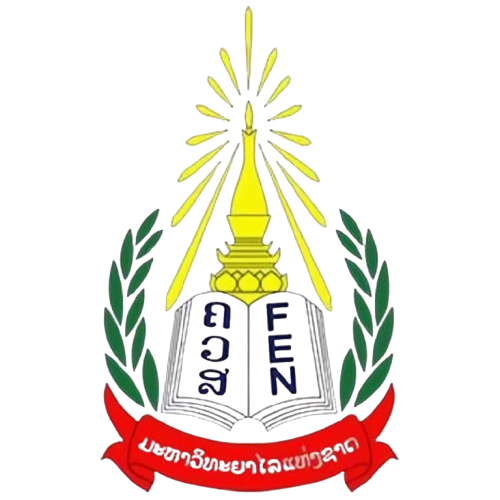


ສາທາລະນະລັດ ປະຊາທິປະໄຕ ປະຊາຊົນລາວ

ສັນຕິພາບ ເອກະລາດ ປະຊາທິປະໄຕ ເອກະພາບ ວັດທະນາຖາວອນ



ມະຫາວິທະຍາໄລແຫ່ງຊາດ

ຄະນະວິສະວະກຳສາດ

ພາກວິຊາວິສະວະກຳຄອມພິວເຕີສອນໂດຍ ປຕ ອຈ ລັດທິດາ ຄົມສອນລະສິນ

ຮຽບຮຽງ: ທ້າວ ພຸດສະຫວັນ ເເຈ້ງສະຫວ່າງ

Email: [nosaya2021rs@gmail.com](mailto:nosaya2021rs@gmail.com)

Number:205804516

ບົດສະຫຼຸບວິຊາໄມໂຄໂປຣເຊັດເຊີເເລະໄມໂຄຄອມພິວເຕີ.

ບົດທີ 1 Blink

1.Blink ເເມ່ນຫຍັງ:

A blue circuit board with black and white text

Description automatically generated

Blink ຄືໂປຣເຈັກຫຼືຊຸດຄຳສັ່ງທີ່ໃຊ້ໃນ (tinker cad) ເປັນ platform ການອອກເເບບ 3D ເເລະການຈຳລອງວົງຈອນ (Circuit simulation) ໂດຍໃນສ່ວນຂອງ Blink

ໝາຍເຖີງການເຮັດໄຟ led ກະພິບ (blinking led) ໂດຍໃຊ້ microcontroller ເຊັ່ນ Arduino ໃນການຄວມຄຸມການກະພິບຂອງ led ໃນ tinker cad

ເຮົາສາມາດສ້າງ ວົງຈອນນີ້ໄດ້ຢ່າງງ່າຍດາຍ ໂດຍການໃຊ້ເຄື່ອງມືຈຳລອງເພື່ອທົດສອບໂປຣເຈັກກ່ອນທີ່ຈະໄປໃຊ້ງານໂຕຈິງ

ສະຫຼຸບໂປຣເຈັກ blink ໃຊ້ຫຼັກການງ່າຍໆໃນການຂຽນ code ເຊັ່ນໃຫ້ led ຕິດເເລະດັບຕາມເວລາທີ່ກຳໜົດ

ເຊິ່ງເປັນໂປຣເຈັກເລີ່ມຕົ້ນທີ່ດີສຳຫຼັບຜູ້ທີ່ຮຽນຮູ້ການຂຽນໂປຣເເກຣມ microcontroller ເຊັ່ນ Arduino ຜ່ານ tinker cad.

* 1. Blink ການເຮັດວຽກ:

1.11: ການເຂົ້າສູ່ລະບົບ Tinkercad:

ເຂົ້າສູ່ລະບົບ tinker cad ເເລະເຂົ້າສູ່ລະບົບດ້ວຍບັນຊີ Autodesk ຂອງເຈົ້າຂອງເອງ

1.12: ສ້າງໂປຣເຈັກໃຫມ່:

ກົດທີ່ “create new design” ເພື່ອສ້າງໂປຣເຈັກ 3D ໃຫມ່ຫຼືເລືອກ “circuits” ເພື່ອທຳງານກັບວົງຈອນອິເລັກທຣໍນິກ .

1.13 ເພີ່ມ Arduino ເເລະສ່ວນປະກອບ:

- ໃນໂໝດ “circuits” ໃຫ້ລາກ Arduino (ເຊັ່ນ Arduino uno) ມາວາງໃນພື້ນທີ່ທຳງານ

- ເພີ່ມ led ເເລະຕົວຕ້ານທານ (resistor)

ລົງໃນວົງຈອນ ໂດຍຕໍ່ led ເຂົ້າກັບຂາ digital ຂອງ

Arduino (ເຊັ່ນ ຂາ 13) ເເລະຕໍ່ຕົວຕ້ານທານ

ເພື່ອຈຳກັດກະເເສໄຟຟ້າ.

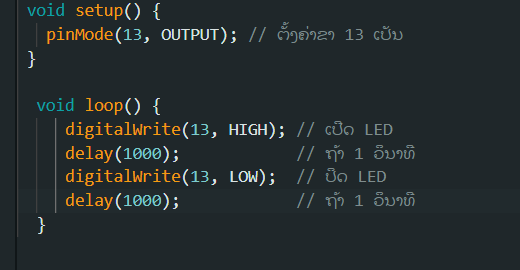
1.14 ຂຽນ code blink:

- ກົດທີ່ປຸ່ມ code ເພື່ອເປີດຫນ້າຕ່າງຂຽນ

Code

Tinker cad ມີຕົວຢ່າງ code ພື້ນຖານສຳຫຼັບການເຮັດໃຫ້ led ກະພິບ ເຊິ່ງເຮົາສາມາດໃຊ້ໄດ້ທັນຫຼືຈະຂຽນ code ເອງກະໄດ້.

