

Pre-Lab #64

Electronics Club

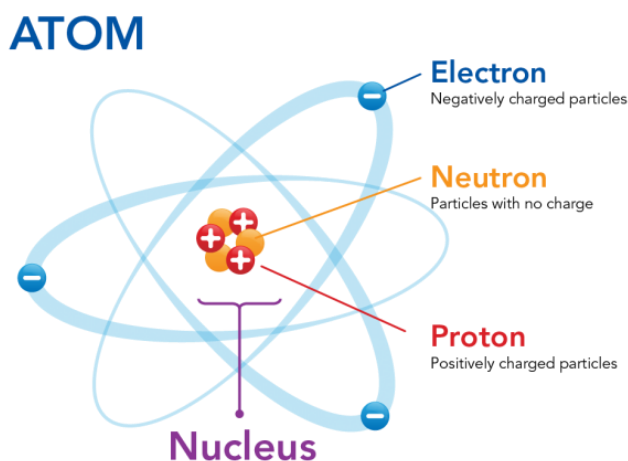
School of Engineering

King Mongkut's Institute of Technology Ladkrabang

ทฤษฎีพื้นฐานทางไฟฟ้า

ไฟฟ้าคืออะไร (Electric)

ถ้าพูดถึงคำว่า “ไฟฟ้า” เราก็คงจะคุ้นหูกันดี ไม่ว่าจะเป็นอะไร ๆ ก็ต้องใช้ไฟฟ้าถึงจะทำงานได้ ไม่ว่าจะเป็น โทรศัพท์ วิทยุ คอมพิวเตอร์ พัดลม ตู้เย็น โคมไฟ โทรศัพท์ รถบังคับ ฯลฯ แล้วเราเคยนึกสงสัยหรือไม่ว่าเจ้า “ไฟฟ้า” นั้นมันคืออะไร มาจากไหน รูปร่างหน้าตาเป็นอย่างไร ก่อนอื่นเราต้องทำความเข้าใจกันก่อนว่า ไฟฟ้านั้นมีอยู่ที่ไหน ทุกคนคงทราบดีว่าวัตถุประกอบไปด้วยอะตอมจำนวนมาก โดยอะตอมนั้นจะประกอบด้วย นิวเคลียสและอิเล็กตรอนซึ่งมีประจุเป็นลบ โดยที่อิเล็กตรอนนั้นจะโคจรรอบนิวเคลียสอย่างมีกฎเกณฑ์ที่แน่นอนภายในนิวเคลียสก็จะประกอบด้วยโปรตอนซึ่งมีประจุเป็นบวกและนิวตรอนไม่มีประจุไฟฟ้าจำนวนโปรตอนจะเท่ากับจำนวนอิเล็กตรอนจึงทำให้วัตถุทั่วไป มีค่าของประจุทางไฟฟ้าเป็นกลางแต่ถ้าหากวัตถุใดมีอิเล็กตรอนเกินหรือขาดไปก็จะทำให้เกิดค่าทางไฟฟ้าขึ้นมาทันที



รูปแสดงโครงสร้างของอะตอม

กระแสไฟฟ้าคืออะไร (I)

หากจะนิยามภาพการไหลของกระแสไฟฟ้า (Current) อาจจะเปรียบเทียบได้กับการไหลของกระแสน้ำในท่อประปา กระแสไฟฟ้าก็คือ การเคลื่อนที่ของประจุไฟฟ้า โดยประจุไฟฟ้านั้นมีสองชนิดคือ ชนิดบวก กับ ชนิดลบ โดยปกติแล้วการเคลื่อนที่จะเป็นการเคลื่อนที่ของประจุไฟฟ้าลบ (อิเล็กตรอน) แต่ทิศทางการเคลื่อนที่นั้นไม่ใช่ทิศทางการไหลของกระแส ซึ่ง “ทิศทางการไหลของกระแสจะตรงข้ามกับการเคลื่อนที่ของประจุไฟฟ้าลบ” ซึ่งเป็นข้อตกลงกันมาตั้งแต่สมัยแรกๆ ที่ค้นพบ

โดยขนาดของกระแสไฟฟ้า นั้น กำหนดได้จากปริมาณของประจุไฟฟ้าที่ไหล ผ่านจุดหนึ่งจุดใด ในเส้นลวดใน 1 วินาที มีหน่วยเป็น แอมแปร์ (Ampere: A)

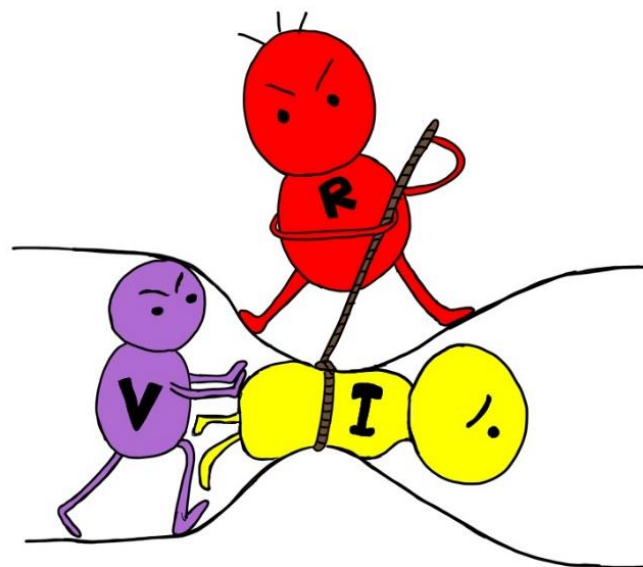
ความต่างศักย์ไฟฟ้าคืออะไร (V)

ความแตกต่างของศักย์ไฟฟ้าระหว่างจุด 2 จุด ในวงจรที่มีกระแสไฟฟ้า คือ พลังงานไฟฟ้าต่อหนึ่งประจุที่สูญเสียไปในระหว่างจุด 2 จุด เราเรียกว่า *ความต่างศักย์ไฟฟ้า* ซึ่งเป็นผลงานการค้นพบของท่านโวลตา (ผู้ค้นพบหลักการของเซลล์ไฟฟ้าซึ่งต่อมาก็มีการสร้างแหล่งจ่ายแรงดันไฟฟ้าชนิดต่างๆ ออกมา) หากจะยกตัวอย่างก็เช่น ระหว่างขั้วบวกและลบของถ่านไฟฉาย

หน่วยของความต่างศักย์ไฟฟ้า คือ จูลต่อคูลอมบ์หรือเรียกว่า โวลต์ (Volt) และเขียนแทนด้วย V

ความต้านทานไฟฟ้าคืออะไร (R)

ความต้านทานไฟฟ้า คือ ความยากง่ายของการไหลผ่านของกระแสไฟฟ้า ซึ่งเป็นคุณสมบัติของวัตถุต่างๆ ซึ่งจะเกิดกับวัตถุทุกชนิด โดยที่ความต้านทานมีหน่วยเป็น โอห์ม (*Ohm ใช้สัญลักษณ์ Ω*) ความต้านทาน 1 โอห์ม หมายถึง ความต้านทานของเส้นลวดที่ยอมให้กระแสไหลผ่านได้ 1 แอมแปร์ เมื่อใส่แรงดันไฟฟ้าเข้าไป 1 โวลต์ วัตถุที่ยอมให้ไฟฟ้าไหลผ่านได้ง่าย เรียกว่า *ตัวนำไฟฟ้า (Conductor)* ได้แก่ พวกโลหะ เช่น ทอง เงิน ทองแดง อะลูมิเนียม เหล็ก เป็นต้น วัตถุที่ไม่ยอมให้กระแสไฟฟ้าไหลผ่านได้ง่าย เรียกว่า *ฉนวนไฟฟ้า (Insulator)* ได้แก่ ยาง พลาสติก แก้ว เป็นต้น นอกจากนี้ยังมีวัตถุประเภทหนึ่งที่มีคุณสมบัติอยู่ระหว่างตัวนำและฉนวนไฟฟ้า เรียกว่า *สารกึ่งตัวนำ (Semiconductor)*



รูปแสดงความสัมพันธ์ระหว่าง V, I และ R

สัญลักษณ์ทางอิเล็กทรอนิกส์

สัญลักษณ์เหล่านี้ นื่อง ๆ อาจเคยเห็นกันมาบ้างแล้ว แต่อาจสงสัยอยู่ว่ามันคืออะไร ซึ่งในที่นี้ จะยกมาเฉพาะแบบที่พบเห็นกันได้ง่าย ๆ และใช้ในวงจรทั่วไปๆเป็นตัวอย่าง ซึ่งได้แก่

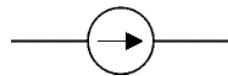
แหล่งจ่ายกระแสสลับ

-



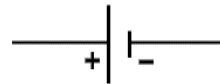
แหล่งจ่ายกระแสตรง

-



เซลล์ไฟฟ้า หรือแบตเตอรี่

-



สายดิน (Ground, GND)

-



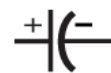
ตัวต้านทาน (R)

-



ตัวเก็บประจุ (C)

-



ตัวเหนี่ยวนำ (L)

-



ไดโอด

-



ซีเนอร์ไดโอด

-



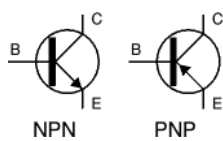
ไดโอดเปล่งแสง (LED)

-



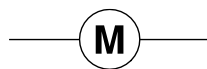
ทรานซิสเตอร์

-



มอเตอร์

-



หลอดไฟ

-



โวลต์มิเตอร์

-



แอมป์มิเตอร์

-



โอห์มมิเตอร์

-

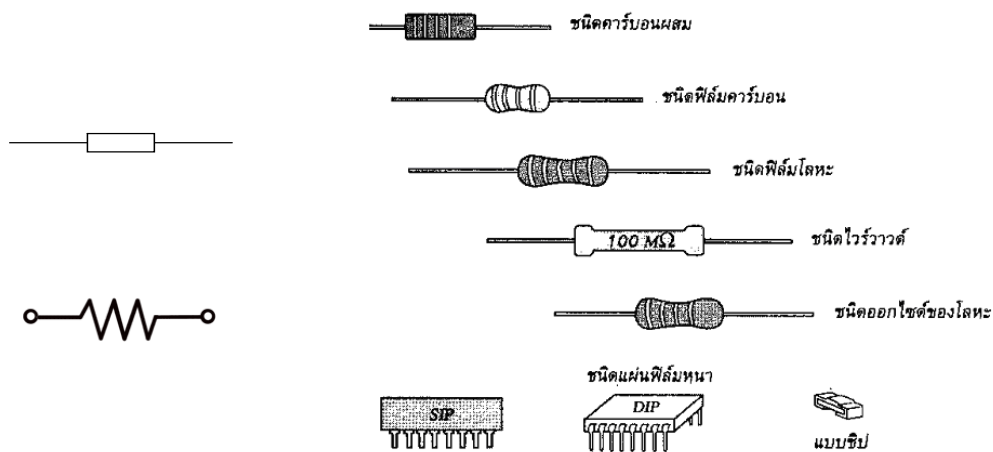


ตัวต้านทาน (Resister)

เป็นอุปกรณ์ Electronic พื้นฐานของวงจรไฟฟ้า ทำหน้าที่เป็นตัวต้านทานการไหลของกระแสไฟฟ้า ในวงจรไฟฟ้า โดยส่วนมากจะสร้างขึ้นจากโลหะผสมต่าง ๆ เช่น คาร์บอน เซรามิก เป็นต้น มีหน่วยเป็น โอห์ม สัญลักษณ์ Ω (โอเมก้า)

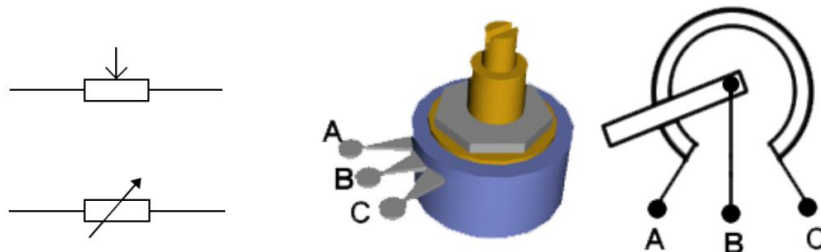
ตัวต้านทานแบ่งเป็น 2 ประเภท

1. ความต้านทานคงที่ (Fix Resister) ทำมาจากสารผสมของคาร์บอน หรือ เซรามิกจะใช้แถบสีคาดเอาไว้บนตัวต้านทานเอาไว้บอกค่าความต้านทานนั้น ๆ



