

เครื่องวัดคุณสมบัติอุปกรณ์กึ่งตัวนำโดยใช้ต้นทุนต่ำ  
LOW – COST SEMICONDUCTOR DEVICES CURVE TRACER  
EQUIPMENT



โดย

นายภูริวัชร

พุ่มพวง

นายสารัตน์

พรวิเชียรวงศ์

นางสาวสุมินตรา

สงคะมัย

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2565

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องวัดคุณสมบัติอุปกรณ์กึ่งตัวนำโดยใช้ต้นทุนต่ำ  
LOW – COST SEMICONDUCTOR DEVICES CURVE TRACER EQUIPMENT

โดย

นายภูริวัชร พุ่มพวง	รหัส 62010730
นายสารัตน์ พรวิเชียรวงศ์	รหัส 62010934
นางสาวสุมินตรา สงคะมัย	รหัส 62010979

อาจารย์ที่ปรึกษา

รศ. ดร. พิพัฒน์ พรหมมี

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
ภาควิชาวิศวกรรมโทรคมนาคม  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2565

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2565

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เครื่องวัดคุณสมบัติอุปกรณ์กึ่งตัวนำโดยใช้ต้นทุนต่ำ

LOW – COST SEMICONDUCTOR DEVICES CURVE TRACER EQUIPMENT

ผู้จัดทำ

1. นายภูริวัชร พุ่มพวง 62010730
2. นายสารัตน์ พรวิเชียรวงศ์ 62010934
3. นางสาวสุมินตรา สงคะมัย 62010979



อาจารย์ที่ปรึกษา

(รศ. ดร. พิพัฒน์ พรหมมี)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

การดำเนินโครงการเรื่อง “เครื่องวัดคุณสมบัติอุปกรณ์กึ่งตัวนำโดยใช้ต้นทุนต่ำ” จะไม่ สามารถสำเร็จลุล่วงไปได้ด้วยดี หากไม่ได้รับความช่วยเหลือ และความอนุเคราะห์อย่างยิ่ง จาก อาจารย์ที่ปรึกษา คือ รศ.ดร. พิพัฒน์ พรหมมี ที่ได้คอยให้คำปรึกษา คำสั่งสอนและคำแนะนำ และยัง คอยตรวจสอบ แก้ไขข้อบกพร่องทุกขั้นตอนของการจัดทำโครงการขึ้นนี้ รวมถึงสนับสนุน สถานที่ เครื่องมือ และอุปกรณ์ต่างๆ ที่ใช้ระหว่างการจัดทำโครงการ รวมทั้งพี่นักศึกษาปริญญาโท และปริญญาเอก ภาควิชาวิศวกรรมโทรคมนาคมของห้อง T108-A ที่กรุณาให้คำแนะนำ คำปรึกษา และแนวทางการแก้ไขปัญหาที่เป็นประโยชน์ต่อการศึกษา คำนคว่าวิจัยให้โครงการนี้สำเร็จ สมบูรณ์ ยิ่งขึ้น

ขอกราบขอบพระคุณ บิดา มารดา และครอบครัว ที่ให้ความรัก ความห่วงใย และ เป็น กำลังใจที่สำคัญเสมอมาและที่สำคัญคือสนับสนุนให้โอกาสทางด้านการศึกษามีค่ายิ่งแก่ ผู้จัดทำ

นายภูริวัชร พุ่มพวง

นายสารัตน์ พรวิเชียรวงศ์

นางสาวสุมินตรา สงคะมัย

ผู้จัดทำ

**เครื่องวัดคุณสมบัติอุปกรณ์กึ่งตัวนำโดยใช้ต้นทุนต่ำ**  
**LOW – COST SEMICONDUCTOR DEVICES CURVE TRACER**  
**EQUIPMENT**

โดย	นายภูริวัชร	พุ่มพวง	62010730
	นายสารรัตน์	พรวิเชียรวงศ์	62010934
	นางสาวสุมินตรา	สงคัมย์	62010979

**อาจารย์ที่ปรึกษา** รศ.ดร.พิพัฒน์ พรหมมี

**บทคัดย่อ**

โครงการนี้นำเสนอการออกแบบและสร้างเครื่องวัดคุณสมบัติอุปกรณ์กึ่งตัวนำ เพื่อทำการสร้างอุปกรณ์เครื่องวัดคุณสมบัติอุปกรณ์กึ่งตัวนำแบบต้นทุนต่ำ โดยใช้ Microcontroller ร่วมกับวงจรอิเล็กทรอนิกส์โดยแสดงเป็นกราฟคุณสมบัติของกระแสและแรงดันตกคร่อมในอุปกรณ์อิเล็กทรอนิกส์จำพวกตัวต้านทาน โดยทำการวัดแรงดันตกคร่อมและประมวลผลด้วย Microcontroller โดยสร้างสัญญาณพัลส์จากอุปกรณ์ Arduino Mega2560 และทำการแปลงสัญญาณด้วยวงจร DAC โดยวิธีวัดอุปกรณ์ Semiconductor ต้องใส่อุปกรณ์ที่ต้องการวัดลงไปในตำแหน่งที่กำหนดภายในวงจร วงจรจะสร้างสัญญาณที่ใช้ในการวัดตัวอุปกรณ์จากนั้นดูผลการวัดกราฟคุณสมบัติบนหน้าจคอมพิวเตอร์

**ABSTRACT**

This project presents a circuit design to study the operation of a semiconductor property measurement circuit to create a low-cost curve tracer and apply a microcontroller with a curve tracer that can be used to measure the current and voltage across diodes, resistors, and 2 PIN semiconductor devices by generating a counter signal from the Arduino Mega2560, converting the signal with a DAC circuit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

and measuring the voltage across it. When you receive the device, check the circuit for ready-to-use 2 PIN semiconductors. that you want to measure into the specified position within the circuit and press the calibrate switch to have a generator signal counter used to measure the device



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

	หน้า
กิตติกรรมประกาศ	I
บทคัดย่อ	II
สารบัญ	III
สารบัญรูป	VI
สารบัญตาราง	X
<b>บทที่ 1</b>	
<b>บทนำ</b>	<b>1</b>
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตของโครงการปริญญานิพนธ์	1
<b>บทที่ 2</b>	
<b>ทฤษฎีและหลักการที่เกี่ยวข้อง</b>	<b>3</b>
2.1 ศึกษาหลักการและวิธีการทำงานของวงจร R2R Ladder	3
2.2 ศึกษาหลักการและวิธีการทำงานของ Arduino IDE	4
2.2.1 ส่วนของฮาร์ดแวร์	4
2.2.2 ส่วนของซอฟต์แวร์	4
2.2.3 ส่วนของฟังก์ชันการทำงาน	4
2.3 ศึกษาหลักการและวิธีการทำงานของบอร์ด Arduino	5
2.3.1 Arduino Mega 2560	5
2.4 ศึกษาการออกแบบวงจรในโปรแกรม Proteus	8
2.5 ศึกษาการทำงานของ IC4040	7
2.5.1 ลำดับเวลา (Timing Sequence)	10
2.6 ศึกษาการทำงานของอุปกรณ์ LF147/LF347	11
2.7 ศึกษาการทำงานของ Device Under Test 2 ขา (DUT)	12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
2.7.1 การทำงานของ Diode	12
2.7.2 ไดโอดในทางอุดมคติ (Ideal Diode)	13
2.7.3 ประเภทของไดโอด	14
2.8 ศึกษาการทำงานของทรานซิสเตอร์ (Transistor)	20
2.8.1 วงจรทรานซิสเตอร์	20
2.9 ศึกษาการทำงานของวงจรขยายในงานเครื่องมือวัด (Instrument Amplifier Circuit)	23
<b>บทที่ 3 การออกแบบและการจัดทำปฏิญญานิพนธ์</b>	<b>24</b>
3.1 การออกแบบโครงสร้างโดยรวม	24
3.1.1 ออกแบบภาพรวมเครื่องวัดคุณสมบัติอุปกรณ์กึ่งตัวนำโดยใช้ ต้นทุนต่ำ	25
3.2 การออกแบบและสร้างวงจร Curve Tracer	26
3.2.1 วงจร R2R Ladder 4 Bits และ 8 Bits	26
3.2.2 วงจร Instrument Amplifier	26
3.2.3 วงจรเปลี่ยนกราฟจาก Unipolar เป็น Bipolar	27
3.3 การออกแบบเครื่องวัดคุณสมบัติกึ่งตัวนำโดยใช้ต้นทุนต่ำ (ชิ้นงาน)	29
3.3.1 ออกแบบลายปริ้นวงจร PBC	29
3.3.2 ออกแบบหน้าต่างแสดงผล GUI (Graphic User Interface)	32
3.3.3 เขียนโปรแกรมควบคุมการเจนเนอร์เรทสัญญาณโดยใช้ Arduino IDE	33
3.4 เครื่องมือที่ใช้ในการทดลอง	33
3.5 การจัดเก็บผลทดลอง	34

## สารบัญ (ต่อ)

	หน้า
<b>บทที่ 4 ผลการทดลอง</b>	<b>35</b>
4.1 การทดสอบแรงดันจากวงจร R2R Ladder	35
4.2 ทดสอบวงจร Non - Inverting Amplifier (Unipolar to Bipolar)	36
4.3 ทดสอบวงจรในส่วนการ Reset วงจรด้วย Pull up switch	37
4.4 ทดสอบวัสดุอุปกรณ์ DUT (Semiconductor 2 ขา)	37
4.4.1 ทดสอบวัสดุอุปกรณ์ Resistor (ตัวต้านทาน)	37
4.4.2 ทดสอบวัสดุอุปกรณ์ Diode	38
4.4.3 ทดสอบวัสดุอุปกรณ์ Zener Diode	39
4.5 ทดสอบวัสดุอุปกรณ์ (DUT) Semiconductor 3 ขา	39
4.5.1 ทดสอบวัสดุอุปกรณ์ Transistor แบบ NPN	40
4.5.2 ทดสอบวัสดุอุปกรณ์ Transistor แบบ PNP	40
4.6 ทดสอบวัสดุอุปกรณ์ (DUT) แสดงผลบนหน้าจอ UI	41
<b>บทที่ 5 สรุปผลและข้อเสนอแนะ</b>	<b>47</b>
5.1 สรุปผล	47
5.2 ข้อเสนอแนะ	47
<b>บรรณานุกรม</b>	<b>48</b>
<b>ภาคผนวก</b>	<b>51</b>

## สารบัญรูป

รูปที่	หน้า
2.1	3
2.2	5
2.3	7
2.4	8
2.5	9
2.6	9
2.7	10
2.8	11
2.9	12
2.10	13
2.11	14
2.12	15
2.13	15
2.14	16
2.15	17
2.16	17
2.17	19
2.18	19
2.19	21

## สารบัญรูป (ต่อ)

	หน้า
รูปที่	
2.20 การทำงานของทราสซิสเตอร์	22
2.21 วงจรขยายในงานเครื่องมือวัด (Instrument Amplifier Circuit)	23
3.1 บล็อกไดอะแกรมหลักการทำงานของเครื่องวัดคุณสมบัติอุปกรณ์กึ่งตัวนำโดยใช้ต้นทุนต่ำ	24
3.2 การทำงานเครื่องวัดคุณสมบัติอุปกรณ์กึ่งตัวนำโดยใช้ต้นทุนต่ำ	25
3.3 วงจร R2R Ladder 4 Bits	26
3.4 วงจร R2R Ladder 8 bits	26
3.5 วงจร Instrument Amplifier	27
3.6 วงจรขยายสัญญาณ Unipolar เป็น Bipolar	27
3.7 วงจร Curve Tracer ที่ได้ทำการต่อจริง	28
3.8 Mockup 3D ของเครื่องวัดคุณสมบัติกึ่งตัวนำโดยใช้ต้นทุนต่ำ	29
3.9 ลายปริ้นวงจร Curve tracer ในส่วนอุปกรณ์ DUT 3 ขา และวงจรเจเนอเรเตอร์ $V_C$ และ $I_B$	30
3.10 ลายปริ้นวงจร Curve tracer ในส่วนอุปกรณ์ DUT 2 ขา และสวิตช์โหมด	30
3.11 วงจร Curve tracer	31
3.12 ชิ้นงานเครื่องวัดคุณสมบัติกึ่งตัวนำต้นทุนต่ำ	31
3.13 ภาพรวมหน้าต่าง GUI	32
3.14 โค้ดในส่วนที่ใช้ในการเจเนอเรตสัญญาณ 8 bits	33
4.1 แสดงผลของสัญญาณ Output	35
4.2 กราฟสัญญาณ Ramp 256 step (Bipolar)	36
4.3 ผลทดสอบการ Reset วงจรด้วย Pull up switch	37
4.4 ผลทดสอบวัดอุปกรณ์ Resistor $1k\Omega$	38

## สารบัญรูป (ต่อ)

	หน้า
(ก) สัญญาณแรงดัน และ กระแสที่วัดได้จากวงจรถ่าย Curve Tracer	
(ข) กราฟใน Mode XY	
4.5 ผลทดสอบวัสดุอุปกรณ์ Diode 1N4148	38
(ก) สัญญาณแรงดันและกระแสที่วัดได้จากวงจรถ่าย Curve Tracer	
(ข) กราฟใน Mode XY	
4.6 ผลทดสอบวัสดุอุปกรณ์ Zener Diode 1N4733 (5.1V)	39
(ก) สัญญาณแรงดันและกระแสที่วัดได้จากวงจรถ่าย Curve Tracer	
(ข) กราฟใน Mode XY	
4.7 ผลทดสอบวัสดุอุปกรณ์ Transistor NPN 2N3904	40
(ก) สัญญาณแรงดัน $V_{CE}$ และ กระแส $I_C$ จากวงจรถ่าย Curve Tracer	
(ข) กราฟใน Mode XY	
4.8 ผลทดสอบวัสดุอุปกรณ์ Transistor NPN 2N3904	41
(ก) สัญญาณแรงดัน $V_{CE}$ และ กระแส $I_C$ จากวงจรถ่าย Curve Tracer	
(ข) กราฟใน Mode XY	
4.9 กราฟแสดงผลผ่านหน้าจอบริการ UI ของตัวต้านทาน	42
4.10 กราฟแสดงผลผ่านหน้าจอบริการ UI ของ Diode 1N4148	42
4.11 กราฟแสดงผลผ่านหน้าจอบริการ UI ของ Diode 1N4001	43
4.12 กราฟแสดงผลผ่านหน้าจอบริการ UI ของ Zener Diode 1N4733 (5.1V)	43
4.13 กราฟแสดงผลผ่านหน้าจอบริการ UI ของ Zener Diode 1N4737 (7.5V)	44
4.14 กราฟแสดงผลผ่านหน้าจอบริการ UI ของ Transistor PNP 2N3906	44
4.15 กราฟแสดงผลผ่านหน้าจอบริการ UI ของ Transistor PNP BC557	45
4.16 กราฟแสดงผลผ่านหน้าจอบริการ UI ของ Transistor NPN BC5478	46
4.17 กราฟแสดงผลผ่านหน้าจอบริการ UI ของ Transistor NPN BC - 337	46

## สารบัญตาราง

ตารางที่	หน้า
1.1	คุณสมบัติของซีเนอร์ไดโอด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 1

### บทนำ

#### 1.1 ความเป็นมาและความสำคัญของปัญหา

เนื่องจากเครื่องวัดคุณสมบัติอุปกรณ์กึ่งตัวนำ (Curve Tracer) ตามท้องตลาดในปัจจุบัน มีราคาที่สูงเข้าถึงได้ยาก ซึ่งประสิทธิภาพในการทำงานนั้นเป็นไปตามราคา ในส่วนเครื่องวัดอุปกรณ์กึ่งตัวนำในช่วงที่มีราคาใกล้เคียงกับผู้จัดทำที่ได้ทำการสร้างขึ้น จะมีฟังก์ชันในการทำงานน้อย เช่น สามารถวัดอุปกรณ์ semiconductor ได้เพียง 2 ขาและต้องต่อพอร์ตเข้ากับ Oscilloscope เพื่อดูการแสดงผลและ Oscilloscope มีน้ำหนักที่เยอะจึงไม่สะดวกในการใช้งานสำหรับการพกพาทางผู้จัดทำได้เล็งเห็นถึงปัญหาเหล่านี้ จึงได้ศึกษาค้นคว้าหลักการเกี่ยวกับวงจรที่เกี่ยวข้องกับการทำงานของเครื่องวัดคุณสมบัติอุปกรณ์กึ่งตัวนำ และทำการศึกษาในเชิงทฤษฎีของอุปกรณ์ Semiconductor เพื่อดูคุณสมบัติการทำงานตามลักษณะเฉพาะของอุปกรณ์ จากนั้นทางผู้จัดทำได้นำอุปกรณ์ Microcontroller มาประยุกต์ใช้งาน ร่วมกับอุปกรณ์อิเล็กทรอนิกส์ที่หาได้ในท้องตลาด และมีราคาที่ถูกลง ซึ่งมีกระบวนการทำงานโดยการกำเนิดสัญญาณพัลส์จากอุปกรณ์ Arduino Mega2560 และทำการแปลงสัญญาณโดย วงจร R2R Ladder (DAC) จากนั้นนำวงจร Instrument Amplifier มาประยุกต์ใช้ในการวัดกระแส และ แรงดันตกคร่อมในอุปกรณ์ Semiconductor และแสดงผลผ่านหน้าต่าง UI ในโปรแกรม Python เพื่อให้ไม่กระทบต่อวงจรหรือมีผลกระทบน้อยที่สุด

#### 1.2 วัตถุประสงค์

- 1) เพื่อศึกษาการทำงานของวงจร Curve Tracer
- 2) เพื่อทำการสร้างอุปกรณ์ใช้งาน Curve Tracer แบบต้นทุนต่ำ
- 3) เพื่อประยุกต์การใช้งานอุปกรณ์ Microcontroller กับ Curve Tracer

#### 1.3 ขอบเขตของปริญญานิพนธ์

- 1) ประยุกต์ใช้งานอุปกรณ์ Microcontroller เพื่อทำการวัดแรงดันตกคร่อม โดยสร้างสัญญาณพัลส์จากอุปกรณ์ Arduino Mega2560 และทำการแปลงสัญญาณโดยวงจร DAC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 2) ใช้วัดกระแสและแรงดันตกคร่อมในอุปกรณ์อิเล็กทรอนิกส์จำพวก ทราานซิสเตอร์ ไดโอด และตัวต้านทาน
- 3) ต่อยอดการสร้างอุปกรณ์ Curve Tracer เพื่อให้ใช้งานใน Lab ปฏิบัติการได้จริง



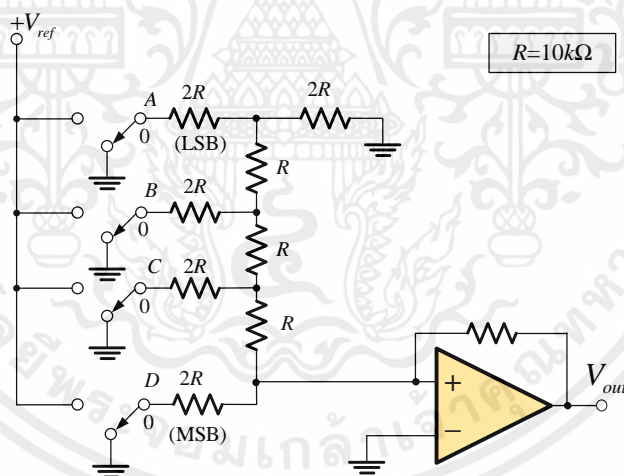
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

### ทฤษฎีและหลักการที่เกี่ยวข้อง

#### 2.1 ศึกษาหลักการและวิธีการทำงานของวงจร R2R Ladder

การที่วงจรแปลงสัญญาณดิจิทัลเป็นแอนะล็อก เมื่อมีอินพุตที่เป็นจำนวนบิต ( $n$  bit) มาก ทำให้จึงต้องเพิ่มตัวต้านทานมากขึ้น จึงมีการออกแบบวงจรที่ใช้ค่าความต้านทานน้อยๆ ซึ่งวงจร R2R Ladder DAC เป็นวงจร DAC ที่ใช้อุปกรณ์และตัวต้านทานที่ต่อเป็นโครงข่ายซึ่งเรียกว่าโครงข่าย Ladder ที่เชื่อมต่อแบบขนานและแบบอนุกรมซึ่งทำหน้าที่เป็นตัวแบ่งแรงดันที่เชื่อมต่อระหว่างกัน โดยตัวต้านทานที่ใช้มีเพียง 2 ค่า คือ  $R$  กับ  $2R$  เพื่อสะดวกในการออกแบบ ซึ่งจะมีแรงดันที่เอาต์พุตจะขึ้นอยู่กับค่าดิจิทัลเอาต์พุตของ Microcontroller [1] ดังแสดงในรูปที่ 2.1 แตกต่างจากตัวแปลงแอนะล็อกเป็นดิจิทัลจะรวมอยู่ในชิปเช่น Arduino หรือ Raspberry Pi เพื่อความสะดวกในการเขียนโปรแกรมและการใช้งาน [2, 3]



รูปที่ 2.1 วงจรแสดง DAC 4 บิต แบบ R2R Ladder

สำหรับสมการคำนวณค่าแรงดันแอนะล็อกเอาต์พุตคือสมการ

$$V_{out} = -\frac{V_{ref}}{8} \times B$$

เมื่อ  $B$  = ค่าเลขฐานสิบของอินพุตดิจิทัล 4 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.2 ศึกษาหลักการและวิธีการทำงานของ Arduino IDE

Arduino เป็นแพลตฟอร์มอิเล็กทรอนิกส์แบบโอเพนซอร์สบนพื้นฐานของฮาร์ดแวร์และซอฟต์แวร์ที่ง่ายต่อการใช้งาน สามารถนำไปประยุกต์ใช้งานได้หลายอย่าง เช่น ควบคุมอุปกรณ์อิเล็กทรอนิกส์ขนาดเล็ก อ่านค่าเซ็นเซอร์วัดสภาพแวดล้อมต่างๆแล้วแสดงค่าที่เซ็นเซอร์สามารถอ่านได้ออกมาทางจอแสดงผล นำไปประยุกต์เข้าเป็นชิ้นงานทางอิเล็กทรอนิกส์เพื่ออำนวยความสะดวกในการใช้ชีวิตประจำวัน เป็นต้น

ปัจจุบัน Arduino ถือได้ว่าเป็นแพลตฟอร์มที่ได้รับความนิยมสูงจากทั่วโลก เนื่องจากราคาของตัวบอร์ด Arduino ไม่ค่อยสูงมาก เป็นโอเพนซอร์สทั้งทางด้านฮาร์ดแวร์และซอฟต์แวร์ คอมมูนิตีและฟอรัมในการถามตอบเรื่องเกี่ยวกับการใช้งานร่วมกับอุปกรณ์อิเล็กทรอนิกส์ต่างๆ ตัวอย่างโปรแกรมเบื้องต้นและไลบรารีสำหรับใช้งาน และอื่นๆ ในแพลตฟอร์มของ Arduino ประกอบไปด้วยส่วนหลักๆ 2 ส่วน

### 2.2.1 ส่วนของฮาร์ดแวร์

บอร์ด Arduino เป็นวงจรอิเล็กทรอนิกส์ขนาดเล็กที่มีไอซีหรือชิปประมวลผลของไมโครคอนโทรลเลอร์ (Micro Controller Unit : MCU) ประกอบอยู่กับวงจรอิเล็กทรอนิกส์อื่นๆ ตัวบอร์ดของ Arduino จะมีอยู่หลายโมเดลให้เลือกใช้ โดยบอร์ดแต่ละโมเดลจะมีความแตกต่างกันในเรื่องของสเปก จำนวนพอร์ตต่างๆสำหรับใช้งาน พื้นที่ในการเขียนโปรแกรม ขนาดของบอร์ด เป็นต้น

### 2.2.2 ส่วนของซอฟต์แวร์

ซอฟต์แวร์ Arduino หรือที่เรียกว่า Arduino IDE (Integrated Development Environment) เป็นเครื่องมือที่ใช้ในการพัฒนาโปรแกรมบนแพลตฟอร์ม Arduino และอัปโหลดโปรแกรมที่พัฒนาสำเร็จแล้ว ลงบนตัวบอร์ด

### 2.2.3 ส่วนของฟังก์ชันการทำงาน

2.2.3.1 ฟังก์ชัน Setup จะเริ่มทำงานเป็นอันดับแรกเมื่อ Arduino เริ่มทำงาน และคำสั่งที่ถูกเขียนลงไปนั้น จะทำงานเพียงครั้งเดียวเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3.2 ฟังก์ชัน Loop จะเริ่มทำงานทันทีเมื่อฟังก์ชัน Setup ทำงานเสร็จ และคำสั่งที่ถูกเขียนลงไปนั้น จะทำงานตั้งแต่คำสั่งแรก ไล่ไปจนถึงคำสั่งสุดท้าย แล้วกลับมาที่คำสั่งแรก ไล่ไปจนถึงคำสั่งสุดท้าย วนซ้ำแบบนี้ไปเรื่อย ๆ ตลอดการทำงานของ Arduino [4, 5]

