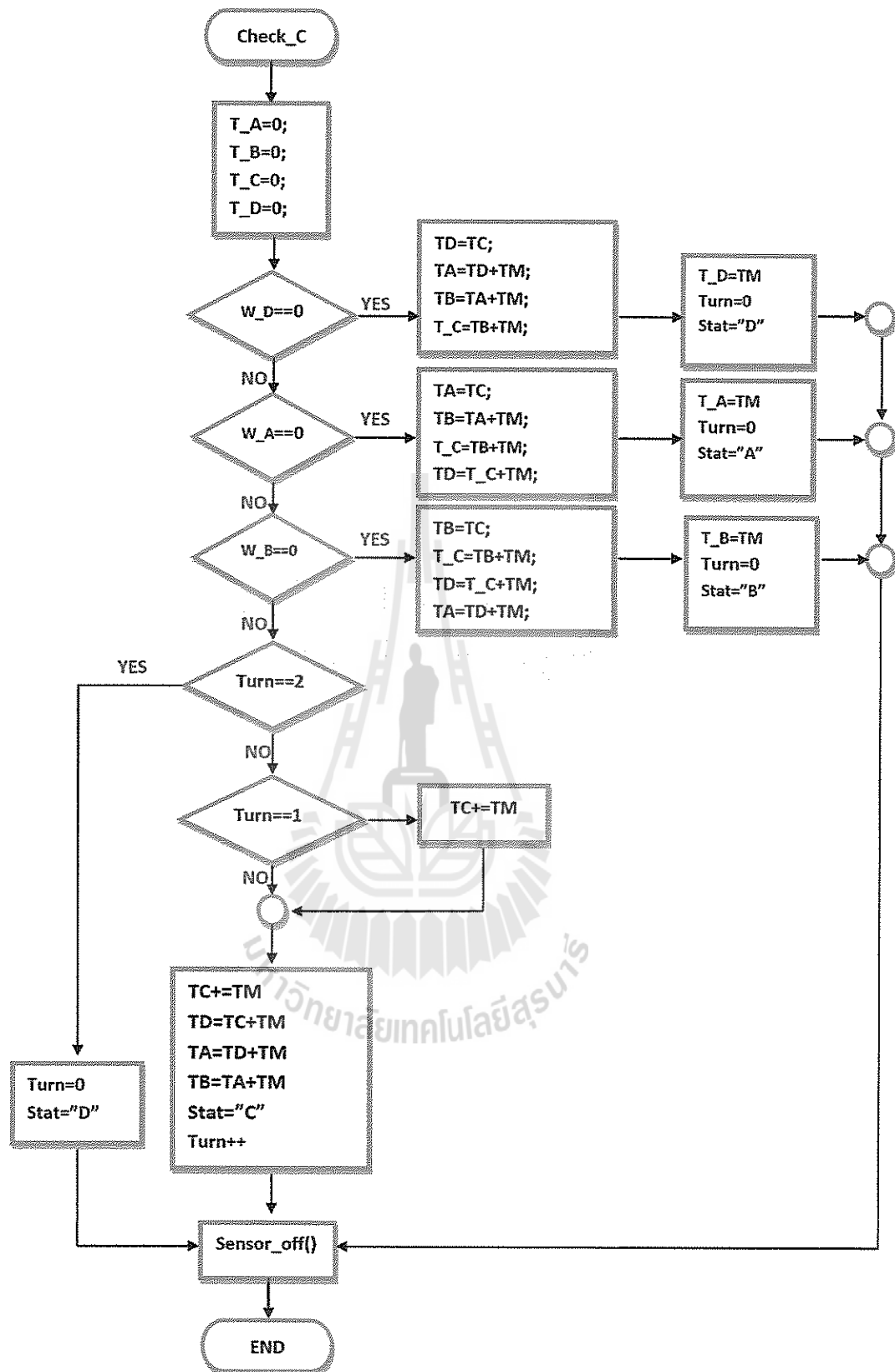
รูปที่ 3.8 แสดงการทำงาน โปรแกรมย่อย Class D



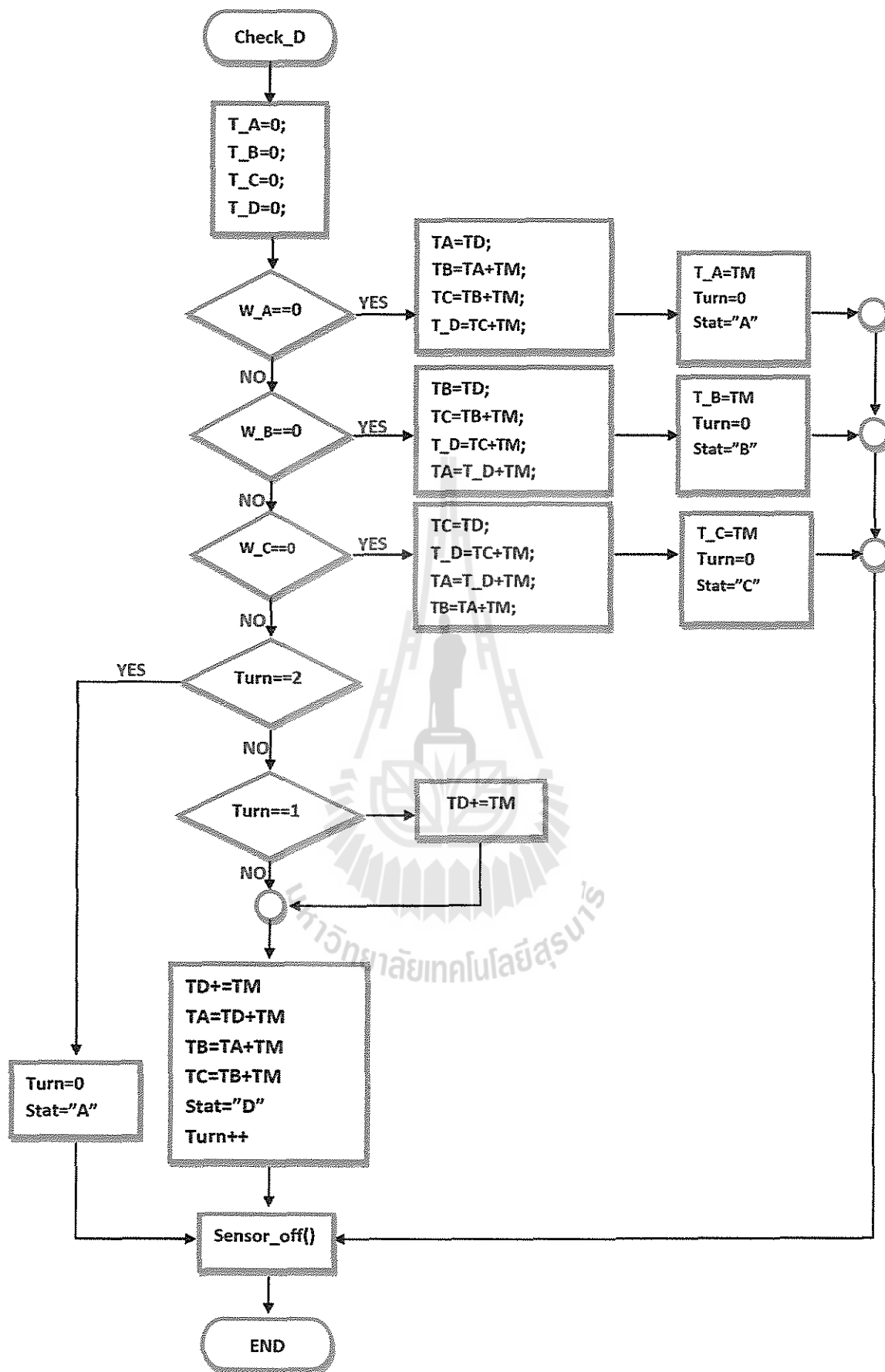
รูปที่ 3.9 แสดงการทำงานของโปรแกรมย่อย Check A



รูปที่ 3.10 แสดงการทำงานของโปรแกรมย่อย Check B



รูปที่ 3.11 แสดงการทำงานของโปรแกรมย่อย Check C



รูปที่ 3.12 แสดงการทำงานของโปรแกรมย่อย Check D

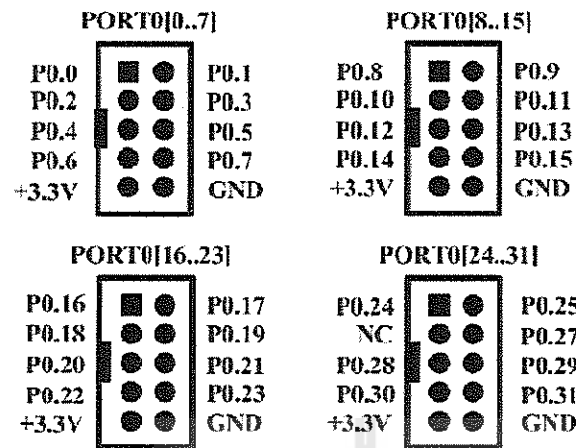
3.5 ไมโครคอนโทรลเลอร์ที่ใช้

การควบคุมสัญญาณไฟจราจรจากไมโครคอนโทรลเลอร์ โดยใช้ไมโครคอนโทรลเลอร์ ARM7TDMI-S Core รุ่น LPC2368 ซึ่งเหตุผลที่ได้เลือกใช้ไมโครคอนโทรลเลอร์ตัวนี้ เนื่องจากเป็นไมโครคอนโทรลเลอร์ 16/32-Bit ขนาด 144 Pin โดยมีคุณสมบัติเด่นคือมีจำนวน I/O ที่มากนอกจากนั้น GPIO ยังสามารถรองรับสัญญาณที่เป็น 5V ได้ ทำให้การใช้งานง่ายและไม่ต้องใช้อุปกรณ์เสริมเพิ่มเติม

คุณสมบัติของบอร์ด

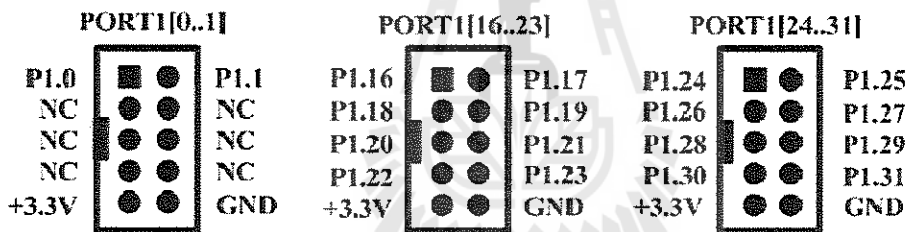
1. ใช้ MCU ตระกูล ARM7TDMI-S เบอร์ LPC2214 ของ Philips ซึ่งเป็น MCU ขนาด 16/32-Bit
2. ใช้ Crystal 19.6608 MHz โดย MCU สามารถประมวลผลด้วยความเร็วสูงสุดที่ 58.9824 MHz เมื่อใช้งานร่วมกับ Phase-Locked Loop (PLL) ภายในตัว MCU
3. รองรับการโปรแกรมแบบ In-System Programming (ISP) และ In-Application Programming (IAP) ผ่านทาง On-Chip Boot-Loader Software ทางพอร์ต RS232-CH0
4. พอร์ต JTAG 20PIN สำหรับ Real Time Debugging จำนวน 1 พอร์ต
5. พอร์ต LCD มาตรฐาน ETT 14PIN จำนวน 1 พอร์ต
6. พอร์ต GPIO ขนาด 10PIN จำนวน 16 พอร์ต มาตรฐาน ETT
7. มีขนาดหน่วยความจำโปรแกรมแบบ Flash 256KB และหน่วยความจำข้อมูลแบบ RAM 16KB (พร้อม External Memory Interface)
8. VIN +5V Power Supply
9. จำนวน GPIO สูงสุดถึง 112 I/O Pins (เฉพาะ GPIO รองรับสัญญาณที่เป็น 5V ได้) ซึ่งขาสัญญาณ GPIO จะมีการใช้งานร่วมกันของ Function อื่นๆอีก ดังนี้
 - SPI จำนวน 2 ช่อง, I2C 1 ช่อง, 8-Channel 10 Bit A/D Converter จำนวน 8 ช่อง ระดับสัญญาณ 0V - 3.3V เท่านั้น
 - UART แบบ Full-Duplex จำนวน 2 ช่อง คือ RS232-CH0, RS232-CH1 มาตรฐาน 4 Pin ETT เป็นสัญญาณระดับ RS232 Level
 - Timer 32-bit จำนวน 2 ช่อง (4 Input Capture / 4 Output Compare), Watchdog Timer, Real Time Clock, PWM 6 Output

- หมายเลข 6, 3, 14 และ 10 คือ GPIO0 แบ่งเป็น 4 พอร์ต ดังรูป (เฉพาะ GPIO สามารถรองรับสัญญาณที่เป็น 3.3V และ 5V ได้)



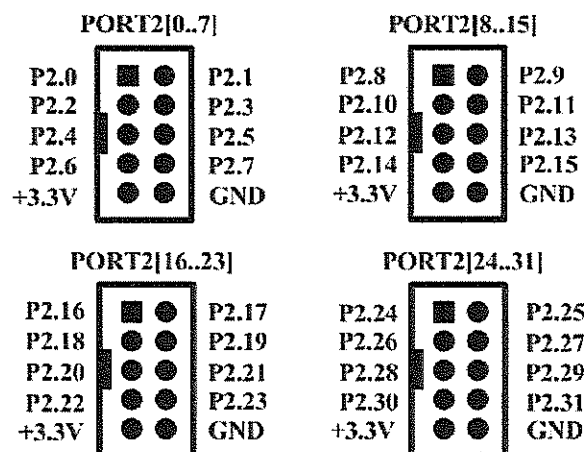
รูปที่ 3.14 แสดง GPIO0 4 พอร์ต

- หมายเลข 2, 7 และ 9 คือ GPIO1 แบ่งเป็น 3 พอร์ต ดังรูป (เฉพาะ GPIO สามารถรองรับสัญญาณที่เป็น 3.3V และ 5V ได้)



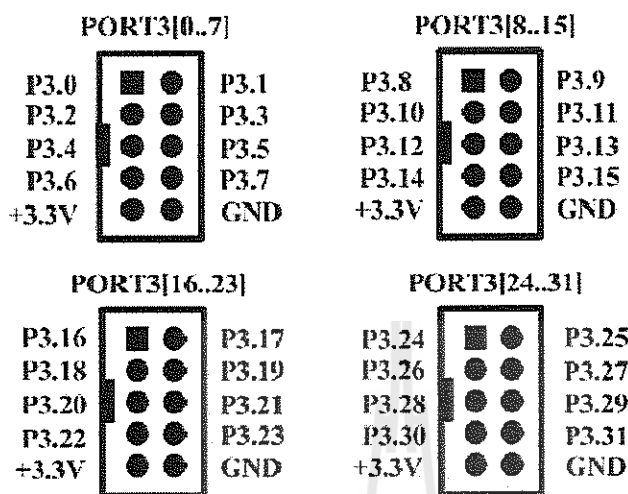
รูปที่ 3.15 แสดง GPIO1 3 พอร์ต

- หมายเลข 1, 17, 13 และ 12 คือ GPIO2 แบ่งเป็น 4 พอร์ต ดังรูป (เฉพาะ GPIO สามารถรองรับสัญญาณที่เป็น 3.3V และ 5V ได้)



รูปที่ 3.16 แสดง GPIO2 4 พอร์ต

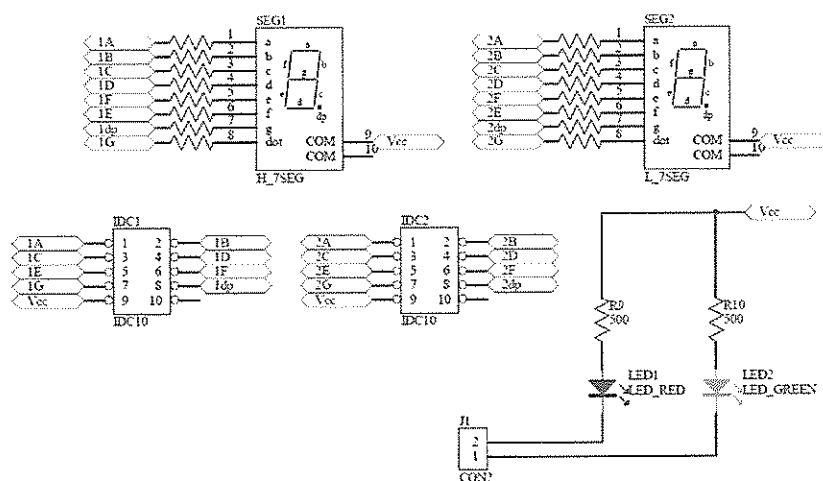
หมายเลข 4, 5, 8 และ 11 คือ GPIO3 แบ่งเป็น 4 พอร์ต ดังรูป (เฉพาะ GPIO สามารถรองรับสัญญาณที่เป็น 3.3V และ 5V ได้)