รูปและสัญลักษณ์ของตัวต้านทานคงที่

2. ความต้านทานที่ปรับค่าได้ (Variable Resister) หรือเรียกว่า รีโอสแตท (Rheostat) ตัวต้านทานแบบนี้สามารถปรับค่าได้ในช่วงที่ไม่เกินความต้านทานที่มันทำได้โดยเลื่อนแขนปรับให้แตะเส้นลวดที่ตำแหน่งต่าง ๆ กัน ใช้ทำเป็น Volume ในอุปกรณ์ Electronics



รูปสัญลักษณ์ของตัวต้านทานปรับค่าได้

รหัสแถบสีของตัวต้านทาน

สำหรับความต้านทานชนิด 3 แถบสี จะแสดงสั้ไว้บนตัวต้านทานโดยแต่ละสีจะมีค่าความต้านทานที่แน่นอนเป็น มาตรฐานสากลกำกับอยู่ ในการอ่านรหัสสีให้เทียบแถบสีกับค่าที่บอกข้างล่างนี้

วิธีอ่านรหัสสี

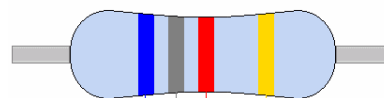
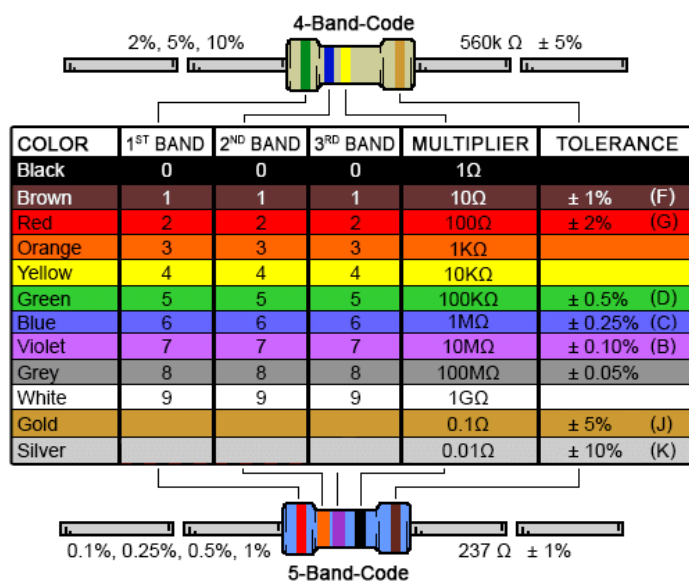
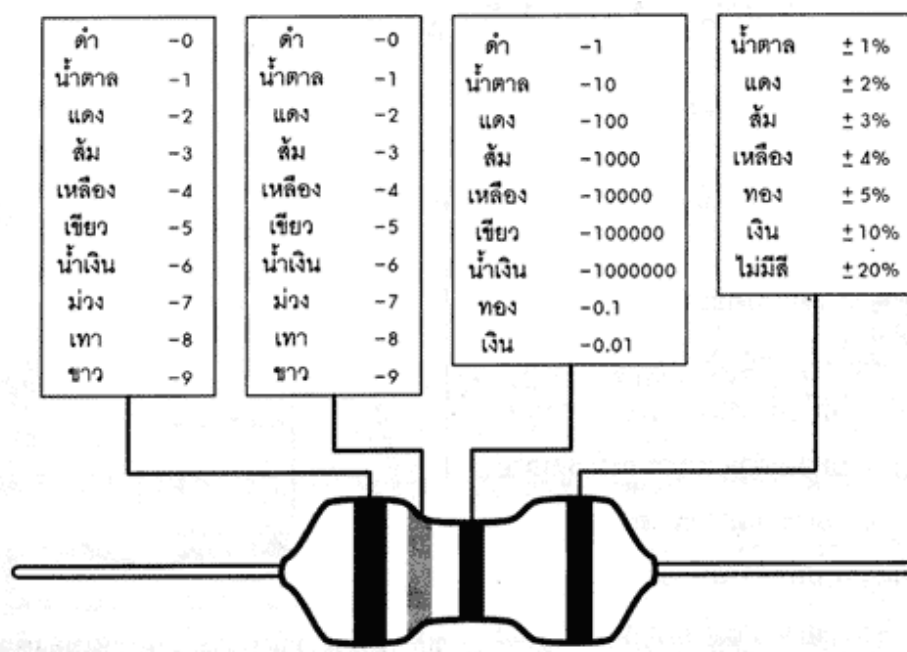
แถบสีที่1 : มีค่าเป็นหลักสิบ

แถบสีที่2 : มีค่าเป็นหลักหน่วย

แถบสีที่3 : เป็นค่าเลขยกกำลังตามตัวเลขนั้น ๆ

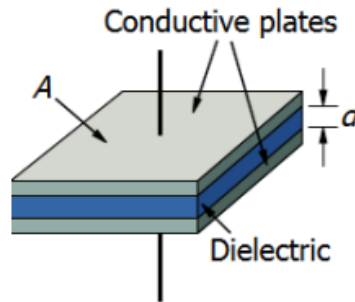
แถบสีที่4 : เป็นค่าความผิดพลาดของตัวต้านทาน

จะได้ในรูปดังนี้ $12 \times 10^3 \pm 5\%$



ตัวเก็บประจุ (Capacitor)

ตัวเก็บประจุ (Capacitor หรือ Condenser) มีหน่วยเป็นฟารัด (F) นิยมใช้หน่วยเล็ก ๆ เช่น F , μF , pF ทำหน้าที่เก็บสะสมพลังงานในรูปสนามไฟฟ้า ประกอบด้วย แผ่นตัวนำสองแผ่นเรียกว่าแผ่น 1 เฟลต คั่นด้วย แผ่นไดอิเล็กทริก ซึ่งทำด้วยฉนวนไฟฟ้า

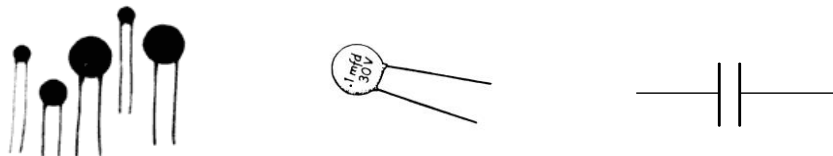


รูปส่วนประกอบของตัวเก็บประจุ

ตัวเก็บประจุนั้นมีอยู่หลายชนิดมากมาย ซึ่งแบ่งออกได้เป็น 2 ประเภท คือ

1. ตัวเก็บประจุแบบค่าคงที่

1.1 ชนิดเซรามิก ตัวเก็บประจุชนิดเซรามิก ส่วนใหญ่ตัวเก็บประจุชนิดนี้มีค่าน้อยกว่า 1 ไมโครฟารัด และเป็นตัวเก็บประจุชนิดที่ไม่มีขั้ว สามารถทนแรงดันได้ประมาณ 50-100 โวลต์ ค่าความจุของตัวเก็บประจุชนิดเซรามิกที่มีใช้กันในปัจจุบันอยู่ในช่วง 1 พิโคฟารัด ถึง 0.1 ไมโครฟารัด



รูปและสัญลักษณ์ของตัวเก็บประจุชนิดเซรามิก

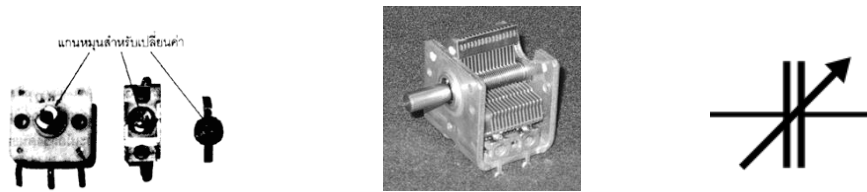
1.2 ตัวเก็บประจุชนิดอิเล็กโทรไลติก ตัวเก็บประจุชนิดนี้มีขั้วที่แน่นอนพิมพ์ติดไว้ด้านข้างตัวถัง ถ้าป้อนแรงดันให้กับตัวเก็บประจุผิดขั้วก็อาจเกิดความเสียหายกับตัวมันและอุปกรณ์ที่ประกอบร่วมกับตัวมันได้ ขั้วของตัวเก็บประจุชนิดนี้สังเกตได้ง่ายๆ คือ ขาที่ยาวจะเป็นขั้วบวก และขาที่สั้นจะเป็นขั้วลบ และยังต้องระวังการป้อนแรงดันให้ตัวเก็บประจุ โดยจะมีอัตราทนแรงดันพิมพ์ติดไว้หน่วยเป็นโวลต์ V



รูปและสัญลักษณ์ของตัวเก็บประจุชนิดอิเล็กโทรไลติก

2. ตัวเก็บประจุแบบปรับค่าได้

เราสามารถปรับค่าความจุของตัวเก็บประจุได้ ด้วยการหมุนแกนของตัวเก็บประจุ ค่าความจุก็จะเปลี่ยนไปตามมุมที่หมุน โดยมากใช้ตัวเก็บประจุแบบปรับค่าได้ ในวงจรเลือกคลื่นวิทยุทั่ว ๆ ไป



รูปและสัญลักษณ์ของตัวเก็บประจุปรับค่าได้

ไดโอด และทรานซิสเตอร์ (Diode & Transistor)

ไดโอด (Diode)

เป็นอุปกรณ์อิเล็กทรอนิกส์ที่สร้างโดยใช้สารกึ่งตัวนำ 2 ชนิดคือ ชนิด P ซึ่งเป็นขั้ว Anode (A) และชนิด N ซึ่งเป็นขั้ว Cathode (K) นำมาต่อกันทำให้กระแสไฟฟ้าสามารถไหลผ่านได้ทางเดียว คือ จาก A ไป K เท่านั้น



ไดโอดแบบอื่น ๆ

ซีเนอร์ไดโอด (Zener Diode)

เป็นไดโอดที่ใช้กับวงจรไบอัสย้อนกลับ (แต่ถ้าต่อไบอัสตรงก็จะเหมือนกับไดโอดธรรมดา) ปกติแล้วไดโอดทั่วไปเมื่อไบอัสย้อนกลับจะไม่มีกระแสไหลผ่านตัวไดโอดได้เลย แต่ซีเนอร์ไดโอดเมื่อไบอัสย้อนกลับจะมีกระแสไหลผ่านได้ ถ้าแรงดันตกคร่อมไดโอดมีค่าๆ หนึ่ง ซึ่งค่านี้เรียกว่า “แรงดันพังทลาย” ค่านี้จะขึ้นอยู่กับชนิดของซีเนอร์ไดโอด



โฟโตไดโอด (Photo Diode)

เป็นไดโอดที่อาศัยแสงจากภายนอกผ่านเลนส์ ซึ่งฝังตัวอยู่ระหว่างรอยต่อ P-N เพื่อกระตุ้นให้ไดโอดทำงาน



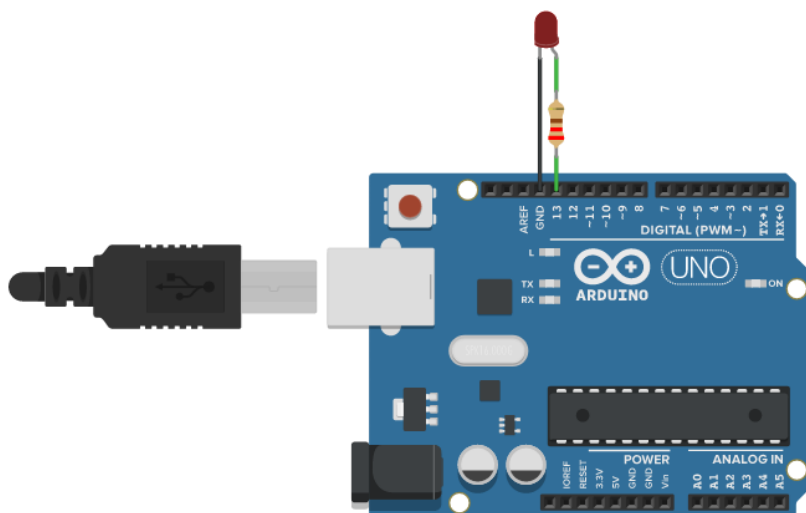
ไดโอดเปล่งแสง (Light Emitting Diodes; LED)