## 2.3 ศึกษาหลักการและวิธีการทำงานของบอร์ด Arduino

### 2.3.1 Arduino Mega 2560

Arduino Mega 2560 เป็นบอร์ดที่ใช้ชิป ATmega2560 เป็นไมโครคอนโทรลเลอร์หลัก เป็นบอร์ดที่ต่อยอดมาจาก Arduino Uno R3 มี Digital Input/Output 54 ขา สามารถใช้เป็น Output แบบ PWM ได้ 15 ขา มี Analog Inputs 16 ขา มี UARTs (Hardware Serial Ports) 4 ขา ทำงานที่ความถี่ 16 MHz สามารถเชื่อมต่อกับคอมพิวเตอร์ด้วยสายเคเบิล USB หรือใช้ Adaptor AC – to – DC เพื่อเริ่มต้นใช้งานและมีปุ่ม Reset สามารถต่อเข้ากับ Shields ที่ออกแบบเพื่อใช้งานกับ Arduino Duemilanove หรือ Diecimila

Arduino Mega 2560 เป็นบอร์ดที่มีขาสำหรับต่อใช้งานเยอะ จึงเป็นบอร์ดที่นิยมใช้กับโปรเจกต์ที่ต้องการใช้งาน Sensor จำนวนมาก และเนื่องจากบอร์ดมีหน่วยความจำค่อนข้างสูงจึงเหมาะกับงานที่ต้องการเก็บข้อมูลมาก ๆ [5]

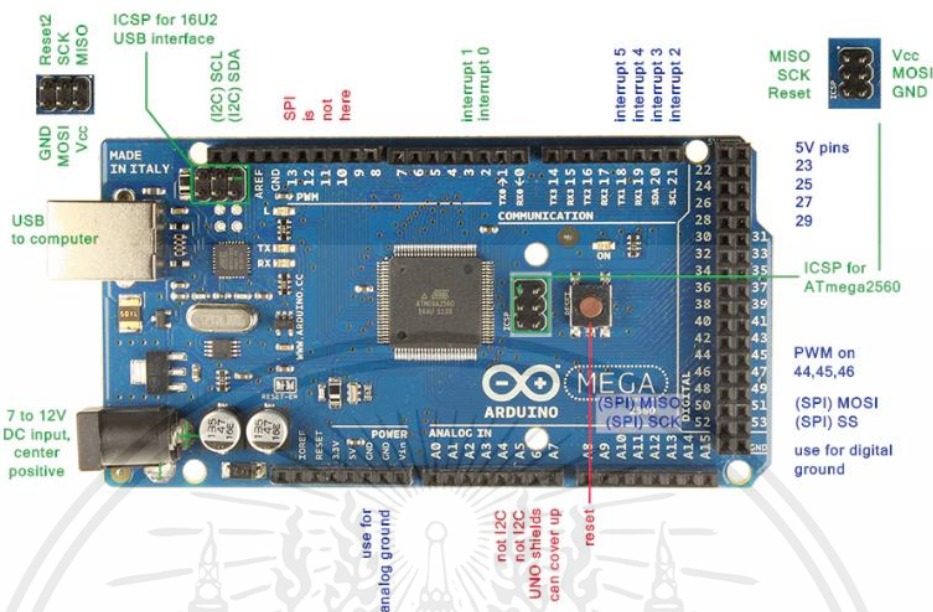


รูปที่ 2.2 บอร์ด Arduino Mega 2560

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยมีคุณสมบัติของ Arduino Mega 2560 ดังนี้

- 1) External Interrupts จะมี PIN 2 (Interrupt 0) , 3 (Interrupt 1) , 18 (Interrupt 5) , 19 (Interrupt 4) , 20 (Interrupt 3) , 21 (Interrupt 2) PINs เหล่านี้สามารถที่จะกำหนดค่าที่เรียก Interrupt ในค่าต่างๆ ขอบขาขึ้นและลงหรือเปลี่ยนแปลงค่า
- 2) PWM 2 ถึง 13 และ 44 ถึง 46 ให้ Output PWM Output 8-bits
- 3) SPI 50 (MISO) , 51 (MOSI) , 52 (SCK) , 53 (SS) ใช้สำหรับรองรับการสื่อสารแบบ SPI โดยที่ไม่เกี่ยวข้องกับ ICSP header ซึ่งจะมีลักษณะคล้ายกับ Uno, Duemilanove และ Diecimila
- 4) LED 13 เป็น Build-in LED ที่เชื่อมต่อกับ Digital PIN 13 เมื่อ PIN มีค่าเป็น HIGH LED จะติด แต่เมื่อ PIN เป็น LOW LED จะดับ
- 5) TWI 20 (SDA) และ 21 (SCL) รองรับการเชื่อมต่อแบบ TWI (I2C)
- 6) บอร์ด Mega2560 มี 16 Analog Inputs แต่ละ PINs ให้ความละเอียด 10 bits
- 7) AREF แรงดันอ้างอิง สำหรับ Analog Input
- 8) Reset ใช้ในการ Reset ไมโครคอนโทรลเลอร์ โดยทั่วไปจะใช้โดยการเพิ่มปุ่ม Reset ใว้บน Shield เพื่อป้องกันปุ่มที่อยู่บนบอร์ด [6]



รูปที่ 2.3 ส่วนประกอบของ PIN Out บอร์ด Arduino Mega 2560

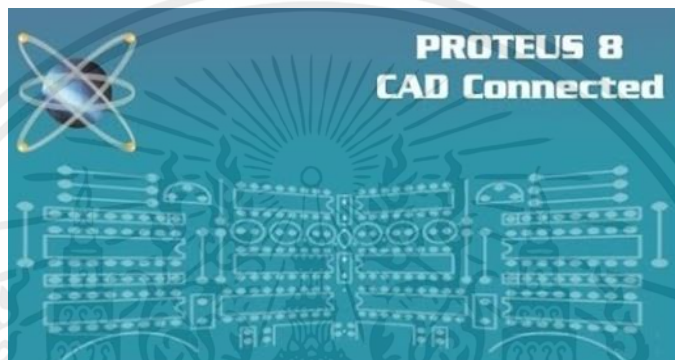
## 2.4 ศึกษาการออกแบบวงจรในโปรแกรม Proteus

โปรแกรม Proteus หรือ Proteus VSM (Virtual System Modeling) เป็นโปรแกรมที่พัฒนาขึ้นโดยบริษัท แล็บเซ็นเตอร์อิเล็กทรอนิกส์ จำกัด (Lab center Electronics Ltd.) ที่ประเทศอังกฤษ โปรแกรม Proteus มีชื่อเต็มว่า Lab center Electronics Proteus ซึ่งภายในโปรแกรมจะประกอบด้วยส่วนประกอบหลัก 2 ส่วน คือ ISIS และ ARES โปรแกรม Proteus จะมีอยู่หลายเวอร์ชันให้เลือกใช้งาน ซึ่งเวอร์ชันในปัจจุบัน คือ เวอร์ชัน 8

โปรแกรม Proteus เป็นโปรแกรมที่มีความสามารถในการจำลองการทำงานของวงจรอิเล็กทรอนิกส์ได้หลากหลายรูปแบบ ไม่ว่าจะเป็นแบบอนาล็อกและแบบดิจิทัล หรือทั้งแบบอนาล็อกและ ดิจิตอลผสมกัน นอกจากนี้ Proteus ยังสามารถออกแบบลายวงจรพิมพ์ (PCB) ได้อีกด้วย จุดเด่นของ โปรแกรม Proteus ที่เป็นที่ยอมรับและชื่นชอบก็คือ การจำลองการทำงานของวงจรอิเล็กทรอนิกส์ ที่ใช้ ไมโครคอนโทรลเลอร์ตระกูลต่าง ๆ ไม่ว่าจะเป็น PIC , MCS-51 , AVR และ ARM เป็นต้น ทำให้นักเขียน โปรแกรมหรือโปรแกรมเมอร์สามารถตรวจสอบได้ว่าโปรแกรม หรือซอสโค้ด (Source Code) ที่เขียนขึ้นมา นั้น สามารถสนับสนุนกับวงจรฮาร์ดแวร์ที่ต่อได้หรือไม่ ถ้าโปรแกรม (Source Code)ที่เขียนขึ้น ไม่สนับสนุน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กับวงจรฮาร์ดแวร์ที่ต่อโปรแกรมเมอร์ก็จะทำการพัฒนาโปรแกรม (Source Code) ที่เขียนขึ้นใหม่ หรือ ปรับปรุงวงจรฮาร์ดแวร์ใน Proteus จนกว่าโปรแกรมที่เขียนขึ้นและฮาร์ดแวร์ที่ต่อ สามารถสนับสนุนซึ่งกัน และกัน ทำให้การสร้างโครงการต่าง ๆ สามารถประหยัดเวลาและค่าใช้จ่ายเป็นอย่างมาก เพราะในอดีตการ เขียนโปรแกรมขึ้นมา นั้น จะต้องต่อวงจรจริงเพื่อทดสอบ ทำให้เสียเวลาและค่าใช้จ่ายมาก ในกรณีที่วงจรถ่ายและโปรแกรมที่เขียนขึ้นไม่สนับสนุนซึ่งกันและกัน [7]



รูปที่ 2.4 หน้าต่างเข้าใช้งานและโลโก้ของโปรแกรม Proteus

## 2.5 ศึกษาการทำงานของ IC4040

ในทางเทคนิคแล้ว IC4040 เป็นชิปตัวนับระลอกคลื่นไบนารี 12 ขั้นตอน กล่าวง่ายๆ ว่า อุปกรณ์จะสร้างเอาต์พุตความถี่ Delay ที่คำนวณได้เพื่อตอบสนองต่อทุกพัลส์ที่ใช้กับอินพุต CLK Delay นี้จะเพิ่มขึ้นในอัตรา  $2^n$  โดยที่  $n$  คือลำดับพินเอาต์พุตในลำดับของเอาต์พุต ซึ่งคุณสมบัติหลักและข้อกำหนดของ IC4040 ได้ตั้งนับเฟอ์เต็ม 12 เอาต์พุตซึ่งแบ่ง CLK Input ที่อัตรา  $2^n$  โดยที่  $n$  เป็นลำดับพินเอาต์พุตเริ่มต้นจาก Q0 จนถึง Q11 การจัดลำดับเอาต์พุตข้างต้นเกิดขึ้นเพื่อตอบสนองต่อทุกขอบที่ตกลงมาของนาฬิกาที่ใช้กับพิน CP เป็นอินพุตของ CLK IC จะตอบสนองแม้กับพัลส์นาฬิกาที่ตกลงมาอย่างช้าๆ อย่างมีประสิทธิภาพเช่นกัน อินพุตจะรีเซ็ตต้นแบบแบบอะซิงโครนัสและ MR จะรีเซ็ตเอาต์พุตทั้งหมดให้เป็นศูนย์เมื่อใช้ลอจิกสูง ในขณะที่ลอจิกต่ำคงที่จึ่งช่วยให้ IC ทำงานต่อไปได้ IC สามารถทำงานได้อย่างสมบูรณ์ด้วย  $V_{dd}$  ที่ต่ำถึง 3V และคงคุณสมบัติการทำงานคงที่แม้ในแรงดันไฟฟ้าประมาณ 15V [8]

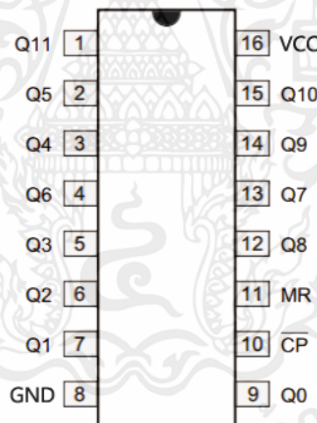
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 อุปกรณ์ IC4040

ซึ่งพารามิเตอร์สำหรับ IC4040 มีดังนี้

- 1) แรงดันไฟจ่าย ( $V_{dd}$ ) : โดยปกติระหว่าง 3V - 15V , 18V เป็นขีดจำกัดสูงสุด
- 2) แรงดันไฟฟ้าขาเข้า ( $V_{dd}$ ) : แรงดันไฟฟ้าที่อาจใช้กับอินพุตเช่น CP, MR ฯลฯ โดยทั่วไปควรต่ำกว่า  $V_{dd}$  หรือความดันที่ใช้มากที่สุด ก็คือ  $V_{dd} + 0.5V$
- 3) กระแสไฟในการทำงานที่เหมาะสมที่สุด : 50mA เนื่องจากมีเอาต์พุตที่เกี่ยวข้องจำนวนมาก



รูปที่ 2.6 Pinout ของ IC4040

จากรูปที่ 4 ได้แสดงการกำหนดค่าพินเอาต์ของ IC4040 ซึ่งมี Pinout Q0 ถึง Q11 เป็นเอาต์พุตของตัว IC ดังนี้

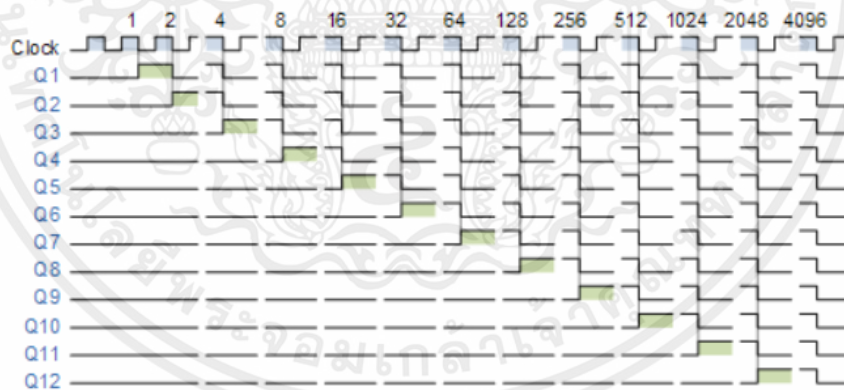
- 1)  $V_{ss}$  เป็น PIN ขา Ground
- 2)  $V_{dd}$  เป็น PIN ขาบวก (ไฟบวก)
- 3) MR เป็น Reset PIN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4) CP เป็น Input นาฬิกา

### 2.5.1 ลำดับเวลา (Timing Sequence)

การวิเคราะห์ลำดับเวลาเอาต์พุตของ IC4040 ดังที่แสดงในแผนภาพต่อไปนี้ จะสามารถเข้าใจรายละเอียดต่อไปนี้ ทรานซิชันที่อินพุต MR สูง เอาต์พุต IC จะไม่ตอบสนอง ทันทีที่มันเหลือน้อย IC จะเริ่มตอบสนองและนับ CLK ที่อินพุต CP ฟินเอาต์พุตแรก Q0 สูงขึ้นหลังจาก CLK  $2^n$  ที่ CP ( $2^0 = 1$ ) หมายความว่า Q0 จะสูงขึ้นที่ขอบตกของพัลส์แรกและต่ำลงเพื่อตอบสนองต่อขอบขาของ CLK ต่อมาเป็นต้น ในทำนองเดียวกัน Q1 จะสูงขึ้นหลังจาก  $2^1 = 2$  ซึ่งหมายความว่า จะสูงขึ้นทันทีที่ตรวจพบขอบตกของ CLK ที่ 2 และลดลงที่ขอบตกของ CLK ที่ 4 ต่อมาเป็นต้น เหมือนกัน Q2 จะสูงขึ้นและต่ำลงหลังจาก  $2^2 =$  ขอบตกของ CLK ที่ 4 และอื่น ๆ ลำดับข้างต้นจะดำเนินต่อไปจนถึง Q11 เพื่อตอบสนองต่อสัญญาณนาฬิกาที่ต่อเนื่องที่ CP หมายความว่าถ้าสมมติว่า CP ไวเวอร์คล็อกด้วยพัลส์ 1Hz Q11 จะสูงขึ้นหลังจาก 211 วินาทีหรือหลังจาก 2048 วินาทีซึ่งเท่ากับ 34 นาทีโดยประมาณ ลองนึกภาพช่วงดีเลย์ที่ทำได้โดยการเพิ่มอินพุตนาฬิกาเป็นวินาทีหรืออาจจะเป็นนาที่



รูปที่ 2.7 ลำดับเวลาของ IC4040

จากการวิเคราะห์รายละเอียดข้างต้นของข้อมูล IC4040 เราสามารถสรุปได้ว่า โดยทั่วไปแล้ว IC นั้นเหมาะสำหรับการใช้งานทั้งหมดที่เกี่ยวข้องกับข้อกำหนดในการแบ่งความถี่ หรือข้อกำหนดในการสร้างช่วงเวลา Delay ดังนั้นจึงเหมาะอย่างยิ่งสำหรับการใช้งานวงจรแบ่งความถี่ ตัวจับเวลาระยะเวลา ไพกะทริบ และการใช้งานอื่นๆ ที่คล้ายคลึงกัน [9]

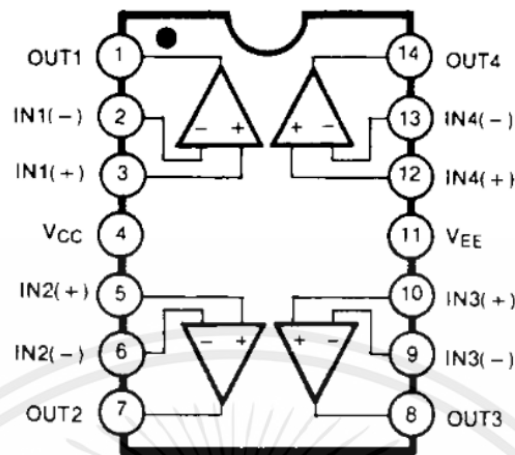
## 2.6 ศึกษาการทำงานของอุปกรณ์ LF147/LF347

LF147/LF347 เป็นเครื่องขยายสัญญาณอินพุตแบบ Quad JFET ความเร็วสูงที่ประหยัดค่าใช้จ่าย พร้อมระบบตัดต่อแรงดันไฟฟ้า (BI-FET II technology II) อุปกรณ์นี้ต้องการกระแสไฟต่ำ และยังรักษาผลิตภัณฑ์แบนด์วิดท์ที่เพิ่มขึ้นอย่างมากและอัตราการแกว่งที่รวดเร็ว นอกจากนี้ อุปกรณ์ป้อนข้อมูล JFET แรงดันสูงที่ตรงกันยังให้ค่า Bias ในการป้อนข้อมูลต่ำมากและกระแสออฟเซต LF147 เข้ากันได้กับ LM148 มาตรฐาน พีเจอร์รี่ช่วยให้นักออกแบบสามารถอัพเกรดประสิทธิภาพโดยรวมของ LF148 และ LM124 ที่มีอยู่ได้ทันที [11]

LF147 อาจใช้ในแอปพลิเคชัน เช่น อินทิเกรเตอร์ความเร็วสูง , Fast D/A Converters , วงจรตัวอย่างและโฮลด์และวงจรอื่น ๆ หลายวงจรที่ต้องการแรงดันออฟเซตอินพุตต่ำกระแส Bias อินพุตต่ำ , อิมพีแดนซ์อินพุตสูง , อัตราการแกว่งสูง และแบนด์วิดท์กว้าง อุปกรณ์มีเสียงรบกวนต่ำและแรงดันไฟฟ้าออฟเซต



รูปที่ 2.8 อุปกรณ์ Op – Amp LF347N



รูปที่ 2.9 PIN OUT และส่วนประกอบภายในของ Op – Amp LF347

คุณสมบัติของอุปกรณ์ Op – Amp LF347 มีดังนี้

- 1) แรงดัน  $V_{CC} : \pm 18 V$
- 2) แรงดันไฟฟ้าขาเข้าแบบอนุพันธ์  $V_{I(DIFF)} : 30 V$
- 3) ช่วงของแรงดันขาเข้า  $V_I : \pm 15 V$
- 4) ระยะเวลาเอาต์พุตของ Short Circuit ใช้ได้อย่างต่อเนื่อง
- 5) การกระจายของพลังงาน  $P_D : 570 mW$  [11]

## 2.7 ศึกษาการทำงานของ Device Under Test 2 ขา (DUT)

### 2.7.1 การทำงานของ Diode

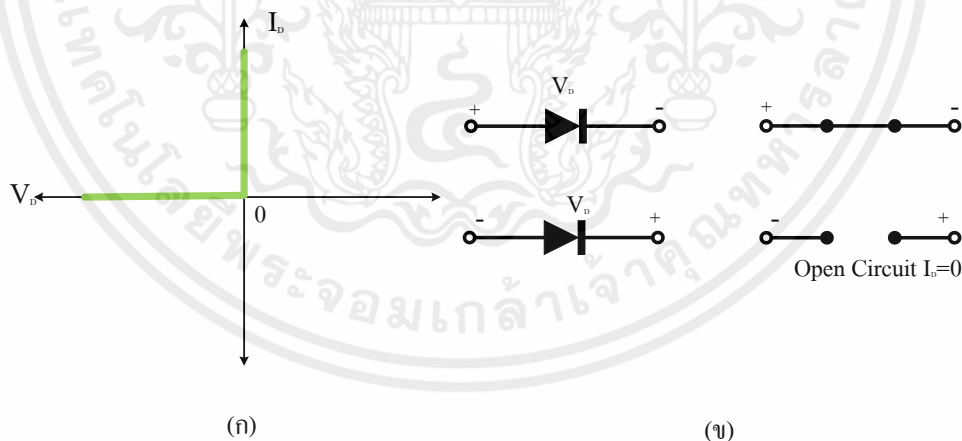
ไดโอดเป็นสารกึ่งตัวนำซิลิคอนรอยต่อ P – N ได้พัฒนามาเป็นอุปกรณ์อิเล็กทรอนิกส์เรียกว่า ไดโอด ไดโอดมี 2 ขั้ว ขั้วที่ขึ้นสารชนิด P เรียกว่า แอโนด Anode : (A) และขั้วที่ขึ้นสารชนิด N เรียกว่า แคโทด (Cathode : K) สัญลักษณ์ของไดโอดเปรียบเทียบกับสารกึ่งตัวนำ 1 รอยต่อ (P - N) ดังนั้นแรงดันที่ตกคร่อมไดโอดเมื่อไดโอดนำกระแสจึงเท่ากับ 0.6V มันคือแรงดันตกคร่อมรอยต่อ P – N นั่นเอง ไดโอดเกิดจากการนำสารกึ่งตัวนำชนิด N ติดเข้ากับสารกึ่งตัวนำชนิด P เชื่อมสายไฟเข้ากับขั้วไฟฟ้าทั้งสอง เมื่อยังไม่มีกระแสให้แรงดันไฟฟ้าอิเล็กทรอนิกส์จาก N จะเคลื่อนที่ข้ามรอยต่อไปที่ P เกิดโซนดีพลีชัน (Depletion) ชั้นโซนนี้เปรียบเทียบกับกำแพง

ป้องกันการเคลื่อนที่ของอิเล็กตรอน ถ้าโชนนี้มีขนาดใหญ่ขึ้นการเคลื่อนที่ของอิเล็กตรอนอิสระจะยากขึ้นและอาจทำให้อิเล็กตรอนหยุดการเคลื่อนที่ได้

อย่างไรก็ตามถ้าควบคุมให้โชนนี้เล็กลง การเคลื่อนที่ก็จะง่ายขึ้น อิเล็กตรอนอิสระจาก  $N$  เคลื่อนที่ข้ามรอยต่อไปลงหลุมที่  $P$  ทำให้เกิดโชนตีพลีชั่น เป็นฉนวนกั้นการไหลของอิเล็กตรอน เพื่อจะทำให้อิเล็กตรอนสามารถเคลื่อนที่ผ่านโชนนี้ได้ง่ายขึ้น เราต้องทำให้โชนนี้แคบลง โดยการต่อขั้ว  $N$  ของไดโอดเข้ากับขั้วลบของแบตเตอรี่ และขั้วบวกเข้ากับขั้ว  $P$  ทำให้อิเล็กตรอนอิสระใน  $N$  ถูกดันด้วยแรงดันทางไฟฟ้า ส่วนโฮลขั้ว  $P$  จะถูกดันด้วยแรงทางไฟฟ้าเช่นเดียวกัน ถ้าเราให้แรงดันทางไฟฟ้ามักพอ โชนนี้จะแคบจนหายไป และอิเล็กตรอนอิสระสามารถเคลื่อนที่ผ่านรอยต่อได้อย่างง่ายดาย เหมือนกับไม่มีแรงเสียดทาน หรือความต้านทานในทางกลับกัน ถ้าต่อขั้วลบเข้ากับ  $P$  และขั้วบวกเข้ากับ  $N$  การไหลของอิเล็กตรอนจะเป็นไปได้ยาก เพราะการเคลื่อนที่เป็นไปในทิศทางตรงกันข้าม โชนตีพลีชั่นจะหนาขึ้น เป็นกำแพงกั้นการไหลของกระแสไฟฟ้าอิเล็กตรอนและโฮลไม่สามารถเคลื่อนที่ได้อย่างอิสระ

### 2.7.2 ไดโอดในทางอุดมคติ (Ideal Diode)

ไดโอดในอุดมคติมีลักษณะเหมือนสวิตช์ที่สามารถนำกระแสไหลผ่านได้ในทิศทางเดียว



รูปที่ 2.10 กราฟไดโอดในอุดมคติ

จากรูปที่ 2.10 แสดงให้เห็นว่าถ้าต่อขั้วแบตเตอรี่ให้เป็นแบบไป้อตรงไดโอดจะเปรียบเป็นเสมือนกับสวิตช์ที่ปิด (Close Switch) หรือไดโอดลัดวงจร (Short Circuit)  $I_D$  ไหลผ่านไดโอด