รูปที่ 3.17 แสดง GPIO3 4 พอร์ต

3.6 บอร์ดแสดงผล 7-Segment

บอร์ดแสดงผลนี้จะทำงานในลักษณะของวงจร Common Anode เมื่อไมโครคอนโทรลเลอร์จ่ายแรงดันไฟฟ้า 0 โวลต์ ตัวอุปกรณ์ 7-Segment และ LED จะทำงาน สำหรับบอร์ดแสดงผลนี้จะประกอบไปด้วย 7-Segment จำนวนสองหลัก คือ H_7SEG , L_7SEG และ LED แสดงไฟเขียวและไฟแดง ดังแสดงในรูปที่ 48



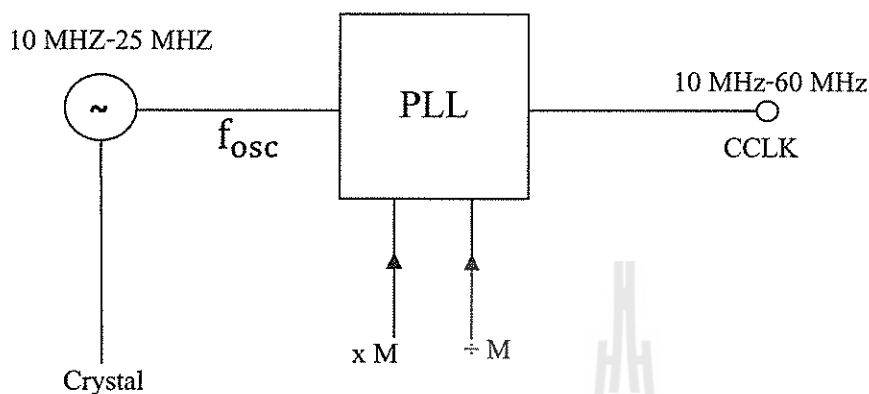
รูปที่ 3.18 บอร์ดแสดงผล 7-Segment

3.7 การเขียนโปรแกรม

ภาษาที่ใช้ในการเขียนโปรแกรมระบบสัญญาณไฟจากรณี เขียนโดยภาษา C ที่เขียนบนโปรแกรม

Keil uVision3 สำหรับ ARM7 ต้องมีการกำหนดค่าเริ่มต้นให้แก่ไมโครคอนโทรลเลอร์ดังนี้

- การกำหนดค่าความถี่ออสซิลเลเตอร์ให้กับไมโครคอนโทรลเลอร์



รูปที่ 3.19 ฟังก์ชัน PLL ภายในไมโครคอนโทรลเลอร์ ARM7

วงจร PLL จะรับค่าความถี่สัญญาณนาฬิกาจากคริสตัลออสซิลเลเตอร์ที่ถูกควบคุมค่าโดยคริสตัลภายนอก แล้วนำมาคูณด้วยค่าคงที่ M ให้เป็นความถี่ 10 MHz - 60 MHz โดยใช้วงจร Current Controlled Oscillator (CCO) ทำหน้าที่คูณความถี่ ค่าตัวคูณ M นี้มีค่าได้จาก 1 ถึง 32 (จากภาคปฏิบัติพบว่าไม่ควรเกิน 6) วงจร CCO ทำงานในช่วงความถี่ 156 MHz ถึง 320 MHz ดังนั้นภายในรูปของ CCO จะต้องมีวงจรหารค่าอีกตัวหนึ่งตัวเพื่อให้เอาต์พุตของ PLL ผลิตความถี่ได้ตามค่าที่ต้องการ ค่าตัวหารซึ่งก็คือ P กำหนดค่าได้เป็น 2, 4, 8 หรือ 16 เมื่อไมโครคอนโทรลเลอร์ถูกรีเซ็ต วงจร PLL จะถูกปิดการทำงาน ซึ่งต้องใช้ซอฟต์แวร์ตั้งเปิดการทำงานของมัน ตัวโปรแกรมจะต้องกำหนดค่าตัวคูณ M ตัวหาร P และกระตุ่นการทำงานของ PLL และรอให้ PLL ล็อกความถี่ได้ก่อน จึงจะสั่งต่อให้ PLL เป็นสัญญาณนาฬิกาของไมโครคอนโทรลเลอร์ ARM7 ค่าเวลาในการเซต PLL เป็น 100 μ s

เอาต์พุตของ PLL จำนวนได้ดังนี้

$$\text{CCLK} = M * f_{osc} \quad \text{หรือ} \quad \text{CCLK} = f_{cco} / (2 * P)$$

ค่าความถี่ cco จำนวนได้ดังนี้

$$f_{cco} = \text{CCLK} * 2 * P \quad \text{หรือ} \quad f_{cco} = f_{osc} * 2 * M * P$$

ค่า P เมื่อนำไปคูณแล้ว f_{cco} จะมีเงื่อนไขดังนี้

$$156 \text{ MHz} < f_{cco} < 320 \text{ MHz}$$

ในที่นี้ใช้แผงวงจร ET-JR ARM7 LPC2214 ใช้ Crystal 19.6608 MHz

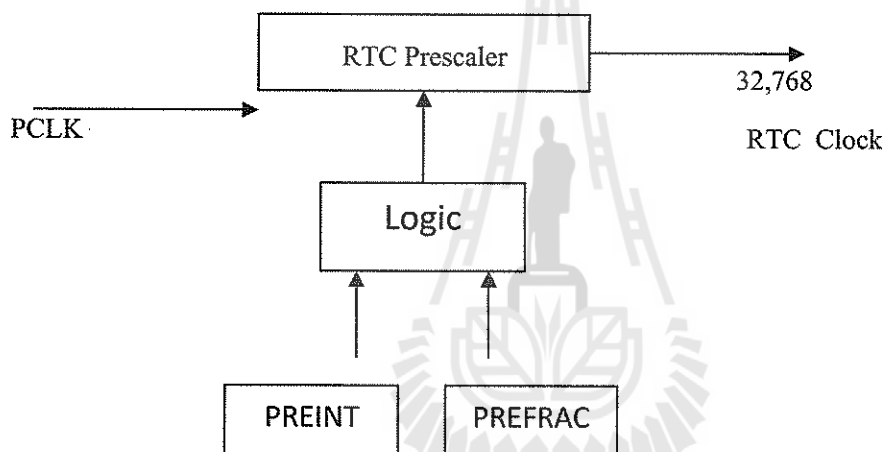
ดังนั้นได้ $f_{osc} = 19.6608 \text{ MHz}$ ต้องการความถี่ 58.9824 MHz จะได้ตัวคูณ $M=5$ และ $P = 2$ จะ

ได้ $CCLK = 5 \times 19.6608 \text{ MHz} = 58.9824 \text{ MHz}$

การกำหนดค่าเริ่มต้นให้กับ โมดูลนาฬิกาเวลาจริง

ในการทำงานของโมดูล RTC จำเป็นต้องมีสัญญาณนาฬิกาควบคุมจังหวะการทำงาน โดยในไมโครคอนโทรลเลอร์ตระกูล LCP2000 รุ่นเก่า นำค่าความถี่ของ PCLK มาหารค่าให้ได้ความถี่ $32,768 \text{ Hz}$ จึงไม่จำเป็นต้องต่อคริสตัลภายนอกเพื่อกำเนิดความถี่ แต่ภายในไมโครคอนโทรลเลอร์รุ่นเก่า เช่น LPC2119 ไม่มีวงจรแบ็คอัพแบตเตอรี่เพื่อจ่ายไฟเลี้ยงให้โมดูล RTC ทำงานตลอดเวลาเพื่อป้องกันค่าเวลาที่ตั้งไว้หายไปเมื่อไม่มีไฟป้อนให้วงจรทำให้ไม่สามารถปิดไฟเลี้ยงแผงวงจรในการเขียนโปรแกรมต้องเขียนโปรแกรมให้ทำงานในโหมดประหยัดพลังงานให้มากที่สุด

ภายใน RTC มีวงจรหารความถี่จาก PCLK ค่าใดๆก็ได้ให้มีความถี่ $32,768 \text{ Hz}$ โดยควบคุมการหารความถี่ได้จากรีจิสเตอร์ PREINT และ PREFRAC ดังแสดงในรูปที่ 3.20



รูปที่ 3.20 แสดงวงจร RTC

ค่ารีจิสเตอร์ PREINT เป็นค่าตัวหารที่เป็นเลขจำนวนเต็ม และ PREFRAC เป็นค่าตัวหารส่วนที่เป็นเศษเหลือ โดยเขียนเป็นสมการได้ดังนี้

$$PREINT = (\text{int})(PCLK / 32768) - 1$$

$$PREFRAC = PCLK - ((PREINT + 1) \times 32768)$$

ตัวอย่างในแผงวงจร ET - ARM STAMP LPC2119 ใช้คริสตัลความถี่ 19.6608 MHz ใช้วงจร PLL คูณ 3 ให้ได้ค่า $CCLK = 58.9824 \text{ MHz}$ กำหนดค่าหารความถี่ได้ $PCLK = 29.4912 \text{ MHz}$ จึงคำนวณค่าของ PREINT ได้ดังนี้

$$PREINT = (29,491,200 / 32768) - 1 = 899 \text{ ลงตัวพอดี ซึ่งในกรณีนี้ค่า}$$

ของ PREFRAC = 0

ตัวอย่างของแผงวงจรอื่น ที่มีค่า $PCLK = 30 \text{ MHz}$ จะคำนวณค่าได้ดังนี้

$$PREINT = (\text{int})(30,000,000 / 32768) - 1 = 914$$

$$PREFRAC = 30,000,000 - ((914 + 1) \times 32768) = 17,280$$

เมื่อคำนวณได้ค่าแล้วให้นำไปเขียนลงในรีจิสเตอร์ PREINT และ PREFRAC แล้วให้ตั้ง enable โมดูล RTC ก็จะพร้อมทำงานแล้ว ในกรณีของของแผงวงจร ET-ARM STAMP LPC2119 สามารถเขียนค่าได้ดังนี้

```
PREINT = 0 × 00000383; // Set RTC prescaler
```

```
PREFRAC = 0 × 00000000;
```

```
CCR = 0 × 00000001; // Start the RTC
```

สำหรับแผงวงจร JX-2148 หรือ CR-JR ARM7 USB-LPC2148 ที่มีการต่อใช้ความถี่จากวงจรออสซิลเลเตอร์ ไม่จำเป็นต้องกำหนดค่าของ PREINT และ PREFRAC แต่จะต้องกำหนดที่บิตที่ 4 ของรีจิสเตอร์ CCR ให้มีค่าเป็น 1 เพื่อนำค่าความถี่ของวงจรออสซิลเลเตอร์ภายนอกป้อนให้โมดูล RTC

การกำหนด Port ใช้งาน

P0[31...24]	P0 [23...16]	P0 [15...8]	P0 [7...0]
	LED_R&G		Switch input

P1 [31...24]	P1 [23...16]	P1 [15...8]	P1 [7...1]	0
AH	AL			Sc

P2 [31...24]	P2 [23...16]	P2 [15...8]	P2 [7...0]
	Sensor input	BH	BL

P3 [31...24]	P3 [23...16]	P3 [15...8]	P3 [7...0]
DH	DL	CH	CL

Switch input = อ่านค่าจากปุ่มกดด้วยมือ

Sensor input = อ่านค่าจากตัว Sensor

LED_R&G = แสดงผลสัญญาณไฟ เขียว/แดง ทั้งหมด

AH = แสดง 7-Segment หลักรับบนของเส้นทาง A

AL = แสดง 7-Segment หลักรับล่างของเส้นทาง A

BH = แสดง 7-Segment หลักรับบนของเส้นทาง B

BL = แสดง 7-Segment หลักรับล่างของเส้นทาง B

CH	=	แสดง 7-Segment หลักบนของเส้นทาง C
CL	=	แสดง 7-Segment หลักล่างของเส้นทาง C
DH	=	แสดง 7-Segment หลักบนของเส้นทาง D
DL	=	แสดง 7-Segment หลักล่างของเส้นทาง D
SC	=	ควบคุมการทำงานของ Sensor

การกำหนดค่าตัวแปรต่างๆ

TM =15	TA_buff="สำรองค่าเวลาในเส้นทาง A"
TA ="ค่าเวลาในเส้นทาง A"	TB_buff="สำรองค่าเวลาในเส้นทาง B"
TB ="ค่าเวลาในเส้นทาง B"	TC_buff="สำรองค่าเวลาในเส้นทาง C"
TC ="ค่าเวลาในเส้นทาง C"	TD_buff="สำรองค่าเวลาในเส้นทาง D"
TD ="ค่าเวลาในเส้นทาง D"	turn_buff="นับรอบวนซ้ำ"
Stat ="ค่าสถานะการทำงาน"	
stat_buff ="สำรองค่าสถานะการทำงาน"	

โปรแกรมย่อย แสดงผล 7-Segment

```

void SHOW_TIME ()
{
  unsigned short
  dis[100]={0x003f,0x0006,0x00ab,0x008f,0x0096,0x009d,0x00bd,0x0007,0x00bf,0x009f,
0x063f,0x0606,0x06ab,0x068f,0x0696,0x069d,0x06bd,0x0607,0x06bf,0x069f,
0xab3f,0xab06,0xabab,0xab8f,0xab96,0xab9d,0xabbd,0xab07,0xabbf,0xab9f,
0x8f3f,0x8f06,0x8fab,0x8f8f,0x8f96,0x8f9d,0x8fbd,0x8f07,0x8fbf,0x8f9f,
0x963f,0x9606,0x96ab,0x968f,0x9696,0x969d,0x96bd,0x9607,0x96bf,0x969f,
0x9d3f,0x9d06,0x9dab,0x9d8f,0x9d96,0x9d9d,0x9dbd,0x9d07,0x9dbf,0x9d9f,
0xbd3f,0xbd06,0xbdab,0xbd8f,0xbd96,0xbd9d,0xbdbd,0xbd07,0xbdbf,0xbd9f,
0x073f,0x0706,0x07ab,0x078f,0x0796,0x079d,0x07bd,0x0707,0x07bf,0x079f,
0xbf3f,0xbf06,0xbfab,0xbf8f,0xbf96,0xbf9d,0xbfbf,0xbf07,0xbfbf,0xbf9f,
0x9f3f,0x9f06,0x9fab,0x9f8f,0x9f96,0x9f9d,0x9fbd,0x9f07,0x9fbf,0x9f9f};

  IOSET1|=0xFFFF0000;
  buff_A = dis[TA]<<16;
  IOCLR1 |= buff_A;
  IOSET2|=0x0000FFFF;
  buff_B = dis[TB];
  IOCLR2 |= buff_B;
  IOSET3|=0xFFFFFFFF;
  buff_C=dis[TC];
  buff_D=dis[TD]<<16;
  buff_D&=0xFFFF0000;
  IOCLR3 |= buff_C;
  IOCLR3 |= buff_D;
}

```

โปรแกรมย่อย หน่วงเวลา 1 วินาที


```

void delay_sec()
{
    Second = CTIME0 & MASKSEC;
    Last_Second = Second;
    while(Last_Second == Second){Second = CTIME0 & MASKSEC;}
}