คุณสมบัติจะเหมือนไดโอดธรรมดาทั่วไป จะแตกต่างจากไดโอดธรรมดาตรงที่เมื่อเราจ่ายไฟให้แล้ว มันจะเปล่งแสงออกมาความสว่างและสีของแสงจะแตกต่างกันไปแล้วแต่ชนิดของสารที่ใช้ทำ

การดูขั้วให้ดูที่ขา ซึ่งขาที่ยาวกว่าจะเป็น A (+) และขาที่สั้นกว่าจะเป็น K (-) หรือดูรอยบากที่ตัวหลอด ขาตรงรอยบากนั้นจะเป็นขั้ว K (-)



รูปเปรียบเทียบโครงสร้างของ LED



ในการต่อ LED ให้นำขา K(-) ลง GND

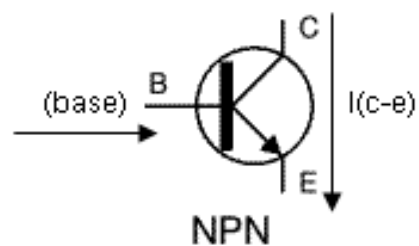
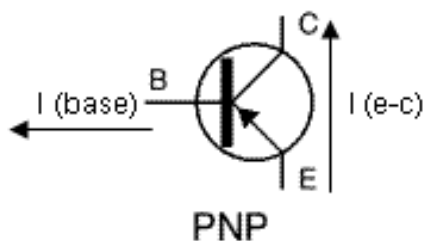
ทรานซิสเตอร์ (Transistor)



สร้างจากสารกึ่งตัวนำ 2 ชนิดเหมือนกับไดโอด จะแตกต่างกันที่ไดโอดมีเพียงหนึ่งรอยต่อคือ P-N (มี 2 ขาคือ A, K) แต่ทรานซิสเตอร์มี 2 รอยต่อคือ P-N-P หรือ N-P-N และมี 3 ขาคือ

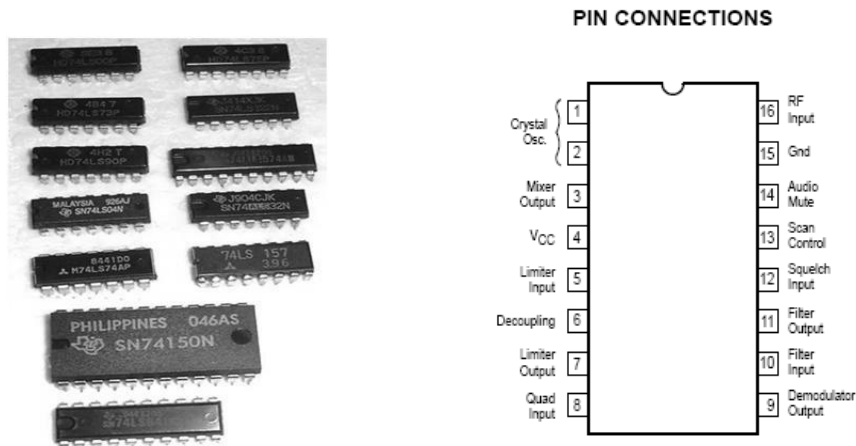
ขาเบส (Base; B) เป็นขาที่เราใช้ในการกระตุ้นให้ทรานซิสเตอร์ทำงานโดยการจ่ายกระแส (I_{base}) ให้กับทรานซิสเตอร์ที่ขาเบส P-N-P ทำงานด้วยโฟลบ (กระแสไหลออก) และ N-P-N ทำงานด้วยไฟบวก (กระแสไหลเข้า)

ขาคอลเล็กเตอร์ (Collector; C) และ ขาอิมิตเตอร์ (Emitter; E) เป็นขาที่มีการไหลของกระแสในทิศทางเดียว เมื่อขาเบสถูกกระตุ้น P-N-P กระแสไหลจากขาอิมิตเตอร์ไปขาคอลเล็กเตอร์ N-P-N กระแสไหลจากขาคอลเล็กเตอร์ไปขาอิมิตเตอร์



ไอซี (Integrated Circuit, IC)

IC หรือวงจรรวม (Integrated Circuit) หมายถึง วงจรที่นำเอาไดโอด, ทรานซิสเตอร์, ตัวต้านทาน, ตัวเก็บประจุ และองค์ประกอบวงจรต่าง ๆ มาประกอบรวมกันบนแผ่นวงจรขนาดเล็ก ในปัจจุบันแผ่นวงจรนี้จะทำด้วยแผ่นซิลิคอน บางที่อาจเรียก ชิป (Chip) อุปกรณ์ทั้งหมดนี้ เราเรียกมันว่า IC ซึ่งส่วนประกอบสำคัญในอุปกรณ์อิเล็กทรอนิกส์เกือบทุกชนิด

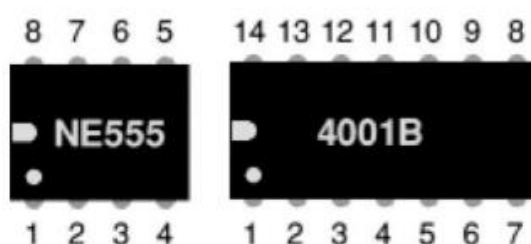


ส่วนประเภทของ IC สามารถแบ่งออกได้เป็น 2 ประเภทใหญ่ ๆ คือ

1. **Analog IC หรือ Linear IC** ใช้ขยายแรงเคลื่อนไฟฟ้าที่ไม่คงที่ ซึ่ง IC ชนิดนี้มีหลายชนิด เช่น เครื่องขยาย เครื่องผลิตความถี่ (Oscillator) เครื่องปรับแรงเคลื่อนไฟฟ้า (Regulator) วงจรสื่อสาร ตลอดจนวงจรที่ใช้ในวิทยุโทรทัศน์ทั่วไป

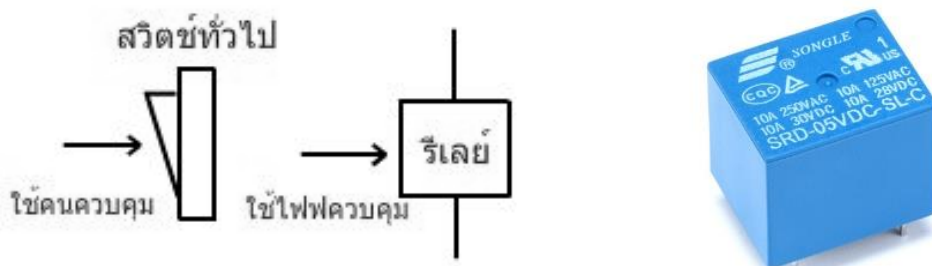
2. **Digital IC หรือ Logic IC** เป็นวงจรที่กำเนิดสัญญาณเพียงแรงเคลื่อนไฟฟ้า 2 ระดับ IC ชนิดนี้ส่วนมากจะเป็น Microprocessor, หน่วยความจำ (Memory), Microcontroller และ Chip อื่น ๆ อีกมากมาย

การนับขาของ IC มีความจำเป็นตรงที่เราจะต้องรู้ว่าขาที่เท่าไร เอาไว้ทำอะไร โดยวิธีการนับขานั้น เราจะดูจากรอยบาก หรือร่องของ IC แล้วเราจะให้ขาซ้ายบนสุดด้านล่างของด้านที่มีร่องอยู่เป็นขาที่ 1 และขาที่ 2 ไต่กันลงมาในทิศวนเข็มนาฬิกา



รีเลย์ (Relay)

รีเลย์ เป็นอุปกรณ์ที่เปลี่ยนพลังงานไฟฟ้าให้เป็นพลังงานแม่เหล็ก เพื่อใช้ในการดึงดูดหน้าสัมผัสของคอนแทคให้เปลี่ยนสถานะ โดยการป้อนกระแสไฟฟ้าให้กับขดลวด เพื่อทำการปิดหรือเปิดหน้าสัมผัสคล้ายกับสวิตช์อิเล็กทรอนิกส์

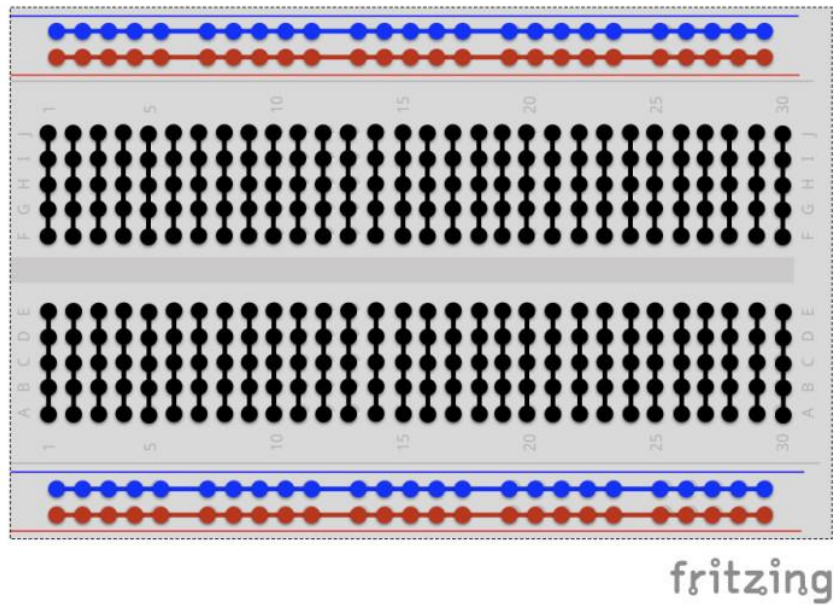


หลักการทำงานของรีเลย์

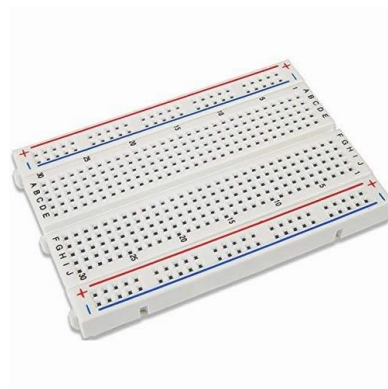
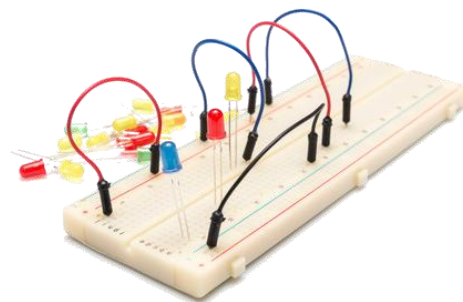
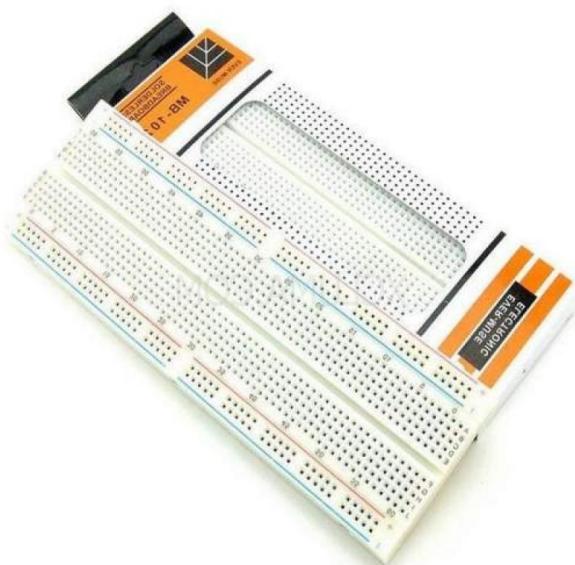
เมื่อไม่จ่ายกระแสไฟฟ้าที่ขดลวดรีเลย์ทำให้จุดร่วมต่ออยู่กับหน้าสัมผัสแบบปกติปิด (NC) แต่เมื่อทำการจ่ายกระแสไฟฟ้าให้กับขดลวดรีเลย์ (Coil) จะทำให้เกิดสนามแม่เหล็กรอบขดลวด ซึ่งอำนาจแม่เหล็กชั่วคราวที่เกิดขึ้นมีค่าเพียงพอที่จะชนะแรงสปริง เกิดการเปลี่ยนแปลงที่หน้าสัมผัสทำให้จุดร่วมต่ออยู่กับหน้าสัมผัสแบบปกติเปิด (NO) และเมื่อไม่ได้จ่ายกระแสไฟฟ้า สปริงจะดึงหน้าสัมผัสกลับมาต่อกับหน้าสัมผัสแบบปกติปิด (NC) อีกครั้ง