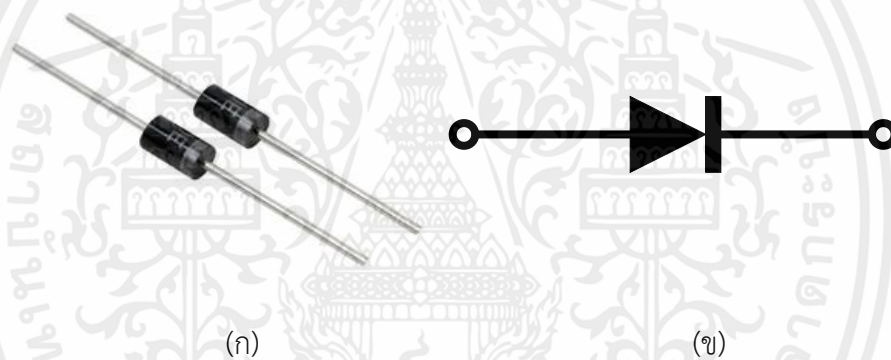
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ได้ แต่ถ้าต่อขั้วแบตเตอรี่แบบไขว้กลับ ไดโอดจะเปรียบเป็นเสมือนสวิตช์เปิด (Open Switch) หรือเปิดวงจร (Open Circuit) ทำให้  $I_D$  เท่ากับศูนย์ [12]

### 2.7.3 ประเภทของไดโอด

#### 2.7.3.1 ไดโอดธรรมดา (Normal diode)

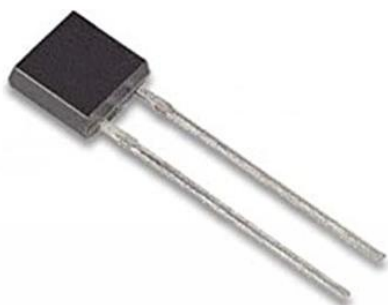
มีรูปร่างหลายแบบแต่ที่นิยมใช้ในวงจรอิเล็กทรอนิกส์จะมี 2 ขั้วลักษณะคล้ายตัวต้านทานทำหน้าที่ควบคุมการไหลของกระแสไฟฟ้าให้ไหลไปทางเดียวคือกระแสไฟฟ้าจะไหลผ่านไดโอดได้ถ้าต่อขั้ว P กับขั้วบวกและ N กับขั้วลบ แต่ความต้านทานของไดโอดไม่คงที่ เพราะความสัมพันธ์ระหว่างกระแส กับความต่างศักย์ที่คร่อมไดโอดไม่เป็นเส้นตรง



รูปที่ 2.11 (ก) ลักษณะของอุปกรณ์ Normal Diode 1N4007  
(ข) สัญลักษณ์ลักษณะของ Normal Diode

#### 2.7.3.2 ไดโอดวาระกเตอร์หรือวาริแคป (Varactor or Varicap Diode)

ไดโอดวาระกเตอร์หรือวาริแคปเป็นไดโอดที่มีลักษณะพิเศษ คือ สามารถปรับค่าคาปาซิแตนซ์เชื่อมต่อ  $C_t$  ได้โดยการปรับค่าแรงดันไบอัสกลับ ไดโอดประเภทนี้มีโครงสร้างเหมือนกับไดโอดทั่วไป ขณะแรงดันไบอัสกลับ (Reverse Bias Voltage ;  $V_r$ ) มีค่าต่ำ Depletion Region จะแคบลงทำให้  $C_t$  ครอบรอบต่อมีค่าสูง แต่ในทางตรงข้ามถ้าเราปรับ  $V_r$  ให้สูงขึ้น Depletion Region จะขยายกว้างขึ้น ทำให้  $C_t$  มีค่าต่ำ จากลักษณะดังกล่าว เราจึงนำวาริแคปไปใช้ในวงจรปรับความถี่ เช่น วงจรจูนความถี่อัตโนมัติ (Automatic Fine Tuning ; AFC) และวงจรกรองความถี่ซึ่งปรับช่วงความถี่ได้ตามต้องการ (Variable Bandpass Filter) เป็นต้น



(ก)



(ข)

รูปที่ 2.12 (ก) ลักษณะของอุปกรณ์ Varactor or Varicap Diode

(ข) สัญลักษณ์ลักษณะของ Varactor or Varicap Diode

2.7.3.3 ไดโอดเปล่งแสงหรือแอลอีดี (Light Emitting Diode ; LED)

LED เป็นไดโอดที่ใช้สารประเภทแกลเลียมอาร์เซไนด์ฟอสไฟด์ (Gallium Arsenide Phosphide ; GaAsP) หรือสารแกลเลียมฟอสไฟด์ (Gallium Phosphide ; GaP) มาทำเป็นสารกึ่งตัวนำชนิด P และ N แทนสาร Si และ Ge สารเหล่านี้มีคุณลักษณะพิเศษคือสามารถเรืองแสงได้เมื่อได้รับไบอัสตรง การเกิดแสงที่ตัว LED นี้เรียกว่าอิเล็กโทรลูมิเนสเซนซ์ (Electroluminescence) ปัจจุบันนิยมใช้ LED แสดงผลในเครื่องมืออิเล็กทรอนิกส์ เช่น เครื่องคิดเลข , นาฬิกา เป็นต้น



(ก)



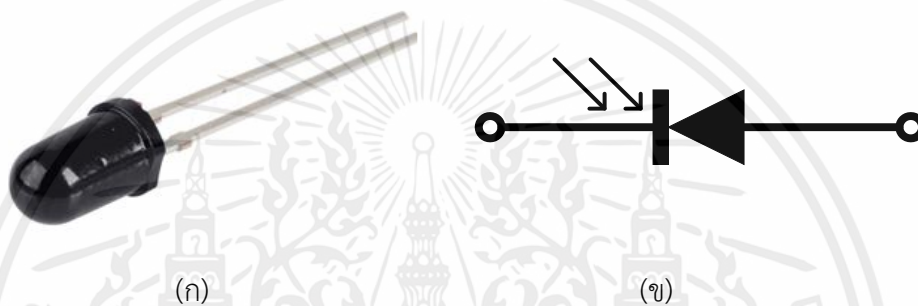
(ข)

รูปที่ 2.13 (ก) ลักษณะของอุปกรณ์ LED (ข) สัญลักษณ์ลักษณะของ LED

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 2.7.3.4 โฟโตไดโอด (Photo Diode)

โฟโตไดโอดเป็นไดโอดที่อาศัยแสงจากภายนอกผ่านเลนส์ ซึ่งฝังตัวอยู่ระหว่างรอยต่อ P - N เพื่อกระตุ้นให้ไดโอดทำงาน การต่อโฟโตไดโอดเพื่อใช้งานจะเป็นแบบไบอัสกลับ ทั้งนี้เพราะไม่ต้องการให้โฟโตไดโอดทำงานในทันทีทันใด แต่ต้องการให้ไดโอดทำงานเฉพาะเมื่อมีปริมาณแสงสว่างมากพอตามที่กำหนดเสียก่อน กล่าวคือเมื่อเลนส์ของโฟโตไดโอดได้รับแสงสว่างจะเกิดกระแสรั่วไหล ปริมาณกระแสรั่วไหลนี้เพิ่มขึ้นตามความเข้มของแสง



(ก)

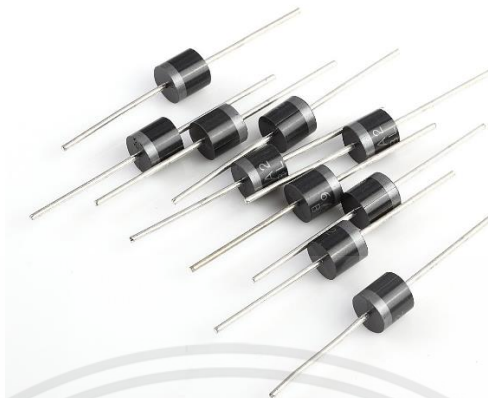
(ข)

รูปที่ 2.14 (ก) ลักษณะของอุปกรณ์ Photo Diode

(ข) สัญลักษณ์ลักษณะของ Photo Diode

#### 2.7.3.5 ไดโอดกำลัง (Power Diode)

ไดโอดกำลัง เป็นไดโอดที่ออกแบบให้บริเวณรอยต่อมีช่วงกว้างมากกว่า ไดโอดทั่วไป เพื่อนำไปใช้กับงานที่มีกำลังไฟฟ้าสูง กระแสสูงและทนต่ออุณหภูมิสูงได้ เช่น ประกอบเป็นวงจรเรียงกระแส ในอิเล็กทรอนิกส์กำลังเป็นต้น จะเห็นได้ว่าเมื่อพิกัดกระแสไฟฟ้ามูลค่าหลายร้อยแอมป์ ทำให้ไดโอดมีอุณหภูมิขณะทำงานสูง โดยทั่วไปจึงนิยมใช้ร่วมกับตัวระบายความร้อน (Heat Sinks) เพื่อเพิ่มพื้นที่ระบายความร้อนภายในตัวไดโอดกำลัง



รูปที่ 2.15 ลักษณะของอุปกรณ์ Photo Diode

#### 2.7.3.6 ซีเนอร์ไดโอด (Zener diode)

ซีเนอร์ไดโอดเป็นไดโอดชนิดไดโอดชนิดพิเศษที่สร้างให้มีการทำงานแตกต่างจากไดโอดเรียงกระแสทั่วไปกล่าวคือ เมื่อให้ Bias ตรงกับซีเนอร์ไดโอดจะมีการทำงานที่เหมือนกับไดโอดทั่วไปคือนำกระแสได้และมีแรงดันตกคร่อมซีเนอร์ไดโอดขณะได้รับ Bias ตรง (Forward Bias) เท่ากับ 0.6V และเมื่อซีเนอร์ไดโอดได้รับ Bias กลับ (Reverse Bias) ถึงค่าแรงดันที่กำหนดไว้ (กำหนดค่าแรงดันขึ้นในกระบวนการสร้างซีเนอร์ไดโอด เช่น 2.2V 5.1V 6V 10V 12V เป็นต้น) ซีเนอร์ไดโอดจะนำกระแสได้และจะเกิดแรงดันตกคร่อมตัวมันคงที่เท่ากับค่าแรงดันที่กำหนดจากบริษัทผู้ผลิต ซึ่งลักษณะของซีเนอร์ไดโอดแสดงดังรูป



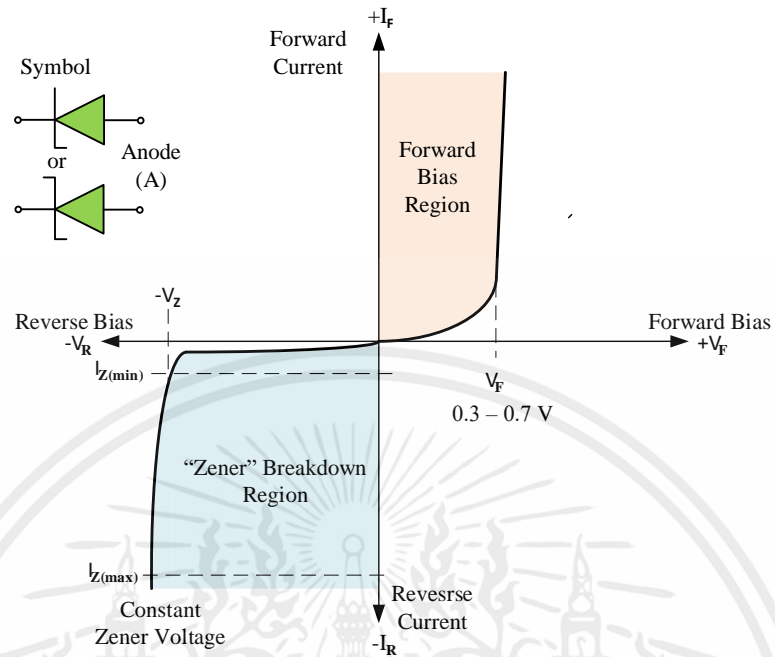
รูปที่ 2.16 ลักษณะของอุปกรณ์ซีเนอร์ไดโอด (Zener Diode)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

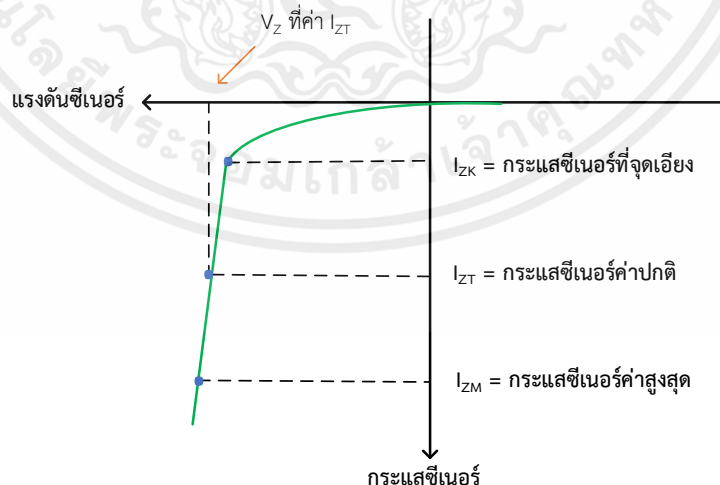
### 1) คุณสมบัติของซีเนอร์ไดโอด

การพังทลายของซีเนอร์ไดโอด (Zener Breakdown) การพังทลายของไดโอดแบ่งเป็น 2 ชนิด คือ การพังทลายแบบอะวาลานซ์ หมายความว่าเมื่อไดโอดได้รับ Bias กลับ (Reverse Bias) ที่แรงดันสูงมากๆจะทำให้มีกระแสไหลย้อนกลับผ่านไดโอดจำนวนมาก ทำให้รอยต่อของไดโอดทะลุและใช้งานไม่ได้อีก การพังทลายอีกแบบหนึ่งคือ การพังทลายแบบซีเนอร์เป็นการพังทลายที่เกิดขึ้นกับแรงดัน Bias กลับค่าต่างๆซึ่งกำหนดได้จากการโดปสารกึ่งตัวนำที่ใช้สร้างเป็นซีเนอร์ไดโอด การพังทลายแบบซีเนอร์ไดโอดนี้จะมีการแสไหลผ่านซีเนอร์ไดโอดจำนวนหนึ่ง ซึ่งต้องรักษาไม่ให้เกิดค่าพิกัดสูงสุดและจะทำให้เกิดสถานะที่แรงดันตกคร่อมซีเนอร์ไดโอดมีค่าคงที่เรียกว่า แรงดันซีเนอร์ คุณสมบัติข้อนี้สามารถนำซีเนอร์ไดโอดไปสร้างเป็นวงจรควบคุมแรงดันไฟตรงจากแหล่งจ่ายไฟฟ้ากระแสตรงให้มีค่าแรงดันคงที่ได้ ซีเนอร์ไดโอดที่มีอยู่ในทั่วไปมีขนาดแรงดันซีเนอร์ตั้งแต่ 1.8V ถึง 200V

คุณสมบัติของการพังทลาย (Breakdown Characteristics) เมื่อพิจารณาจากกราฟลักษณะของการพังทลายของซีเนอร์ไดโอดเมื่อได้รับ Bias กลับ ดังรูปที่ 2.18 เมื่อเพิ่มแรงดัน Bias กลับจนถึงค่าแรงดันซีเนอร์  $V_Z$  จะเกิดกระแสไหลผ่านซีเนอร์ไดโอดมากขึ้น ที่จุดเอียงของกราฟจะมีกระแสไหลผ่านซีเนอร์ไดโอดเท่ากับ  $I_{ZK}$  และถ้าซีเนอร์ไดโอดได้รับแรงดันสูงขึ้นไปอีกกระแสจะเพิ่มขึ้น แต่แรงดันซีเนอร์จะคงที่ ถ้าเพิ่มกระแสเกินกว่าค่ากระแสซีเนอร์สูงสุด  $I_{ZM}$  แรงดันซีเนอร์จะไม่คงที่ [13, 14]



รูปที่ 2.17 กราฟลักษณะกระแสและแรงดันของซีเนอร์ไดโอด และสัญลักษณ์ของซีเนอร์ไดโอด  
 ดังนั้นการนำซีเนอร์ไดโอดไปใช้ในการควบคุมให้แรงดันไฟฟ้ากระแสตรงให้คงที่ โดยใช้ค่าแรงดันซีเนอร์นั้น จึงต้องออกแบบวงจรควบคุมให้มีกระแสซีเนอร์ไดโอดช่วงระหว่างค่ากระแส  $I_{ZK}$  ถึงค่า  $I_{ZM}$  สำหรับกระแส  $I_{ZT}$  หมายถึงค่ากระแสทดสอบค่าแรงดันซีเนอร์ซึ่งเป็นค่ากระแสที่พิกัดของแรงดันซีเนอร์ตามค่าที่ผู้ผลิตกำหนด [15, 16]



รูปที่ 2.18 กราฟลักษณะกระแสและแรงดันของซีเนอร์ไดโอดเมื่อรับ Bias กลับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 1 คุณสมบัติของซีเนอร์ไดโอด

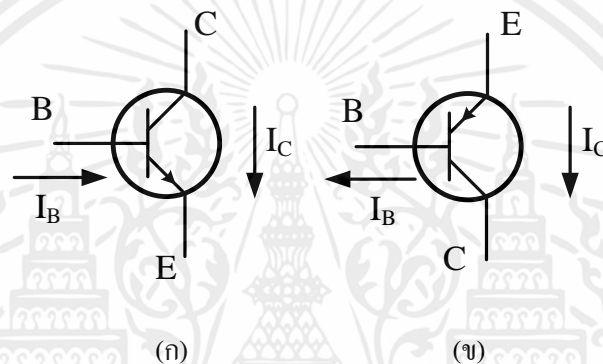
ชนิด	ระดับของแรงดันซีเนอร์		ความต้านทานต่อไฟฟ้าสลับสูงสุดของซีเนอร์			กระแสไหลย้อนกลับ		กระแสที่ $T_A = 25^\circ\text{C}$	กระแสซีเนอร์สูงสุด
	$V_Z$ (V)	$I_{ZT}$ (mA)	$r_{zjT}$ ( $\Omega$ )	$r_{zjk}$ ( $\Omega$ )	$I_{ZK}$ (mA)	$I_R$ ( $\mu\text{A}$ )	$V_R$ (V)	$I_R$ (mA)	$I_{ZM}$ (mA)
1N4728	3.3	76	10	400	1.0	150	1	1375	275
1N4729	3.6	69	10	400	1.0	100	1	1260	252
1N4730	3.9	64	9	400	1.0	100	1	1190	234
1N4731	4.3	58	9	400	1.0	50	1	1070	217
1N4732	4.7	53	7	500	1.0	10	1	970	193
1N4733	5.1	49	8	550	1.0	10	1	890	178
1N4734	5.6	45	5	600	1.0	10	2	810	162
1N4735	6.2	41	2	700	1.0	10	3	730	146
1N4736	6.8	37	3.5	700	1.0	10	4	660	133
1N4737	7.5	34	4.0	700	0.5	10	6	605	121
1N4738	8.2	31	4.5	700	0.5	10	5	550	110
1N4739	9.1	28	5.0	700	0.5	10	7	500	100
1N4740	10	25	7	700	0.25	10	7.6	454	91

## 2.8 ศึกษาการทำงานของทรานซิสเตอร์ (Transistor)

### 2.8.1 วงจรทรานซิสเตอร์

ตัวทรานซิสเตอร์นั้นถูกสร้างโดยมีพื้นฐานมาจากตัวไดโอด ดังนั้นส่วนที่เป็นรอยต่อระหว่างเบส (Base) กับอีมิเตอร์ (Emitter) หรือเรียกว่า Base-Emitter Junction จะไม่นำกระแสจนกว่าไบอัสตรง (Forward Bias) มีจะมีค่าแรงดันสูงกว่า 0.7 โวลต์ และหากกระแสที่ไหลผ่านตัวทรานซิสเตอร์มากเกินไป ก็จะทำให้ทรานซิสเตอร์ ร้อนจัดและทำงานผิดปกติได้

โดยทั่วไปแล้วทรานซิสเตอร์จะมีโครงสร้าง 2 ชนิด คือ แบบ NPN และ PNP เนื่องจากทรานซิสเตอร์โครงสร้างภายในถูกผลิตขึ้นแบบนี้ ดังนั้นในการเลือกใช้งานทรานซิสเตอร์ จะต้องเลือกใช้งานให้ถูกต้องด้วยขาของทรานซิสเตอร์มี 3 ขา และมีชื่อเรียกคือ ขาเบส (Base : B), ขาคอนเนคเตอร์ (Collector : C) และ ขาอีมิเตอร์ (Emitter : E) โดยสัญลักษณ์ของทรานซิสเตอร์แบบ NPN และ PNP ดังแสดงได้ดังรูปที่ 2 (ก) และ 2 (ข) ซึ่งโครงสร้างแบบ NPN สังเกตว่าสัญลักษณ์ทรานซิสเตอร์หัวลูกศรจะพุ่งออก และ โครงสร้างแบบ PNP สังเกตว่าสัญลักษณ์ทรานซิสเตอร์หัวลูกศรจะพุ่งเข้า



รูปที่ 2.19 สัญลักษณ์ของทรานซิสเตอร์  
(ก) NPN (ข) PNP

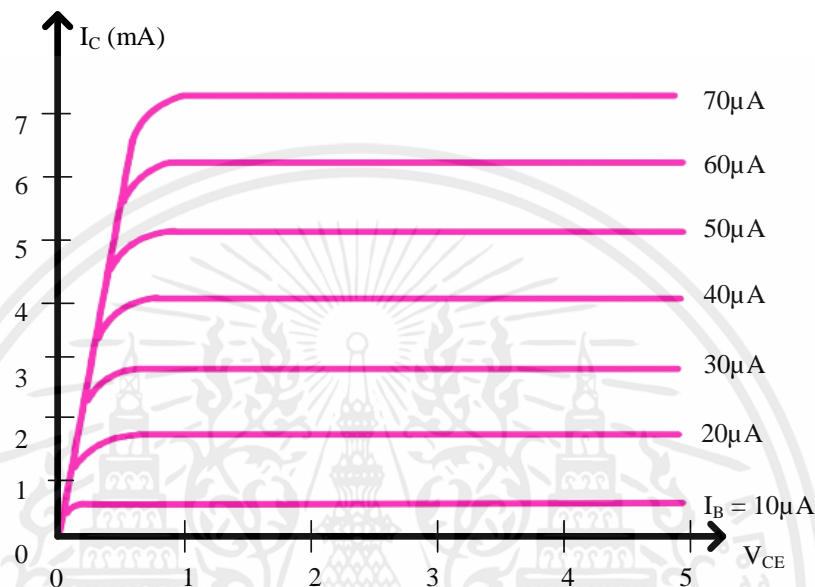
#### 2.8.1.1 การทำงานของทรานซิสเตอร์ NPN

เมื่อมีกระแสไฟฟ้าไบอัสเข้าไปที่ขา B เล็กน้อย ทรานซิสเตอร์ก็จะอยู่ในสภาวะทำงาน ยอมให้กระแสไฟฟ้าที่มากกว่าหลายเท่าไหลผ่านขา C ไปยังขา E แต่ในทางตรงกันข้าม แต่ถ้าไม่มีกระแสไฟฟ้าที่ขา B เลย ทรานซิสเตอร์จะอยู่ในสภาวะ Cut-Off หรือเป็นการบล็อกไม่ให้กระแสไฟฟ้าไหลจากขา C ไป E ได้ (แบบ NPN นี้ ขา E ทำหน้าที่เป็นกราวด์)

#### 2.8.1.2 การทำงานของทรานซิสเตอร์ PNP

ทรานซิสเตอร์แบบนี้จะทำงานแบบตรงกันข้ามกับ NPN คือ โดยขา C จะทำหน้าที่เป็นกราวด์แทนขา E ในทรานซิสเตอร์ NPN เมื่อมีไม่มีกระแสไฟฟ้าที่ขา B ทรานซิสเตอร์จะทำการบล็อกไม่ให้กระแสไฟฟ้าไหลผ่านขา E ไปยัง C ได้ แต่ถ้ามีกระแสไหล

ออกจากขา B (กระแสติดลบ) ทรานซิสเตอร์ PNP ยอมให้กระแสไฟฟ้าที่มากกว่าไหลผ่านจากขา E ไปขา C [17]



รูปที่ 2.20 การทำงานของทรานซิสเตอร์

ทรานซิสเตอร์มีโหมดการทำงานอยู่ 4 โหมดได้แก่

1. Active Mode หรือ Active Region เป็นโหมดที่มีการทำงานแบบกระแสไฟฟ้าที่ไหลผ่านขา CE แบบเดียวกันกับกระแสไฟฟ้าที่ขา B คือถ้ายังมีกระแสไฟฟ้ามากระตุ้นที่ขา B มีค่ามากเท่าไร กระแสที่ CE ก็จะมีค่ามากๆ (แต่จะมากไม่เกินแหล่งจ่าย  $V_{CC}$ )
2. Cut-Off Mode หรือ Cut-Off Region คือโหมดที่ไม่มีการทำงานหรือหยุดการทำงานซึ่งในช่วงนี้จะไม่มีการเสมาที่ขา B ซึ่งก็จะไม่มีกระแสที่ CE ด้วยทรานซิสเตอร์จะทำหน้าที่เปรียบเสมือนเป็นสวิตช์ที่เปิดวงจร
3. Saturation Mode หรือ Saturation Region คือโหมดอิ่มตัวซึ่งจะคล้ายๆกับ Active Mode แต่ในโหมดนี้จะมีการจ่ายกระแสไฟฟ้าที่ขา B จนทำให้เกิดการอิ่มตัว ซึ่งก็จะส่งผลทำให้ทรานซิสเตอร์ทำหน้าที่เปรียบเสมือนเป็นสวิตช์ปิดวงจรแบบสนิทหรือ Shot Circuit และทำ