```

รายละเอียดโปรแกรมทั้งหมด

```

#include <LPC22xx.H>
#define uchar unsigned char
#define uint unsigned int
//Input pin mark
#define ID 0x00000008
#define IC 0x00000004
#define IB 0x00000002
#define IA 0x00000001
#define IALL (IA|IB|IC|ID)
//Output pin mark
#define XALL 0x00000001
#define RD 0x00010000
#define RC 0x00010000
#define RB 0x00020000
#define RA 0x00020000
#define GD 0x00040000
#define GC 0x00040000
#define GB 0x00080000
#define GA 0x00080000
#define OALL (RA|RB|RC|RD|GA|GB|GC|GD)
//pin contron
#define XALL_OFF() IOCLR1 |= XALL
#define XALL_ON() IOSET1 |= XALL
#define RA_ON() IOCLR0 |= RA
#define RA_OFF() IOSET0 |= RA
#define RB_ON() IOCLR0 |= RB
#define RB_OFF() IOSET0 |= RB
#define RC_ON() IOCLR0 |= RC
#define RC_OFF() IOSET0 |= RC
#define RD_ON() IOCLR0 |= RD
#define RD_OFF() IOSET0 |= RD
#define GA_ON() IOCLR0 |= GA
#define GA_OFF() IOSET0 |= GA
#define GB_ON() IOCLR0 |= GB
#define GB_OFF() IOSET0 |= GB
#define GC_ON() IOCLR0 |= GC
#define GC_OFF() IOSET0 |= GC
#define GD_ON() IOCLR0 |= GD
#define GD_OFF() IOSET0 |= GD
#define MASKSEC 0x3F
unsigned char BIT_STEP;
unsigned long int buff_A,buff_B,buff_C,buff_D,IO_buff,IO_buff_1,IO_buff_2;
unsigned int Second,Last_Second;
short TA,TB,TC,TD,TM,stat,stat_buff,TA_buff,TB_buff,
TC_buff,TD_buff,turn_buff,TX,temp;

void SHOW_TIME()
{

```

```

unsigned short
dis[100]={0x003f,0x0006,0x00ab,0x008f,0x0096,0x009d,0x00bd,0x0007,0x00bf,0x009f,
0x063f,0x0606,0x06ab,0x068f,0x0696,0x069d,0x06bd,0x0607,0x06bf,0x069f,
0xab3f,0xab06,0xabab,0xab8f,0xab96,0xab9d,0xabbd,0xab07,0xabbf,0xab9f,
0x8f3f,0x8f06,0x8fab,0x8f8f,0x8f96,0x8f9d,0x8fbd,0x8f07,0x8fbf,0x8f9f,
0x963f,0x9606,0x96ab,0x968f,0x9696,0x969d,0x96bd,0x9607,0x96bf,0x969f,
0x9d3f,0x9d06,0x9dab,0x9d8f,0x9d96,0x9d9d,0x9dbd,0x9d07,0x9dbf,0x9d9f,
0xbd3f,0xbd06,0xbdab,0xbd8f,0xbd96,0xbd9d,0xbdbd,0xbd07,0xbdbf,0xbd9f,
0x073f,0x0706,0x07ab,0x078f,0x0796,0x079d,0x07bd,0x0707,0x07bf,0x079f,
0xbf3f,0xbf06,0xbfab,0xbf8f,0xbf96,0xbf9d,0xbfbD,0xbf07,0xbfbf,0xbf9f,
0x9f3f,0x9f06,0x9fab,0x9f8f,0x9f96,0x9f9d,0x9fbd,0x9f07,0x9fbf,0x9f9f};

IOSET1|=0xFFFF0000;
buff_A = dis[TA]<<16;
IOCLR1 |= buff_A;
IOSET2|=0x0000FFFF;
buff_B = dis[TB];
IOCLR2 |= buff_B;
IOSET3|=0xFFFFFFFF;
buff_C=dis[TC];
buff_D=dis[TD]<<16;
buff_D&=0xFFFF0000;
IOCLR3 |= buff_C;
IOCLR3 |= buff_D;
}
void RED_ALL()
{
    RA_ON();
    RB_ON();
    RC_ON();
    RD_ON();
    GA_OFF();
    GB_OFF();
    GC_OFF();
    GD_OFF();
}
void delay_sec()
{
    Second = CTIME0 & MASKSEC;
    Last_Second = Second;
    while(Last_Second == Second){Second = CTIME0 & MASKSEC;}
}
int main()
{
    PINSEL0 = 0x00000000;
    PINSEL1 = 0x00000000;
    PINSEL2 = 0x00000030;
    IODIRO = 0xFFFFFFFF00;
    IODIR1 = 0xFFFF0001;
    IODIR2 = 0xFF00FFFF;
    IODIR3 = 0xFFFFFFFF;
    PREINT = 0x00000399;
    PREFRAC = 0x00000000;
    CCR &= 0x00;
    CCR |= 0x02;
    CCR &= 0xFD;
    CCR |= 0x01;

    BIT_STEP=0;
    TX=5;
    TM=15;
    TA=TM;
}

```



```

        TA_buff=TD+TM-TX;
        TB=TA_buff+TM+TX;
        TC_buff=TM;
        turn_buff=0;
        stat_buff='C';
    }
    else if ((IO_buff & ID) == BIT_STEP)
    {
        TD=TA;
        TA_buff=TD+TM-TX;
        TB=TA_buff+TM+TX;
        TC=TB+TM;
        TD_buff=TM;
        turn_buff=0;
        stat_buff='D';
    }
    else if (turn_buff <= 2)
    {
        TA_buff=0;
        TA_buff=TM+TM;
        if (turn_buff == 1)
        {
            TA_buff+=TM;
        }
        temp=TA;
        TA+= TA_buff;
        TB=TA;
        TC=TB+TM;
        TD=TC+TM;
        turn_buff++;
        IO_buff = 0x00000000;
        IO_buff_1 = 0x00000000;
        IO_buff_2 = 0x00000000;
        if (turn_buff >= 3)
        {
            TA=temp;
            TB=TA;
            TC=TB+TM;
            TD=TC+TM;
            TA_buff=TD+TM-TX;
            TB_buff=TM;
            stat_buff='B';
        }
    }
    }
    XALL_OFF();
}
}
stat=stat_buff;
TA+=TA_buff;
TB+=TB_buff;
TC+=TC_buff;
TD+=TD_buff;
turn_buff=0;
GA_OFF();
RED_ALL();
break;

case 'B' :
    RED ALL();

```

```

RB_OFF();
GB_ON();
IO_buff = 0x00000000;
IO_buff_1 = 0x00000000;
IO_buff_2 = 0x00000000;
turn_buff=0;
while (TB>=1)
{
SHOW_TIME();
delay_sec();
TA--;
TB--;
TC--;
TD--;

if (TB==7)
{XALL_ON();}

if ((TB<=6) && (TB>=5))
{
IO_buff_1 |= ~IOPIN2;
IO_buff_2 |= (IO_buff_1>>16);
IO_buff_2 &= 0x0000000F;
IO_buff |= IOPIN0;
IO_buff &= 0x0000000F;
IO_buff &= IO_buff_2;
}

if (TB==TX)
{
TA_buff=0;
TB_buff=0;
TC_buff=0;
TD_buff=0;

if ((IO_buff & IC) == BIT_STEP )
{
TC=TB;
TD=TC+TM;
TA=TD+TM;
TB_buff=TA+TM-TX;
TC_buff=TM;
turn_buff=0;
stat_buff='C';
}
else if ((IO_buff & ID) == BIT_STEP)
{
TD=TB;
TA=TD+TM;
TB_buff=TA+TM-TX;
TC=TB_buff+TM+TX;
TD_buff=TM;
turn_buff=0;
stat_buff='D';
}
else if ((IO_buff & IA) == BIT_STEP)
{
TA=TB;
TB_buff=TA+TM-TX;
TC=TB_buff+TM+TX;
TD=TC+TM;
}
}

```

```

        TA_buff=TM;
        turn_buff=0;
        stat_buff='A';
    }
    else if (turn_buff <= 2)
    {
        TB_buff=0;
        TB_buff=TM+TM;
        if (turn_buff == 1)
        {
            TB_buff+=TM;
        }
        temp=TB;
        TB+=TB_buff;
        TC=TB;
        TD=TC+TM;
        TA=TD+TM;
        turn_buff++;
        IO_buff = 0x00000000;
        IO_buff_1 = 0x00000000;
        IO_buff_2 = 0x00000000;
        if (turn_buff >= 3)
        {
            TB=temp;
            TC=TB;
            TD=TC+TM;
            TA=TD+TM;
            TB_buff=TA+TM-TX;
            TC_buff=TM;
            turn_buff=0;
            stat_buff='C';
        }
    }
    XALL_OFF();
}
}

stat=stat_buff;
TA+=TA_buff;
TB+=TB_buff;
TC+=TC_buff;
TD+=TD_buff;
turn_buff=0;
GB_OFF();
RED_ALL();
break;

case 'C' :
    RED_ALL();
    RC_OFF();
    GC_ON();
    IO_buff = 0x00000000;
    IO_buff_1 = 0x00000000;
    IO_buff_2 = 0x00000000;
    turn_buff=0;
    while (TC>=1)
    {
        SHOW_TIME();
        delay_sec();

        TA--;
        TB--;

```

```

TC--;
TD--;

if (TC==7)
{XALL_ON();}
if ((TC<=6)&&(TC>=5))
{
IO_buff_1 |= ~IOPIN2;
IO_buff_2 |= (IO_buff_1>>16);
IO_buff_2 &= 0x0000000F;
IO_buff |= IOPIN0;
IO_buff &= 0x0000000F;
}

if (TC==TX)
{
TA_buff=0;
TB_buff=0;
TC_buff=0;
TD_buff=0;
IO_buff &= IO_buff_2;
if ((IO_buff & ID) == BIT_STEP )
{
TD=TC;
TA=TD+TM;
TB=TA+TM;
TC_buff=TB+TM-TX;
TD_buff=TM;
turn_buff=0;
stat_buff='D';
}
else if ((IO_buff & IA) == BIT_STEP)
{
TA=TC;
TB=TA+TM;
TC_buff=TB+TM-TX;
TD=TC_buff+TM+TX;
TA_buff=TM;
turn_buff=0;
stat_buff='A';
}
else if ((IO_buff & IB) == BIT_STEP)
{
TB=TC;
TC_buff=TB+TM-TX;
TD=TC_buff+TM+TX;
TA=TD+TM;
TB_buff=TM;
turn_buff=0;
stat_buff='B';
}
}
else if (turn_buff <= 2)
{
TC_buff=TM+TM;
if (turn_buff == 1)
{
TC_buff+=TM;
}
temp=TC;
TC+=TC_buff;
TD=TC;
}

```

```

        TA=TD+TM;
        TB=TA+TM;
        turn_buff++;
        IO_buff = 0x00000000;
        IO_buff_1 = 0x00000000;
        IO_buff_2 = 0x00000000;
        if (turn_buff >= 3)
        {
            TC=temp;
            TD=TC;
            TA=TD+TM;
            TB=TA+TM;
            TC_buff=TB+TM-TX;
            TD_buff=TM;
            turn_buff=0;
            stat_buff='D';
        }
    }
    XALL_OFF();
}
}
stat=stat_buff;
TA+=TA_buff;
TB+=TB_buff;
TC+=TC_buff;
TD+=TD_buff;
turn_buff=0;
GC_OFF();
RED_ALL();
break;

case 'D' :
    RED_ALL();
    RD_OFF();
    GD_ON();
    IO_buff = 0x00000000;
    IO_buff_1 = 0x00000000;
    IO_buff_2 = 0x00000000;
    turn_buff=0;
    while (TD>=1)
    {
        SHOW_TIME();
        delay_sec();
        TA--;
        TB--;
        TC--;
        TD--;

        if (TD==7)
        {XALL_ON();}

        if ((TD<=6) && (TD>=5))
        {
            IO_buff_1 |= ~IOPIN2;
            IO_buff_2 |= (IO_buff_1>>16);
            IO_buff_2 &= 0x0000000F;
            IO_buff |= IOPIN0;

            IO_buff &= 0x0000000F;
        }
    }
}

```



```

if (TD==TX)
{
    TA_buff=0;
    TB_buff=0;
    TC_buff=0;
    TD_buff=0;
    IO_buff &= IO_buff_2;
    if ((IO_buff & IA) == BIT_STEP )
    {
        TA=TD;
        TB=TA+TM;
        TC=TB+TM;
        TD_buff=TC+TM-TX;
        TA_buff=TM;
        turn_buff=0;
        stat_buff='A';
    }
    else if ((IO_buff & IB) == BIT_STEP)
    {
        TB=TD;
        TC=TB+TM;
        TD_buff=TC+TM-TX;
        TA=TD_buff+TM+TX;
        TB_buff=TM;
        turn_buff=0;
        stat_buff='B';
    }
    else if ((IO_buff & IC) == BIT_STEP)
    {
        TC=TD;
        TD_buff=TC+TM;
        TA=TD_buff+TM;
        TB=TA+TM;
        TC_buff=TM;
        turn_buff=0;
        stat_buff='C';
    }
    else if (turn_buff <= 2)
    {
        TD_buff=TM+TM;
        if (turn_buff == 1)
        {
            TD_buff+=TM;
        }
        temp=TD;
        TD+=TD_buff;
        TA=TD;
        TB=TA+TM;
        TC=TB+TM;
        turn_buff++;
        IO_buff = 0x00000000;
        IO_buff_1 = 0x00000000;
        IO_buff_2 = 0x00000000;
        if (turn_buff >= 3)
        {
            TD=temp;
            TA=TD;

```

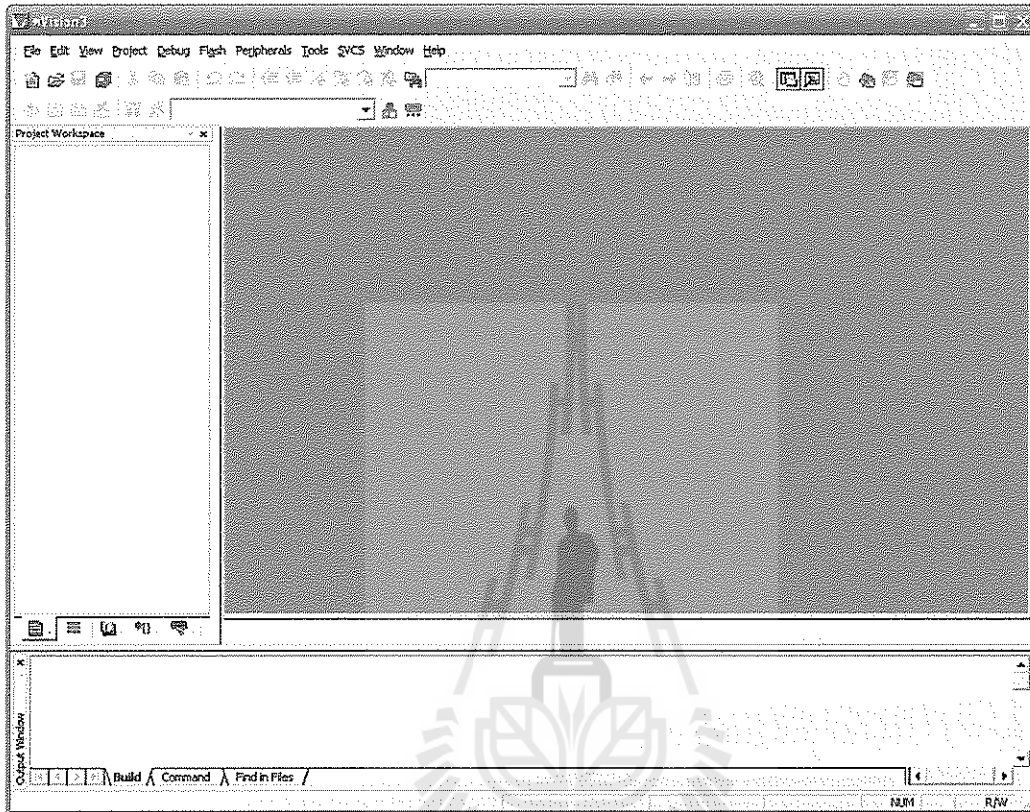
```

        TB=TA+TM;
        TC=TB+TM;
        TD_buff=TC-TX;
        TA_buff=TM;
        turn_buff=0;
        stat_buff='A';
    }
}
XALL_OFF();
}
}
stat=stat_buff;
TA+=TA_buff;
TB+=TB_buff;
TC+=TC_buff;
TD+=TD_buff;
turn_buff=0;
GD_OFF();
RED_ALL();
break;
default:
stat='A';
TA=TM;
TB=TA;
TC=TB+TM;
TD=TC+TM;
turn_buff=0;
break;
}
}
}