แผงต่อวงจรหรือโปรโตบอร์ด (Protoboard)



เป็นแผงพลาสติกที่มีการจัดแบ่งกลุ่มโดยภายในแต่ละกลุ่มบรรจุแผงโลหะตัวนำปลอดสนิม แล้วทำการเจาะรูบนพลาสติกนั้น เพื่อให้สามารถนำสายไฟขนาดเล็กเสียบเข้าไปสัมผัสกับแผงโลหะ ในขณะเดียวกันแผงโลหะดังกล่าวก็จะทำการบีบสายไฟนั้นให้แน่นอยู่กับที่ เมื่อผู้ใช้งานต้องการปลดสายไฟออกก็ออกแรงดึงเพียงเล็กน้อย หน้าสัมผัสของแผงโลหะก็จะคลายออก ทำให้สายไฟฟ้าสามารถหลุดออกจากจุดต่อนั้นได้



เครื่องมือในการวัดทางไฟฟ้าและการทดลอง

ชนิดของเครื่องมือวัดไฟฟ้า เครื่องมือวัดไฟฟ้านี้เราสามารถแบ่งชนิดตามลักษณะของการทำงาน ได้ เป็น 2 ชนิดคือ

1. เครื่องมือวัดไฟฟ้าที่ใช้หลักการของเข็มเคลื่อนที่บนสเกลของหน้าปัด เรียกว่า เครื่องมือวัดไฟฟ้าแบบอนาล็อก
2. เครื่องมือวัดไฟฟ้าที่แสดงผลของการวัดออกมาเป็นตัวเลขเรียกว่า เครื่องมือวัดไฟฟ้าแบบดิจิตอล

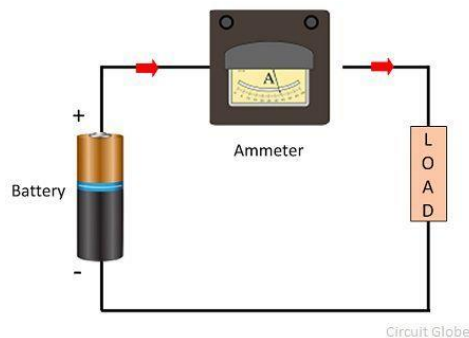


ดิจิตอลมัลติมิเตอร์



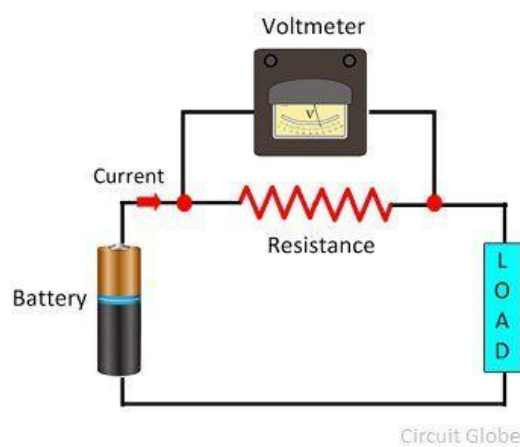
อนาล็อกมัลติมิเตอร์

การวัดกระแส



นำมัลติมิเตอร์ ไปอนุกรมกับอุปกรณ์ที่วัด

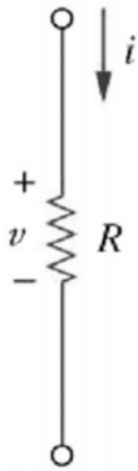
การวัดความต่างศักย์



นำมัลติมิเตอร์ ไปคร่อมขนานกับอุปกรณ์ที่จะวัด

ทฤษฎีทางไฟฟ้า

กฎของโอห์ม(Ohm's Law)

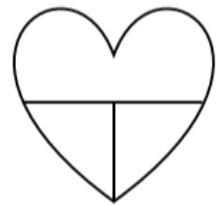


V = แรงดันไฟฟ้า หรือ ค่าความต่างศักย์ระหว่างจุด 2 จุด มีหน่วย โวลต์ (V)
 I = กระแสไฟฟ้า มีหน่วย แอมแปร์ (A)
 R = ค่าความต้านทาน มีหน่วย โอห์ม (Ω)

กระแสไฟฟ้าจะแปรผันตรงกับแรงดันไฟฟ้า $I \propto V$

กระแสไฟฟ้าจะแปรผกผันกับค่าความต้านทาน $I \propto \frac{1}{R}$

$$V = IR$$

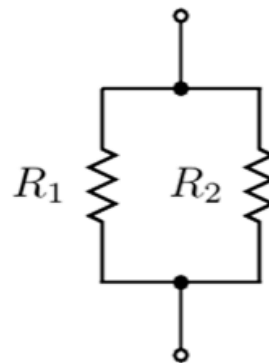


การหาค่าความต้านทานรวม

แบบอนุกรม

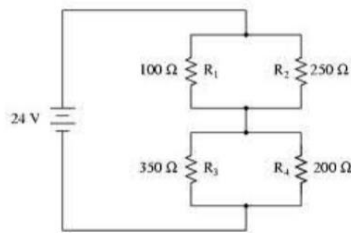


แบบขนาน

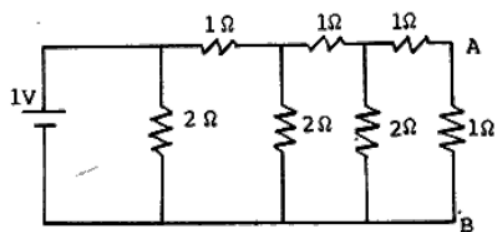


แบบฝึกหัด

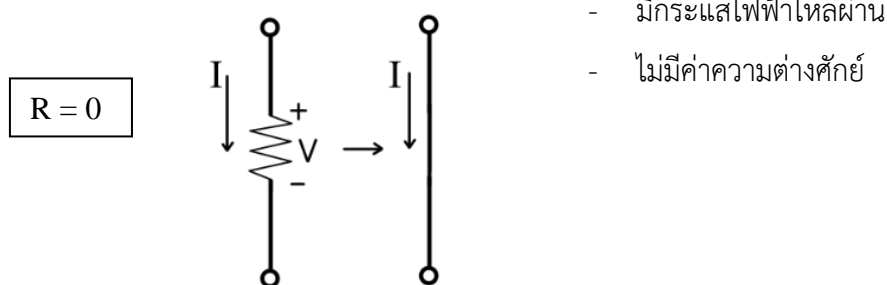
1. จงคำนวณหากระแสรวมของวงจร กระแสที่ไหลผ่าน R_2



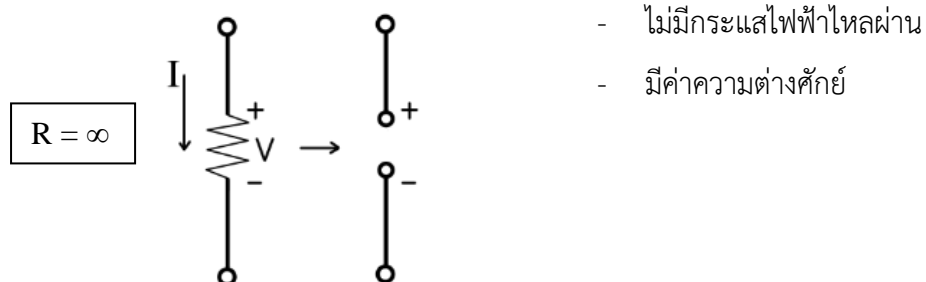
2. หาความต่างศักย์ระหว่างจุด A และ B



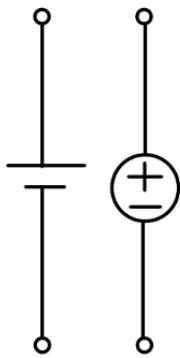
เมื่อ $R = 0$ จะเปรียบเป็น Short circuit



เมื่อ $R = \infty$ จะเปรียบเป็น Open circuit



ตัวจ่ายพลังงาน(Active Elements)



Voltage source



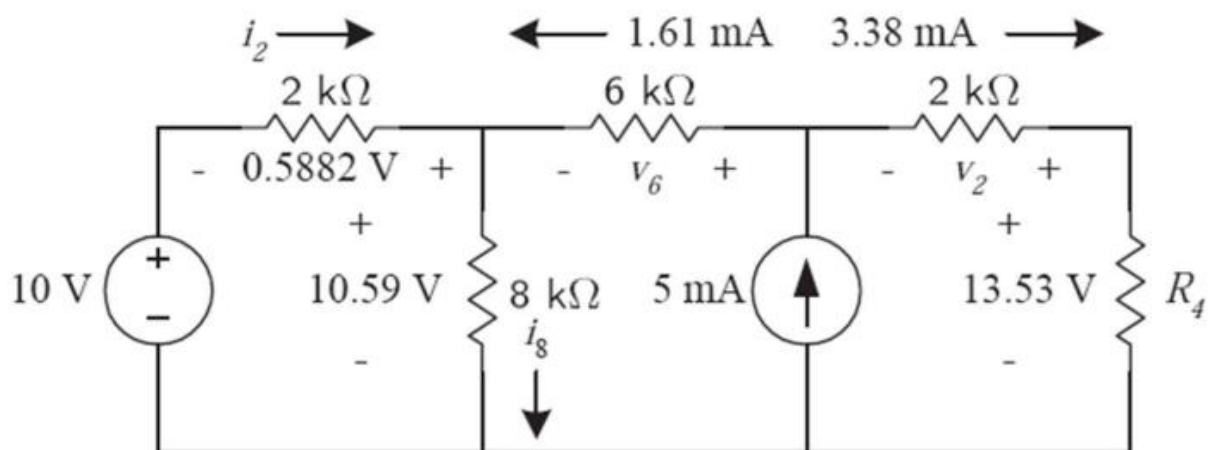
Current source

ตัวรับพลังงาน(Passive Elements)



Resistor

โจทย์



$$i_2 =$$

$$V_6 =$$

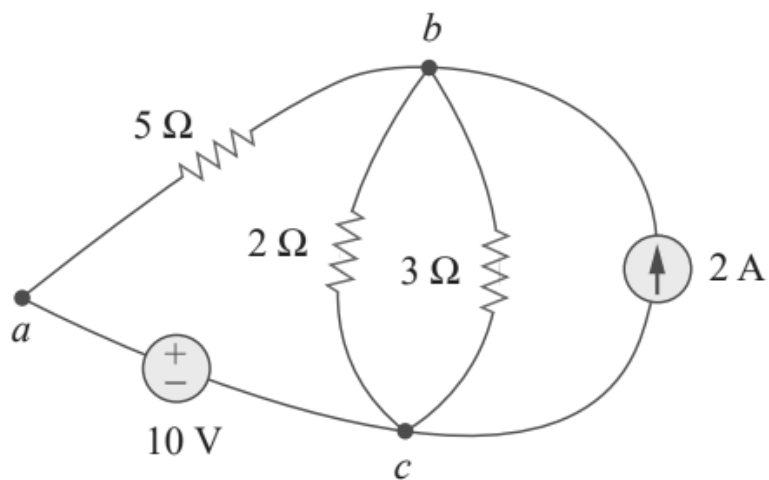
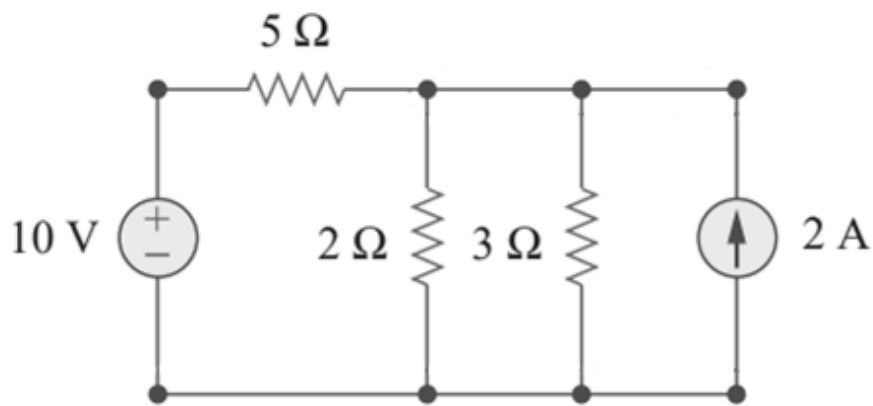
$$R_4 =$$