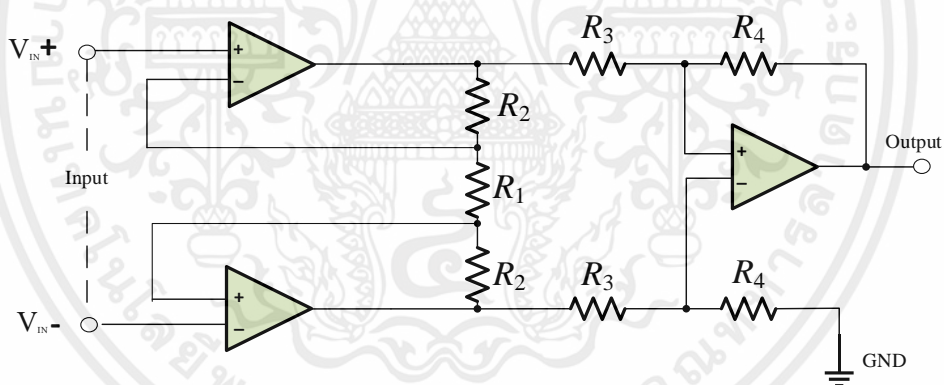
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้ทำกระแสไฟฟ้าไหลผ่านระหว่างขา CE ได้มากที่สุด หรือจะได้รับแรงดันจากแหล่งจ่ายโดยตรง แรงดัน CE มีค่าเท่ากับแรงดัน  $V_{CC}$  ซึ่งเป็นโหมดที่นิยมใช้ เพราะ LOAD จะได้รับกระแสสูงสุด

4. Reverse-Active Mode การทำงานในโหมดนี้จะคล้ายกับ Active Mode แต่ใน โหมดนี้ กระแสไฟฟ้าจะไหลจากขา E ไปขา C แทนซึ่งจะมีใช้ในบางกรณีเท่านั้น [17]

## 2.9 ศึกษาการทำงานของวงจรถยายในงานเครื่องมือวัด (Instrument Amplifier Circuit)

การเชื่อมต่อวงจรถยายในงานเครื่องมือวัดจำเป็นต้องกระทบกับวงจรที่ทำงานอยู่น้อยที่สุด หรือไม่กระทบเลย หรือไม่เกิด Loading effect โดยหลักการคือวงจรจะต้องอินพุตเป็นแบบดิฟเฟอเรนเชียลและมีอิมพีแดนซ์สูงมาก วงจรนี้เรียกว่าวงจรถยายเครื่องมือวัด (Instrument Amplifier Circuit) ดังแสดงในรูปที่ 2.21 ซึ่งจะเห็นว่า มีออปแอมป์ในวงจรส่วนหน้าอีก 2 ตัว และจะเห็นได้ว่า สัญญาณจะถูกป้อนเข้าที่อินพุตของออปแอมป์



รูปที่ 2.21 วงจรถยายในงานเครื่องมือวัด (Instrument Amplifier Circuit)  
สมการอัตราขยายของวงจรสามารถเขียนได้ดังนี้

$$V_o = \frac{R_4}{R_3} \left[ 1 + 2 \frac{R_2}{R_1} \right] (V_{in+} - V_{in-})$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยจากสมการจะเห็นว่า ค่าอัตราขยายของวงจรถูกขึ้นอยู่กั้อัตราส่วนระหว่าง  $R_2$  และ  $R_1$  ดังนั้นเมื่อกำหนดให้  $R_2$  มีอัตราขยายของวงจรถูกค่าคงที่จะขึ้นอยู่กับค่าความต้านทานของ  $R_1$  เพียงอย่างเดียวทำให้สามารถปรับอัตราขยายของวงจรถูกได้ [19, 20]



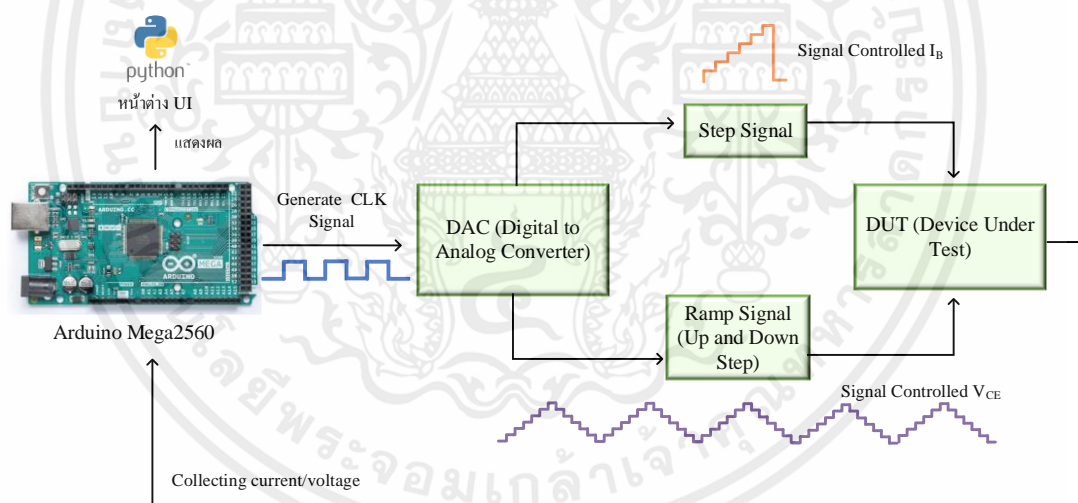
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### บทที่ 3

#### การออกแบบและการจัดทำปฏิญานิพนธ์

##### 3.1 การออกแบบโครงสร้างโดยรวม

ในการสร้างและออกแบบวงจรทำปฏิญานิพนธ์เล่มนี้ ผู้จัดทำได้ทำเครื่องวัดคุณสมบัติอุปกรณ์กึ่งตัวนำโดยใช้ต้นทุนต่ำ ให้การทำงานช่วงการวัด นั้นไม่เกิน  $\pm 10V$  กระแสคอลเลคเตอร์ไม่เกิน  $10mA$  และกระแสเบสไม่เกิน  $100mA$  ซึ่งในหัวข้อนี้ได้นำเสนอในส่วนของ การออกแบบระบบวงจรที่ใช้ในเครื่องวัดอุปกรณ์กึ่งตัวนำ 2 ขาและ 3 ขา โดยภาพรวมการทำงานของระบบสามารถแสดงดังในรูป 3.1 โดยการทำงานของระบบเมื่อ Arduino Mega2560 สร้างสัญญาณพัลส์ไปยัง DAC (Digital to Analog Converter) เพื่อทำการเปลี่ยนสัญญาณพัลส์เป็นสัญญาณ Ramp 16 และ 256 steps และทำการวัดอุปกรณ์ DUT ( Device Under Test) จากนั้นทำการบันทึกค่าและประมวลผลโดยใช้ Arduino Mega2560



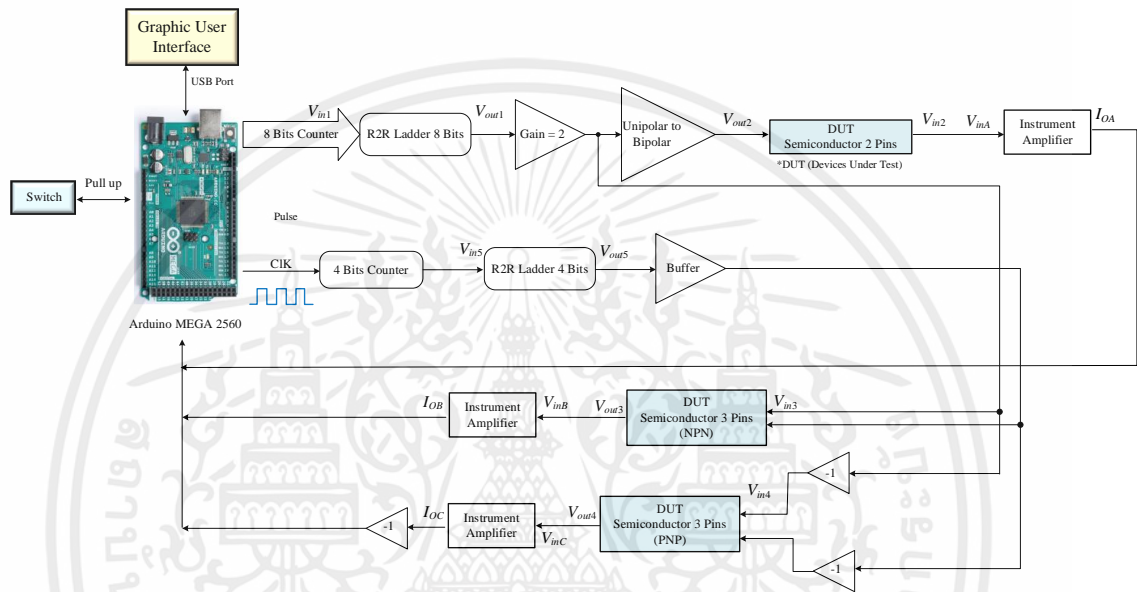
รูปที่ 3.1 บล็อกไดอะแกรมหลักการทำงานของเครื่องวัดคุณสมบัติอุปกรณ์กึ่งตัวนำโดยใช้ต้นทุนต่ำ

##### 3.1.1 ออกแบบภาพรวมเครื่องวัดคุณสมบัติอุปกรณ์กึ่งตัวนำโดยใช้ต้นทุนต่ำ

จากรูปที่ 3.2 แสดงการทำงานภาพรวมของวงจร Curve Tracer แบ่งออกเป็น 2 ส่วน ส่วนแรกจะเป็นการวัดอุปกรณ์ DUT 2 ขาโดยที่ Arduino Mega2560 ที่ทำหน้าที่เจนเนอร์เรท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณ Counter ซึ่งแรงดัน  $V_{in1}$  เป็นสัญญาณ 8 Bits ที่ถูกเจเนอเรทไปยังวงจร R2R Ladder 8 Bits เพื่อทำการแปลงสัญญาณ Digital ( $V_{in1}$ ) เป็น Analog Sawtooth (Ramp) หรือ  $V_{out1}$  จากนั้นทำการขยายสัญญาณโดยใช้ Gain เท่ากับ 2 และเปลี่ยนกราฟสัญญาณ Unipolar เป็น Bipolar จากนั้นนำแรงดัน  $V_{in2}$  ที่เป็น Bipolar ผ่านอุปกรณ์ DUT 2 ขา และ วัดสัญญาณกระแส  $I_{OA}$  ทางอ้อมผ่าน R-Sensor ด้วย Instrument Amp ( $V_{inA}$ )



รูปที่ 3.2 การทำงานเครื่องวัดคุณสมบัติอุปกรณ์กึ่งตัวนำโดยใช้ต้นทุนต่ำ

ในส่วนที่ 2 เป็นวงจรวัดอุปกรณ์ DUT 3 ขา ได้นำเอาอินพุตสัญญาณนาฬิกา  $V_{in5}$  (CLK) จาก Arduino Mega2560 และวงจร 4 บิต Counter เปลี่ยนให้เป็นสัญญาณขั้นบันได 16 Step โดยใช้วงจร R2R Ladder ซึ่งนำแรงดัน  $V_{out5}$  (0 – 10V) เพื่อกำหนดให้เป็นกระแสไบอัส  $I_B$  มาใช้งานในการวัดอุปกรณ์ Transistor NPN เมื่อผ่านอุปกรณ์ DUT 3 ขา จะวัดค่าแรงดัน  $V_{CE}$  จาก  $V_{in3}$  และ วัดสัญญาณกระแสไบอัส  $I_{OB}$  ทางอ้อมผ่าน R-Sensor ด้วย Instrument Amp

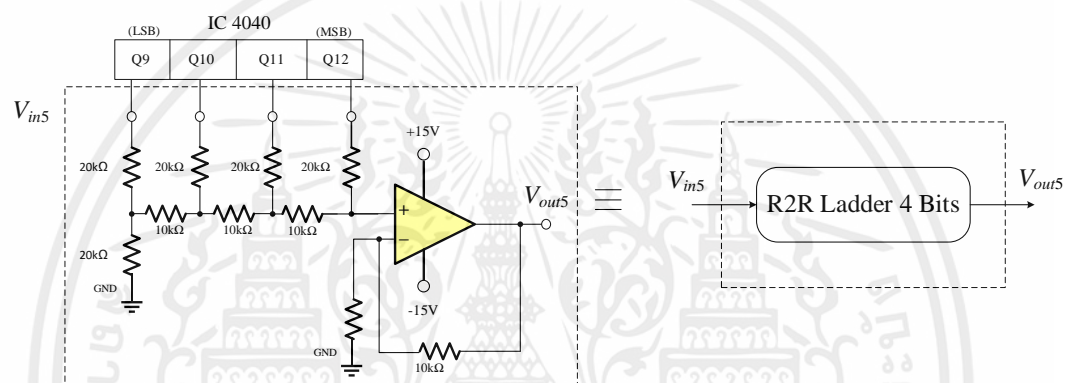
ในส่วนการวัดอุปกรณ์ Transistor PNP มีการทำงานเช่นเดียวกับอุปกรณ์ Transistor NPN ต่างกันที่สัญญาณ  $V_{in4}$  รวมทั้งแรงดัน  $V_{out5}$  (กระแสไบอัส  $I_B$ ) จะถูกแปลงให้เป็นลบเสียก่อน ส่วนการวัดกระแสไบอัส และ แรงดัน  $V_{CE}$  ก็จะใช้แบบเดียวกับ NPN สัญญาณ และ กลับค่าสัญญาณวัดได้ก่อนเข้าประมวลผล ให้กระแสที่วัดออกมาเป็นบวกและทำการเก็บค่าที่ Arduino Mega2560 จากนั้นแสดงผลผ่านหน้าต่าง UI (User Interface)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

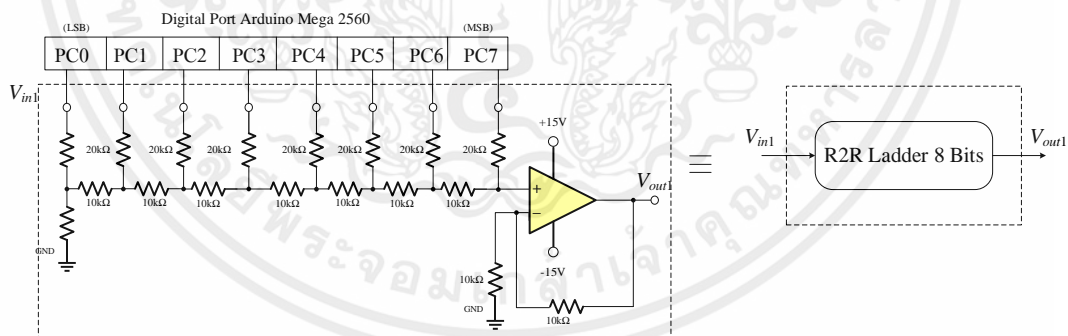
## 3.2 การออกแบบและสร้างวงจร Curve Tracer

### 3.2.1 วงจร R2R Ladder 4 Bits และ 8 Bits

ในการออกแบบวงจร Curve Tracer จะใช้วงจร R2R Ladder ที่ทำหน้าที่เป็นวงจรแปลงดิจิตอลเป็นแอนะล็อก ซึ่งจะใช้ออปแอมป์และตัวต้านทานที่ต่อเป็นโครงข่าย Ladder โดยใช้ตัวต้านทาน 2 ค่า คือ  $10k\Omega$ ,  $20k\Omega$  และใช้ออปแอมป์เบอร์ LF347 โดยทำการออกแบบโปรแกรม Proteus แสดงดังรูปที่ 8 และ รูปที่ 9



รูปที่ 3.3 วงจร R2R Ladder 4 Bits

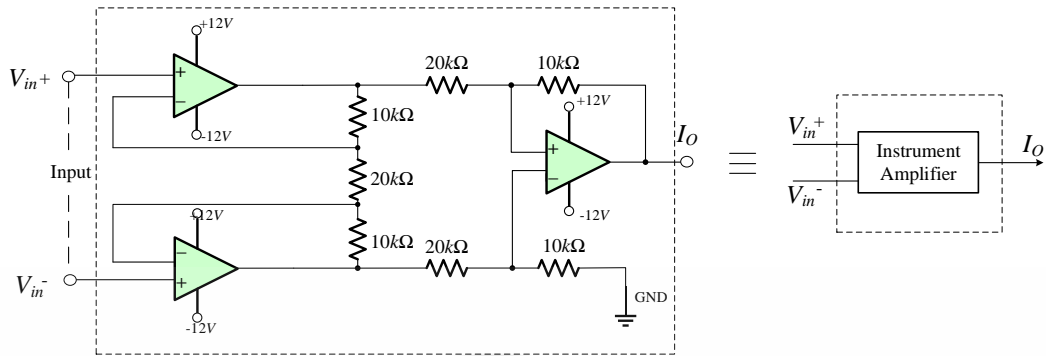


รูปที่ 3.4 วงจร R2R Ladder 8 bits

### 3.2.2 วงจร Instrument Amplifier

วงจร Instrument Amplifier มีอัตราการจัดสัญญาณโหมดร่วมสูง (Common Mode Rejection Ratio) และนำมาใช้งานในส่วนของการวัดแรงดันตกคร่อมของโหลด  $100\Omega$  เพื่อที่จะนำมาทดแทนเป็นกระแส  $I_C$  ในการวัดประสิทธิภาพของตัวอุปกรณ์เทียบกับ  $V_C$

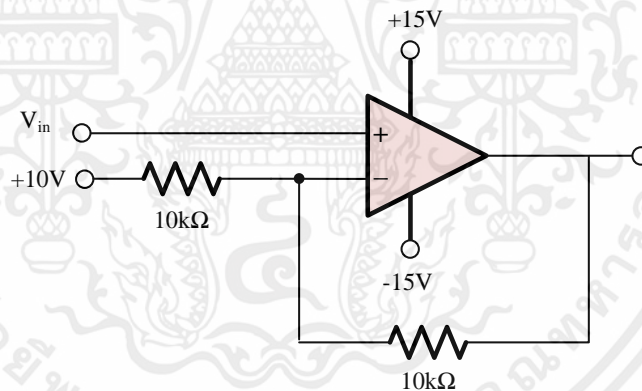
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 วงจร Instrument Amplifier

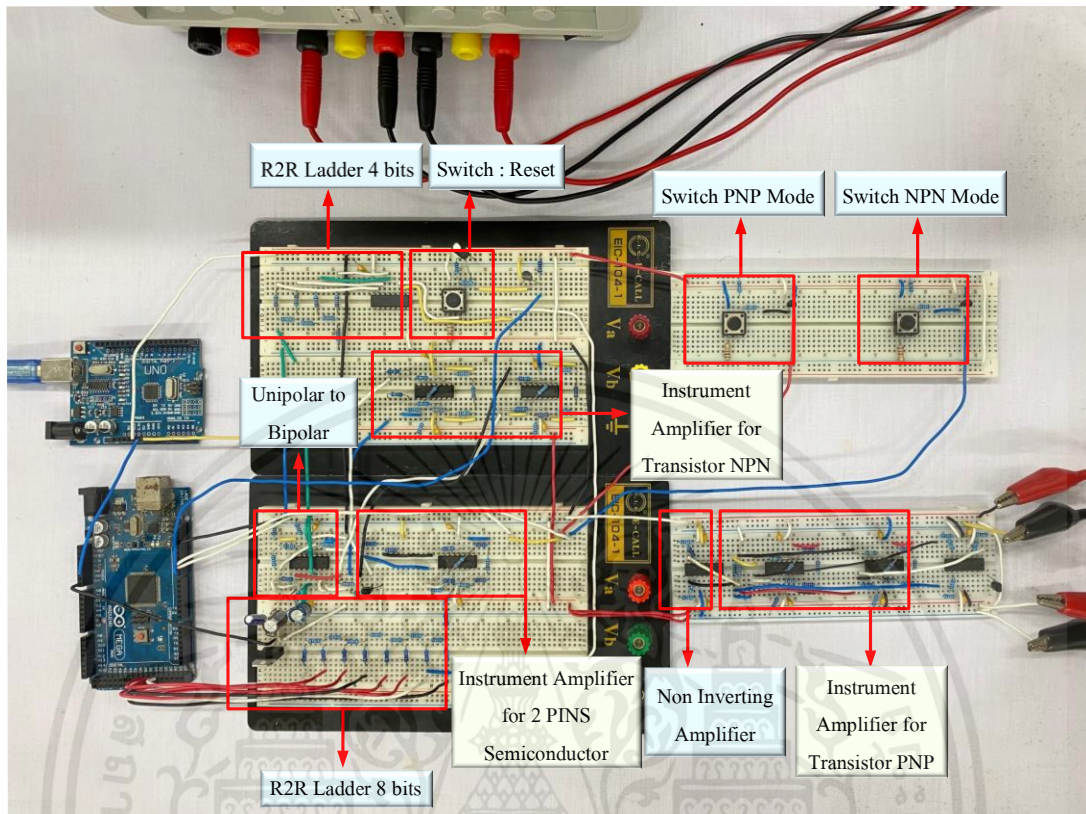
### 3.2.3 วงจรเปลี่ยนกราฟจาก Unipolar เป็น Bipolar

โดยการเพิ่มออปแอมป์เบอร์ LF347 อีกตัวหลังวงจร R2R Ladder 8 bits ทำการขยายสัญญาณโดยใช้วงจรขยายให้มี Gain เท่ากับ 2 เมื่อผ่านวงจรจะได้สัญญาณ Analog ที่มีแรงดัน 0V ถึง 10V จากนั้นทำการส่งสัญญาณเข้าวงจร Unipolar to Bipolar โดยใช้ไฟเลี้ยง +15V และ -15V จากนั้นทำการป้อน DC Offset +10V เข้าไปที่ขาลบของออปแอมป์



รูปที่ 3.6 วงจรขยายสัญญาณ Unipolar เป็น Bipolar

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

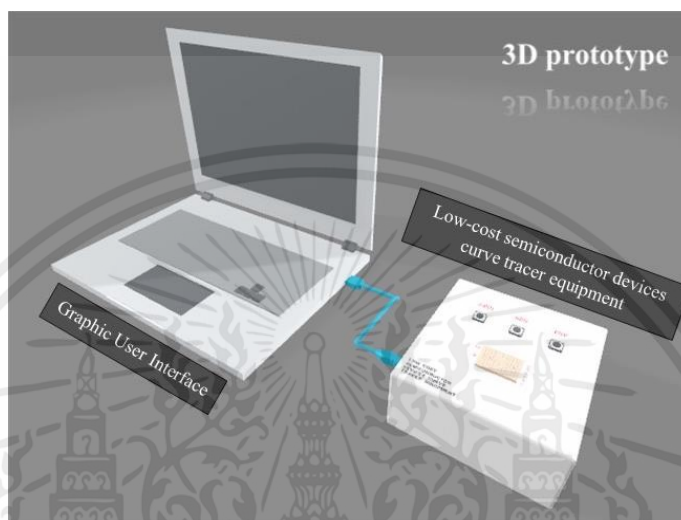


รูปที่ 3.7 วงจร curve tracer ที่ได้ทำการต่อจริง

จากรูปที่ 3.7 จะเห็นได้ว่า Arduino Mega2560 มีการเจนเนอร์เนตสัญญาณ 8 bits ให้กับวงจร R2R Ladder 8 bits ซึ่ง IC4040 ได้ต่อระหว่าง PIN 36 ของ Arduino Mega2560 และ PIN 10 (CP) ทำหน้าที่เป็น Input CLK ให้กับตัวไอซีและทำการต่อ PIN 5, 10, 12, 13 และ 14 เข้ากับวงจร R2R Ladder 4 bits เพื่อทำการเจนเนอร์เนตสัญญาณ 16 step ในส่วนของการต่อวงจรจริงได้ทำการเพิ่ม Arduino Uno เข้ามาเพื่อทำการลด Noise ในการเจนเนอร์เนตสัญญาณของ Arduino Mega2560 และเพิ่มสวิตช์ในการเปลี่ยนโหมดอุปกรณ์ semiconductor 3 ขา (PNP Mode และ NPN Mode)

### 3.3 การออกแบบเครื่องวัดคุณสมบัติกึ่งตัวนำโดยใช้ต้นทุนต่ำ (ชิ้นงาน)

การออกแบบเครื่องวัดคุณสมบัติกึ่งตัวนำโดยใช้ต้นทุนต่ำ ได้ใช้โปรแกรม Tinker Cad ซึ่งเป็นโปรแกรมที่ใช้สำหรับออกแบบโมเดล 3 มิติที่ทำงานบน Web Browser