```

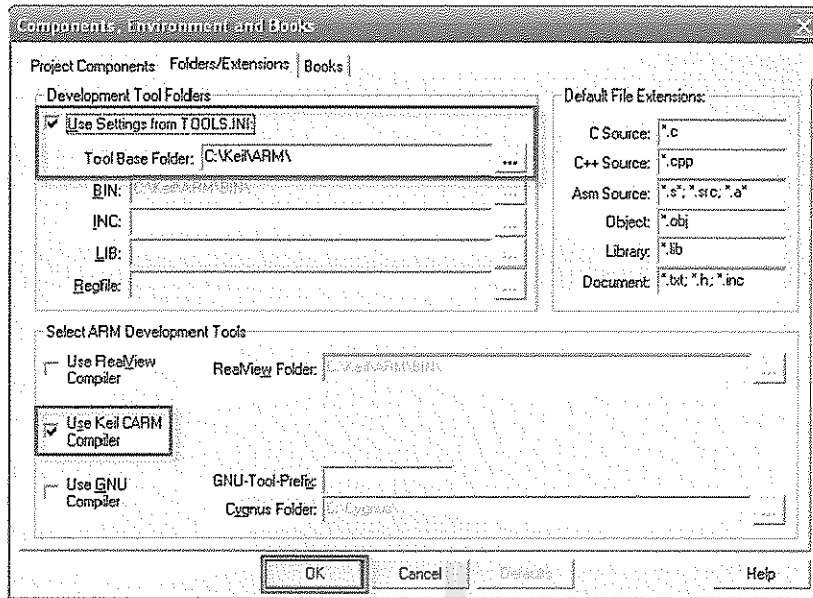
การสร้าง Project File ของ Keil ARM

1. เปิดโปรแกรม Keil uVision3 ซึ่งเป็นโปรแกรม Text Editor ของ Keil-CARM ใช้สำหรับใช้ในการเขียน โปรแกรมที่เป็น Source Code ภาษาซี โดยจะมีลักษณะดังรูปที่ 3.21



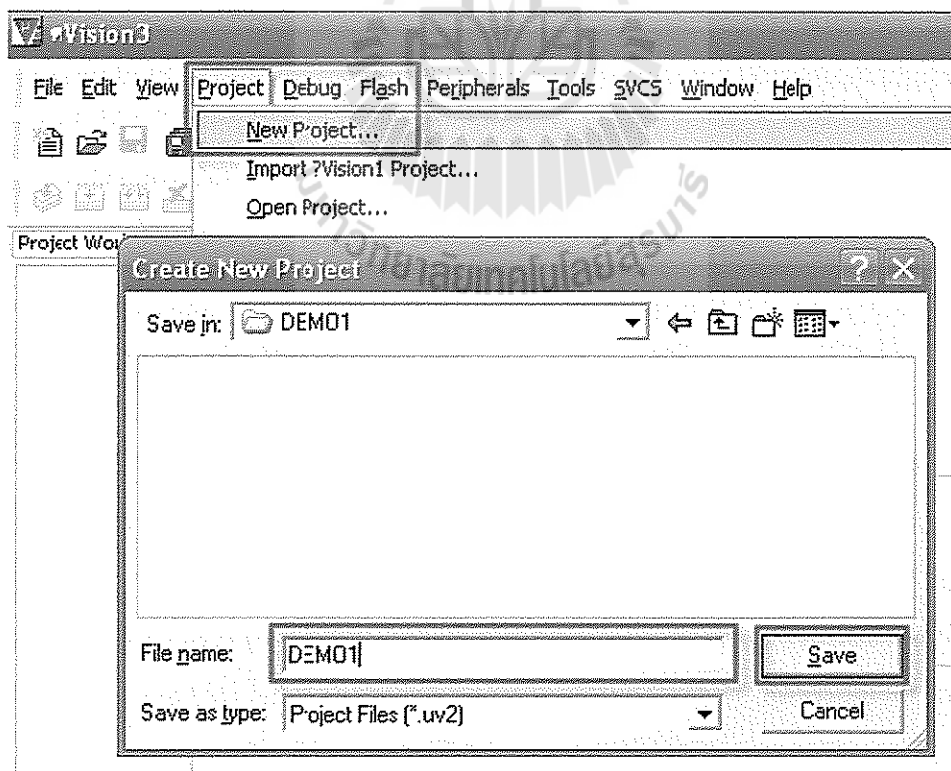
รูปที่ 3.21

2. ทำการกำหนดค่าตัวเลือกในการแปลคำสั่งของ uVision3 ให้ใช้งานกับโปรแกรม Keil uVision3 และ Keil-CARM โดยให้เลือกคลิกเมาส์ที่เมนูคำสั่ง Project ► Components ,Environment, Books... จากนั้นให้เลือกค่าตัวเลือกสำหรับกำหนดการใช้งาน Compiler จากหัวข้อ Select ARM Development Tools ซึ่งจะมีค่าตัวเลือกอยู่ 3 แบบ คือ Use Keil-CARM Tools ,Use GNU Tools และ Use ARM Tools โดยให้เลือกเป็น “Use Keil ARM Tools” จากนั้นให้ทำการกำหนดตำแหน่ง Folder สำหรับเก็บค่าตัวเลือกการทำงานของโปรแกรม Keil ARM ซึ่งตามปรกติแล้วจะเป็น “C:\Keil\ARM” แต่ถ้าติดตั้ง Keil ไว้ที่อื่นก็ต้องปรับเปลี่ยนให้ถูกต้องตามความเป็นจริงด้วยดังรูปที่ 3.22



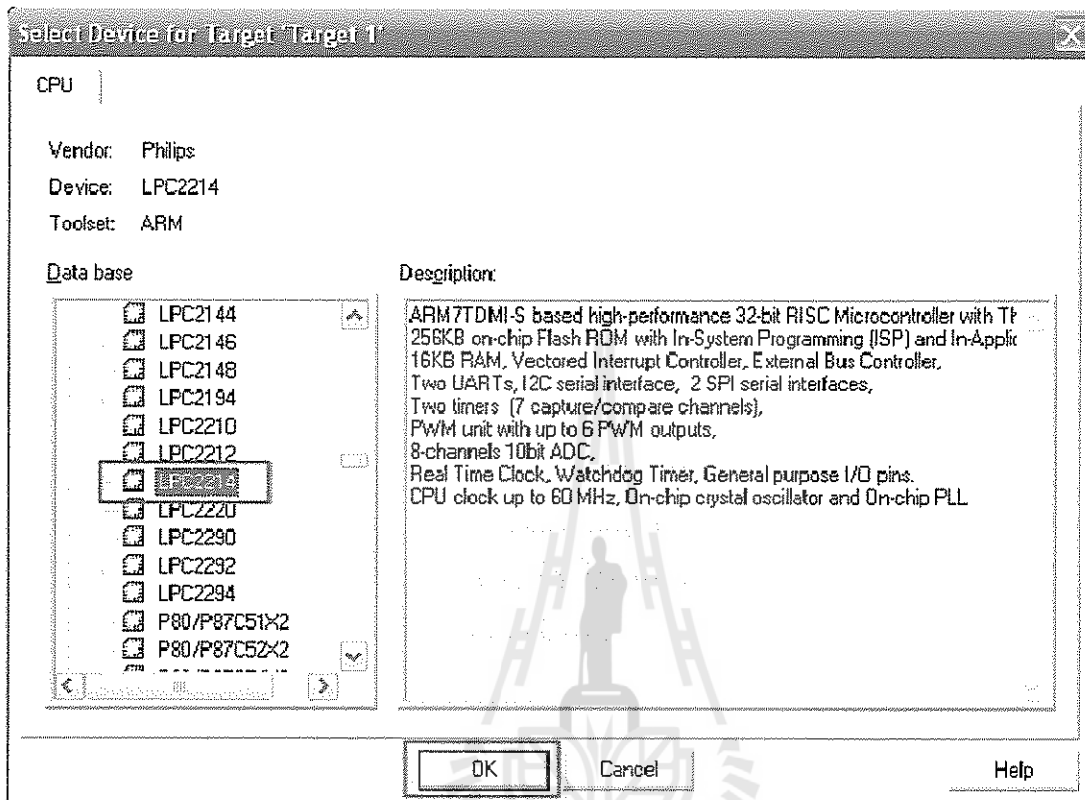
รูปที่ 3.22

3. ทำการสร้าง Project File ขึ้นมาใหม่ โดยเรียกเมนูคำสั่ง Project → New Project จากนั้นให้เลือกกำหนดหรือสร้างตำแหน่ง Folder ที่จะบันทึก Project File พร้อมกับกำหนดชื่อ Project File ตามต้องการ เช่น ถ้าต้องการสร้าง Project File ชื่อ DEMO1 โดยเก็บไว้ใน Folder ชื่อ DEMO1 ก็สามารถกำหนดตำแหน่ง Folder และชื่อ Project File ได้เอง โดยเมื่อกำหนดชื่อในช่อง File name แล้วให้เลือก Save เพื่อบันทึก Project File ไว้ดังรูปที่ 3.23



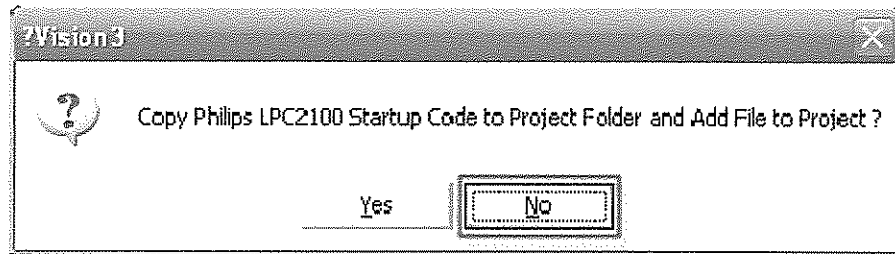
รูปที่ 3.23

หลังจากกำหนดชื่อและตั้ง Save Project File แล้ว โปรแกรมจะรอให้ผู้ใช้ทำการกำหนด เบอร์ MCU ที่ จะใช้งานใน Project ที่ตั้ง Save นั้น ซึ่งในกรณีที่ใช้งานกับบอร์ด ET-ARM7 LPC2114 นั้น ให้เลือกกำหนด เป็น MCU ของ Philips เบอร์ LPC2114 แล้วเลือก OK ดังรูปที่ 3.24



รูปที่ 3.24

หลังจากเลือกกำหนดเบอร์ของ MCU เป็นที่เรียบร้อยแล้ว ในขั้นตอนนี้โปรแกรมจะรอให้ผู้ใช้ยืนยันว่า ต้องการจะทำการ Copy ไฟล์ Startup ของ Keil เพื่อใช้งานกับ MCU ของ Philips มาใช้ใน Project ด้วยหรือไม่ โดย Startup ไฟล์จะเป็นส่วนของการกำหนดค่าเริ่มต้นการทำงานให้กับ MCU เช่น การกำหนดค่า Stack และการกำหนดค่าการทำงานให้กับ Phase-Lock-Loop ต่างๆ ก่อนที่จะเริ่มต้นทำงานตามโปรแกรมที่เราเขียนขึ้น ไม่เช่นนั้นแล้วโปรแกรมที่เราเขียนขึ้นมานั้นจะต้องเพิ่มคำสั่งในการเตรียมการทำงานส่วนเหล่านี้ให้ MCU เองทั้งหมด แต่เนื่องจากไฟล์ Startup ของ Keil-ARM นั้น เป็นไฟล์ภาษาแอสเซมบลี ซึ่งกำหนดค่าการทำงานไว้กับชุดพัฒนาของ Keil เอง ดังนั้นข้อกำหนดและการกำหนดค่าบางอย่างจะมีความแตกต่างกันอยู่กับค่าที่ต้องการ สำหรับ บอร์ด “ET-ARM7 LPC2114” ไม่สามารถใช้งานไฟล์ Startup ได้ทันที ต้องมีการแก้ไขค่าตัวเลือกใหม่ดังนั้นก่อนที่จะใช้โปรแกรม Keil-CARM ในการแปลคำสั่งให้ นั้น ผู้ใช้จะต้องเข้าไปแก้ไขไฟล์ Startup ใหม่โดยต้องกำหนดรูปแบบให้ถูกต้องตรงกับความต้องการของบอร์ด “ET-ARM7 LPC2114” ด้วย ดังนั้นในที่นี้ขอแนะนำให้เลือก “No” เพื่อไม่ให้ Keil uVision3 ทำการ Copy ไฟล์ Startup ของ Keil-CARM มาใช้ใน Project นี้ด้วยดังรูปที่ 3.25

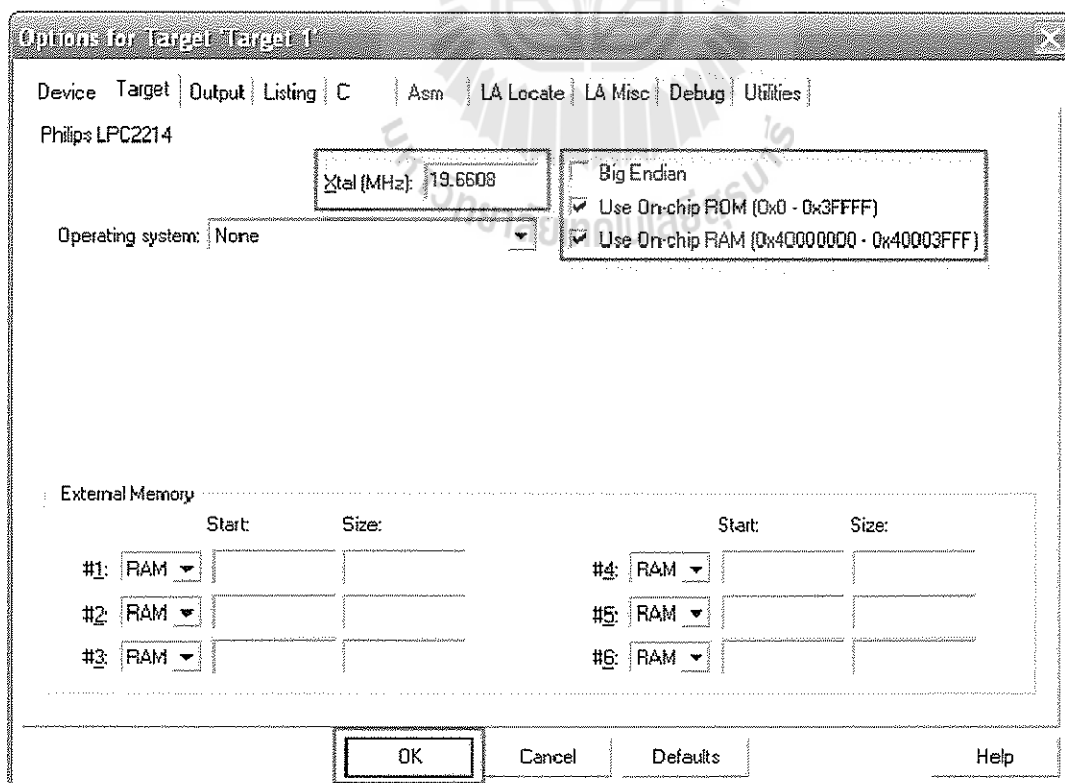


รูปที่ 3.25

4. ให้ทำการ Copy File ชื่อ “Startup.s” ที่ทาง อีทีที จัดเตรียมไว้ใน CD-ROM ซึ่งเก็บไว้ใน Example ชื่อ “Startup.s” มาไว้ในตำแหน่ง Folder เดียวกันกับ Project File ที่สร้างขึ้นมาใหม่โดยไฟล์ “Startup.s” จะเป็นไฟล์ซึ่งบรรจุคำสั่งภาษาแอสเซมบลีของ ARM7 สำหรับทำหน้าที่กำหนดค่าเริ่มต้นการทำงานที่จำเป็นให้กับ MCU ซึ่งได้แก่การ กำหนดค่า Stack ให้กับ MCU การ Initial Phase-Lock-Loop การกำหนดค่าให้กับ MAM Function และการกำหนดตำแหน่ง Vector ต่างๆของ MCU สำหรับใช้งานร่วมกับบอร์ด “ET-ARM7 LPC2114” ซึ่งถ้าสั่ง Add ไฟล์ “Startup.s” จาก Keil หรือ Copy ไฟล์ดังกล่าวมาจากแหล่งอื่นๆ อาจมีการทำงานของโปรแกรมใน Startup ไม่เหมือนกัน ซึ่งจะส่งผลกระทบต่อการทำงานของโปรแกรมที่จะเขียนขึ้นด้วย