Node, Branches and loops

Branch คือ อุปกรณ์ไฟฟ้า เช่น ตัวต้านทาน ตัวจ่ายแรงดันไฟ เป็นต้น

Node คือ จุดร่วมระหว่าง Branch 2 ตัวขึ้นไป

Loop คือ วงจรปิดที่มีเส้นทางกระแสน้ำวนกลับมาที่เดิม



Branch =

Node =

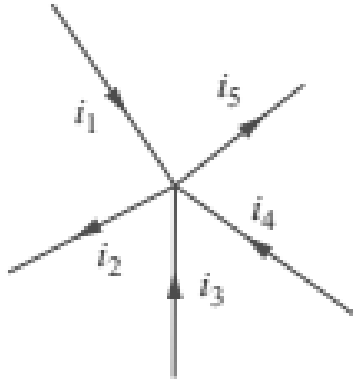
Loop =

กฎของเคอร์ชอฟฟ์ (Kirchhoff's Law)

Kirchhoff's Law แบ่งได้ 2 สูตร

1.Kirchhoff's current law (KCL)

ผลรวมกระแสที่ผ่าน ณ จุด node มีค่าเท่ากับ 0



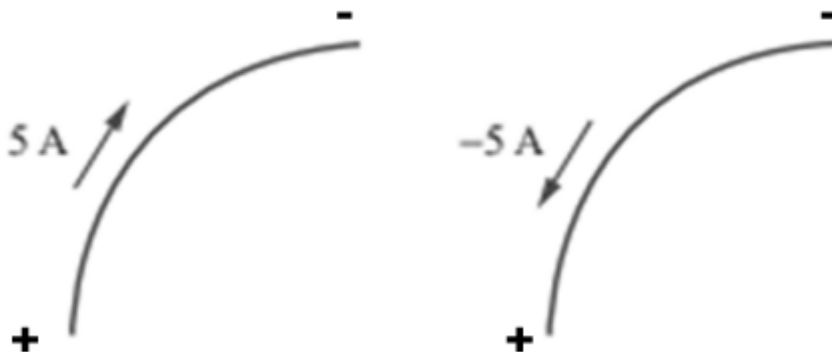
$$\sum_{n=1}^N i_n = 0$$

กล่าวได้ว่า กระแสที่ไหลเข้าจุด node มีค่าเท่ากับ กระแสที่ไหลออกจุด node

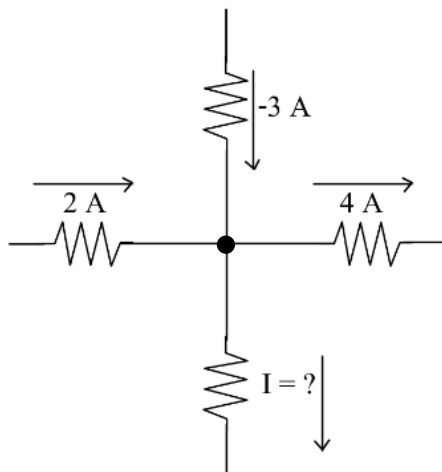
กระแสไหลเข้า = กระแสไหลออก

เพิ่มเติม

ทิศทางของกระแสไฟฟ้าจะไหลจาก ศักย์สูงไปศักย์ต่ำ

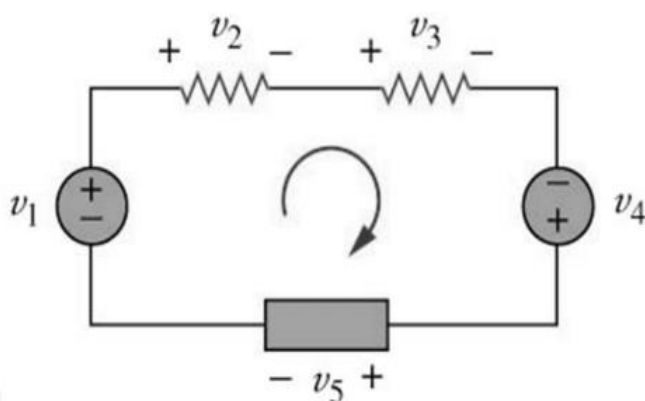


จงหาค่า I และมีทิศทางไหลเข้าหรือไหลออกจาก node



2. Kirchhoff's voltage law (KVL)

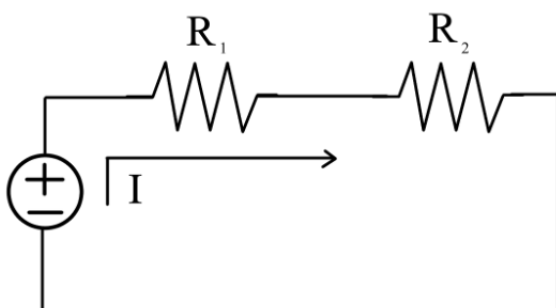
ใน 1 loop ผลรวมแรงดันในวงจรจะมีค่าเท่ากับ 0



$$V_2 + V_3 - V_4 + V_5 - V_1 = 0$$

$$\sum_{m=1}^M v_n = 0$$

การต่อวงจรอนุกรม

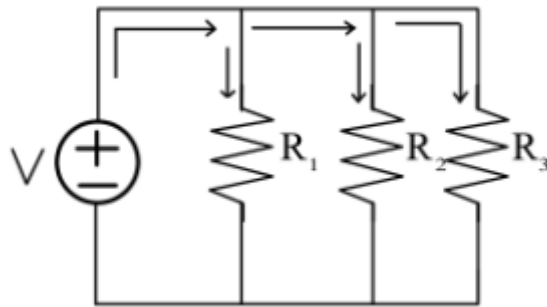


$$R_{\text{รวม}} = R_1 + R_2 + \dots + R_n$$

$$V_{\text{รวม}} = V_1 + V_2 + \dots + V_n$$

! จะมีค่าเท่ากันทั้งวงจร

การต่อวงจรขนาน



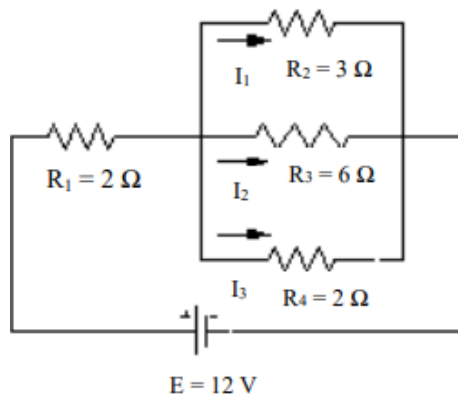
$$\frac{1}{R_{\text{รวม}}} = \frac{1}{R_1} + \frac{1}{R_2} + \dots + \frac{1}{R_n}$$

$$I_{\text{รวม}} = I_1 + I_2 + \dots + I_n$$

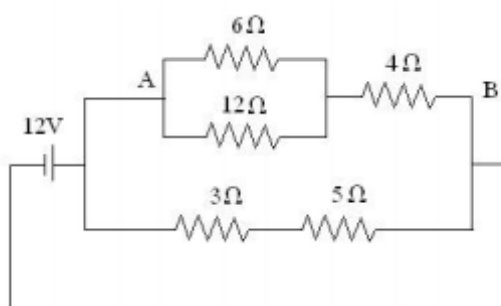
V จะมีค่าเท่ากันในวงจรขนาน

โจทย์

1. จงหากระแสรวมของวงจร กระแสที่ไหลผ่าน R2 และความต่างศักย์ที่ตกคร่อม R1



2. จงหากระแสรวมของวงจร กระแสที่ไหลผ่าน R = 4 Ω และความต่างศักย์ที่ตกคร่อม R = 6 Ω



การวิเคราะห์วงจรแบบโนด(Nodal Analysis)

เป็นการวิเคราะห์วงจรโดยใช้ แรงดัน ณ จุด Node เป็นตัวแปรเพื่อหาค่าต่างๆในวงจร

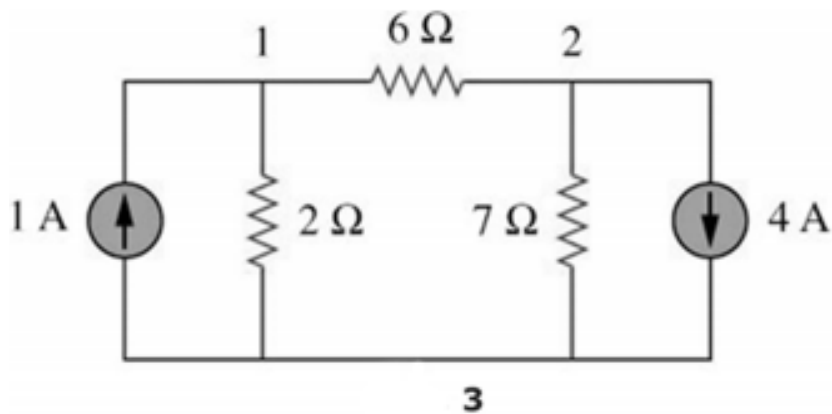
ขั้นตอนในการวิเคราะห์วงจรแบบโนด

- 1.กำหนดnodeอ้างอิง เป็น ground (มีศักย์เป็น 0) และ กำหนด nodeแรงดัน
- 2.พิจารณา nodeแรงดัน โดยใช้กฎ KCL เขียนสมการ เพื่อหาสมการผลรวมกระแส ณ จุด nodeแรงดัน จะได้สมการออกมา $N - 1$ สมการ เมื่อ $N =$ จำนวน Node ทั้งหมด

$$\text{กระแสจะไหลจากศักย์สูงไปศักย์ต่ำ} \quad I = \frac{V_{higher} - V_{lower}}{R}$$

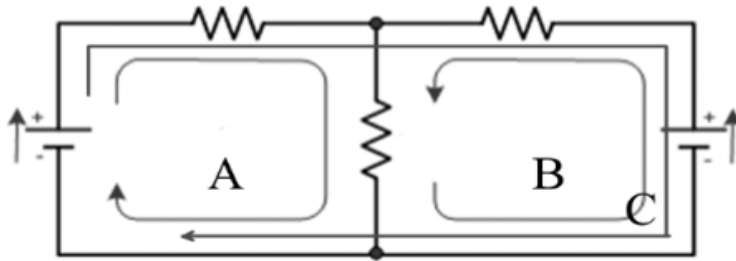
- 3.คำนวณเพื่อหาค่า แรงดัน ณ จุดโนดต่างๆ

หา V_1 และ V_2



การวิเคราะห์วงจรแบบเมช(Mesh Analysis)

เป็นการวิเคราะห์วงจรโดยใช้ กระแสที่ไหลภายในเมช เป็นตัวแปรเพื่อหาค่าต่างๆในวงจร
Mesh คือ ลูปที่ไม่มีลูปอยู่ภายใน

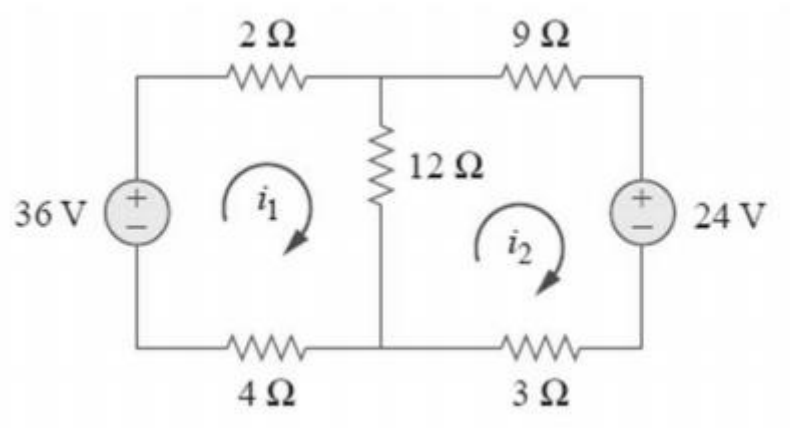


จากภาพ ลูป A และ ลูป B เป็นเมชเนื่องจากไม่มีลูปอื่นอยู่ข้างใน

ขั้นตอนในการวิเคราะห์วงจรแบบเมช

1. กำหนดกระแสไฟฟ้าในแต่ละเมช
2. ใช้กฎ KVL เพื่อหาสมการผลรวมของแรงดันแต่ละเมช
3. แก้สมการเพื่อหา กระแสไฟฟ้าที่ไหลในแต่ละเมช