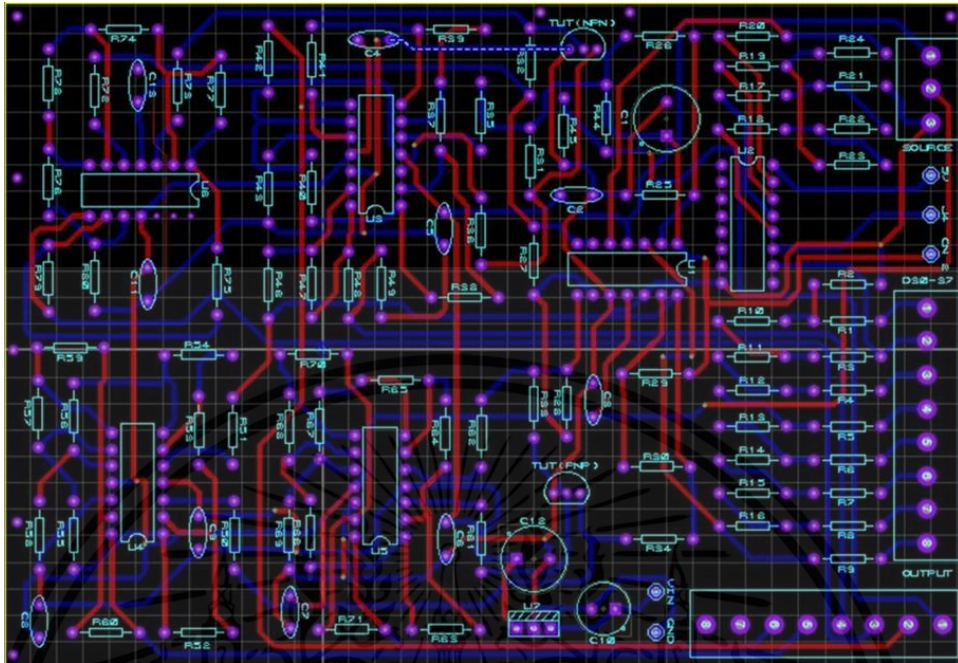


รูปที่ 3.8 Mockup 3D ของเครื่องวัดคุณสมบัติกึ่งตัวนำโดยใช้ต้นทุนต่ำ

จากรูปที่ 3.8 Mockup 3D แสดงให้เห็นถึงรายละเอียดของชิ้นงาน ซึ่งตัวกล่องมีขนาด 146 x 157 x 67 mm ประกอบไปด้วยสวิตช์ 3 ตัว สำหรับการ Reset และเลือกโหมดในการใช้งานและ Photo bread ใช้เป็นตำแหน่งการวัดอุปกรณ์ซึ่งชิ้นงานมี Port USB สำหรับการต่อเข้ากับ Graphic User Interface

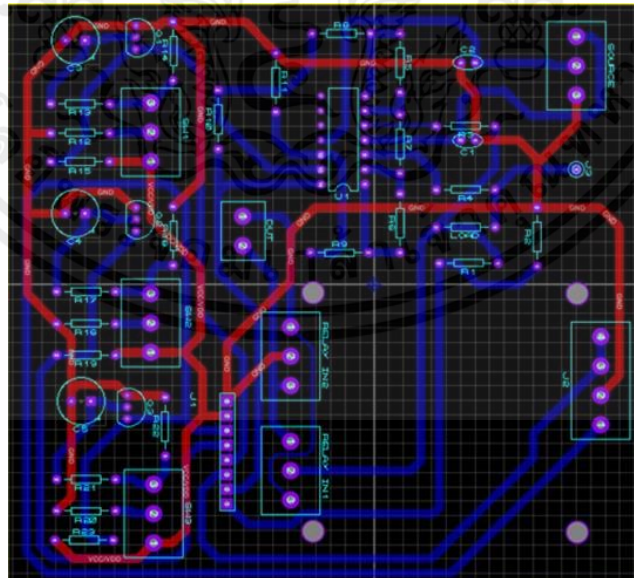
#### 3.3.1 ออกแบบลายปริ้นวงจร PBC

ในการออกแบบลายปริ้นวงจรจริงโดยใช้ PCB Design Mode (Proteus) งานออกแบบนั้นจะใช้แผ่นทองแดงทั้งสองด้านเนื่องจากความซับซ้อนของวงจรและมีการโยงสายการเชื่อมต่อของอุปกรณ์ในแต่ละส่วนที่ซับซ้อน



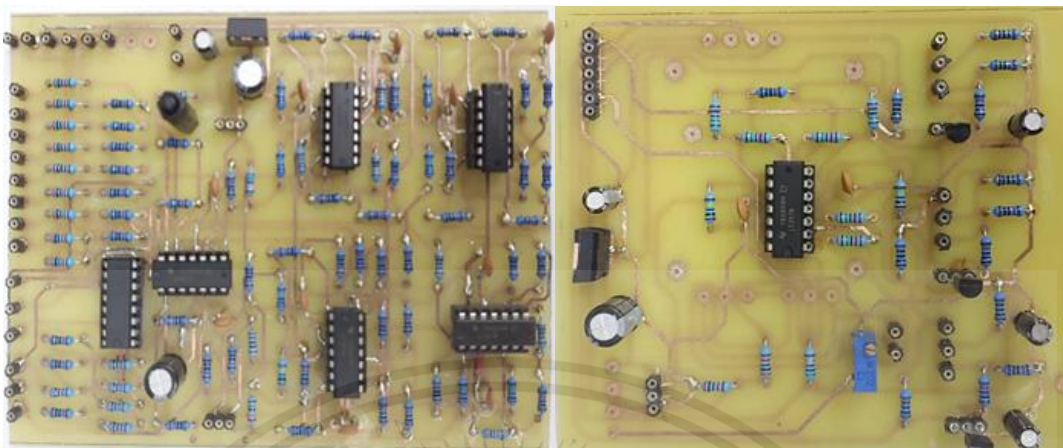
รูปที่ 3.9 ลายปิ้งวงจร Curve tracer ในส่วนอุปกรณ์ DUT 3 ขา และวงจรเจนเนอร์เรท  $V_C$  และ  $I_B$

จากรูปที่แสดงดังรูปที่ 3.9 และที่ 3.10 จะมีการเชื่อมต่อของอุปกรณ์ทั้งหมด 2 เลเยอร์โดยจะแบ่งออกเป็น Top (สีแดง) และ Bottom (สีฟ้า)



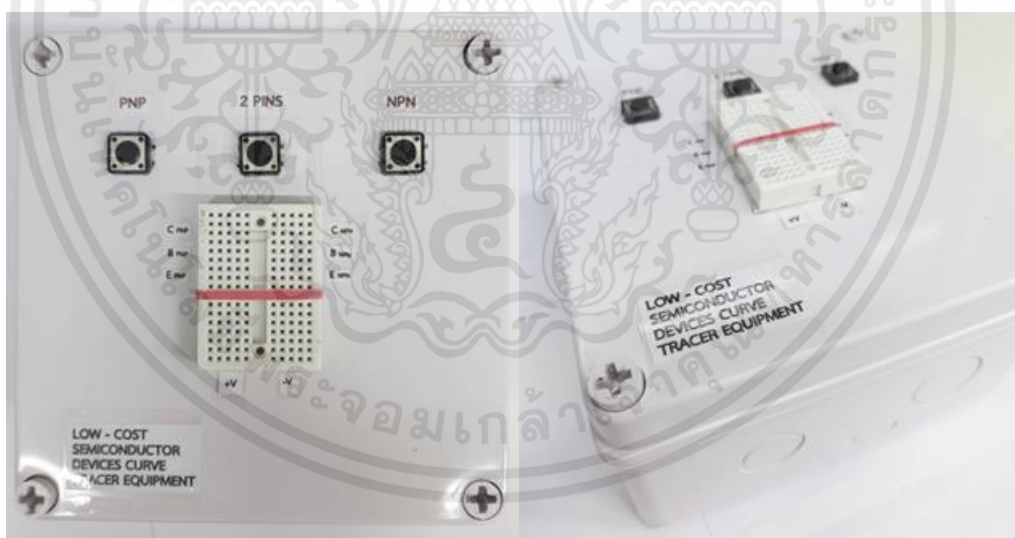
รูปที่ 3.10 ลายปิ้งวงจร Curve tracer ในส่วนอุปกรณ์ DUT 2 ขา และสวิตช์โหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 วงจร Curve tracer

จากรูปที่ 3.11 ชิ้นงานเครื่องวัดคุณสมบัติกึ่งตัวนำต้นทุนต่ำประกอบไปด้วยสวิทช์ โหมดการทำงาน 3 โหมด ได้แก่ โหมด NPN PNP และ 2 ขา (2 PIN) ในส่วนชุดโฟโต้บอร์ดสำหรับ วัดอุปกรณ์จะมีตำแหน่งในการวางขาของอุปกรณ์ 3 ขา เพื่อความสะดวกและลดความซับซ้อนต่อการใช้งาน

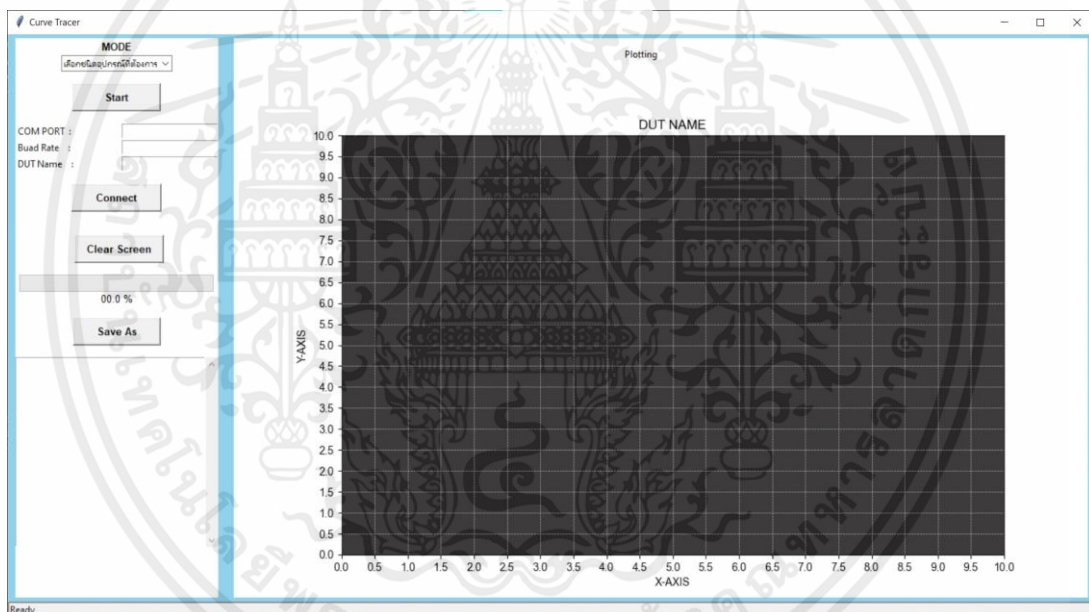


รูปที่ 3.12 ชิ้นงานเครื่องวัดคุณสมบัติกึ่งตัวนำต้นทุนต่ำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.2 ออกแบบหน้าต่างแสดงผล GUI (Graphic User Interface)

หน้าต่าง GUI ประกอบไปด้วยส่วนแสดงกราฟที่มีการรับค่าจาก Arduino Mega2560 และกำหนดชื่อกราฟโดยสามารถพิมพ์ชื่ออุปกรณ์บอกได้ในส่วน DUT Name และกำหนด Com Port และ Baud Rate ให้ตรงกับอุปกรณ์ที่ใช้งาน ในส่วนปุ่ม Clear (เป็นส่วนของการเคลียร์กราฟ) และปุ่ม Start เป็นส่วนการ Run การทำงานของโปรแกรม และเลือก Mode ของอุปกรณ์ที่จะทำการวัด Resistor, Diode, Zener Diode, NPN และ PNP และเมื่อทำการ Run การใช้งานอุปกรณ์ เพื่อแสดงผลเสร็จสามารถทำการบันทึกกราฟได้โดยการกดปุ่ม Save As ดังรูปที่ 3.14



รูปที่ 3.13 ภาพรวมหน้าต่าง GUI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.3 เขียนโปรแกรมควบคุมการเจนเนอร์เรทสัญญาณโดยใช้ Arduino IDE

```

1 #define InputPin_2leg 9
2 int j=1;
3 void setup() {
4   // put your setup code here, to run once:
5   pinMode(InputPin_2leg, INPUT);
6   pinMode(31, OUTPUT);
7   pinMode(32, OUTPUT);
8   pinMode(33, OUTPUT);
9   pinMode(34, OUTPUT);
10  pinMode(35, OUTPUT);
11  pinMode(36, OUTPUT);
12  pinMode(37, OUTPUT);
13 }
14
15
16 void loop() {
17   // put your main code here, to run repeatedly:
18   //Gen Counter
19   if(j!=1){
20
21     for ( int i = 0; i <= 255; i++) {
22
23       PORTC = i;
24       delay(1);
25
26       for ( int i = 255; i >= 0; i--) {
27
28         PORTC = i;
29         delay(1);
30         j++;
31       }
32     } else (digitalRead(InputPin_2leg)==LOW)
33     {j=0;
34     delay(3000);}
35 }

```

รูปที่ 3.13 โค้ดในส่วนที่ใช้ในการเจนเนอร์เรทสัญญาณ 8 bits

โดยให้ค่า  $i$  หรือค่าแรงดันในวงจรเปลี่ยนเป็น 8 bits ตั้งแต่ 0 – 256 โดยที่ออกมาที่ PIN OUT 31 – 37 ที่ต่อเข้ากับวงจร R2R Ladder

## 3.4 เครื่องมือที่ใช้ในการทดลอง

ในโครงงานนี้มีอุปกรณ์และเครื่องมือที่ใช้ในการทดลองดังนี้

### 3.4.1 Arduino Mega2560

ใช้สำหรับประมวลผลการทำงานของอุปกรณ์แต่ละตัวและทำการเจนเนอร์เรทสัญญาณพัลส์ (CLK)

### 3.4.2 ออปแอมป์เบอร์ LF347

ใช้ในการขยายสัญญาณหลังจากวงจร R2R Ladder และทำการเปลี่ยนสัญญาณจาก Unipolar เป็น Bipolar

### 3.4.3 IC4040

เป็นวงจรหารสัญญาณที่ถูกเจเนอเรตมาจาก Arduino Mega2560

### 3.4.4 ตัวต้านทาน 10KΩ และ 20KΩ

### 3.4.5 ตัวเก็บประจุ

### 3.4.6 Switch กดติดกดดับ

ใช้กดเมื่อต้องการ Reset และเลือกโหมดของวงจร Curve Tracer

## 3.5 การจัดเก็บผลทดลอง

### 3.5.1 การทดสอบแรงดันของวงจร R2R Ladder ร่วมกับ Arduino Mega2560

ทำการทดสอบการทำงานของวงจร R2R Ladder เมื่อทำการต่อเข้ากับ PIN OUT 31 – 37 และทำการเจเนอเรตสัญญาณ 8 Bits ไปยังวงจร เพื่อทำการแปลงสัญญาณดิจิทัลเป็น แอนะล็อก (Ramp Signal 256 Step) ในส่วนของ IC4040 ทำหน้าที่เป็น Counter ที่ถูกเจเนอเรต เรทไปยังวงจร R2R Ladder 4 Bits และเป็นไปตามทฤษฎีที่ได้ศึกษามา

### 3.5.2 ทดสอบวงจร Curve Tracer ร่วมกับออปแอมป์เบอร์ LF347 (วงจร Non-inverting )

เป็นการทดสอบสัญญาณที่ได้ถูกเจเนอเรตออกมาเมื่อแรงดันไหลผ่านออปแอมป์ที่ได้ทำการใส่ DC Offset +10V เข้าไปที่ขาลบของออปแอมป์ สัญญาณที่ได้คือสัญญาณ Ramp ที่เป็น Bipolar เพื่อให้สอดคล้องกับวงจรที่ได้ทำการออกแบบ

### 3.5.3 ทดสอบการ Reset ร่วมกับวงจร Curve Tracer

ทดสอบการทำงานของวงจรได้จากสัญญาณ Ramp ที่ถูกปล่อยออกมาและดูค่าจาก กราฟบนหน้าจอ Oscilloscope ให้มีความสอดคล้องกับทฤษฎีที่ได้ทำการศึกษามา และทดสอบ อุปกรณ์ให้มีความเหมาะสมกับการต่อวงจรที่ได้ออกแบบมา

### 3.5.4 การทดสอบวัตต์อุปกรณ์ DUT 2 และ 3 ขา และวิเคราะห์ผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการทดสอบอุปกรณ์ DUT 2 และ 3 ขาในการต่อวงจรลงจริงและวิเคราะห์ผล  
เทียบกับทฤษฎีที่ได้ทำการศึกษามา

### 3.5.5 การแสดงผลผ่านหน้าต่าง UI

เป็นการทดสอบอุปกรณ์และทำการแสดงผลการทดลองผ่านหน้าต่าง UI เพื่อแสดง  
คุณสมบัติทางไฟฟ้าของอุปกรณ์สารกึ่งตัวนำที่ทำการทดลอง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

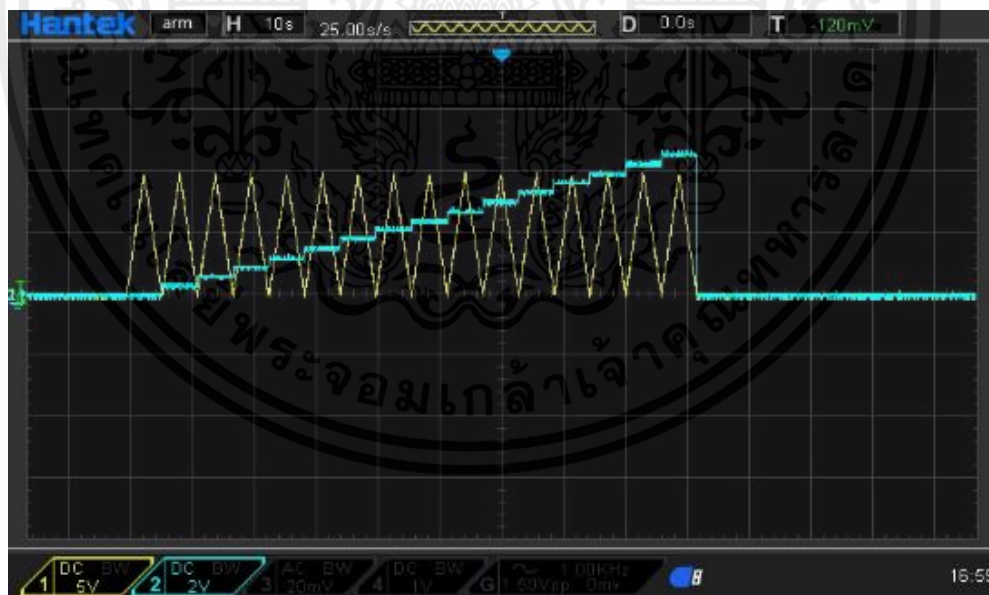
## บทที่ 4

### ผลการทดลอง

จากการทดลองการวัดสัญญาณจากอุปกรณ์ semiconductor โดยทำการแบ่งผลการทดลองเป็น 3 ส่วน ส่วนแรกจะเป็นการวัดจากอุปกรณ์แบบ 2 ขาคอน (Resistor, Diode, และ Zener Diode) ส่วนที่สองจะเป็นการวัดอุปกรณ์ Semiconductor 3 ขา (Transistor ชนิด NPN และ PNP) และส่วนสุดท้ายแสดงหน้าจอแสดงผล (UI) และภาพรวมชิ้นงานจริง

#### 4.1 การทดสอบแรงดันจากวงจร R2R Ladder

การทดสอบวงจรจริง โดยทำการเจนเนอเรตสัญญาณพัลส์ (CLK) ซึ่งต่อจาก PIN OUT 31 – 37 ของ Arduino Mega2560 โดยการป้อน Logic 0 และ +5V ไปยังวงจร R2R Ladder 4 Bits และ 8 Bits ดังรูปที่ 8 และ 9 เพื่อทำการแปลงสัญญาณให้เป็น Step หรือขั้นบันไดตั้งแต่ 0 – 16 (สีฟ้า) และ 0 – 256 (สีเหลือง) หลังจากนั้นทำการต่อ Buffer หรือออปแอมป์ เพื่อเป็นการขยายสัญญาณเท่ากับ  $R_i$  และ  $R_f$  มีค่าเท่ากับ  $1k\Omega$



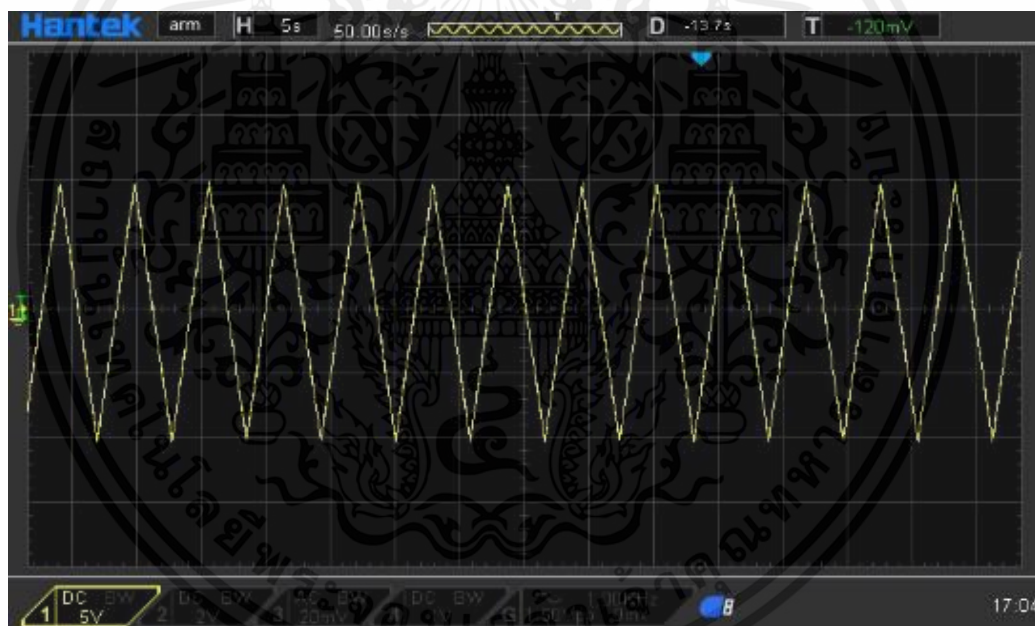
รูปที่ 4.1 แสดงผลของสัญญาณ Output

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 12 เป็นการแสดงผลของสัญญาณ Output จากการเจเนอเรตสัญญาณ 8 bit ของ Arduino Mega2560 ซึ่งจะสังเกตเห็นว่าสัญญาณพัลส์เปลี่ยนเป็นสัญญาณ Ramp เป็น 16 step (สี่ฟ้า) และ 256 step (สี่เหลือง) ที่มีขนาดสัญญาณ 10V และจะเห็นได้ว่าสัญญาณทั้งสองมีความสอดคล้องกัน (Synchronized) เป็นอย่างดี

#### 4.2 ผลการทดสอบวงจร Non-inverting (Unipolar to Bipolar)

จากสัญญาณ 8 bit ที่แสดงดังรูปที่ 4.1 เมื่อทำการต่อวงจร Non Inverting Amplifier จากนั้นทำการต่อตัวออปแอมป์เบอร์ LM347 หลังวงจร R2R Ladder 8 bit เพื่อให้กราฟเปลี่ยนจาก Unipolar เป็น Bipolar โดยป้อน DC Offset +10V ที่ขาลบของออปแอมป์ซึ่งแสดงดังรูปที่ 4.2

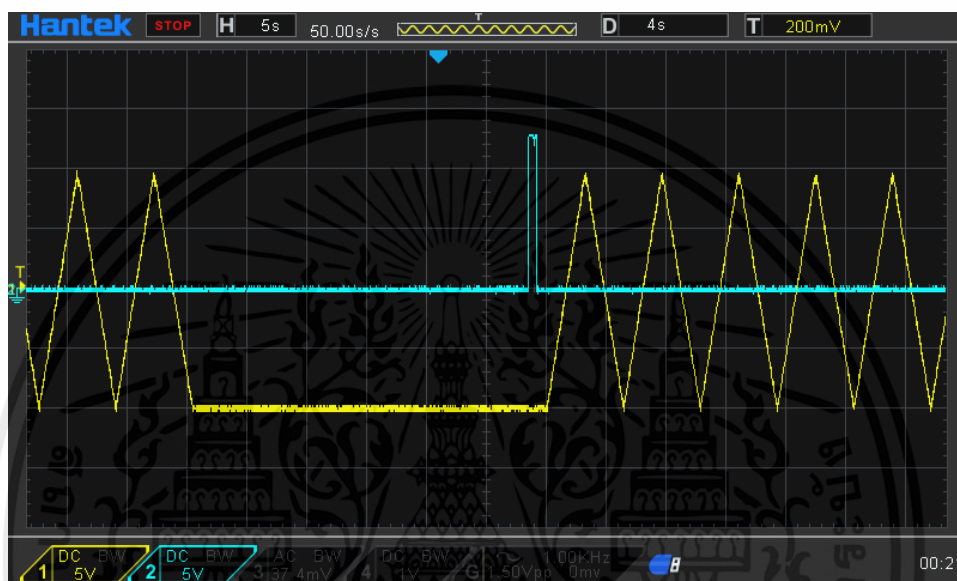


รูปที่ 4.2 สัญญาณ Ramp 256 step (Bipolar)

จากรูปที่ 4.2 เป็นการแสดงผลของสัญญาณ Output จากการเจเนอเรตสัญญาณ 8 bit ของ Arduino Mega2560 ซึ่งจะสังเกตเห็นว่าสัญญาณพัลส์ได้เปลี่ยนเป็นสัญญาณ Ramp เป็น 256 step ที่มีแรงดัน +10V และ -10V คือกราฟที่เป็น Bipolar