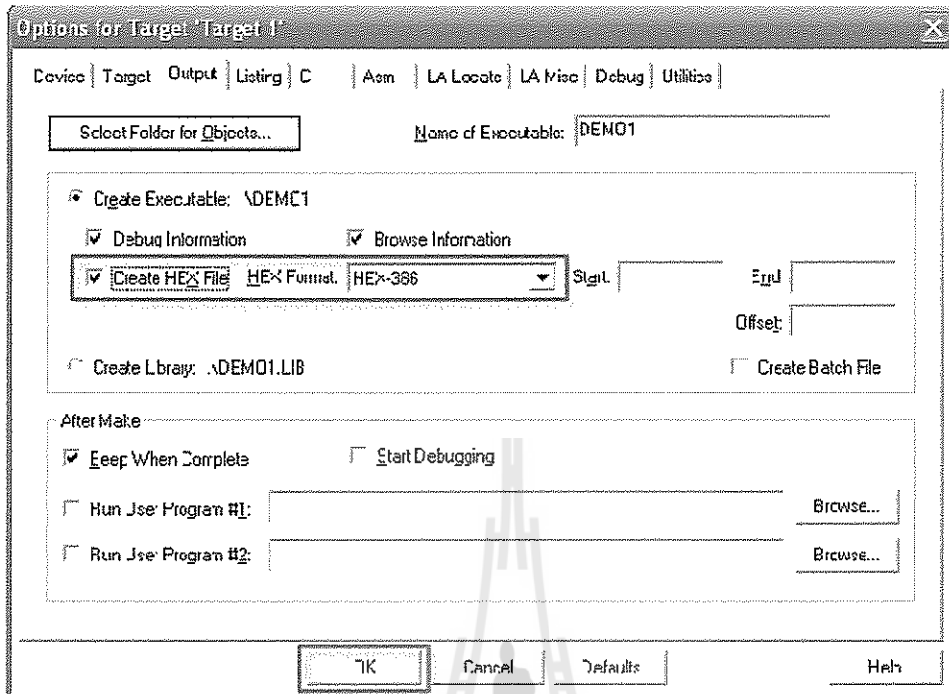
5. ให้ทำการกำหนดค่า Option ของ Project File โดยเลือกเมนูคำสั่ง Project → Option for Target 'Target 1' จากนั้นเลือกที่ Tab ของ Target เพื่อกำหนดค่าของ MCU Target โดยให้กำหนดดังนี้

5.1 X-TAL ให้กำหนดเป็น 19.6608 MHz พร้อมกับเลือกกำหนดให้ใช้หน่วยความจำที่มีอยู่ใน MCU เป็นเงื่อนไขในการแปลโปรแกรมของ Keil-CARM ด้วย ดังรูปที่ 3.26



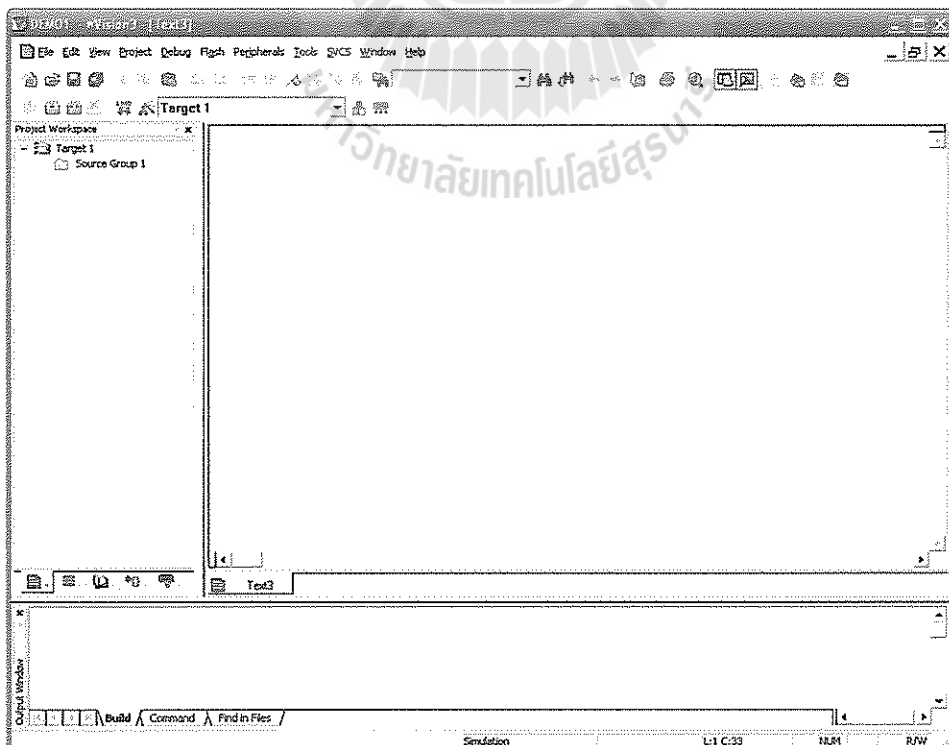
รูปที่ 3.26

5.2 Output ให้เลือกคลิกเมาส์ที่ค่าตัวเลือก Create HEX File พร้อมกับเลือกกำหนดรูปแบบของ Hex ให้เป็นแบบ HEX-386 แล้วเลือก OK ดังรูปที่ 3.27



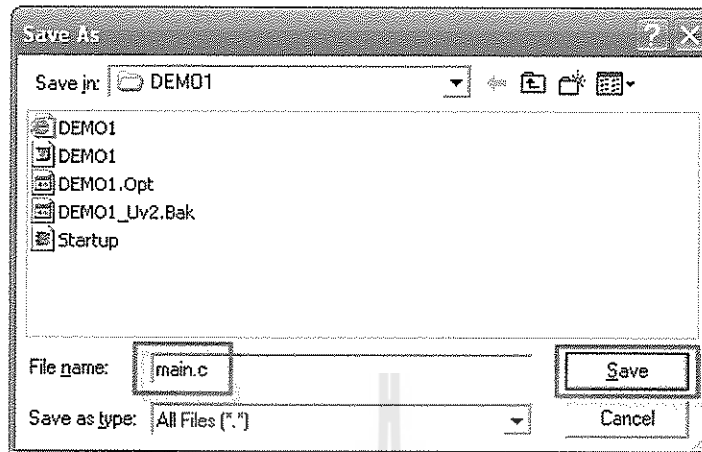
รูปที่ 3.27

6. เริ่มต้นเขียน Source Code ภาษาซี โดยให้เลือกคลิกเมาส์ที่เมนูคำสั่ง File → New... ซึ่งจะได้อพื้นที่ในการเขียน Text File เกิดขึ้นมา โดยในครั้งแรกจะกำหนดชื่อตามค่า Default เป็น "Text1" ดังรูปที่ 3.28



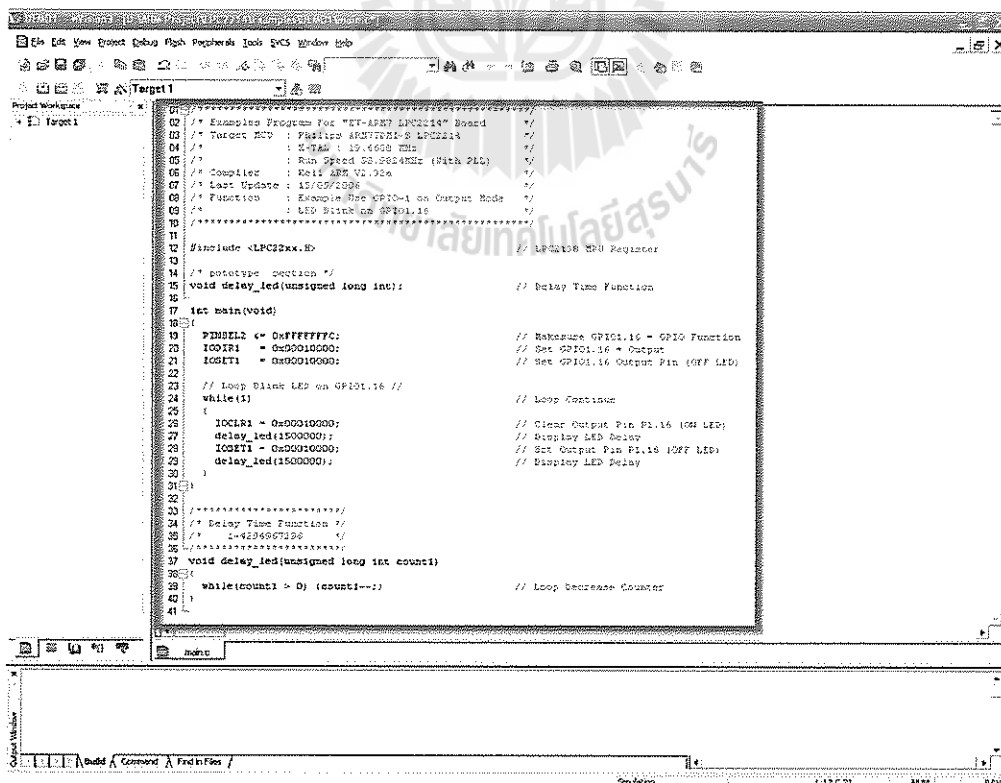
รูปที่ 3.28

หลังจากพิมพ์คำสั่งภาษาซีเสร็จเรียบร้อยแล้วตามต้องการแล้ว ให้สั่ง Save ไฟล์ดังกล่าว โดยต้องกำหนด เป็นไฟล์ที่มีนามสกุลเป็น “.C” ในที่นี้ขอแนะนำให้สั่ง Save โดยใช้คำสั่ง File → Save As... แล้วกำหนด ชื่อและนามสกุลของไฟล์เป็น “main.c” ดังรูปที่ 3.29



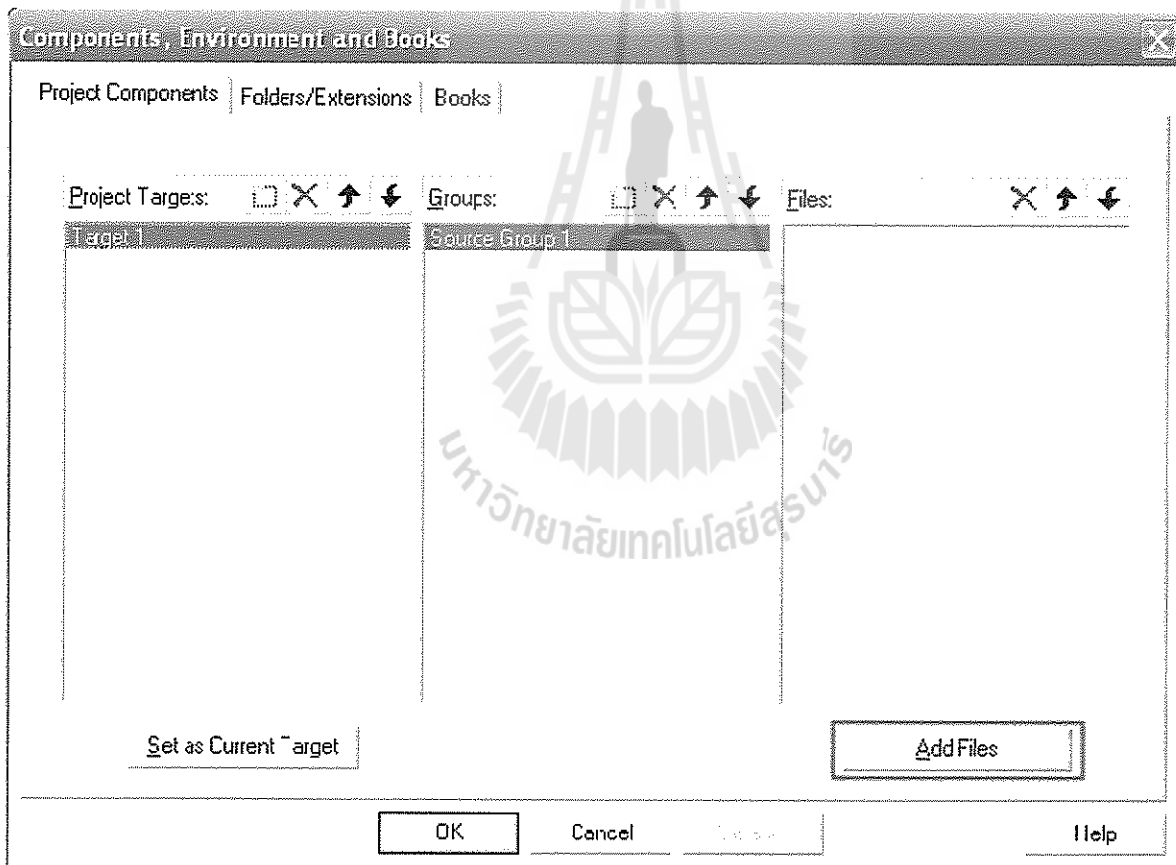
รูปที่ 3.29

ซึ่งหลังจากที่สั่ง Save ไฟล์เป็น “main.c” แล้วจะเห็นว่าลักษณะสีของตัวอักขระต่างๆ ในโปรแกรมจะเกิดการเปลี่ยนแปลงไปตามหน้าที่ เช่น Comment, ตัวแปร และ คำสั่ง เป็นต้น ซึ่งส่วนนี้เป็นข้อดีของ Keil uVision3 ซึ่งสามารถแยกและแสดงตัวอักขระได้อย่างเป็นหมวดหมู่ ทำให้ง่ายต่อการอ่านโปรแกรมด้วย ดังรูปที่ 3.30

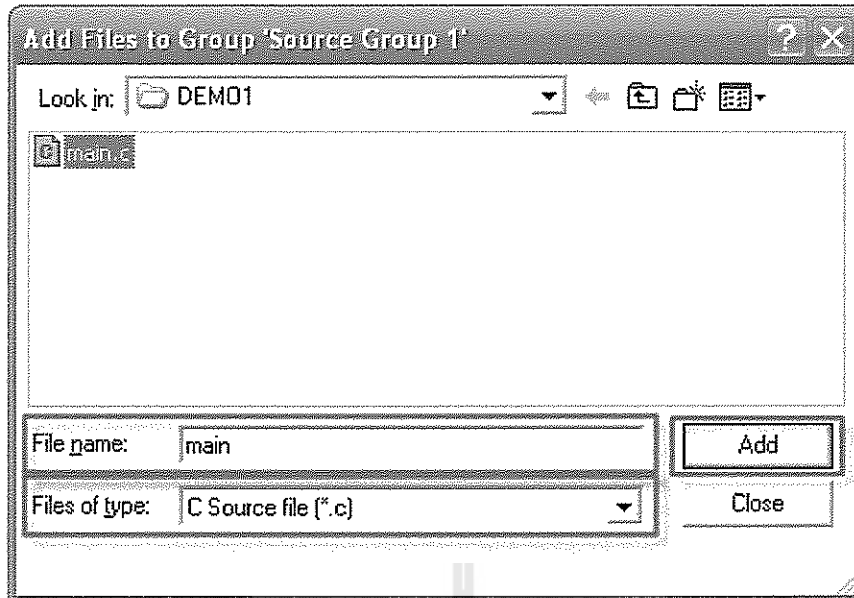


รูปที่ 3.30

7. ทำการสั่ง Add File ต่างๆเข้ากับ Project File โดยให้เลือกคลิกเมาส์ที่คำสั่ง Project —> Components, Environment, Books... จากนั้นให้เลือกที่ Tab Project Components แล้วเลือกที่ Add File ที่ต้องการจะเพิ่มเข้าไปใช้งานร่วมกับ Project File โดยในครั้งแรกให้เลือก Files of type เป็น “C Source files(*.c)” ซึ่งจะปรากฏชื่อไฟล์ต่างๆที่เป็น Source Code ภาษาซีให้เห็น โดยในที่นี้ให้เลือกคลิกเมาส์ที่ไอคอนของไฟล์ชื่อ “main.c” แล้วเลือก Add เพื่อสั่งเพิ่มไฟล์ชื่อ “Startup.s” เข้าไปรวมกับ Project Files ที่เราสร้างไว้ จากนั้นให้เลือกกำหนด File of type ใหม่เป็น “ASM Source files(*.s*;*.src;*.a*)” ซึ่งจะปรากฏชื่อของไฟล์ Startup.s ให้เห็นในช่องแสดงชื่อไฟล์ ให้ทำการคลิกเมาส์ที่ไอคอนของไฟล์ “Startup.s” แล้วเลือก Add เพื่อสั่งเพิ่มไฟล์ชื่อ “Startup.s” เข้าไปรวมกับ Project Files ที่เราสร้างไว้เมื่อทำการสั่ง Add ไฟล์ชื่อ “main.c” และ “Startup.s” ให้กับ Project File เป็นที่เรียบร้อยแล้วให้เลือกที่ Close เพื่อเป็นการสิ้นสุดการสั่ง Add File ซึ่งจะได้ดังรูปที่ 3.31 และ รูปที่ 3.32