จงหา I_1 และ I_2



ทฤษฎีดิจิทัล

ระบบตัวเลขที่เราใช้กันมาตลอดและคุ้นเคยกันดีประกอบด้วยตัวเลขทั้งหมด 10 ตัวคือเลข 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 เป็นเลขที่มนุษย์ใช้สื่อสาร บอกขนาด ปริมาณ เรียกว่า เลขฐานสิบ (Decimal Number) และต่อมา ด้วยเทคโนโลยีทางคอมพิวเตอร์ที่ก้าวหน้าขึ้น จึงเกิดการทำงานของคอมพิวเตอร์ที่มีระบบการทำงานที่เรียกว่า “ดิจิทัล” ซึ่งเป็นระบบที่ทำงานด้วยแรงดันไฟฟ้า 2 ค่า คือ (0 กับ 1) หรือ (ปิด กับ เปิด) เป็นต้น ระบบเลขฐานที่นิยมใช้ในระบบดิจิทัลมีอยู่ 4 ระบบ คือ

- ระบบเลขฐานสอง (Binary number system) ประกอบด้วยเลข 2 ตัว คือ 0 1
- ระบบเลขฐานแปด (Octal number system) ประกอบด้วยเลข 8 ตัว คือ 0 1 2 3 4 5 6 7
- ระบบเลขฐานสิบ (Decimal number system) ประกอบด้วยเลข 10 ตัว คือ 0 1 2 3 4 5 6 7

8 9

- ระบบเลขฐานสิบหก (Hexadecimal number system) ประกอบด้วยเลข 16 ตัว 0 1 2 3 4 5 6 7 8 9 A B C D E F เมื่อ A = 10, B = 11, C = 12, D = 13, E = 14, F = 15

การแปลงเลขฐาน

การแปลงเลขฐาน 2 เป็นเลขฐาน 10

แปลงเลขฐาน 2 เป็น เลขฐาน 10 ทำได้โดยการ เพิ่มค่าให้กับตำแหน่งเลขทีละ 1 แล้วจับรวมกัน เช่น

$$\begin{array}{r} \text{Ex1.} \quad \quad \quad 1 \quad 1 \quad 0 \quad 1 \quad 1_2 \\ 2^4 + 2^3 + 0 + 2^1 + 2^0 = 16 + 8 + 2 + 1 \\ = 27_{10} \end{array}$$

$$\begin{array}{r} \text{Ex2.} \quad \quad \quad 1 \quad 1 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0_2 \\ 2^6 + 2^5 + 2^4 + 0 + 0 + 2^1 + 0 = 64 + 32 + 16 + 2 \\ = 114_{10} \end{array}$$

การแปลงเลขฐาน 10 เป็นเลขฐาน 2

$$\begin{array}{r} \text{Ex1.} \quad \quad \quad 39_{10} = 32 + 4 + 2 + 1 = 2^5 + 0 + 0 + 2^2 + 2^1 + 2^0 \\ = 1 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1 \end{array}$$

$$\begin{array}{r} \text{Ex2.} \quad \quad \quad 100_{10} = 64 + 32 + 4 = 2^6 + 2^5 + 0 + 0 + 2^2 + 0 + 0 \\ = 1 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \end{array}$$

การแปลงเลขฐาน 16 เป็นเลขฐาน 10

$$\begin{aligned}\text{Ex1.} \quad 3AE_{16} &= 3 \times 16^2 + 10 \times 16^1 + 14 \times 16^0 \\ &= 768 + 160 + 14 \\ &= 942_{10}\end{aligned}$$

การแปลงเลขฐาน 10 เป็นเลขฐาน 16

Convert 423_{10} to hex:

$$\begin{aligned}\text{Ex1.} \quad \frac{423}{16} &= 26 + \text{remainder of } 7 \\ \frac{26}{16} &= 1 + \text{remainder of } 10 \\ \frac{1}{16} &= 0 + \text{remainder of } 1\end{aligned}$$

$423_{10} = 1A7_{16}$

การแปลงเลขฐาน 16 เป็นเลขฐาน 2

$$\begin{aligned}\text{Ex1.} \quad AF2_{16} &= \begin{array}{ccc} & \text{A} & \text{F} & & 2 \\ & \downarrow & \downarrow & & \downarrow \\ & 1010 & 1111 & & 0010 \end{array} \\ &= 101011110010_2 \\ &= 101011110010_2\end{aligned}$$

การแปลงเลขฐาน 2 เป็นเลขฐาน 16

$$\begin{aligned}\text{Ex1.} \quad 1110100110_2 &= \underbrace{0011}_3 \underbrace{1010}_A \underbrace{0110}_6 \\ &= 3A6_{16}\end{aligned}$$

การแปลงเลขฐาน 2 เป็นเลขฐาน 8

$$\begin{aligned}\text{Ex1.} \quad 1110100110_2 &= \underbrace{001}_1 \underbrace{110}_6 \underbrace{100}_4 \underbrace{110}_6 \\ &= 1646_8\end{aligned}$$

วงจรลอจิก (Logic Circuit)

วงจรเกตเป็นวงจรขนาดเล็กที่สุดของวงจรดิจิทัลและคอมพิวเตอร์ ซึ่งวงจรเกตพื้นฐานมีอยู่ 3 ชนิด คือ AND Gate, OR Gate และ NOT Gate (Inverter)

ในวงจรลอจิกจะแทนด้วยตัวแปร 2 สถานะ คือ 0 และ 1

- **Logic 0:** false, off, low, no, open switch.
- **Logic 1:** true, on, high, yes, closed switch.

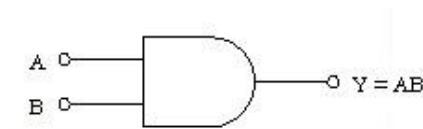
Logic 0	Logic 1
False	True
Off	On
LOW	HIGH
No	Yes
Open switch	Closed switch

การแสดงค่า Logic Gate สามารถแสดงได้ด้วย “ตารางแสดงค่าความจริง” (Truth Table) ซึ่งแสดงค่าความสัมพันธ์ของ Input และ Output ผ่าน Logic

จากการประยุกต์ใช้วงจรเกตพื้นฐาน ทำให้สามารถสร้างเกตใหม่ได้หลายชนิด และใช้งานได้ซับซ้อนมากขึ้น เช่น NAND Gate, NOR Gate, XOR (Exclusive OR Gate), XNOR (Exclusive NOR Gate) และ Buffer Gate เป็นต้น

1) AND Gate

สัญลักษณ์



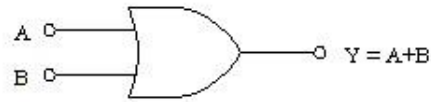
สมการพีชคณิตลอจิก $Y = f(A, B) = A \cdot B$

ตารางค่าความจริง

Input		Output
A	B	$f(A, B) = A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

2) OR Gate

สัญลักษณ์



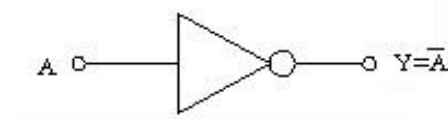
สมการพีชคณิตลอจิก $Y = f(A, B) = A + B$

ตารางค่าความจริง

Input		Output
A	B	$f(A, B) = A + B$
0	0	0
0	1	1
1	0	1
1	1	1

3) NOT Gate

สัญลักษณ์



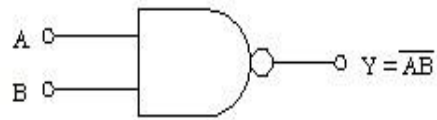
สมการพีชคณิตลอจิก $Y = f(A) = \bar{A}$

ตารางค่าความจริง

Input	Output
A	$f(A) = \bar{A}$
0	1
1	0

4) NAND Gate

สัญลักษณ์



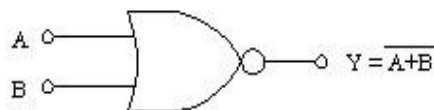
สมการพีชคณิตลอจิก $Y = f(A, B) = \overline{A \cdot B}$

ตารางค่าความจริง

Input		Output
A	B	$f(A, B) = \overline{A \cdot B}$
0	0	1
0	1	1
1	0	1
1	1	0

5) NOR Gate

สัญลักษณ์



สมการพีชคณิตลอจิก $Y = f(A, B) = \overline{A + B}$

ตารางค่าความจริง

Input		Output
A	B	$f(A, B) = \overline{A + B}$
0	0	1
0	1	0
1	0	0
1	1	0

6) XOR Gate (Exclusive OR Gate)

สัญลักษณ์



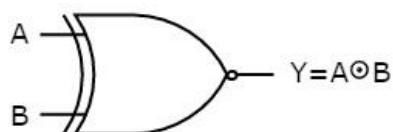
สมการพีชคณิตลอจิก $Y = f(A, B) = A \oplus B$

ตารางค่าความจริง

Input		Output
A	B	$f(A, B) = A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

7) XNOR Gate (Exclusive NOR Gate)

สัญลักษณ์

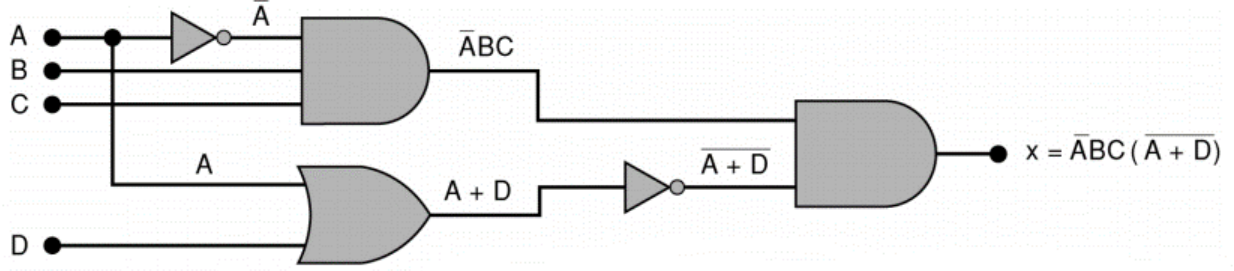


สมการพีชคณิตลอจิก $Y = f(A, B) = \overline{A \oplus B}$

ตารางค่าความจริง

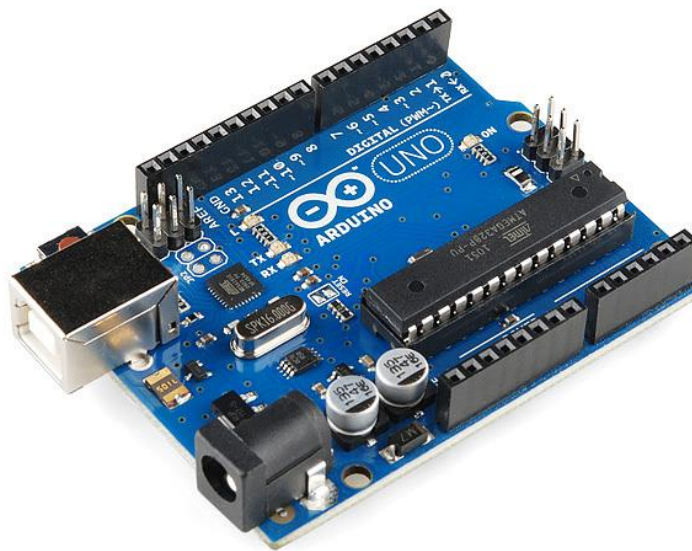
Input		Output
A	B	$f(A, B) = \overline{A \oplus B}$
0	0	1
0	1	0
1	0	0
1	1	1