### 4.3 ทดสอบการ Reset วงจรด้วย Pull up switch

เมื่อกดสวิตช์ Reset จะสังเกตได้จากกราฟที่แสดงดังรูปที่ 14 ซึ่งเกิดกระแส (High) เข้าไปที่ IC4040 ให้เกิดการ Reset และกระแส (Low) ไปที่ Arduino Mega2560 ซึ่งจะเป็นการเจนนอร์เรทสัญญาณใหม่อีกครั้ง เพื่อให้เริ่มการวัดอุปกรณ์ใหม่โดยไม่เกิดความผิดพลาด



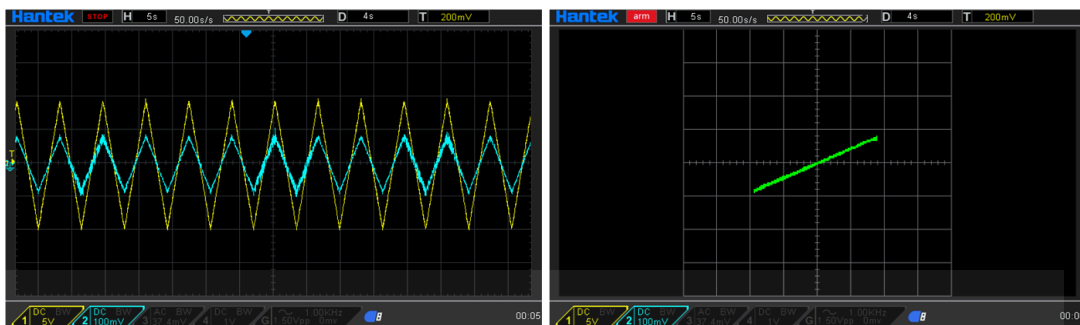
รูปที่ 4.3 ผลทดสอบการ Reset วงจรด้วย Pull up switch

จะสังเกตได้ว่าเมื่อทำการกดสวิตช์ไปแล้วจะเกิดสัญญาณพัลส์เกิดขึ้น จะแสดงให้เห็นว่า วงจร curve tracer ได้ทำการ Reset แล้ว

### 4.4 ทดสอบวัดอุปกรณ์ DUT (Semiconductor 2 ขา)

#### 4.4.1 ทดสอบวัดอุปกรณ์ Resistor (ตัวต้านทาน)

เมื่อทำการทดสอบการวัดอุปกรณ์ตัวต้านทาน สังเกตได้ว่าเมื่อแรงดันเกิดการลดลงจากสัญญาณ  $V_C$  ซึ่งเป็นไปตามคุณสมบัติของตัวต้านทานที่ใช้ในการต้านการไหลของกระแสไฟฟ้า ซึ่งในผลการทดลองได้ใช้ตัวต้านทานขนาด  $1k\Omega$  มีแรงดัน ที่  $10V$  และกระแส  $I_C$  ที่  $-1mA$  ถึง  $1mA$



(ก)

(ข)

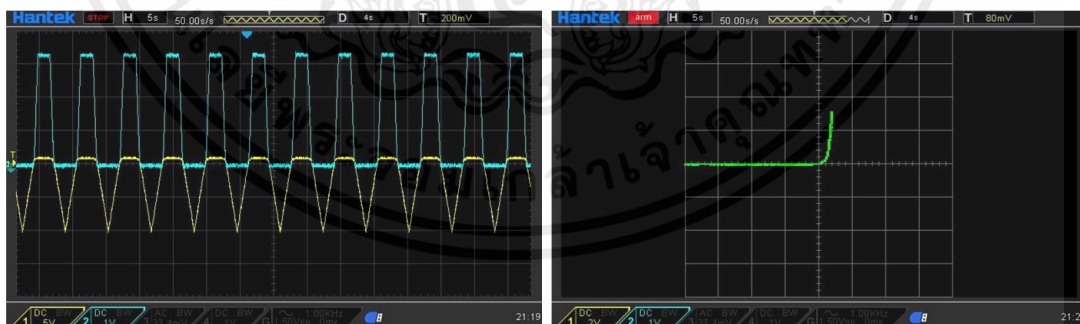
รูปที่ 4.4 ผลทดสอบวัดอุปกรณ์ Resistor 1k $\Omega$

(ก) สัญญาณแรงดัน และ กระแสที่วัดได้จากวงจร Curve Tracer

(ข) กราฟใน Mode XY

#### 4.4.2 ทดสอบวัดอุปกรณ์ Diode

ในหัวข้อนี้จะแสดงการวัดอุปกรณ์ไดโอดดังรูปที่ 16 (ก) เห็นได้ว่าการจ่ายกระแส  $I_C$  ผ่านไดโอด ไดโอดนำกระแสเปรียบเสมือนสวิตช์อยู่ในสถานะต่อวงจร (on) และแรงดันตกคร่อมไดโอดเท่ากับ 0 ในส่วนการไบอัสกลับ ไดโอดจะไม่นำกระแสเปรียบเสมือนสวิตช์อยู่ในสถานะตัดวงจร (off) เมื่อมีแรงดันตกคร่อมไดโอดในระหว่าง 0.6V – 0.7V จากนั้นจะเริ่มมีกระแสไหลผ่านไดโอดมากขึ้น จนกระทั่งมีแรงดันตกคร่อมไดโอดที่มีค่าน้อยกว่า 0.6V ไดโอดจะไม่ยอมให้กระแสไหลผ่าน จากนั้นปรับกราฟให้อยู่ใน Mode XY ดังรูปที่ 16 (ข) ซึ่งพบว่ากราฟเป็นไปตามทฤษฎี



(ก)

(ข)

รูปที่ 4.5 ผลทดสอบวัดอุปกรณ์ Diode 1N4148

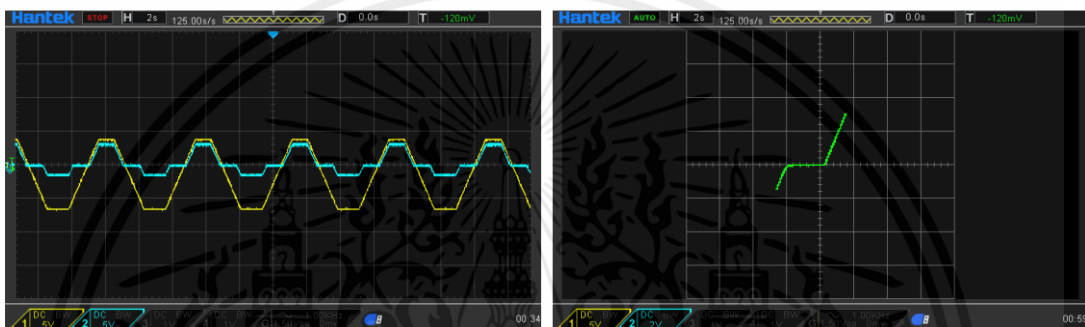
(ก) สัญญาณแรงดันและกระแสที่วัดได้จากวงจร Curve Tracer

(ข) กราฟใน Mode XY

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.4.3 ทดสอบวัตุปกรณ์ Zener Diode

จากการทดสอบวัตุปกรณ์ซีเนอร์ไดโอดดังรูปที่ 4.6 (ก) ซึ่งแสดงให้เห็นถึงการทำงานคล้ายกับไดโอดธรรมดาซึ่งนำกระแสเมื่อมีแรงดันตกคร่อมตัวซีเนอร์ไดโอดเกินกว่า  $0.6\text{V}$  เมื่อจ่ายแรงดันในตัวซีเนอร์ได้มากขึ้นจะนำกระแสเพิ่มขึ้น แต่จะต่างกันเมื่อมีการไบอัสกลับจะต้องจ่ายแรงดันไปถึงแรงดันพังทลายที่  $V_Z$  เท่ากับ  $5.1\text{V}$  (Zener Breakdown) จึงเกิดการนำกระแสและเกิดแรงดันตกคร่อมซีเนอร์คงที่ตามค่า Zener Breakdown ดังรูปที่ 4.6 (ข) ซึ่งขึ้นอยู่กับขบวนการผลิต



(ก)

(ข)

รูปที่ 4.6 ผลทดสอบวัตุปกรณ์ Zener Diode 1N4733 ( $5.1\text{V}$ )

(ก) สัญญาณแรงดันและกระแสจากวงจร Curve Tracer

(ข) กราฟใน Mode XY

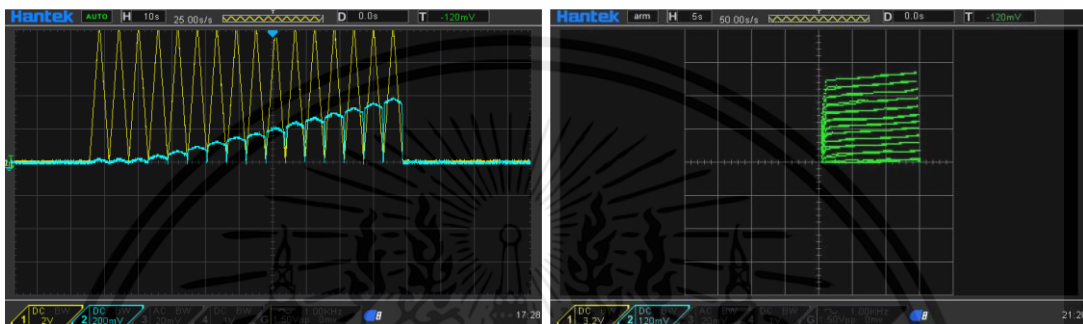
#### 4.5 ทดสอบวัตุปกรณ์ (DUT) Semiconductor 3 ขา

ทดสอบการวัตุปกรณ์ (DUT) Semiconductor 3 ขาโดยต่อ Sensor Resistor ขนาด  $100\Omega$  (เพื่อวัดกระแสคอลเลคเตอร์) และ  $150\text{k}\Omega$  (เพื่อวัดกระแสไบอัส) ก่อนวัตุปกรณ์ Semiconductor เพื่อทำการวัดกระแส  $I_C$  ที่จะเข้าไปในตัวของอุปกรณ์ Semiconductor โดยการวัดกระแส  $I_C$  จะทำการต่อวงจร Inverting Amplifier ตกคร่อมตัว load resistor และวัดแรงดัน  $V_{CE}$  จากนั้นทำการปรับเพิ่มกระแสไบอัสและทำการวัดกระแส  $I_C$  และ  $V_{CE}$  อีกจนจบกระบวนการประมาณ 16 รอบ และตรวจสอบผลในแกน XY

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.5.1 ทดสอบวัดอุปกรณ์ Transistor แบบ NPN

จากรูปที่ 4.7 สังเกตได้ว่ามีกระแส  $I_C$  (สีฟ้า) จะเพิ่มขึ้นในแต่ละ step เทียบกับแรงดัน  $V_{CE}$  (สีเหลือง) สังเกตได้ว่า  $V_{CE}$  จะเข้าสู่สภาวะอิ่มตัว  $V_{CE(sat)}$  เมื่อแรงดันมีค่าประมาณ  $0.7V$  และเพิ่มค่าสูงขึ้นเล็กน้อย (ค่า Depletion region กว้างขึ้น) เมื่อมีกระแสไบอัสสูงขึ้น จึงทำให้กราฟใน Mode XY มีกระแส  $I_B$  เพิ่มขึ้น กระแส  $I_C$  จะเพิ่มขึ้นตามสอดคล้องกับทฤษฎี



(ก)

(ข)

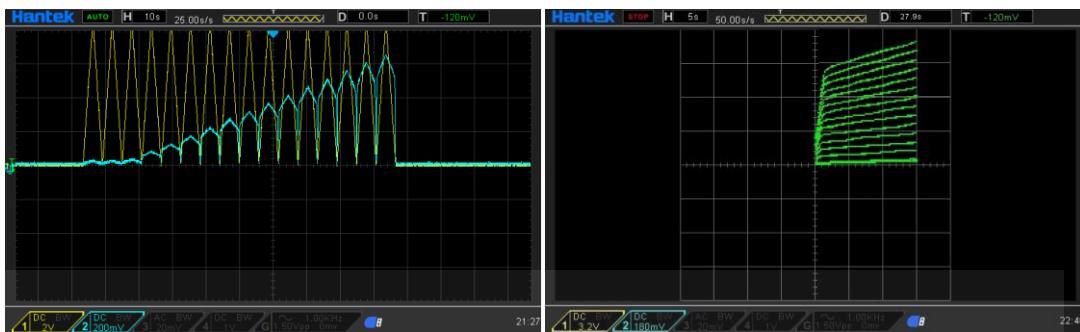
รูปที่ 4.7 ผลทดสอบวัดอุปกรณ์ Transistor NPN 2N3904 [12]

(ก) สัญญาณแรงดัน  $V_{CE}$  และ กระแส  $I_C$  จากวงจร Curve Tracer

(ข) กราฟใน Mode XY

#### 4.5.2 ทดสอบวัดอุปกรณ์ Transistor แบบ PNP

จากรูปที่ 4.8 สังเกตได้ว่ามีกระแส  $I_C$  (สีฟ้า) จะเพิ่มขึ้นในแต่ละ step เทียบกับแรงดัน  $V_{CE}$  (สีเหลือง) สังเกตได้ว่า  $V_{CE}$  จะเข้าสู่สภาวะอิ่มตัว  $V_{CE(sat)}$  เมื่อแรงดันมีค่าประมาณ  $0.7V$  และเพิ่มค่าสูงขึ้นเล็กน้อย (ค่า Depletion region กว้างขึ้น) เมื่อมีกระแสไบอัสสูงขึ้น จึงทำให้กราฟใน Mode XY มีกระแส  $I_B$  เพิ่มขึ้น กระแส  $I_C$  จะเพิ่มขึ้นตามสอดคล้องกับทฤษฎี จากรูปที่ 4.7 (NPN) และ 4.8 (PNP) มีลักษณะคล้ายกัน แต่ PNP จะมีค่า  $V_A$  (Early voltage) ต่ำกว่า จากความชันของกราฟ XY



(ก)

(ข)

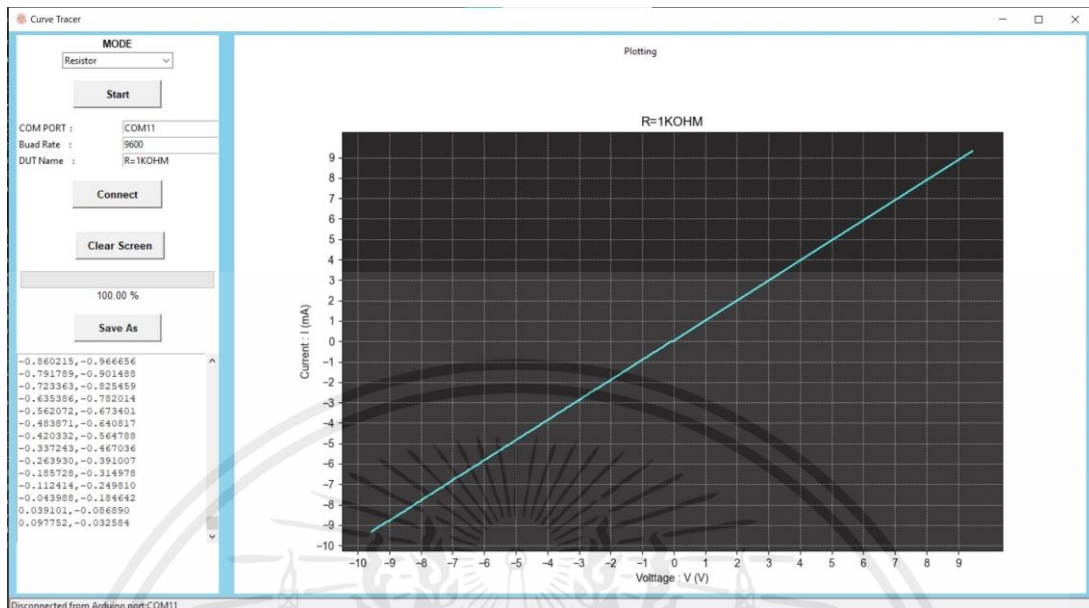
รูปที่ 4.8 ผลทดสอบวัตอุปกรณ์ Transistor NPN 2N3904 [12]  
 (ก) สัญญาณแรงดัน  $V_{CE}$  และ กระแส  $I_C$  จากวงจร Curve Tracer  
 (ข) กราฟใน Mode XY

#### 4.6 ทดสอบวัตอุปกรณ์ (DUT) แสดงผลบนหน้าจอ UI

ในการแสดงผลผ่านหน้าต่าง UI เมื่อทำการเลือก Mode และชนิดอุปกรณ์ semiconductor ที่ต้องการ สามารถกดเลือก Function start เพื่อ Plot กราฟและสามารถลบกราฟโดยการเลือก Function clear แสดงดังรูปที่ 20, 21, 22 และ 23 ซึ่งแสดงถึงการ Plot กราฟของ ตัวต้านทาน, Diode 1N4148, Zener Diode 1N4733, Transistor NPN และ Transistor PNP ตามลำดับ

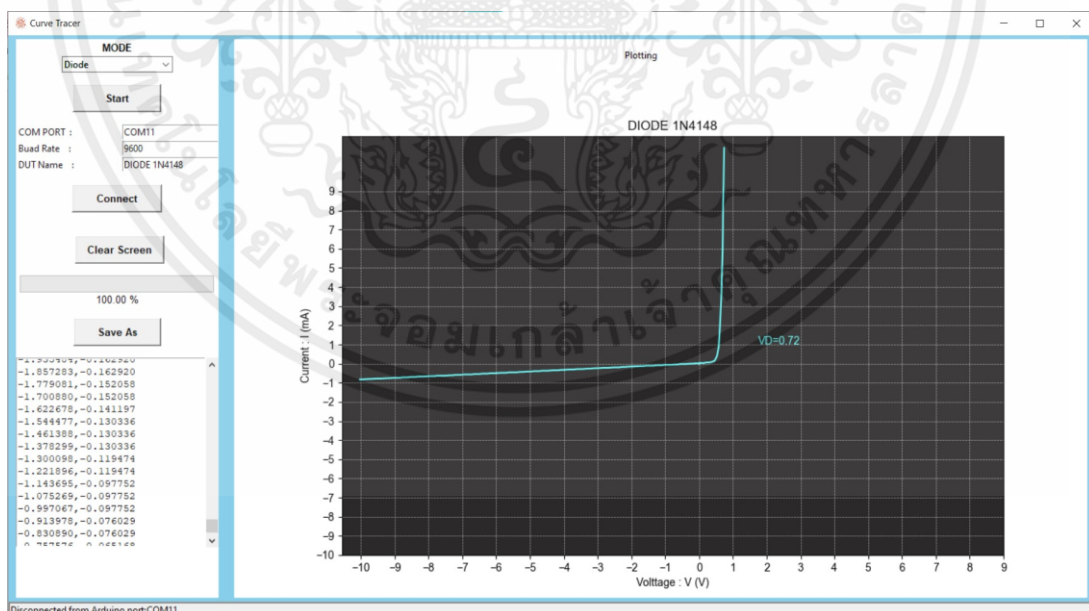
จากกราฟแสดงผลดังรูปที่ 4.9 โดยพิจารณาจากสมการกฎของโอห์ม ซึ่งเห็นได้ว่ากราฟความสัมพันธ์ระหว่างแรงดัน  $V_C$  และกระแส  $I_C$  จะได้ค่าของความต้านทาน  $R_{(DUT)}$  สามารถคำนวณได้จากกราฟความชัน (Slope) ผกผันซึ่งได้จากค่าแรงดัน  $V_C$  ทหารด้วยระยะกระแส  $I_C$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



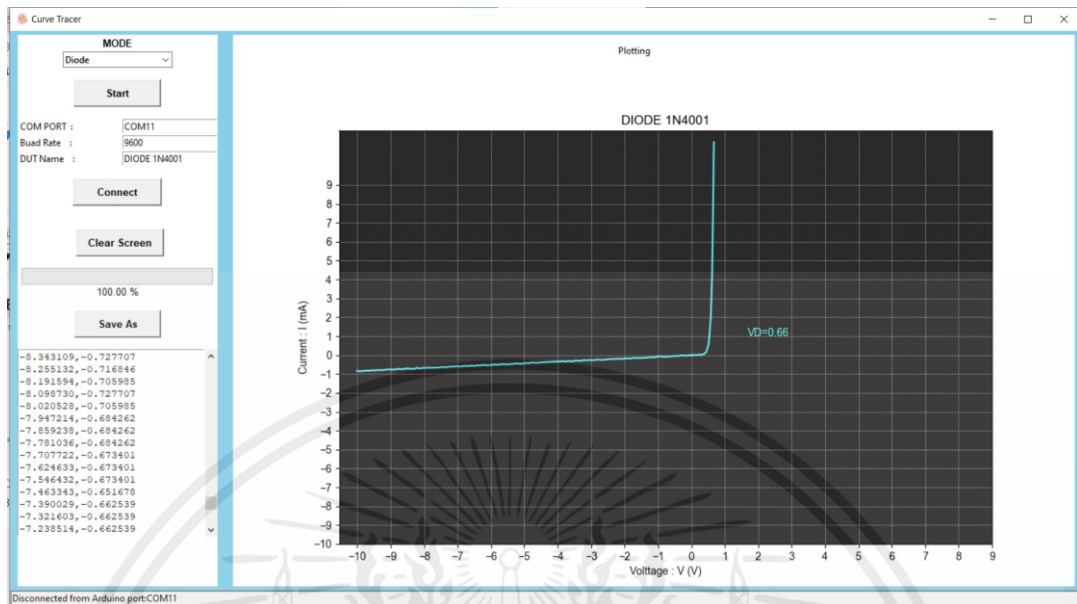
รูปที่ 4.9 กราฟแสดงผลผ่านหน้าจอ UI ของตัวต้านทาน

ในส่วนการวัดคุณสมบัติของไดโอด 1N4148 แสดงดังรูปที่ 4.10 และ 4.11 สังเกตได้ว่าการทำงานการไบอัสตรงที่แรงดัน  $V_C$  ประมาณ 0.6 - 0.7V ส่วนด้านไบอัสกลับ จะไม่มีการนำกระแส ซึ่งผลการทดลองจะสอดคล้องกับรูปที่ 4.5 และ มีความแม่นยำที่ยอมรับได้



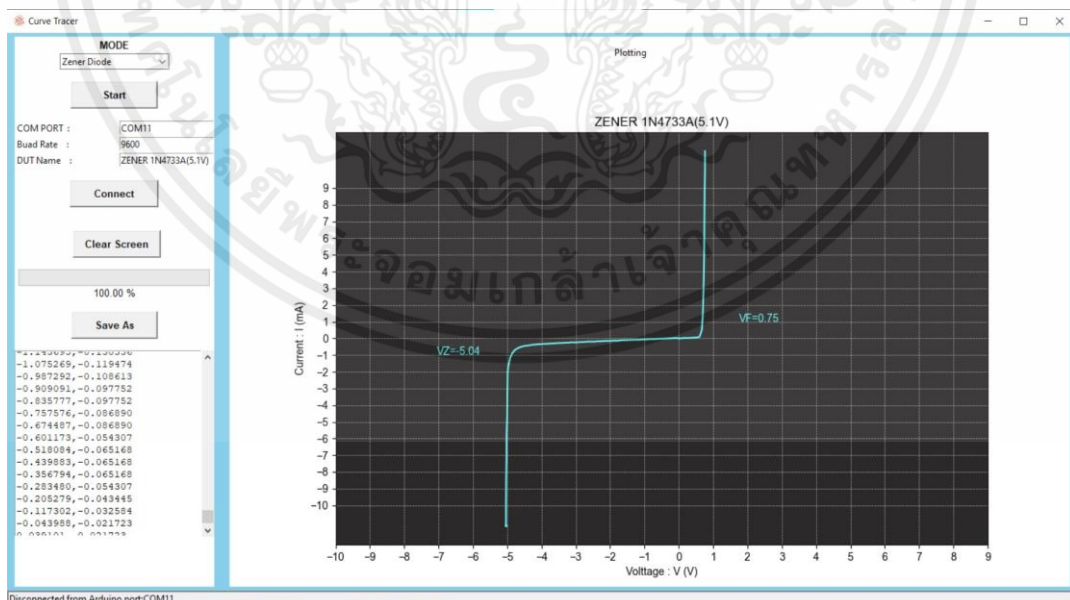
รูปที่ 4.10 กราฟแสดงผลผ่านหน้าจอ UI ของ Diode 1N4148