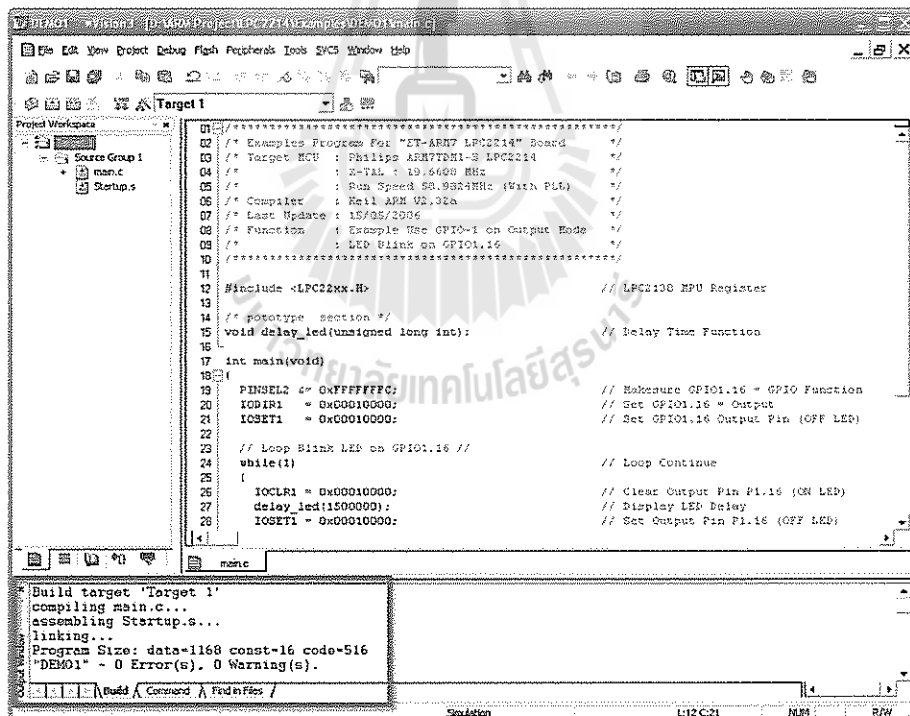


รูปที่ 3.31



รูปที่ 3.32

8. ให้ทำการสั่งแปล โปรแกรมที่เราเขียนขึ้นเรียบร้อยแล้ว โดยให้คลิกเมาส์ที่เมนูคำสั่ง Projects → Rebuild all target files ซึ่ง โปรแกรม Keil uVision3 จะทำการสั่งให้โปรแกรม Keil-CARM ทำการแปล คำสั่งให้ทันที่ดังรูปที่ 3.33



รูปที่ 3.33

ซึ่งหลังจากสั่งแปล โปรแกรมแล้ว ได้ผลถูกต้องและไม่เกิดข้อผิดพลาดใดๆขึ้น (0 Error และ 0 Warning) ก็ได้ Hex File ซึ่งมีชื่อเหมือนกันกับชื่อของ Project File ที่สร้างไว้ ซึ่งผู้ใช้สามารถนำ Hex File ดังกล่าวไปทำการ Download ให้กับ MCU ได้ทันที

บทที่ 4

ผลการทดสอบโปรแกรม

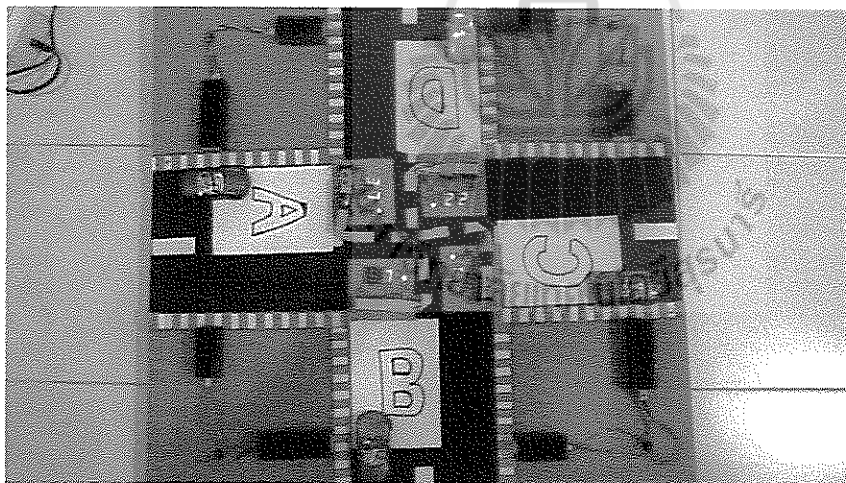
4.1 บทนำ

หลังจากที่ในบทที่แล้ว ได้มีการออกแบบโปรแกรมเพื่อให้สามารถใช้ควบคุมระบบสัญญาณไฟจราจร แบบอัตโนมัติ โดยมีตัว Sensor เป็นตัวแปรในการประมวลผลและอัปเดตลงบอร์ดเรียบร้อยแล้ว ในบทนี้จะได้กล่าวถึงการแสดงผลการออกแบบระบบสัญญาณไฟจราจรอัตโนมัติ

4.2 การทดสอบการใช้งานโปรแกรม“แสดงผลระบบสัญญาณไฟจราจรอัตโนมัติ”

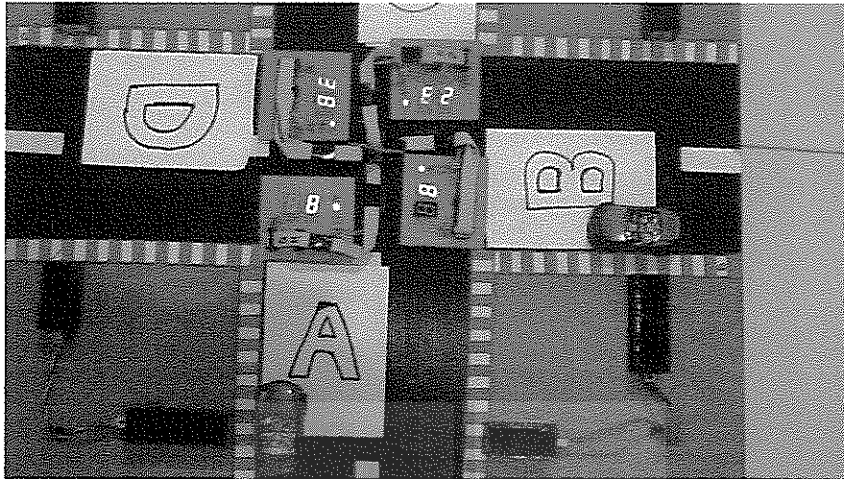
การทดสอบการใช้งานโปรแกรมแสดงผลระบบสัญญาณไฟจราจรอัตโนมัติ โดยเราจะอัปเดตโปรแกรมลงบนบอร์ดไมโครคอนโทรลเลอร์แล้วแสดงผลออกทาง 7-Segment และ LED เพื่อตรวจสอบว่าไมโครคอนโทรลเลอร์สามารถควบคุมสัญญาณไฟจราจรได้หรือไม่ โดยมีตัว Sensor เป็นตัวตรวจจับรถคันที่ผ่านไป

1. ในกรณีที่ไม่มีรถทุกเส้นทาง ปล่อยให้เส้นทาง A เป็นเวลา 15 วินาที แล้วปล่อยให้เส้นทาง B เป็นเวลา 15 วินาที แล้วปล่อยให้เส้นทาง C เป็นเวลา 15 วินาที แล้วปล่อยให้เส้นทาง D เป็นเวลา 15 วินาทีดังรูปที่ 4.1



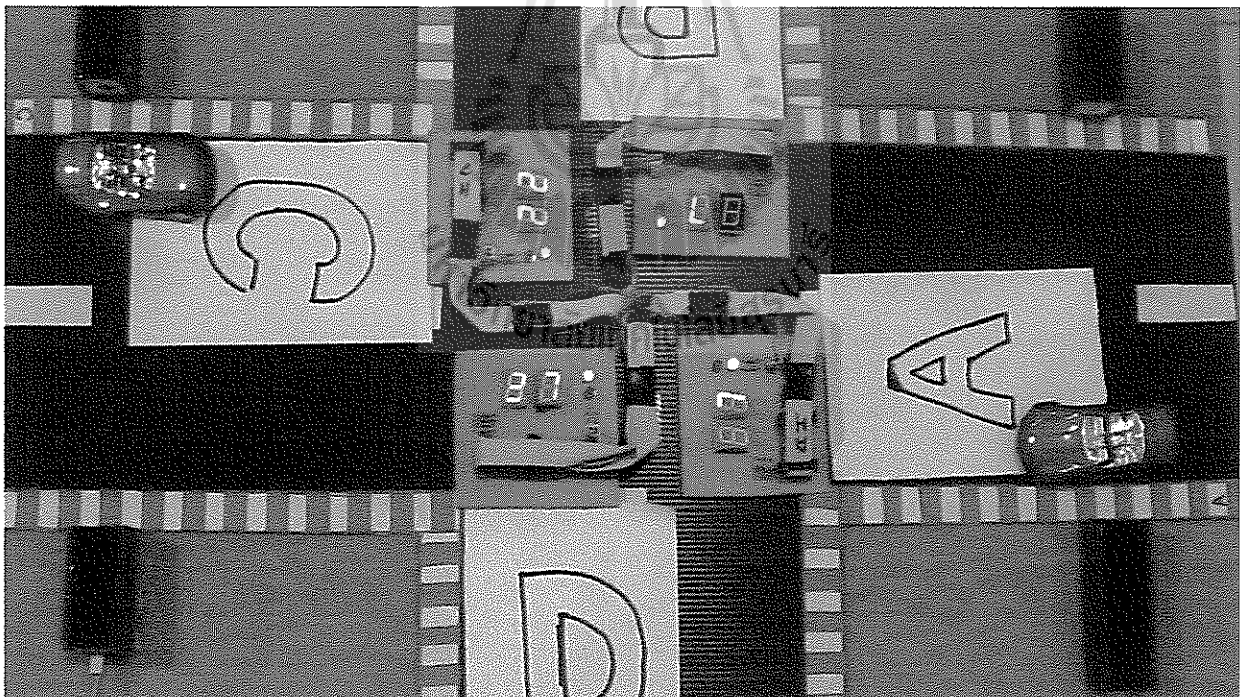
รูปที่ 4.1 แสดงขณะที่มีรถจอด A , B , C และ D

2. ในกรณีที่ A และ B มีรถ ปล่อยเส้นทาง A เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง B เป็นเวลา 15 วินาที
 ดังรูปที่ 4.2



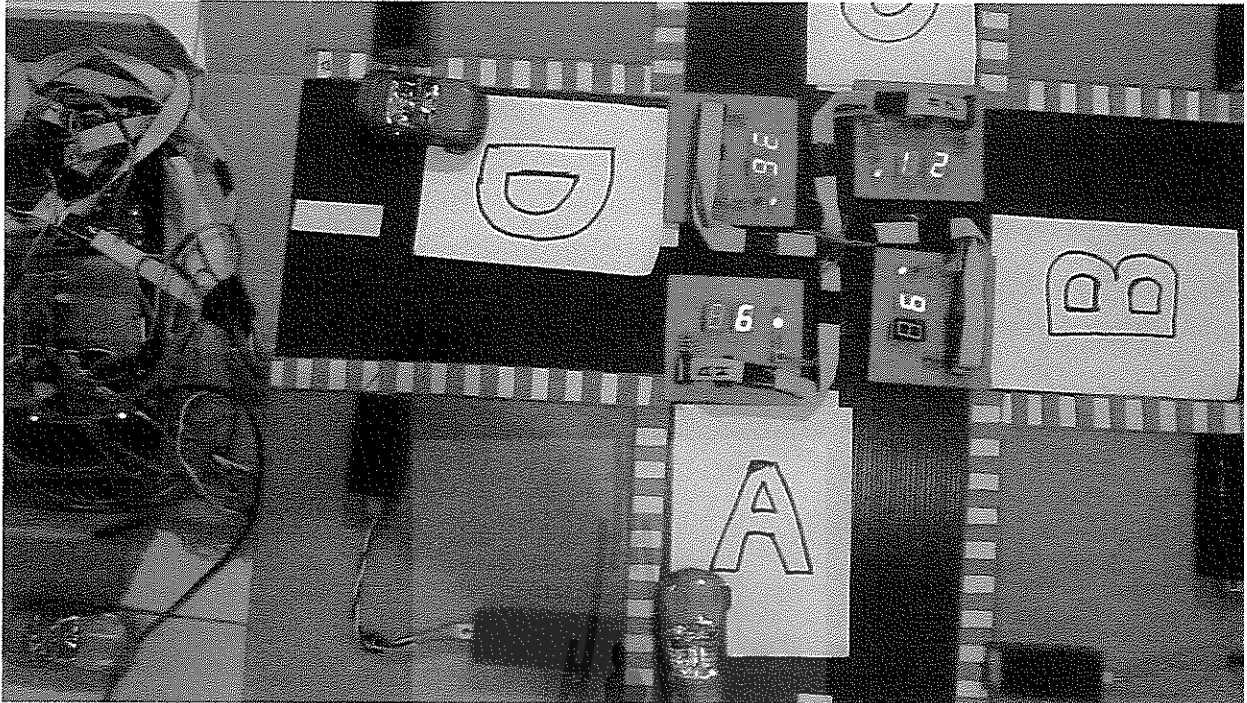
รูปที่ 4.2 แสดงขณะที่มีรถจอด A และ B

3. ในกรณีที่ A และ C มีรถ ปล่อยเส้นทาง A เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง C เป็นเวลา 15 วินาที
 ดังรูปที่ 4.3



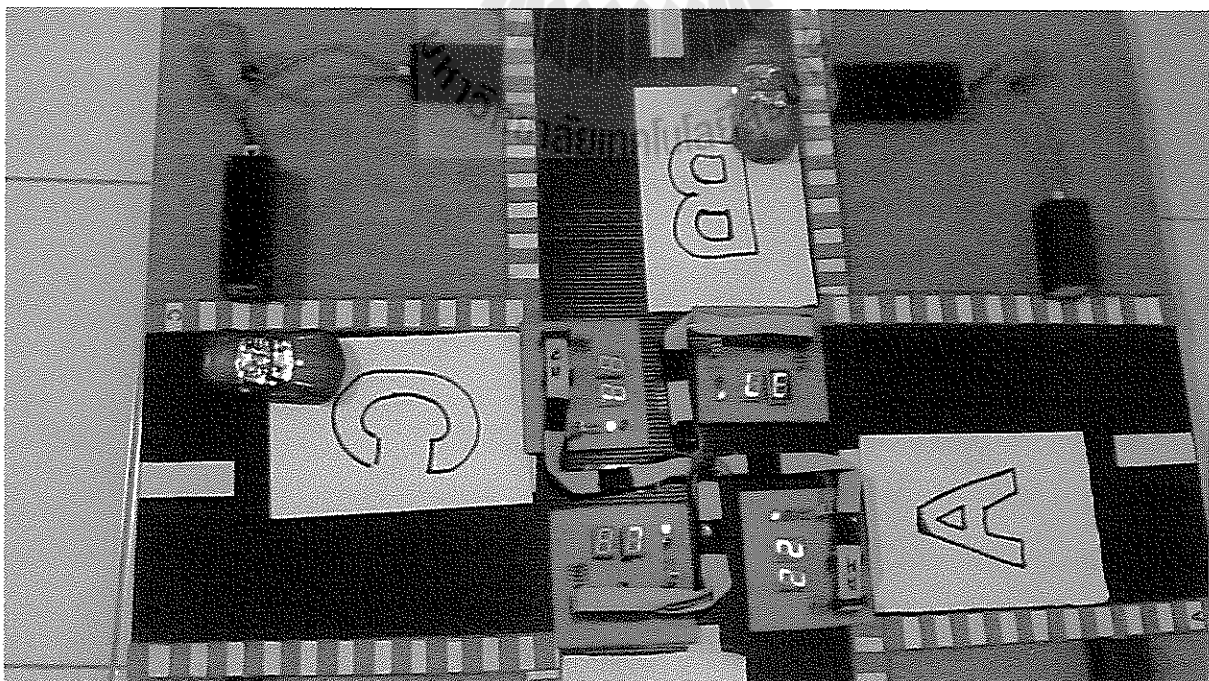
รูปที่ 4.3 แสดงขณะที่มีรถจอด A , C และ D