จงหาค่า Input ที่ทำให้ Output มีค่าเป็น 1



Pre-Lab 64

การใช้งาน Arduino เบื้องต้น



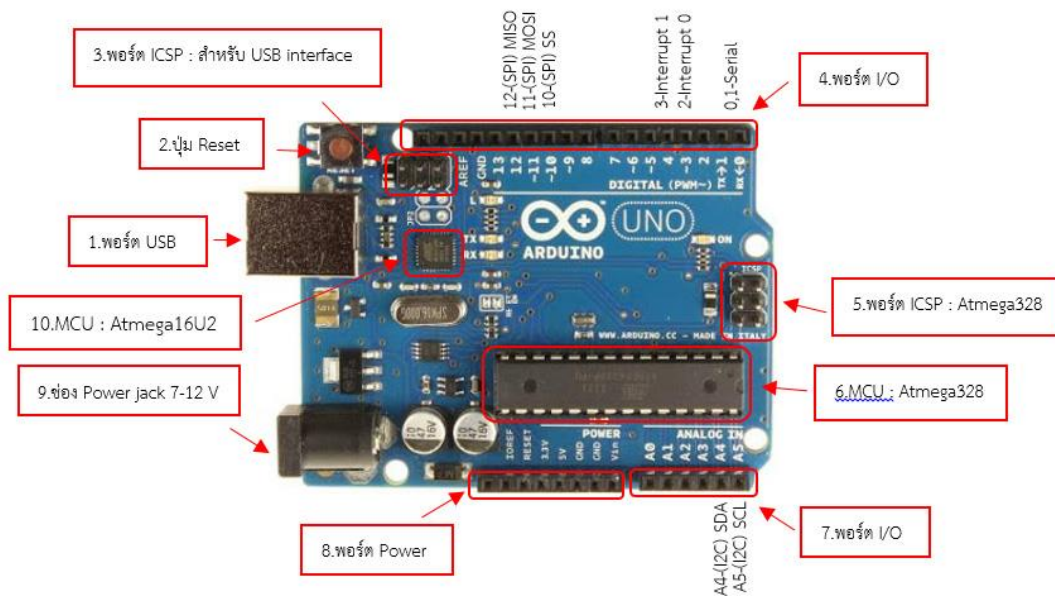
Electronics Club

School of Engineering

King Mongkut's Institute of Technology Ladkrabang

Arduino อ่านว่า (อา-คฺ-อุ-อิ-โน้ หรือ อาคฺยุโน้) เป็นบอร์ดไมโครคอนโทรเลอร์ตระกูล AVR ที่มีการพัฒนาแบบ Open Source คือมีการเปิดเผยข้อมูลทั้งด้าน Hardware และ Software ตัว บอร์ด Arduino ถูกออกแบบมา ให้ใช้งานได้ง่าย ดังนั้นจึงเหมาะสำหรับผู้เริ่มต้นศึกษา ทั้งนี้ผู้ใช้งานยังสามารถดัดแปลง เพิ่มเติมพัฒนาต่อยอดทั้งตัวบอร์ด หรือโปรแกรมต่อได้อีกด้วย

Layout & Pin out Arduino Board (Model: Arduino UNO R3)



(<https://www.mylarduino.net>)

1. USB Port: ใช้สำหรับต่อกับ Computer เพื่ออัปโหลดโปรแกรมเข้า MCU และจ่ายไฟให้กับบอร์ด
2. Reset Button: เป็นปุ่ม Reset ใช้กดเมื่อต้องการให้ MCU เริ่มการทำงานใหม่
3. ICSP Port ของ Atmega16U2 เป็นพอร์ตที่ใช้โปรแกรม Visual Com port บน Atmega16U2
4. I/O Port: Digital I/O ตั้งแต่ขา D0 ถึง D13 นอกจากนี้ บาง Pin จะทำหน้าที่อื่นๆ เพิ่มเติมด้วย เช่น Pin0,1 เป็นขา Tx,Rx Serial, Pin3,5,6,9,10 และ 11 เป็นขา PWM
5. ICSP Port: Atmega328 เป็นพอร์ตที่ใช้โปรแกรม Bootloader
6. MCU: Atmega328 เป็น MCU ที่ใช้บนบอร์ด Arduino
7. I/O Port: นอกจากจะเป็น Digital I/O แล้ว ยังเปลี่ยนเป็น ช่องรับสัญญาณอนาล็อก ตั้งแต่ขา A0-A5
8. Power Port: ไฟเลี้ยงของบอร์ดเมื่อต้องการจ่ายไฟให้กับวงจรภายนอก ประกอบด้วยขาไฟเลี้ยง +3.3 V, +5V, GND, Vin
9. Power Jack: รับไฟจาก Adapter โดยที่แรงดันอยู่ระหว่าง 7-12 V
10. MCU ของ Atmega16U2 เป็น MCU ที่ทำหน้าที่เป็น USB to Serial โดย Atmega328 จะติดต่อกับ Computer ผ่าน Atmega16U2

โครงสร้างพื้นฐานของภาษาซีที่ใช้กับ Arduino นั้นจะประกอบไปด้วย 3 ส่วนใหญ่ด้วยกัน คือ

1. Header ในส่วนนี้จะมีหรือไม่มีก็ได้ ถ้ามีต้องกำหนดไว้ในส่วนเริ่มต้นของโปรแกรม ซึ่งส่วนของ Header ได้แก่ ส่วนที่เป็น Compiler Directive ต่างๆรวมไปถึงส่วนของการประกาศตัวแปร และค่าคงที่ต่างๆที่จะใช้ในโปรแกรม
2. setup() ในส่วนนี้เป็นฟังก์ชันบังคับที่ต้องกำหนดให้มีในทุกๆโปรแกรม ถึงแม้ว่าในบางโปรแกรมจะไม่ต้องการใช้งานก็ยังจำเป็นต้องประกาศไว้ด้วยเสมอ เพียงแต่ไม่ต้องเขียนคำสั่งใดๆไว้ในระหว่างวงเล็บปีกกา {} ที่ใช้เป็นตัวกำหนดของเขตของฟังก์ชัน โดยฟังก์ชันนี้จะใช้สำหรับบรรจุคำสั่งในส่วนที่ต้องการให้โปรแกรมทำงานเพียงรอบเดียวตอนเริ่มต้นทำงานของโปรแกรมครั้งแรกเท่านั้น ซึ่งได้แก่คำสั่งเกี่ยวกับการ Setup ค่าการทำงานต่างๆ เช่น การกำหนดหน้าที่การใช้งานของ PinMode และการกำหนดค่า Baudrate สำหรับใช้งานพอร์ตสื่อสารอนุกรม เป็นต้น
3. loop() เป็นส่วนฟังก์ชันบังคับที่ต้องกำหนดให้มีในทุกๆโปรแกรมเช่นเดียวกันกับฟังก์ชัน setup() โดยฟังก์ชัน loop() นี้จะใช้บรรจุคำสั่งที่ต้องการให้โปรแกรมทำงานเป็นวงรอบซ้ำๆกันไปไม่รู้จบ ซึ่งถ้าเปรียบเทียบกับรูปแบบของ ANSI-C ส่วนนี้ก็คือ ฟังก์ชัน main() นั่นเอง

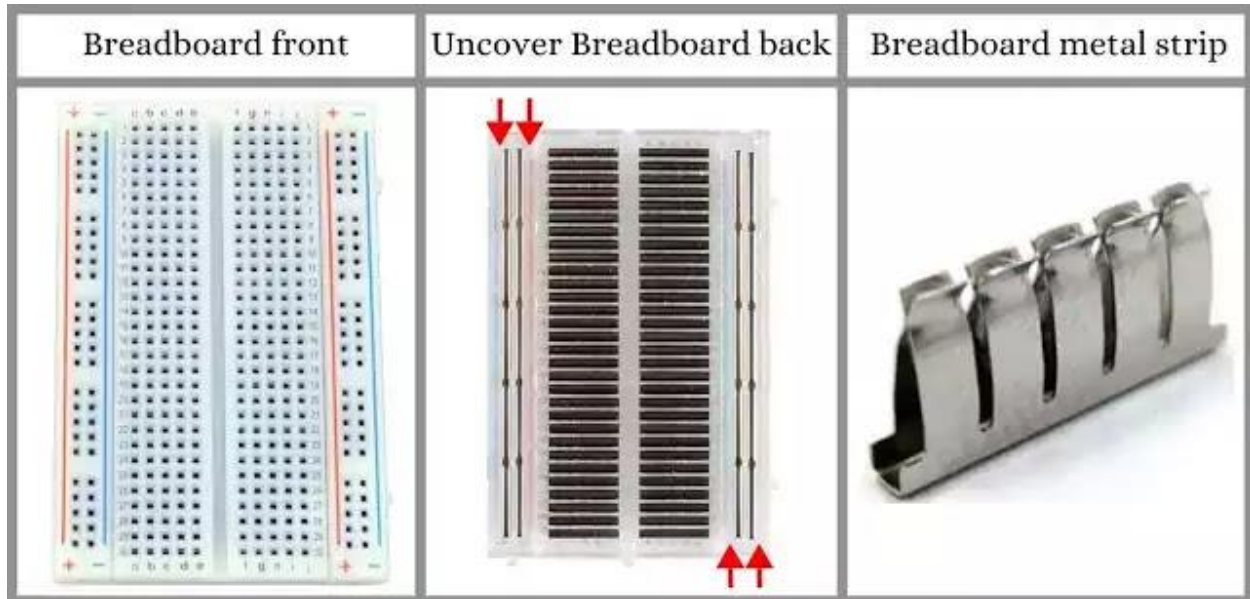
คำสั่งเบื้องต้นในการใช้งาน Arduino

1. pinMode(pin, mode) ใช้กำหนดทิศทางสัญญาณ (I/O direction) ของขาดิจิตัล
2. digitalWrite(pin) ใช้อ่านค่าจากขาดิจิตัลที่ถูกกำหนดให้เป็นอินพุต
3. digitalWrite(pin, logic) ใช้เขียนค่า (LOW/HIGH) ให้ขาดิจิตัลที่ถูกกำหนดให้เป็นเอาต์พุต
4. analogWrite(pin, duty) ใช้สัญญาณ PWM เป็นเอาต์พุต
5. analogRead(pin) ใช้อ่านค่าจากขาแอนะล็อก-อินพุต
6. analogReference(Type) กำหนดระดับแรงดันอ้างอิงสำหรับการอ่านค่าจากขาแอนะล็อก-อินพุต
7. delay(time) รอเวลาให้ผ่านไปตามระยะเวลาที่กำหนด (มิลลิวินาที) ก่อนที่จะทำขั้นตอนต่อไป
8. millis() บอกเวลาที่ผ่านไปหน่วยเป็นมิลลิวินาที นับตั้งแต่โปรแกรมเริ่มต้นทำงาน

อุปกรณ์

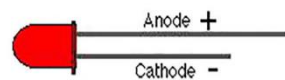
-breadboard หรือเรียกเป็นภาษาไทยว่า แผงต่อวงจร

ภายในรูมีตัวนำไฟฟ้าซึ่งเชื่อมต่อกันในรูปแบบที่มีการกำหนดไว้ เวลาทดลองก็เสียบขาของอุปกรณ์อิเล็กทรอนิกส์ลงไปให้ตัวนำภายในเชื่อมวงจรถึงกัน และอาจใช้สายไฟเสียบลงรูเพื่อเชื่อมวงจรไฟฟ้าได้เช่นกัน ข้อดีของโปรโตบอร์ดคือ ไม่ต้องออกแบบแผงวงจรและไม่ต้องบัดกรี แต่มีข้อเสียคือใช้ทดลองวงจรที่ทำงานที่มีความถี่สูงๆไม่ได้เนื่องมีปัญหาเรื่องสัญญาณรบกวนในวงจร



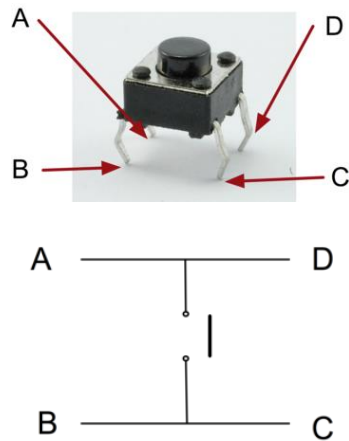
-ไดโอดเปล่งแสง (Light Emitting Diodes; LED)

คุณสมบัติจะเหมือนไดโอดธรรมดาทั่วไป จะแตกต่างจากไดโอดธรรมดาตรงที่เมื่อเราจ่ายไฟให้แล้วมันจะเปล่งแสงออกมาความสว่างและสีของแสงจะแตกต่างกันไปแล้วแต่วิธีการที่ใช้ทำ