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



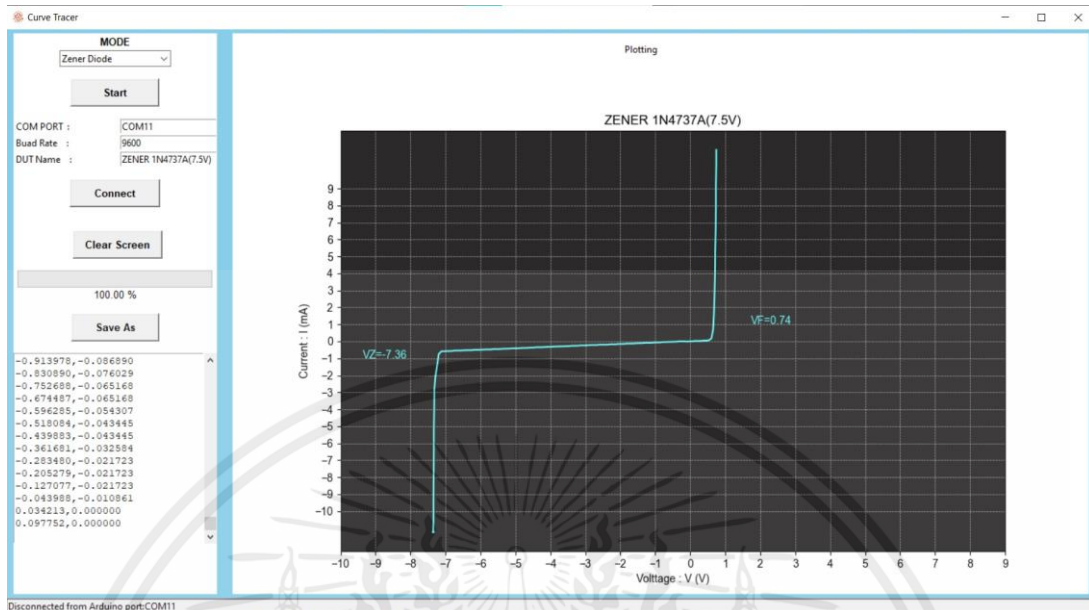
รูปที่ 4.11 กราฟแสดงผลผ่านหน้าจอ UI ของ Diode 1N4001

ในส่วนการวัดคุณสมบัติของ Zener Diode 1N4733 แสดงดังรูปที่ 4.12 และ 4.13 พบว่ามีการทำงานการไบอัสตรงที่แรงดัน  $V_C$  ประมาณ 0.6 – 0.7V ส่วนด้านไบอัสกลับจะมีกระแสไหลเมื่อ  $V_Z$  เท่ากับ 5.1V ตามค่าแรงดันพังทลายที่ได้กำหนดมาจากผู้ผลิต ซึ่งผลการทดลองจะสอดคล้องกับรูปที่ 4.6 และมีความแม่นยำที่ยอมรับได้

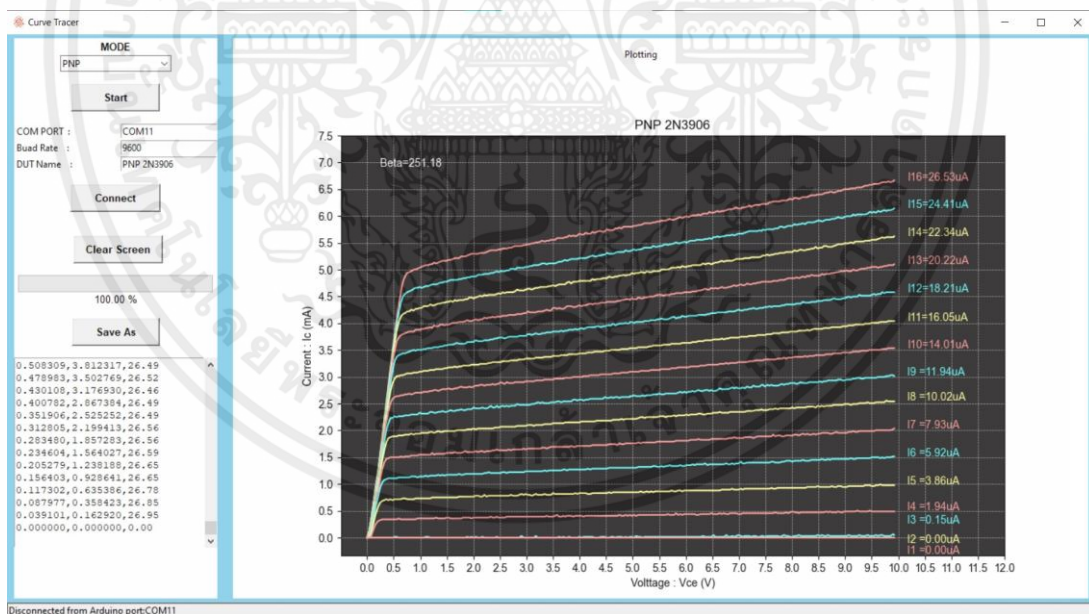


รูปที่ 4.12 กราฟแสดงผลผ่านหน้าจอ UI ของ Zener Diode 1N4733 (5.1V)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

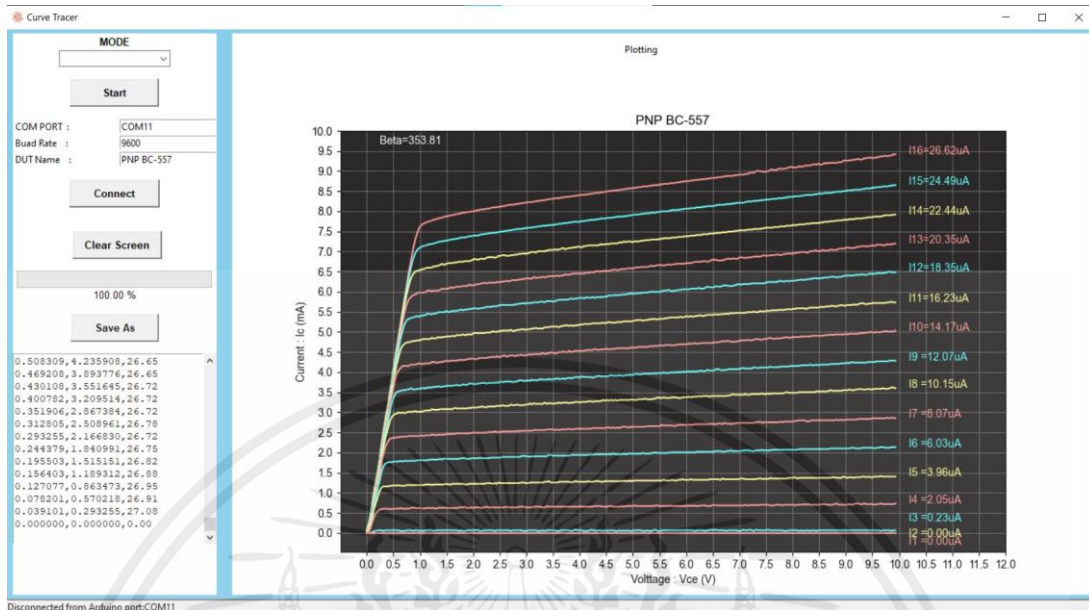


รูปที่ 4.13 กราฟแสดงผลผ่านหน้าจอ UI ของ Zener Diode 1N4737 (7.5V)



รูปที่ 4.14 กราฟแสดงผลผ่านหน้าจอ UI ของ Transistor PNP 2N3906

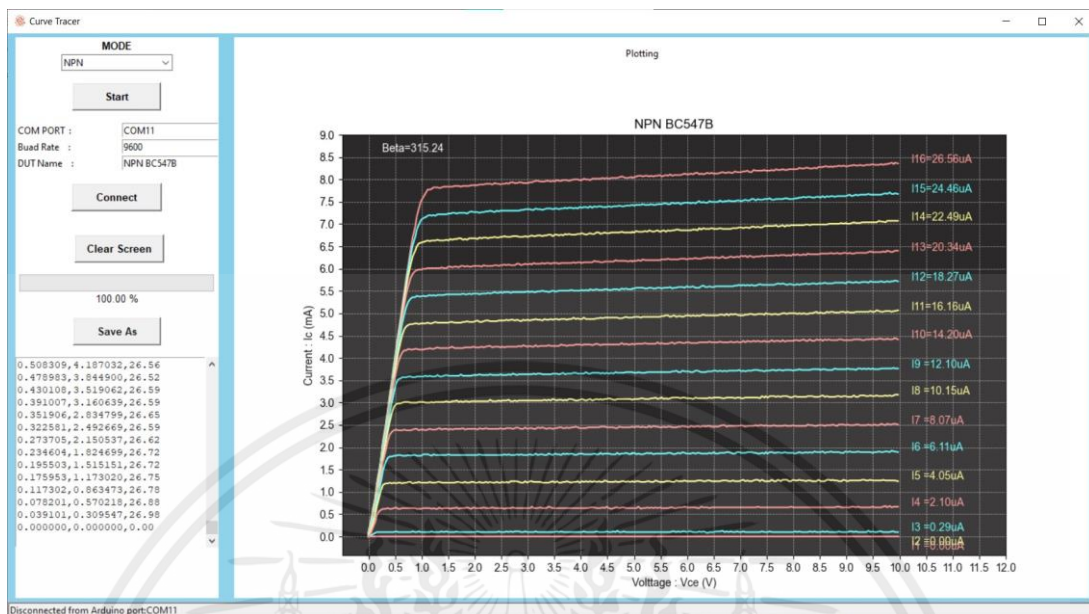
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



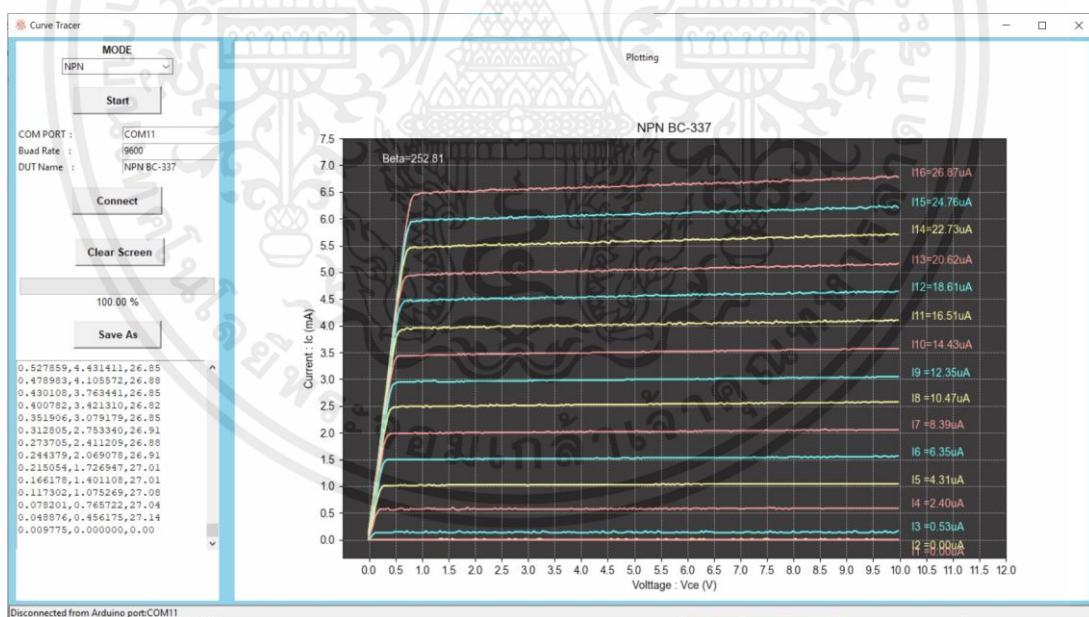
รูปที่ 4.15 กราฟแสดงผลผ่านหน้าจอ UI ของ Transistor PNP BC557

จากรูปที่ 4.14 , 4.15 , 4.16 และ 4.17 เป็นการแสดงกราฟคุณสมบัติของ Transistor PNP (2N3906) , PNP BC557 และ NPN BC548B , NPN BC337 ที่มีการวัดผลจากโปรแกรมซึ่งจะเห็นได้ว่ามีการสร้างสัญญาณ  $V_{CE}$  เริ่มต้นที่จุด (0-10V) และมีการไล่ Step กระแส  $I_B$  เป็นไปตามผลการทดลองดังรูปที่ 4.7 และ 4.8 ซึ่ง NPN (BC548B) จะมีค่าเบต้า (Beta) เท่ากับ 251.18 ซึ่งอยู่ในช่วง 100 – 300 ดังแสดงดังรูปที่ 4. ส่วนค่าเบต้าของ PNP (2N3906) ดังแสดงดังรูปที่ 4.12 มีค่าเท่ากับ 315.14 ตามทฤษฎีในเอกสารข้อมูลของ Transistor NPN BC548B และ Transistor PNP 2N3906

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.16 กราฟแสดงผลผ่านหน้าจอ UI ของ Transistor NPN BC547B



รูปที่ 4.17 กราฟแสดงผลผ่านหน้าจอ UI ของ Transistor NPN BC337

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### สรุปผลและข้อเสนอแนะ

#### 5.1 สรุปผล

งานวิจัยนี้ได้นำเสนอการออกแบบและสร้างเครื่องวัดคุณสมบัติอุปกรณ์สารกึ่งตัวนำต้นทุนต่ำ ซึ่งสามารถนำมาใช้ในการวัดคุณสมบัติของอุปกรณ์อิเล็กทรอนิกส์ต่างๆ ได้แก่ ตัวต้านทาน, ไดโอด, ซีเนอร์ไดโอด, ทรานซิสเตอร์ NPN และ PNP ซึ่งมีองค์ประกอบหลักคือ การสร้างสัญญาณเพื่อปรับค่าแรงดันและวัดค่าของกระแสและสร้างกราฟความสัมพันธ์ระหว่างกระแสและแรงดันเพื่อบ่งบอกคุณสมบัติของอุปกรณ์ที่สนใจ (Device Under Test: DUT) จากนั้นนำมาสร้างวงจรจริงซึ่งได้ผลสรุปว่าเป็นไปตามหลักการ โดยอุปกรณ์หลักที่ใช้ในการสร้างและเก็บค่าความสัมพันธ์ระหว่างกระแสกับแรงดัน คือ Arduino Mega 2560 จากนั้นจะส่งค่าไปส่วนการแสดงผลด้วย GUI (Graphic User Interface) และเมื่อเปรียบเทียบแล้วมีความสอดคล้องกันตามหลักการ และทฤษฎีของคุณสมบัติของตัวอุปกรณ์ที่สนใจและมีราคาต่ำกว่าในท้องตลาด

#### 5.2 ข้อเสนอแนะ

เนื่องจากวงจร Curve Tracer นี้ในการศึกษาของภาคเรียนนี้เป็นการวัดทดสอบอุปกรณ์ Semiconductor 2 และ 3 ขา ซึ่งสามารถพัฒนาต่อให้สามารถวัดทดสอบอุปกรณ์ที่หลากหลาย เช่น Transistor แบบ JFET และ MOSFET เป็นต้น

## บรรณานุกรม

- [1] สมยศ เพ็ญศรีศิริกุล. “บทเรียนคอมพิวเตอร์ช่วยสอนวิชาดิจิตอลประยุกต์”  
[http://pws.npru.ac.th/thawatchait/data/files/Chapter%207\\_ADC%20and%20DAC%20circuits%2001.pdf](http://pws.npru.ac.th/thawatchait/data/files/Chapter%207_ADC%20and%20DAC%20circuits%2001.pdf)
- [2] ElectronicTutorial. “R-2DAC”  
<https://www.electronics-tutorials.ws/combination/r-2r-dac.html>.
- [3] Data sheet: R/2R Series Thick Film Networks. Available Online. BOURNS  
<https://www.bourns.com/pdfs/r2r.pdf>
- [4] SUPPORT THAIEASYELEC. “บทความ Arduino คืออะไร ตอนที่1 แนะนำเพื่อนใหม่ที่ชื่อ Arduino”  
<https://blog.thaieasyelec.com/what-is-arduino-ch1/>
- [5] อาจารย์กรัณวิณัฐ วงษ์ไชยมูล. “สื่อการสอนออนไลน์วิชาเทคนิคการอินเทอร์เฟส”  
<https://sites.google.com/site/karanwinatktech/unit1>
- [6] theeraphong. “Arduino คืออะไร”  
<https://www.tommology.com/2018/08/13/arduino-%e0%b8%84%e0%b8%b7%e0%b8%ad%e0%b8%ad%e0%b8%b0%e0%b9%84%e0%b8%a3/>
- [7] pongkung. “บทที่ 1 เริ่มต้นใช้งาน Proteus”  
<http://know2learning.blogspot.com/2014/06/proteus.html>
- [8] SWAGATAM. “IC 4040 Datasheet, Pinout, Application”  
<https://www.homemade-circuits.com/ic-4040-datasheet-pinout-application/>

- [9] Electronics Tutorials. “4040 Ripple Counter Timing Diagram”  
<https://www.electronics-tutorials.ws/counter/simple-led-flasher.html>
- [10] Texas Instruments. “LF147/LF347 Wide Bandwidth Quad JFET Input Operational Amplifiers datasheet”  
[https://www.ti.com/lit/ds/symlink/lf147.pdf?ts=1664874628661&ref\\_url=https%253A%252F%252Fwww.google.com%252F](https://www.ti.com/lit/ds/symlink/lf147.pdf?ts=1664874628661&ref_url=https%253A%252F%252Fwww.google.com%252F)
- [11] fairchildsemi. “LF347 Datasheet (PDF) - Fairchild Semiconductor”  
<https://pdf1.alldatasheet.com/datasheet-pdf/view/53567/FAIRCHILD/LF347.html>
- [12] FreeWHA. “ซีเนอร์ไดโอด”  
<http://mixsound.orgfree.com/Electronics%20Device/LAB/LAB3.html>
- [13] NAPAT WATJANATEPIN. “ซีเนอร์ไดโอด”  
<https://www.gtech.ac.th/vdo/ELECTRICdoc/%E0%B8%A7%E0%B8%B4%E0%B8%8A%E0%B8%B2%E0%B8%8A%E0%B9%88%E0%B8%B2%E0%B8%87/E-BOOK%20BASIC%20ELECTRIC%20AND%20ELECTRONICS/%E0%B8%AD%E0%B8%B8%E0%B8%9B%E0%B8%81%E0%B8%A3%E0%B8%93%E0%B9%8C%E0%B8%AD%E0%B8%B4%E0%B9%80%E0%B8%A5%E0%B9%87%E0%B8%81%E0%B8%97%E0%B8%A3%E0%B8%AD%E0%B8%99%E0%B8%B4%E0%B8%81%E0%B8%AA%E0%B9%8C/%E0%B8%9A%E0%B8%97%E0%B8%97%E0%B8%B5%E0%B9%88%203%20%E0%B8%8B%E0%B8%B5%E0%B9%80%E0%B8%99%E0%B8%AD%E0%B8%A3%E0%B9%8C%E0%B9%84%E0%B8%94%E0%B9%82%E0%B8%AD%E0%B8%94.pdf>
- [14] ชิตชัย โพธิ์ประภา. “ไดโอด (Diode)”  
<http://phchitchai.wbvschool.net/archives/1312>
- [15] Electrical4U. “Diode: Definition, Symbol, and Types of Diodes”

<https://www.electrical4u.com/diode-working-principle-and-types-of-diode/>

[16] BYJU'S, "Diodes"

<https://byjus.com/physics/diodes/>

[17] Bru. "ทรานซิสเตอร์และการใช้งาน"

<http://blog.bru.ac.th/wp-content/uploads/bp-attachments/9267/%E0%B8%9A%E0%B8%97%E0%B8%97%E0%B8%B5%E0%B9%88-2.pdf>

[18] Electronics Tutorials. "Transistors / Transistor as a Switch"

[https://www.electronics-tutorials.ws/transistor/tran\\_4.html](https://www.electronics-tutorials.ws/transistor/tran_4.html)

[19] Z. Crawford, "Instrumentation Amplifiers", Mar. 21, 2014.

[Online]. Available: <https://www.egr.msu.edu/classes/ece480/capstone/spring14/group11/appnotes.zane.pdf>. [Accessed Oct. 25, 2022].

[20] C. Kitchin, L. Counts, "A DESIGNER'S GUIDE TO INSTRUMENTATION AMPLIFIERS", G3783-30-1/00 (rev. 0), 2000.

Available: ProQuest, <http://pws.npru.ac.th/sarthong/data/files/Instrumentation%20Amplifiers.pdf>. [Accessed November. 15, 2022].

[21] พิพัฒน์ พรหมมี, "วงจรรวมแบบแอนะล็อกสำหรับการสื่อสาร," คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, กรุงเทพฯ, 2017.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ก  
ส่วนโค้ดที่ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โค้ดในส่วนการเจนเนอร์เรตสัญญาณ 8 Bits

```
#define button1 8

#define resetPin 12

void setup()
{
  pinMode(button1,INPUT);

  DDRD = 0xFF; // Port D at Arduino Uno (pin 0-7)
}

void loop()
{
  /* while(digitalRead(button1) == LOW)
  Serial.println("button 1 is pressed");*/
  for ( int i = 0; i <= 255; i++) {
    PORTD = i;
    delay(1);
  }
  for ( int i = 256; i >=0; i--) {
    if(i==256){
      PORTD = i+253;
      delay(1);
    }
  }
  else{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

PORTD = i;

delay(1);

}

}

```

โค้ดในส่วนของการแสดงผลหน้าต่าง GUI

```

#from tkinter import *
from tkinter import ttk
from tkinter import filedialog
from matplotlib.backends.backend_tkagg import ( FigureCanvasTkAgg,
NavigationToolbar2Tk)
from matplotlib.figure import Figure

import numpy as np

import serial

from matplotlib import pyplot as plt

#from tkinter.scrolledtext import *

from tkinter import *

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
from tkinter.scrolledtext import *

root = Tk()

root.title("Curve Tracer")

root.geometry("1400x750+50+0") #W x H + X + Y

#root.maxsize(1400, 750) # specify the max size the window can expand to

root.config(bg="skyblue") # specify background color

arduino_port = " " #serial port of Arduino
DUT_Name= " "
baud_rate =0
ser= serial.Serial()

def upload():

    statusvar.set("Process...")

    sbar.update()

    import time

    time.sleep(1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

statusvar.set("Connected to Arduino port:" + arduino_port)

def progress(currentValue):

    pgb1["value"]=currentValue

def clrf():

    fig = Figure(figsize = (5, 5),dpi = 100)

    ax = fig.add_subplot(111)

    ax.cla()

    graph = FigureCanvasTkAgg(fig, master=right_frame)

    graph.get_tk_widget().grid(row=1,columnspan=10,sticky=N,padx=1,ipadx=300,ipady=
100)

def plot_graph():

    DUT=choice.get()

    if DUT=="Resistor":

        #Label(left_frame,text="เลือก"+DUT).grid(row=10, column=0)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

x=[]

y=[]

currentValue=0

samples = 512#how many samples to collect

pgb1["value"]=currentValue

pgb1["maximum"]=samples

#samples = 512#how many samples to collect

line = 1 #start at 0 because our header is 0 (not real data)

# collect the samples

while line <= samples:
    getData=ser.readline()

    dataString = getData.decode('utf-8')

    data_ar=dataString.rstrip('\n')

    TextBox.insert(END,data_ar+"\n")

    #Pushes the scrollbar and focus of text to the end of the text input.

    TextBox.yview(END)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

data=dataString

print(data)

readings = data.split(",")

#print(readings)

x.append(float(readings[0]))
y.append(float(readings[1]))

line+=1

currentValue=currentValue+1
per="{:.2f}".format((currentValue/512)*100)
stringvar.set(per)
percent.config(text=stringvar.get()+ ' %')
#print(currentValue)
pgb1.after(1, progress(currentValue))

pgb1.update() # Force an update of the GUI

ser.close()

statusvar.set("Disconnected from Arduino port:" + arduino_port)

currentValue=0

pgb1.after(1, progress(currentValue))

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

pgb1.update() # Force an update of the GUI

V_plus=[0]

I_plus=[0]

V_mi=[0]

I_mi=[0]

for i in range(len(x)//256):

    for j in range((i*256),(i*256)+128):

        if (i==0):

            V_plus.append(x[j])

            I_plus.append(y[j])

        elif (i==1):

            V_mi.append(x[j])

            I_mi.append(y[j])

global fig1

fig1 = Figure(figsize = (5, 5),dpi = 100)

line = 1 #start at 0 because our header is 0 (not real data)

# collect the samples

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

while line <= samples:

```

    getData=ser.readline()

    dataString = getData.decode('utf-8')

    data_ar=dataString.rstrip('\n')

    TextBox.insert(END,data_ar+"\n")

    #Pushes the scrollbar and focus of text to the end of the text input.