4. ในกรณีที่มีรถ A และ D มีรถ ปล่อยเส้นทาง A เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง D เป็นเวลา 15 วินาที
 ดังรูปที่ 4.4



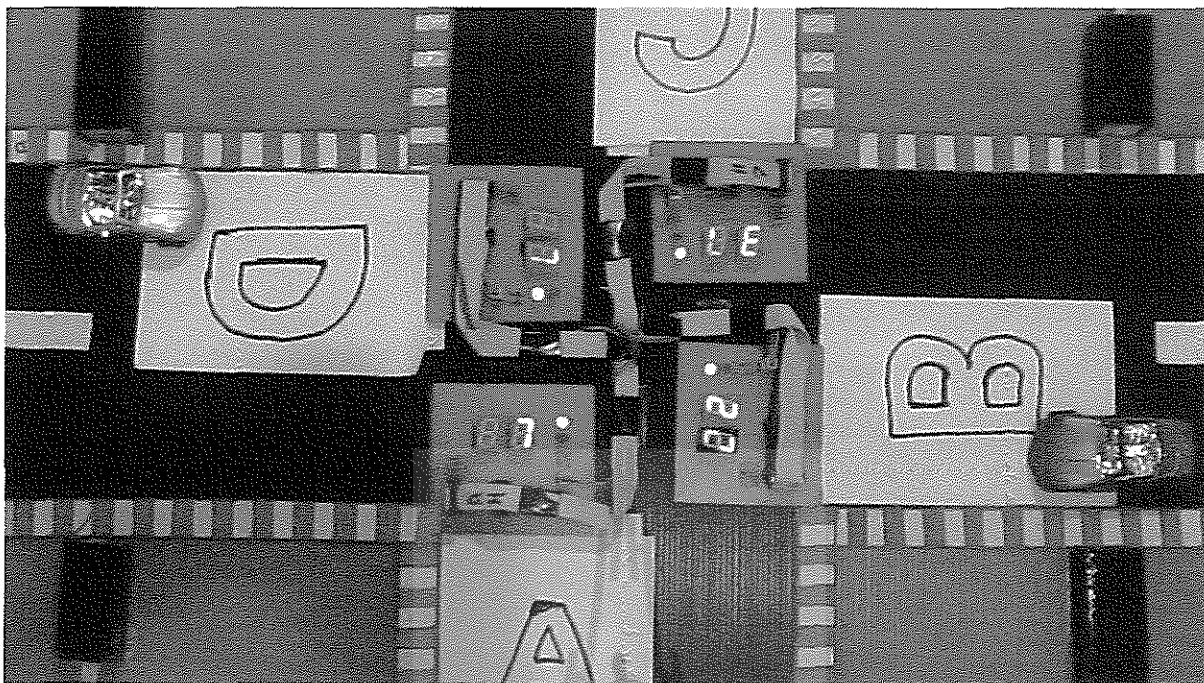
รูปที่ 4.4 แสดงขณะที่มีรถจอด A และ D

5. ในกรณีที่มีรถ B และ C มีรถ ปล่อยเส้นทาง B เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง C เป็นเวลา 15 วินาที
 ดังรูปที่ 4.5



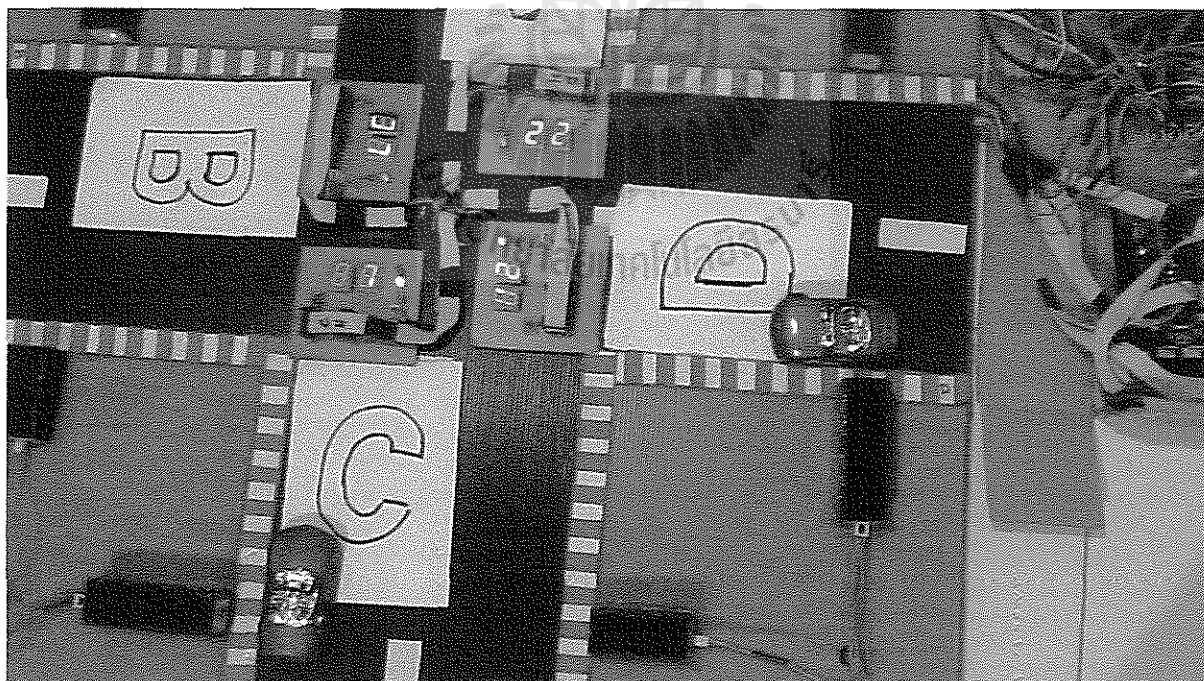
รูปที่ 4.5 แสดงขณะที่มีรถจอด B และ C

6. ในกรณีที่ B และ D มีรอด ปล่อยเส้นทาง B เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง D เป็นเวลา 15 วินาที ดังรูปที่ 4.6



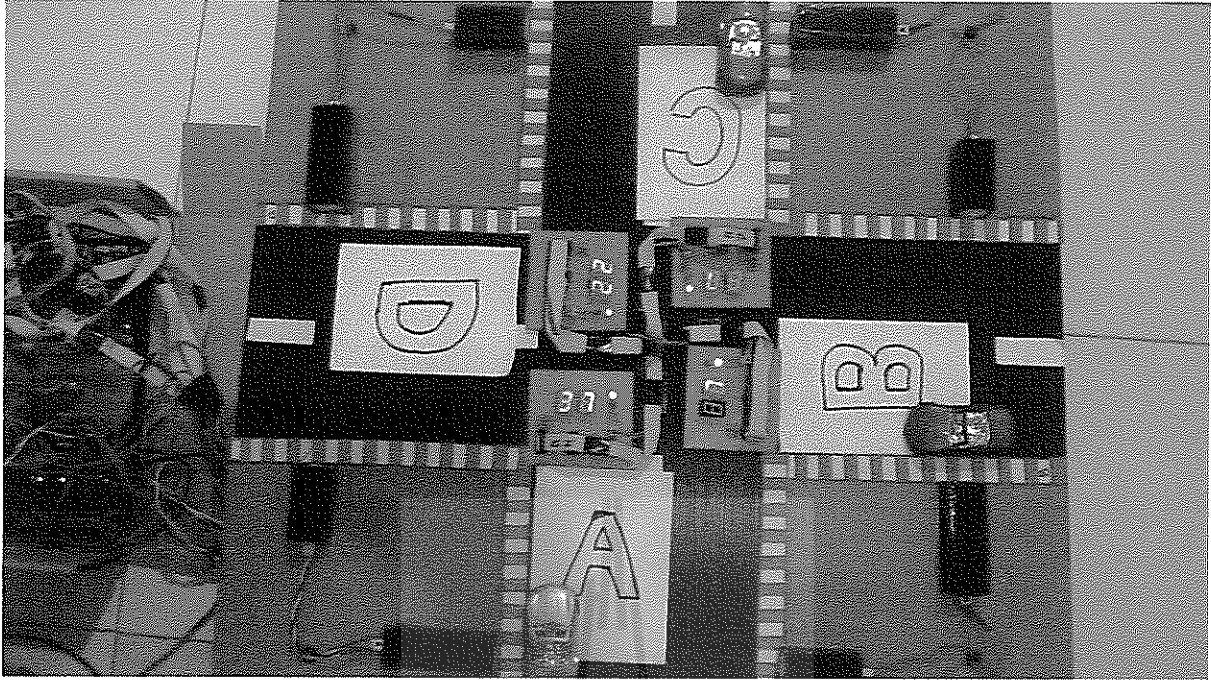
รูปที่ 4.6 แสดงขณะที่มีรอดจุด B และ D

7. ในกรณีที่ C และ D มีรอด ปล่อยเส้นทาง C เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง D เป็นเวลา 15 วินาที ดังรูปที่ 4.7



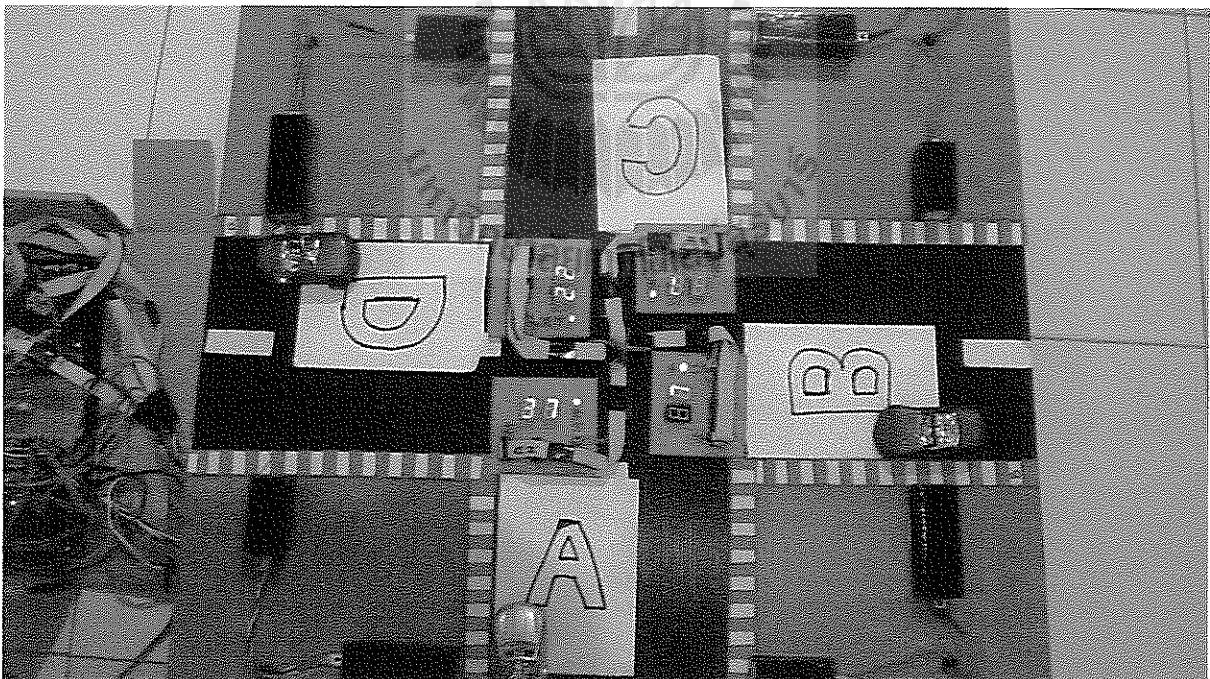
รูปที่ 4.7 แสดงขณะที่มีรอดจุด C และ D

8. ในกรณีที่ A , B และ C มีรอด ปล่อยเส้นทาง A เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง B เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง C เป็นเวลา 15 วินาทีดังรูปที่ 4.8



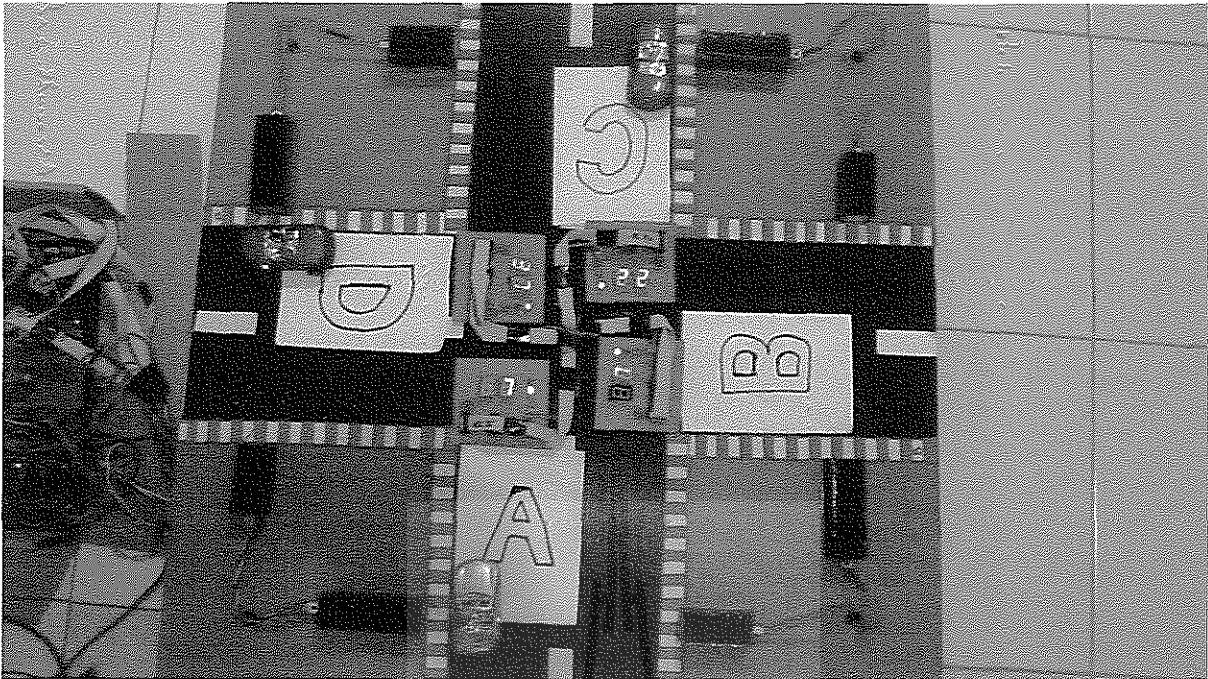
รูปที่ 4.8 แสดงขณะที่มีรถจอด A , B และ C

9. ในกรณีที่ A , B และ D มีรถ ปลดปล่อยเส้นทาง A เป็นเวลา 15 วินาที แล้วปลดปล่อยเส้นทาง B เป็นเวลา 15 วินาที แล้วปลดปล่อยเส้นทาง D เป็นเวลา 15 วินาทีดังรูปที่ 4.9