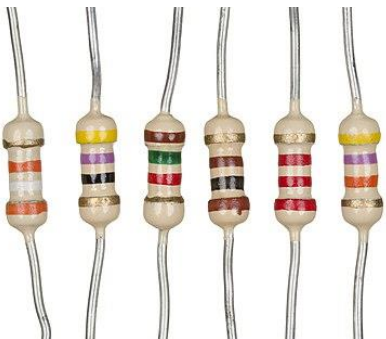
-Push Button Switch หรือ สวิตช์ปุ่มกด

เป็นอุปกรณ์ทางไฟฟ้า ซึ่งทำหน้าที่ตัดและต่อวงจรทางไฟฟ้าและ ใช้ในการควบคุมการทำงานของมอเตอร์ หรือการทำงานของเครื่องจักรต่างๆ เป็นเหมือนอุปกรณ์พื้นฐาน ใช้ได้กับอุตสาหกรรมทั่วไป มีทั้งแบบมีไฟ และทึบแสง



-ตัวต้านทาน (Resistor)

เป็นอุปกรณ์ Electronic พื้นฐานของวงจรไฟฟ้า ทำหน้าที่เป็นตัวต้านทานการไหลของกระแสไฟฟ้า ในวงจรไฟฟ้า โดยส่วนมากจะสร้างขึ้นจากโลหะผสมต่าง ๆ เช่น คาร์บอน เซรามิก เป็นต้น มีหน่วยเป็น โอห์ม สัญลักษณ์ Ω (โอเมก้า)



RESISTOR COLOR CODE GUIDE

4- Band Code

1.0 K Ω \pm 5%

1st 2nd 3rd 4th

Color	1st Band	2nd Band	3rd Band	Decimal Multiplier	Tolerance
Black	0	0	0	1	
Brown	1	1	1	10	\pm 1 %
Red	2	2	2	100	\pm 2 %
Orange	3	3	3	1K	
Yellow	4	4	4	10K	
Green	5	5	5	100K	
Blue	6	6	6	1M	
Violet	7	7	7	10M	
Gray	8	8	8	100,000,000	
White	9	9	9	1,000,000,000	
Gold				0.1	\pm 5 %
Silver				0.01	\pm 10 %
None					\pm 20 %

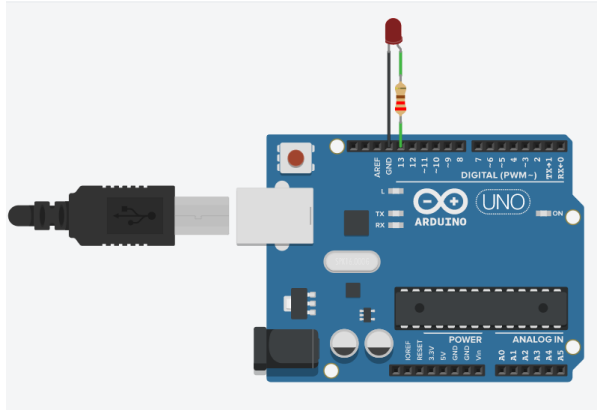
1st 2nd 3rd 4th 5th

5- Band Code

254 Ω \pm 1 %

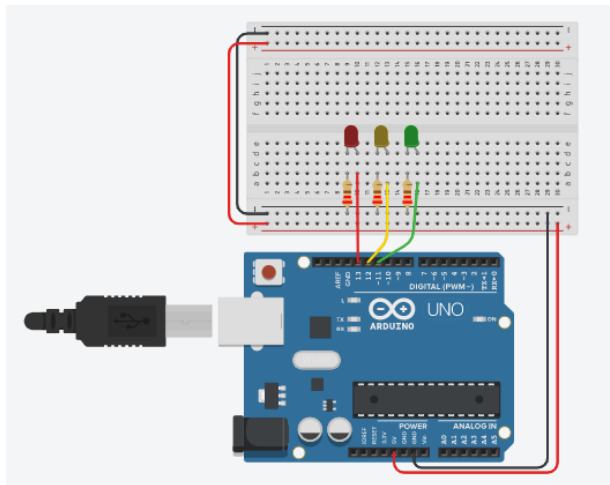
ตัวอย่างการใช้งาน Arduino บน www.tinkercad.com

1.การเขียนโปรแกรมไฟกระพริบ



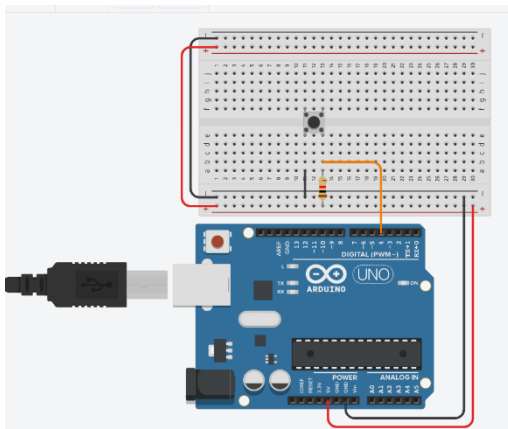
```
1 // C++ code
2 //
3 /*
4  This program blinks pin 13 of the Arduino (the
5  built-in LED)
6  */
7
8 void setup()
9 {
10  pinMode(LED_BUILTIN, OUTPUT);
11 }
12
13 void loop()
14 {
15  // turn the LED on (HIGH is the voltage level)
16  digitalWrite(LED_BUILTIN, HIGH);
17  delay(1000); // Wait for 1000 millisecond(s)
18  // turn the LED off by making the voltage LOW
19  digitalWrite(LED_BUILTIN, LOW);
20  delay(1000); // Wait for 1000 millisecond(s)
21 }
```

2.การเขียนโปรแกรมไฟกระพริบเรียงกัน

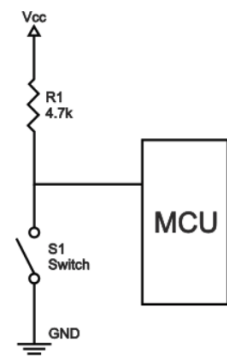


```
1 // C++ code
2 //
3 int animationSpeed = 0;
4
5 void setup()
6 {
7  pinMode(LED_BUILTIN, OUTPUT);
8  pinMode(12, OUTPUT);
9  pinMode(11, OUTPUT);
10 }
11
12 void loop()
13 {
14  animationSpeed = 400;
15  digitalWrite(LED_BUILTIN, HIGH);
16  delay(animationSpeed); // Wait for animationSpeed millisecond(s)
17  digitalWrite(LED_BUILTIN, LOW);
18  delay(animationSpeed); // Wait for animationSpeed millisecond(s)
19  digitalWrite(12, HIGH);
20  delay(animationSpeed); // Wait for animationSpeed millisecond(s)
21  digitalWrite(12, LOW);
22  delay(animationSpeed); // Wait for animationSpeed millisecond(s)
23  digitalWrite(11, HIGH);
24  delay(animationSpeed); // Wait for animationSpeed millisecond(s)
25  digitalWrite(11, LOW);
26  delay(animationSpeed); // Wait for animationSpeed millisecond(s)
27 }
```

3.การต่อสวิตช์แบบ Pull Up

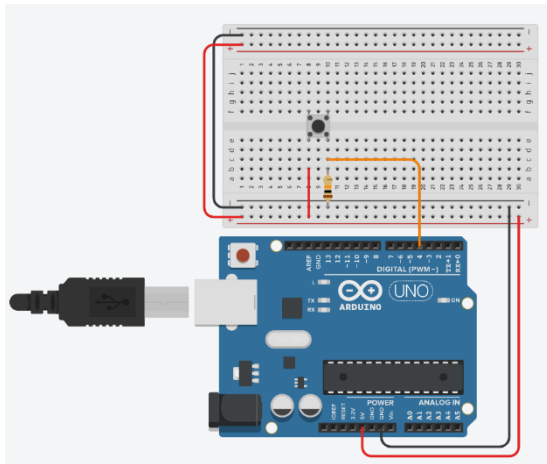


```
1 //Pull Up
2 int SW_1 = 4;
3 int status;
4 void setup()
5 {
6  Serial.begin(9600);
7  pinMode(SW_1, INPUT);
8 }
9
10 void loop()
11 {
12  status = digitalRead(SW_1);
13  Serial.println(status);
14  delay(500);
15 }
```

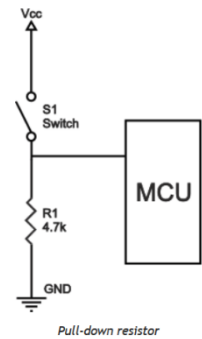


Pull-up resistor circuit

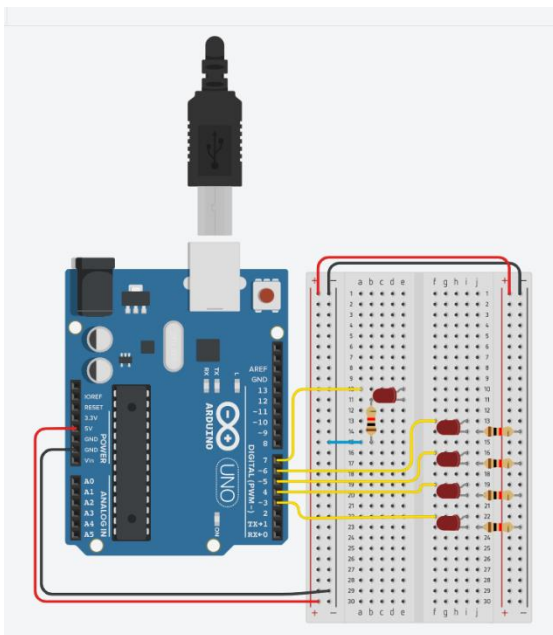
4.การต่อสวิตช์แบบ Pull Down



```
1 //Pull Down
2 int SW_1 = 4;
3 int status;
4 void setup()
5 {
6   Serial.begin(9600);
7   pinMode(SW_1, INPUT);
8 }
9
10 void loop()
11 {
12   status = digitalRead(SW_1);
13   Serial.println(status);
14   delay(500);
15 }
```



5.การวนลูป



```
1 int timer = 200; // The higher the number, the slower the timing.
2
3 void setup() {
4   // use a for loop to initialize each pin as an output:
5   for (int thisPin = 3; thisPin < 8; thisPin++) {
6     pinMode(thisPin, OUTPUT);
7   }
8 }
9
10 void loop() {
11   // loop from the lowest pin to the highest:
12   for (int thisPin = 3; thisPin < 8; thisPin++) {
13     digitalWrite(thisPin, HIGH); // turn the pin on:
14     delay(timer);
15     digitalWrite(thisPin, LOW); // turn the pin off:
16   }
17 }
```


เนื้อหาเพิ่มเติม

การใช้งานคำสั่ง for loop

คำสั่ง for loop เป็นคำสั่งวนซ้ำที่ใช้สำหรับควบคุมให้โปรแกรมทำงานซ้ำๆ ภายใต้เงื่อนไขที่กำหนดเช่นเดียวกับคำสั่ง while loop แต่มันถูกออกแบบมาสำหรับใช้กับการวนซ้ำในจำนวนรอบที่แน่นอน เนื่องจากเราสามารถกำหนดค่าเริ่มต้น เงื่อนไขและการเปลี่ยนแปลงค่าไว้ที่เดียวกัน

```
for (initial; condition; update) {  
    // statements  
}
```

- **initial:** เป็นส่วนที่ใช้กำหนดค่าเริ่มต้นให้กับลูปเพื่อทำงาน ทำงานครั้งแรกก่อนลูปเริ่มทำงาน
- **condition:** เป็นการกำหนดเงื่อนไขเพื่อให้ลูปทำงาน ซึ่งมันจะถูกตรวจสอบการลูปทำงานในแต่ละรอบ
- **update:** เป็นส่วนสำหรับอัปเดตค่าในตัวแปรและอื่นๆ ทำงานเมื่อลูปจบการทำงานในแต่ละรอบ

ตัวอย่างการใช้ for loop ในภาษา C

for_example.c

```
#include <stdio.h>  
  
int main()  
{  
    for (int i = 1; i <= 10; i++) {  
        printf("%d\n", i);  
    }  
    return 0;  
}
```



1
2
3
4
5
6
7
8
9
10

คำสั่ง break

ใช้สำหรับหยุดการทำงานของ loop ในทันทีโดยไม่ต้องให้เงื่อนไขเป็น False ก่อน มันมักจะใช้ในกรณีที่เราต้องการสร้างเงื่อนไขให้โปรแกรมออกจาก loop ที่นอกเหนือจากเงื่อนไขของ loop

ตัวอย่างการใช้ break ในภาษา C

break_example.c

```
#include <stdio.h>

int main()
{
    for (int i = 1; i <= 10; i++) {
        if (i == 6) {
            break;
        }
        printf("%d ", i);
    }
    return 0;
}
```



1 2 3 4 5