    TextBox.yview(END)

    data=dataString

    print(data)

    readings = data.split(",")

    #print(readings)

    x.append(float(readings[0]))

    y.append(float(readings[1]))

    line+=1

    currentValue=currentValue+1

    per="{:.2f}".format((currentValue/512)*100)

    stringvar.set(per)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

percent.config(text=stringvar.get()+ ' %')

#print(currentValue)

pgb1.after(1, progress(currentValue))

pgb1.update() # Force an update of the GUI

ser.close()

statusvar.set("Disconnected from Arduino port:" + arduino_port)

currentValue=0
pgb1.after(1, progress(currentValue))
pgb1.update() # Force an update of the GUI
V_plus=[0]
I_plus=[0]
V_mi=[0]
I_mi=[0]
V_new=[0]
I_new=[0]
VT=0

for i in range(len(x)//256):

    for j in range((i*256),(i*256)+128):

        if (i==0):

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

V_plus.append(x[j])

I_plus.append(y[j])

elif (i==1):

    V_mi.append(x[j])

    I_mi.append(y[j])

for i in range(20):

    V_new.append(V_plus[i])

    I_new.append(I_plus[i])

for i in range(15,20):

    VT+=V_plus[i]

    #print(VT)

Vd=VT/5

#print(V_mi)

fig1 = Figure(figsize = (5, 5),dpi = 100)

ax = fig1.add_subplot(111)

ax.plot(V_new, I_new ,color='#66FFFF',linewidth = 1.5)

ax.plot(V_mi , I_mi ,color='#66FFFF',linewidth = 1.5)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ax.set_facecolor('#2B2929')

ax.grid(linestyle = '--', linewidth = 0.5)

ax.set_xlabel("Voltage : Vce (V)")

ax.set_ylabel("Current : Ic (mA)")

ax.set_title(DUT_Name)

ax.set_yticks(np.arange(-10,10,1))

ax.set_xticks(np.arange(-10,10,1))

ax.text(max(V_new)+1, 1 , 'VD='+ '{:2f}'.format(Vd),color='#66FFFF')

graph = FigureCanvasTkAgg(fig1, master=right_frame)

graph.get_tk_widget().grid(row=1,columnspan=10,sticky=N,padx=1,ipadx=300,ipady=
100)

elif DUT=="Zener Diode":

    #Label(left_frame,text="เลือก"+DUT).grid(row=10, column=0)

    x=[]

    y=[]

    currentValue=0

    samples = 512#how many samples to collect

    pgb1["value"]=currentValue

```

```
pgb1["maximum"]=samples
```

```
line = 1 #start at 0 because our header is 0 (not real data)
```

```
# collect the samples
```

```
while line <= samples:
```

```
    getData=ser.readline()
```

```
    dataString = getData.decode('utf-8')
```

```
    data_ar=dataString.rstrip('\n')
```

```
    TextBox.insert(END,data_ar+"\n")
```

```
    #Pushes the scrollbar and focus of text to the end of the text input.
```

```
    TextBox.yview(END)
```

```
    data=dataString
```

```
    print(data)
```

```
    readings = data.split(",")
```

```
    #print(readings)
```

```
    x.append(float(readings[0]))
```

```
    y.append(float(readings[1]))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

line+=1

currentValue=currentValue+1

per="{:.2f}".format((currentValue/512)*100)

stringvar.set(per)

percent.config(text=stringvar.get()+ ' %')

#print(currentValue)

pgb1.after(1, progress(currentValue))

pgb1.update() # Force an update of the GUI
ser.close()
statusvar.set("Disconnected from Arduino port:" + arduino_port)

currentValue=0

pgb1.after(1, progress(currentValue))

pgb1.update() # Force an update of the GUI

V_plus=[0]

I_plus=[0]

V_mi=[0]

I_mi=[0]

V_new=[0]

I_new=[0]

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

VT=0

Vt=0

for i in range(len(x)//256):

    for j in range((i*256),(i*256)+128):

        if (i==0):

            V_plus.append(x[j])

            I_plus.append(y[j])

        elif (i==1):

            V_mi.append(x[j])

            I_mi.append(y[j])

    for i in range(20):

        V_new.append(V_plus[i])

        I_new.append(I_plus[i])

    for i in range(15,20):

        VT+=V_plus[i]

    #print(VT)

Vf=VT/5

for i in range(124,129):

    Vt+=V_mi[i]

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

# print(i)

# print(V_mi[i])

Vz=Vt/5

fig1 = Figure(figsize = (5, 5),dpi = 100)

ax = fig1.add_subplot(111)
ax.plot(V_new, I_new ,color='#66FFFF',linewidth = 1.5)
ax.plot(V_mi , I_mi ,color='#66FFFF',linewidth = 1.5)
ax.set_facecolor('#2B2929')
ax.grid(linestyle = '--', linewidth = 0.5)
ax.set_xlabel("Voltage : Vce (V)")
ax.set_ylabel("Current : Ic (mA)")
ax.set_title(DUT_Name)
ax.set_yticks(np.arange(-10,10,1))
ax.set_xticks(np.arange(-10,10,1))

ax.text(max(V_new)+1, 1 , 'VF='+ '{:.2f}'.format(Vf),color='#66FFFF')

ax.text(min(V_mi)-2, -1 , 'VZ='+ '{:.2f}'.format(Vz),color='#66FFFF')

graph = FigureCanvasTkAgg(fig1, master=right_frame)

```

```
graph.get_tk_widget().grid(row=1,columnspan=10,sticky=N,padx=1,ipadx=300,ipady=
100)
```

```
elif DUT=="NPN":

    #Label(left_frame,text="เลือก"+DUT).grid(row=10, column=0)

    x=[]
    y=[]
    I=[]
    currentValue=0
    samples = 8192#how many samples to collect
    pgb1["value"]=currentValue
    pgb1["maximum"]=samples
    line = 1 #start at 0 because our header is 0 (not real data)

    # collect the samples

    while line <= samples:

        getData=ser.readline()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

dataString = getData.decode('utf-8')

data_ar=dataString.rstrip('\n')

TextBox.insert(END,data_ar+"\n")

#Pushes the scrollbar and focus of text to the end of the text input.

TextBox.yview(END)

data=dataString

print(data)

readings = data.split(",")
#print(readings)
x.append(float(readings[0]))
y.append(float(readings[1]))
l.append(float(readings[2]))
#print(l)

line+=1

currentValue=currentValue+1

per="{:.2f}".format((currentValue/8192)*100)

stringvar.set(per)

percent.config(text=stringvar.get()+ ' %')

#print(currentValue)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

pgb1.after(1, progress(currentValue))

pgb1.update() # Force an update of the GUI

ser.close()

statusvar.set("Disconnected from Arduino port:" + arduino_port)

currentValue=0

#print(currentValue)

pgb1.after(1, progress(currentValue))

pgb1.update() # Force an update of the GUI

Vce=[0]

I1=[0]

I2=[0]

I3=[0]

I4=[0]

I5=[0]

I6=[0]

I7=[0]

I8=[0]

I9=[0]

I10=[0]

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

l11=[0]

l12=[0]

l13=[0]

l14=[0]

l15=[0]

l16=[0]

lB=[]

i_b=0

#print(len(x))

#print(len(y))

for i in range(len(x)//512):

    for j in range(i*512,(i*512)+256):

        if(i==0):

            Vce.append(x[j])

            l1.append(y[j])

        elif(i==1):

            l2.append(y[j])

        elif(i==2):

            l3.append(y[j])

        elif(i==3):

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

l4.append(y[j])

elif(i==4):

    l5.append(y[j])

elif(i==5):

    l6.append(y[j])

elif(i==6):

    l7.append(y[j])

elif(i==7):

    l8.append(y[j])

elif(i==8):

    l9.append(y[j])

elif(i==9):

    l10.append(y[j])

elif(i==10):

    l14.append(y[j])

elif(i==14):

    l15.append(y[j])

elif(i==15):

    l16.append(y[j])

for i in range(len(l)//512):

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for j in range(i*512,(i*512)+256):

    i_b+=I[j]

i_b=i_b/256

IB.append(i_b)

#global fig

#global ax

#global graph

print(IB)

#print(len(I1))

#print(len(I16))

fig1 = Figure(figsize = (5, 5),dpi = 100)

ax = fig1.add_subplot(111)

ax.set_facecolor('#2B2929')

ax.grid(linestyle = '--', linewidth = 0.5)

# x_new, bspline, y_new

# Plot the new data points

ax.plot(Vce, I16 ,color='#FF9999')

ax.plot(Vce, I15 ,color='#66FFFF')

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ax.plot(Vce, I14 ,color='#FFFF99')

ax.plot(Vce, I13 ,color='#FF9999')

ax.plot(Vce, I12 ,color='#66FFFF')

ax.plot(Vce, I11 ,color='#FFFF99')

ax.plot(Vce, I10 ,color='#FF9999')

ax.plot(Vce, I9 ,color='#66FFFF')

ax.plot(Vce, I8 ,color='#FFFF99')

ax.plot(Vce, I7 ,color='#FF9999')

ax.plot(Vce, I6 ,color='#66FFFF')

ax.plot(Vce, I5 ,color='#FFFF99')

ax.plot(Vce, I4 ,color='#FF9999')

ax.plot(Vce, I3 ,color='#66FFFF')

ax.plot(Vce, I2 ,color='#FFFF99')

ax.plot(Vce, I1 ,color='#FF9999')

ax.text(max(Vce)+0.25,max(I1)-0.3,'I1
='+'{:2f}'.format(IB[0])+uA',color='#FF9999')#,fontname="monospace")

ax.text(max(Vce)+0.25,max(I2)-0.2,'I2
='+'{:2f}'.format(IB[1])+uA',color='#FFFF99')

ax.text(max(Vce)+0.25,max(I3) ,I3 ='+'{:2f}'.format(IB[2])+uA',color='#66FFFF')

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ax.text(max(Vce)+0.25,max(I4) , 'I4 ='+{:2f}'.format(IB[3])+ 'uA',color='#FF9999')
ax.text(max(Vce)+0.25,max(I5) , 'I5 ='+{:2f}'.format(IB[4])+ 'uA',color='#FFFF99')
ax.text(max(Vce)+0.25,max(I6) , 'I6 ='+{:2f}'.format(IB[5])+ 'uA',color='#66FFFF')
ax.text(max(Vce)+0.25,max(I7) , 'I7 ='+{:2f}'.format(IB[6])+ 'uA',color='#FF9999')
ax.text(max(Vce)+0.25,max(I8) , 'I8 ='+{:2f}'.format(IB[7])+ 'uA',color='#FFFF99')
ax.text(max(Vce)+0.25,max(I9) , 'I9 ='+{:2f}'.format(IB[8])+ 'uA',color='#66FFFF')
ax.text(max(Vce)+0.25,max(I10)
,'I10='+{:2f}'.format(IB[9])+ 'uA',color='#FF9999')
ax.text(max(Vce)+0.25,max(I11)
,'I11='+{:2f}'.format(IB[10])+ 'uA',color='#FFFF99')
ax.text(max(Vce)+0.25,max(I12)
,'I12='+{:2f}'.format(IB[11])+ 'uA',color='#66FFFF')
ax.text(max(Vce)+0.25,max(I13)
,'I13='+{:2f}'.format(IB[12])+ 'uA',color='#FF9999')
ax.text(max(Vce)+0.25,max(I14)
,'I14='+{:2f}'.format(IB[13])+ 'uA',color='#FFFF99')
ax.text(max(Vce)+0.25,max(I15)
,'I15='+{:2f}'.format(IB[14])+ 'uA',color='#66FFFF')
ax.text(max(Vce)+0.25,max(I16)
,'I16='+{:2f}'.format(IB[15])+ 'uA',color='#FF9999')
ax.text(min(Vce)+0.25,max(I16)+0.25
,'Beta='+{:2f}'.format(max(I16)/IB[15]*1000),color='#FFFFFF')

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ax.set_xticks(np.arange(min(Vce),max(Vce)+2.2,0.5))

ax.set_yticks(np.arange(min(Vce),max(I16)+1,0.5))

ax.set_xlabel("Voltage : Vce (V)")

ax.set_ylabel("Current : Ic (mA)")

ax.set_title(DUT_Name)

graph = FigureCanvasTkAgg(fig1, master=right_frame)

graph.get_tk_widget().grid(row=1,columnspan=10,sticky=N,padx=1,ipadx=300,ipady=
100)

elif DUT=="PNP":

    #Label(left_frame,text="เลือก"+DUT).grid(row=10, column=0)

    x=[]

    y=[]

    I=[]

    currentValue=0

    samples = 8192#how many samples to collect

    pgb1["value"]=currentValue

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

pgb1["maximum"]=samples

line = 1 #start at 0 because our header is 0 (not real data)

# collect the samples

while line <= samples:

    getData=ser.readline()
    dataString = getData.decode('utf-8')
    data_ar=dataString.rstrip('\n')
    TextBox.insert(END,data_ar+"\n")
    #Pushes the scrollbar and focus of text to the end of the text input.
    TextBox.yview(END)
    data=dataString
    print(data)

    readings = data.split(",")

    #print(readings)

    x.append(float(readings[0]))

    y.append(float(readings[1]))

    l.append(float(readings[2]))

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

line+=1

currentValue=currentValue+1

per="{:.2f}".format((currentValue/8192)*100)

stringvar.set(per)

percent.config(text=stringvar.get()+ ' %')

#print(currentValue)

pgb1.after(1, progress(currentValue))

pgb1.update() # Force an update of the GUI
ser.close()

statusvar.set("Disconnected from Arduino port:" + arduino_port)

currentValue=0

#print(currentValue)

pgb1.after(1, progress(currentValue))

pgb1.update() # Force an update of the GUI

Vce=[0]

I1=[0]

I2=[0]

I3=[0]

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

I4=[0]

I5=[0]

I6=[0]

I7=[0]

I8=[0]

I9=[0]

I10=[0]

I11=[0]

I12=[0]

I13=[0]

I14=[0]

I15=[0]

I16=[0]

IB=[]

i\_b=0

for i in range(len(x)//512):

    for j in range(i\*512,(i\*512)+256):

        if(i==0):

            Vce.append(x[j])

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
l1.append(y[j])
```

```
elif(i==1):
```

```
l2.append(y[j])
```

```
elif(i==2):
```

```
l3.append(y[j])
```

```
elif(i==3):
```

```
l4.append(y[j])
```

```
elif(i==4):
```

```
l5.append(y[j])
```

```
elif(i==5):
```

```
l6.append(y[j])
```

```
elif(i==6):
```

```
l7.append(y[j])
```

```
elif(i==7):
```

```
l8.append(y[j])
```

```
elif(i==8):
```

```
l9.append(y[j])
```

```
elif(i==9):
```

```
l10.append(y[j])
```

```
elif(i==10):
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        l11.append(y[j])

    elif(i==11):

        l12.append(y[j])

    elif(i==12):

        l13.append(y[j])

    elif(i==13):

        l14.append(y[j])

    elif(i==14):

        l15.append(y[j])

    elif(i==15):

        l16.append(y[j])

for i in range(len(l)//512):

    for j in range(i*512,(i*512)+256):

        i_b+=l[j]

        i_b=i_b/256

    IB.append(i_b)

print(IB)

#global fig

#global ax

#global graph

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

fig1 = Figure(figsize = (5, 5),dpi = 100)

ax = fig1.add_subplot(111)

ax.set_facecolor('#2B2929')

ax.grid(linestyle = '--', linewidth = 0.5)

# x_new, bspline, y_new

# Plot the new data points
ax.plot(Vce, I16 ,color='#FF9999')
ax.plot(Vce, I15 ,color='#66FFFF')
ax.plot(Vce, I14 ,color='#FFFF99')
ax.plot(Vce, I13 ,color='#FF9999')
ax.plot(Vce, I12 ,color='#66FFFF')
ax.plot(Vce, I11 ,color='#FFFF99')
ax.plot(Vce, I10 ,color='#FF9999')
ax.plot(Vce, I9 ,color='#66FFFF')
ax.plot(Vce, I8 ,color='#FFFF99')
ax.plot(Vce, I7 ,color='#FF9999')
ax.plot(Vce, I6 ,color='#66FFFF')
ax.plot(Vce, I5 ,color='#FFFF99')

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ax.plot(Vce, I4 ,color='#FF9999')

ax.plot(Vce, I3 ,color='#66FFFF')

ax.plot(Vce, I2 ,color='#FFFF99')

ax.plot(Vce, I1 ,color='#FF9999')

ax.text(max(Vce)+0.25,max(I1)-0.3,'I1
='+'{:2f}'.format(IB[0])+uA',color='#FF9999')#,fontname="monospace")

ax.text(max(Vce)+0.25,max(I2)-0.1,'I2
='+'{:2f}'.format(IB[1])+uA',color='#FFFF99')

ax.text(max(Vce)+0.25,max(I3)+0.2,'I3
='+'{:2f}'.format(IB[2])+uA',color='#66FFFF')

ax.text(max(Vce)+0.25,max(I4) ,,'I4 ='+'{:2f}'.format(IB[3])+uA',color='#FF9999')
ax.text(max(Vce)+0.25,max(I5) ,,'I5 ='+'{:2f}'.format(IB[4])+uA',color='#FFFF99')
ax.text(max(Vce)+0.25,max(I6) ,,'I6 ='+'{:2f}'.format(IB[5])+uA',color='#66FFFF')
ax.text(max(Vce)+0.25,max(I7) ,,'I7 ='+'{:2f}'.format(IB[6])+uA',color='#FF9999')
ax.text(max(Vce)+0.25,max(I8) ,,'I8 ='+'{:2f}'.format(IB[7])+uA',color='#FFFF99')
ax.text(max(Vce)+0.25,max(I9) ,,'I9 ='+'{:2f}'.format(IB[8])+uA',color='#66FFFF')

ax.text(max(Vce)+0.25,max(I10)
,'I10='+'{:2f}'.format(IB[9])+uA',color='#FF9999')

ax.text(max(Vce)+0.25,max(I11)
,'I11='+'{:2f}'.format(IB[10])+uA',color='#FFFF99')

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ax.text(max(Vce)+0.25,max(I12)
,'I12='+ '{:.2f}'.format(IB[11])+ 'uA',color='#66FFFF')

ax.text(max(Vce)+0.25,max(I13)
,'I13='+ '{:.2f}'.format(IB[12])+ 'uA',color='#FF9999')

ax.text(max(Vce)+0.25,max(I14)
,'I14='+ '{:.2f}'.format(IB[13])+ 'uA',color='#FFFF99')

ax.text(max(Vce)+0.25,max(I15)
,'I15='+ '{:.2f}'.format(IB[14])+ 'uA',color='#66FFFF')

ax.text(max(Vce)+0.25,max(I16)
,'I16='+ '{:.2f}'.format(IB[15])+ 'uA',color='#FF9999')

ax.text(min(Vce)+0.25,max(I16)+0.25
,'Beta='+ '{:.2f}'.format(max(I16)/IB[15]*1000),color='#FFFFFF')

ax.set_xticks(np.arange(min(Vce),max(Vce)+2.2,0.5))
ax.set_yticks(np.arange(min(Vce),max(I16)+1,0.5))

ax.set_xlabel("Volltage : Vce (V)")
ax.set_ylabel("Current : Ic (mA)")
ax.set_title(DUT_Name)

graph = FigureCanvasTkAgg(fig1, master=right_frame)

```

```
graph.get_tk_widget().grid(row=1,columnspan=10,sticky=N,padx=1,ipadx=300,ipady=
100)
```

```
#create left & Right frame
```

```
left_frame = Frame(root, bg="white", relief="sunken")
left_frame.grid(row=0, column=0, padx=10, pady=5, sticky="nsew")
right_frame = Frame(root, bg="white", relief="sunken")
right_frame.grid(row=0, column=1, padx=10, pady=5, sticky="nsew")
root.grid_rowconfigure(0, weight=1)
root.grid_columnconfigure(0, weight=1)
root.grid_columnconfigure(1, weight=1000)

#Mode
Label(left_frame,text="MODE",font=('arial',10,'bold'),bg="white").grid(row=2,column=0
)

choice = StringVar(value="เลือกชนิดอุปกรณ์ที่ต้องการวัด")
combo=ttk.Combobox(left_frame,textvariable=choice)
combo.grid(row=3,column=0)

combo["values"]="Resistor","Diode","Zener Diode","NPN","PNP")
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

btn1 =
Button(left_frame,font=('arial',10,'bold'),text="Start",width=13,command=plot_graph,
pady=5).grid(row=4,column=0,pady=15)

def save():

    global fig1

    filename = filedialog.asksaveasfilename(initialfile =
'Untitled.png',defaultextension=".png",filetypes=[("All Files", "*.*"),("Portable Graphics
Format", "*.png")])

    fig1.savefig(filename)

#Graph
lab = Label(right_frame, text="Plotting", bg =
'white').grid(row=0,columnspan=5,sticky=E,padx=500,pady=10)

fig = Figure(figsize = (5, 5),dpi = 100)

plt.rcParams.update({'font.family':'Arial'})

ax = fig.add_subplot(111)

ax.set_xlabel("X-AXIS")

ax.set_ylabel("Y-AXIS")

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ax.set_title("DUT NAME")

ax.set_facecolor('#2B2929')

ax.grid(linestyle = '--', linewidth = 0.5)

#ax.set_xlim(0,10)

ax.set_yticks(np.arange(0,10.5,0.5))

ax.set_xticks(np.arange(0,10.5,0.5))

graph = FigureCanvasTkAgg(fig, master=right_frame)

graph.get_tk_widget().grid(row=1,columnspan=10,sticky=N,padx=1,ipadx=300,ipady=
100)

def connect_port():
    global arduino_port
    global baud_rate
    global ser
    global DUT_Name

    arduino_port=COM.get()

    baud_rate=baud.get()

    DUT_Name=name.get()

    ser = serial.Serial(arduino_port, baud_rate)

    upload()

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

print("Connected to Arduino port:" + arduino_port)

#print(DUT_Name)

def text_clr():

    print(arduino_port)

    print( baud_rate)

    et1.delete(0,END)

    et2.delete(0,END)

    et3.delete(0,END)

    fig = Figure(figsize = (5, 5),dpi = 100)
    ax = fig.add_subplot(111)
    ax.cla()
    ax.grid()
    ax.set_xlabel("X-AXIS")
    ax.set_ylabel("Y-AXIS")
    ax.set_title("DUT NAME")

    ax.set_facecolor('#2B2929')

    ax.grid(linestyle = '--', linewidth = 0.5)

    #ax.set_xlim(0,10)

    ax.set_yticks(np.arange(0,10.5,0.5))

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ax.set_xticks(np.arange(0,10.5,0.5))

graph = FigureCanvasTkAgg(fig, master=right_frame)

graph.get_tk_widget().grid(row=1,columnspan=10,sticky=N,padx=1,ipadx=300,ipady=
100)

statusvar.set("Ready")

stringvar.set('00.0 %')

percent.config(text=stringvar.get())

currentValue=0

pgb1.after(1, progress(currentValue))

pgb1.update() # Force an update of the GUI

TextBox.delete("1.0","end")

ser.close()

Label(left_frame,text="COM PORT :",bg="white").grid(row=5,column=0,sticky=W)

Label(left_frame,text="Buad Rate :",bg="white").grid(row=6,sticky=W)

Label(left_frame,text="DUT Name :",bg="white").grid(row=7,sticky=W)

COM = StringVar()

et1 = Entry(left_frame,textvariable=COM)

```

```

et1.grid(row=5,column=0,sticky=E,padx=0)

buad= IntVar()

et2 = Entry(left_frame,textvariable=buad)

et2.delete(0,END)

et2.grid(row=6,column=0,sticky=E)

name = StringVar()

et3 = Entry(left_frame,textvariable=name)

et3.grid(row=7,column=0,sticky=E,padx=0)

btn2 =
Button(left_frame,font=('arial',10,'bold'),text="Connect",pady=5,command=connect_
port).grid(row=8, column=0,pady=15,padx=10,ipadx=26,sticky=N)

btn3 = Button(left_frame,font=('arial',10,'bold'),text="Clear
Screen",pady=5,command=text_clr).grid(row=9,
column=0,pady=15,padx=70,ipadx=10,sticky=E)

btn4 = Button(left_frame,font=('arial',10,'bold'),text="Save
As",pady=5,command=save).grid(row=12,
column=0,pady=15,padx=70,ipadx=25,sticky=N)

```

```
#progress Bar
```

```
pgb1 = ttk.Progressbar(left_frame, mode="determinate",length=250)
```

```
pgb1.grid(row=10,column=0)
```

```
stringvar=StringVar()
```

```
stringvar.set('00.0 %')
```

```
percent=Label(left_frame,text=stringvar.get(),font=("",10),bg="white",border=0)
```

```
percent.grid(row=11)
```

```
statusvar = StringVar()
```

```
statusvar.set("Ready")
```

```
sbar = Label(root, textvariable=statusvar, relief=SUNKEN, anchor="w")
```

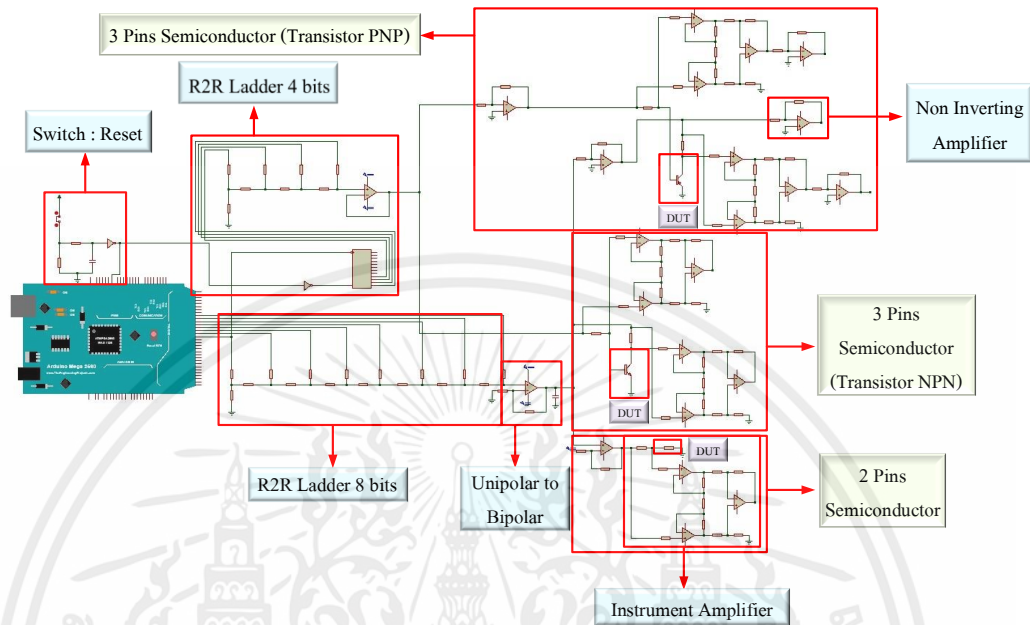
```
sbar.grid(row=12,ipadx=1000,sticky=W,columnspan=15)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## การออกแบบวงจร Curve Tracer ในโปรแกรม Proteus



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้