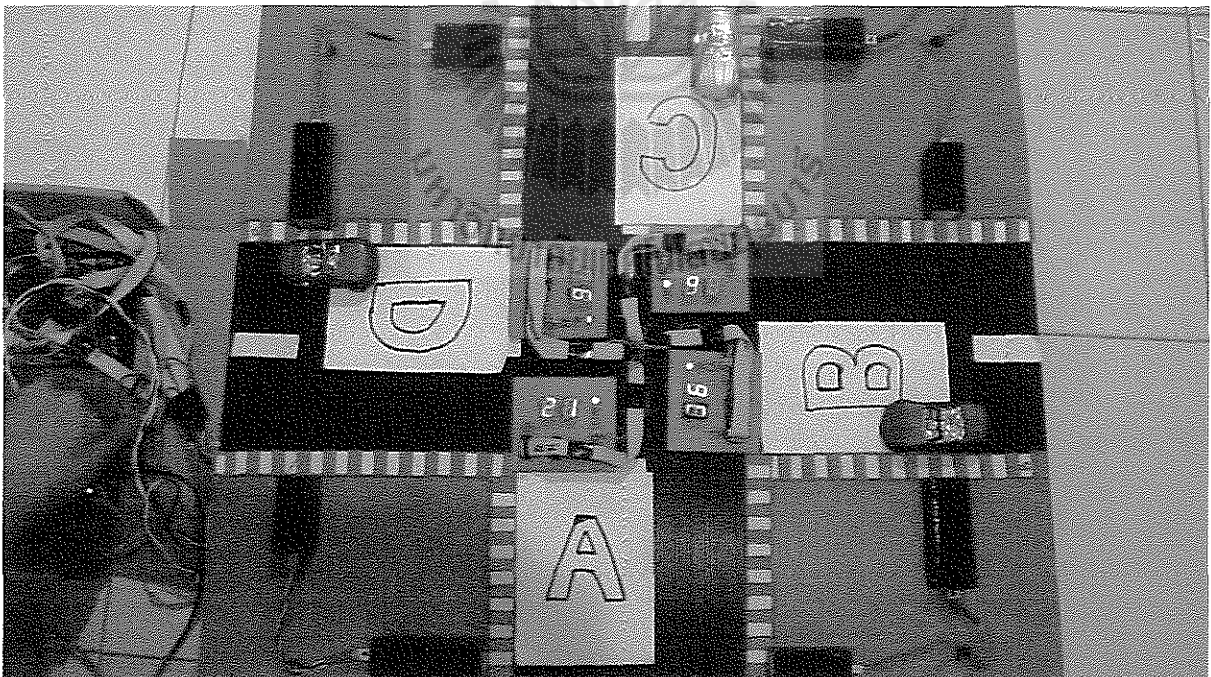
รูปที่ 4.9 แสดงขณะที่มีรถจอด B และ D

10. ในกรณีที่ A , C และ D มีรถ ปลดปล่อยเส้นทาง A เป็นเวลา 15 วินาที แล้วปลดปล่อยเส้นทาง C เป็นเวลา 15 วินาที แล้วปลดปล่อยเส้นทาง D เป็นเวลา 15 วินาทีดังรูปที่ 4.10



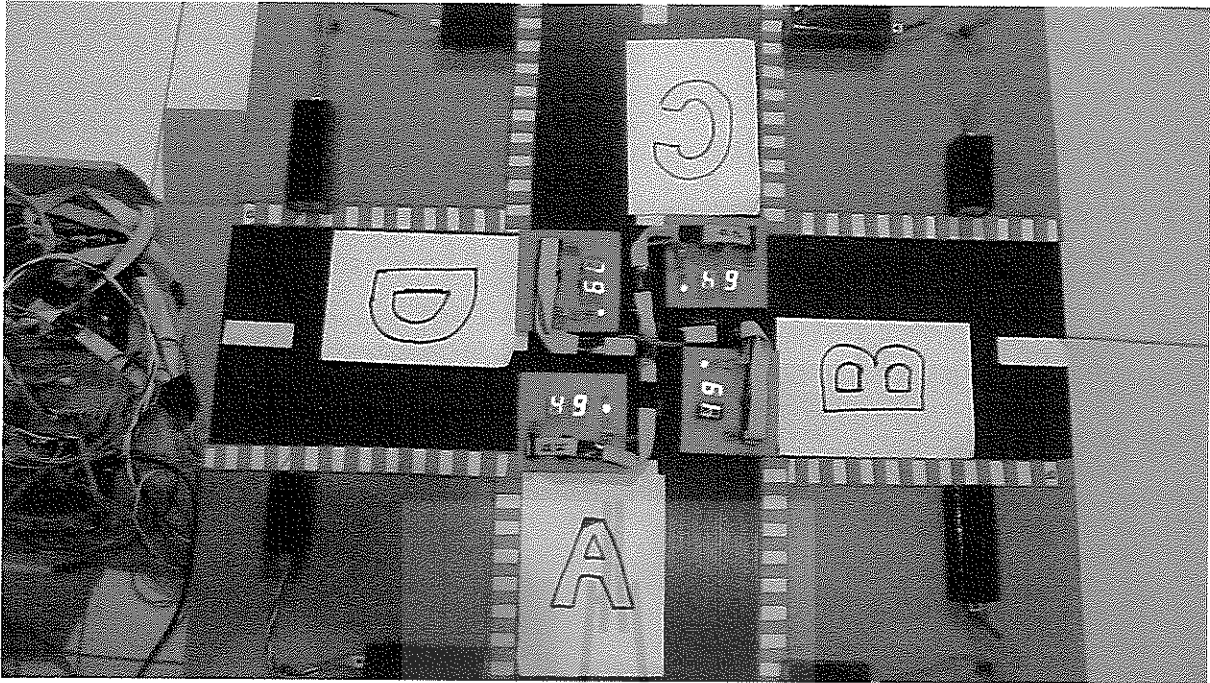
รูปที่ 4.10 แสดงขณะที่มีรถจอด A และ D

11. ในกรณีที่ B , C และ D มีรถ ปล่อยเส้นทาง B เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง C เป็นเวลา 15 วินาที แล้วปล่อยเส้นทาง D เป็นเวลา 15 วินาทีดังรูปที่ 4.11



รูปที่ 4.11 แสดงขณะที่มีรถจอด B , C และ D

12. ในกรณีที่ไม่มีรถผ่านเลย ปล่อยทีละเส้นทาง โดยเพิ่มเวลาสัญญาณไฟเขียวเป็นสามเท่าดังรูปที่ 4.12



รูปที่ 4.12 แสดงการปล่อยสัญญาณไฟเขียวเป็นสามเท่า

4.3 วิเคราะห์ผลการทดลอง

- การประมวลผลของไมโครคอนโทรเลอร์
 1. สามารถควบคุมด้วย switch ได้
 2. สามารถทำงานตามค่าที่ได้รับจาก Sensor ได้
 3. แสดงผลได้ตรงตาม ตารางการทำงานทุกประการ
- การทำงาน Sensor
 1. ไมโครคอนโทรเลอร์สามารถรับค่า Logic จาก Sensor ได้
 2. Sensor มีความไวต่อแสงสีแดงและใช้งานได้ดีในที่สว่างไม่มากนัก
 3. ระยะเวลาใช้งานของ Sensor ขึ้นอยู่กับความแรงของ แสง laser และแสงรบกวนจากสิ่งแวดล้อม

บทที่ 5

สรุปและข้อเสนอแนะ

5.1 บทนำ

โครงการนี้เกี่ยวกับการออกแบบ Sensor ตรวจจับรถ และพัฒนาโปรแกรมควบคุมไฟสัญญาณจราจร โดยมี Microcontroller ประมวลผลซึ่งในปัจจุบันการคมนาคมเป็นส่วนสำคัญต่อการพัฒนาเศรษฐกิจ ดังนั้นเราจึงนำเทคโนโลยีเข้ามาช่วยพัฒนาการคมนาคมให้มีประสิทธิภาพ เพื่อลดต้นทุนและเพิ่มความสะดวกและรวดเร็วต่อการคมนาคม

ปัจจุบันมียานพาหนะในการเดินทาง ไม่ว่าจะเป็นรถยนต์หรือแม้กระทั่งรถจักรยานยนต์ แต่มีเพียงบุคคลส่วนมากไม่นิยมใช้บริการรถประจำทางหรือระบบขนส่งมวลชน จึงทำให้เกิดปัญหาการจราจรขึ้น เช่น ปัญหาการจราจรที่พบได้ ตามเมืองใหญ่ และปัญหาการจราจรตามชนบท ผู้จัดทำโครงการจึงได้ออกแบบโปรแกรมและพัฒนา Sensor ตรวจจับรถ โดยมี Microcontroller ประมวลผล เพื่อเพิ่มประสิทธิภาพการจราจรบนท้องถนน

เนื้อหาในบทนี้จะเป็นการสรุปผลการทำงานของโปรแกรมและตัว Sensor และผลจากพัฒนา Sensor เพื่อตรวจจับการเคลื่อนที่ของรถ โดยใช้ Microcontroller ประมวลผลสามารถนำไปใช้ประโยชน์หรือนำไปทำการพัฒนาต่อไปได้อีก

5.2 สรุปผลการทดสอบ

จากการทดสอบการทำงานของ Sensor มีความสามารถตรวจจับระยะ 3 เมตร สามารถใช้การได้ดี และ Microcontroller สามารถรับค่าจาก Sensor ได้ดี แต่ในความเป็นจริงถนนมีความกว้างมากกว่า 5 เมตร ดังนั้นความสามารถของ Sensor ยังมีข้อจำกัดด้านระยะทางอยู่

ในส่วนของการทำงานของระบบสัญญาณไฟจราจรนี้ จะมีการทำงานตามตารางที่โปรแกรมไว้ โดยมีการใช้ Sensor ตรวจจับรถในเส้นทางที่จอดอยู่ (ติดไฟแดง) ถ้าเส้นทางที่ไม่มีรถเลยก็จะทำการข้ามรายการไป โดยกระทำที่เส้นทางถัดไปที่มีรถจอดรออยู่ ซึ่งในการทำงานในลักษณะนี้จะเป็นการลดระยะเวลาของสัญญาณไฟแดงลงทำให้เกิดความสะดวกและรวดเร็วต่อผู้ใช้ถนนในการเดินทาง

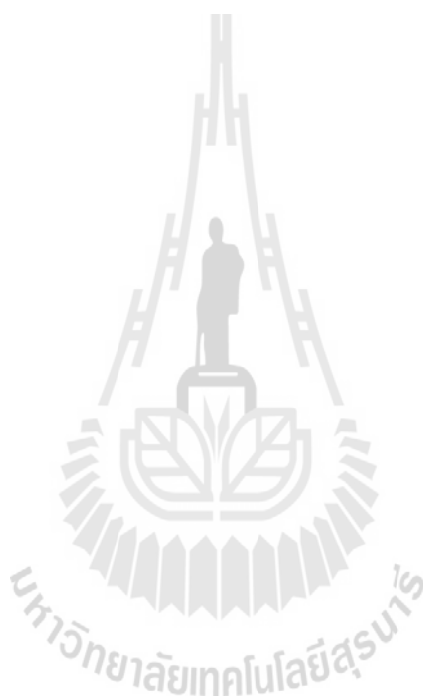
5.3 ปัญหาและอุปสรรค

1. ในการออกแบบ Sensor นั้นยังไม่ได้ คำนึงถึงแสงรบกวนจากภายนอกจึงทำให้เกิดปัญหาในเวลาใช้งานในที่แจ้ง
2. Sensor มีข้อจำกัดในเรื่องของระยะทาง
3. ระบบแสดงผลยังใช้ Port output มากและมีใช้สายไฟจำนวนมากจึงมีโอกาสดังกล่าวให้เกิดปัญหาได้ง่ายเวลาใช้งานจริง

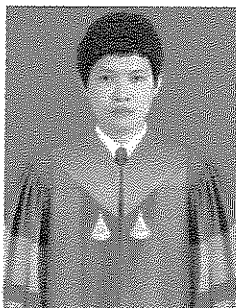
5.4 ข้อเสนอแนะ

1. ในการออกแบบ Sensor นั้นยังไม่ได้ คำนึงถึงแสงรบกวนจากภายนอกจึงทำให้เกิดปัญหาในเวลาใช้งานในที่แจ้ง จึงควรจะมีการศึกษาเรื่องของแสงเพิ่ม และออกแบบโครงสร้างภายนอกของ Sensor ให้มีการรบกวนจากแสงภายนอกให้น้อยที่สุด

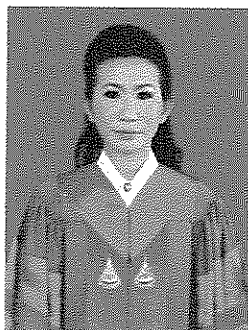
2. ระบบแสดงผลยังใช้ Port output มากและมีใช้สายไฟจำนวนมากจึงมีโอกาสให้เกิดปัญหาได้ง่ายเวลาใช้งานจริง



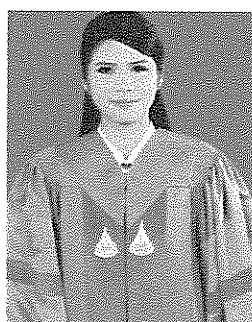
ประวัติผู้เขียน



นายอภิสิทธิ์ พุทธศรี เกิดเมื่อวันที่ 2 มิถุนายน พ.ศ. 2530
ภูมิลำเนาอยู่ที่ ตำบลในเมือง อำเภอเมือง จังหวัดขอนแก่น
สำเร็จการศึกษาระดับมัธยมศึกษาตอนปลาย
จากโรงเรียนขอนแก่นวิทยายน อำเภอเมือง จังหวัดขอนแก่น
เมื่อปี พ.ศ. 2548 ปัจจุบันเป็นนักศึกษาชั้นปีที่ 5
สาขาวิชาวิศวกรรมโทรคมนาคม สำนักวิชาวิศวกรรมศาสตร์
มหาวิทยาลัยเทคโนโลยีสุรนารี จังหวัดนครราชสีมา



นางสาวอัครา โสตก เกิดเมื่อวันที่ 7 เมษายน พ.ศ. 2531
ภูมิลำเนาอยู่ที่ ตำบลผาดอ อำเภอเมือง จังหวัดน่าน
สำเร็จการศึกษา ระดับชั้นมัธยมศึกษาตอนปลาย
จากโรงเรียนท่าวังผาพิทยาคม อำเภอท่าวังผา จังหวัดน่าน
เมื่อปี พ.ศ. 2548 ปัจจุบันเป็นนักศึกษาชั้นปีที่ 5
สาขาวิชาวิศวกรรมโทรคมนาคม สำนักวิชาวิศวกรรมศาสตร์
มหาวิทยาลัยเทคโนโลยีสุรนารี จังหวัดนครราชสีมา



นางสาววิสตา ส่งเสริม เกิดเมื่อวันที่ 3 มิถุนายน พ.ศ. 2530
ภูมิลำเนาอยู่ที่ ตำบลป่าตาล อำเภอเมือง จังหวัดลพบุรี
สำเร็จการศึกษา ระดับชั้นมัธยมศึกษาตอนปลาย
จากโรงเรียนวินิตศึกษา อำเภอเมือง จังหวัดลพบุรี
เมื่อปี พ.ศ. 2548 ปัจจุบันเป็นนักศึกษาชั้นปีที่ 4
สาขาวิชาวิศวกรรมโทรคมนาคม สำนักวิชาวิศวกรรมศาสตร์
มหาวิทยาลัยเทคโนโลยีสุรนารี จังหวัดนครราชสีมา

บรรณานุกรม

- [1] http://eletronics.se-ed.com/contents/140s135/140s135_p07.asp
- [2] http://elec.pnt.rm.ul.ac.th/attachments/015_el_lesson4.pdf
- [3] <http://osioptoelectronics.com/application-notes/AN-Photodiode-Parameters-Characteristics.pdf>
- [4] <http://nopparat.tve.ac.th/index.php?name=knowledge&file=readknowledge&id=22>
- [5] [http://www.phas.ubc.ca/beaudoin/PDS%20Papers/UDT_Phodiode Characteristics.pdf](http://www.phas.ubc.ca/beaudoin/PDS%20Papers/UDT_Phodiode%20Characteristics.pdf)
- [6] <http://www.etteam.com>