ຕົວຢ່າງ code:



1.15 ຈຳລອງການເຮັດວຽກ:

- ຫຼັງຈາກຂຽນ code ເເລ້ວ ໃຫ້ຄຣິກປຸ່ມ

start simulation ເພື່ອເລີ່ມການຈຳລອງ

ເຮົາຈະເຫັນ led ທີ່ຕໍ່ກັບຂາ 13 ຂອງ Arduino ກະພິບຕາມທີ່ຕັ້ງຄ່າໄວ້ໃນ code

1.16 ປັບເເຕ່ງເເລະທົດສອບ:

ເຮົາສາມາດປັບປ່ຽນ code ຫຼືວົງຈອນເພື່ອທົດສອບການທຳງານທີ່ເເຕກຕ່າງອອກໄປເຊັ່ນປ່ຽນຄວາມໄວການກະພິບຫຼືໃຊ້ led ຫຼາຍຕົວ

ຕົວຢ່າງ Link in Tinkercad:

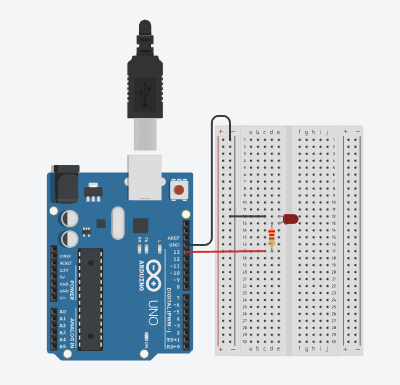
link:

https://www.tinkercad.com/things/0LWlH4WEsVC-0-blink?sharecode=S3q8Pjr8kyig00HoM01p\_wVuP3\_XJTq3WRuQ\_QlWQW4



ບົດທີ2 Experiment Blink

1. Experiment Blink ເເມ່ນຫຍັງ:



ຄືການທົດລອງພື້ນຖານທີ່ເຮັດໃຫ້ led ກະພິບ

ໂດຍໃຊ້ microcontroller ເຊັ່ນ Arduino,

raspberry pi ຫຼື board ອື່ນໆ ການທົດລອງນີ້

ມັກເປັນໜື່ງໃນຂັ້ນຕອນເເລກສຳຫຼັບຜູ້ເລີ່ມຕົ້ນຮຽນຮູ້

ກ່ຽວກັບການຂຽນໂປຣເເກຣມເເລະອິເລັກທຣໍນິກ.

1.1 ການເຮັດວຽກ:

ເປັນໂຄງການພື້ນຖານທີ່ໃຊ້ສຳຫຼັບການຮຽນຮູ້ການຂຽນໂປຮເເກຣມ microcontroller (Arduino) ເເລະການທຳງານຂອງວົງຈອນອິເລັກທຣໍນິກ ໂດຍມີຈຸດມຸ່ງໝາຍເພື່ອໃຫ້ led ກະພິບ (blink)

ຕາມໂປຣເເກຣມທີ່ຂຽນ ຕໍ່ໄປນີ້ເປັນຂັ້ນຕອນເເລະຫຼັກການທຳງານຂອງ Experiment Blink.

1.11 ອຸປະກອນ:

* Arduino: ໃຊ້ເປັນ microcontroller ສຳຫຼັບຄວບຄຸມການທຳງານຂອງວົງຈອນ
* LED: ໃຊ້ອຸປະສະເເດງຜົນ ໂດຍຈະກະພິບຕາມສັນຍານທີ່ສົ່ງມາຈາກ Arduino
* Resistor: ໃຊ້ເພື່ອຈຳກັດກະເເສໄຟຟ້າໃຫ້ເໝາະສົມກັບ led ປ້ອງກັນບໍ່ໃຫ້ led ເສຍຫາຍ
* ສາຍເຊື່ອມຕໍ່: ໃຊ້ສຳຫຼັບເຊື່ອມຕໍ່ລະຫວ່າງ Arduino, led,ເເລະ ຕົວຕ້ານທານ

1.12 ການຕໍ່ວົງຈອນ:

1. ຕໍ່ຂາ​ Anode (ຂາບວກ)ຂອງ led ກັບ digital pin ຂອງ Arduino

(ເຊັ່ນ pin 13).

2. ຕໍ່ຂາ Cathode (ຂາລົບ) ຂອງ led ກັບ ຕົວຕ້ານທານ.

3. ຕໍ່ຕົວຕ້ານທານເຂົ້າກັບ Ground (GND)

ຂອງ Arduino.

1.13 ການຂຽນໂປຣເເກຣມ:

ໂປຣເເກຣມສຳຫຼັບ Experiment Blink ຈະຂຽນດ້ວຍພາສາ Arduino (C/C++)

ເຊິ່ງມີໂຄງສ້າງຫຼັກ 2 ສ່ວນຄື

Setup () ເເລະ loop ():

Code ຕົວຢ່າງ:

A screen shot of a computer program

Description automatically generated

1.14 ຄຳອະທິບາຍ Code:

Pin mode (13, OUTPUT): ຕັ້ງຄ່າ Pin 13

ເປັນໂໝດ OUTPUT ເພື່ອສົ່ງສັນຍານອອກໄປຄວບຄຸມ led .

digital Write (13, HIGH): ສົ່ງສັນຍານ HIGH (ເເຮງດັນ 5V) ໄປທີ່ Pin 13 ເພື່ອເປີດ led.

digital Write (13 LOW): ສົ່ງສັນຍານ LOW

(ເເຮງດັນ 0V) ໄປທີ່ Pin 13 ເພື່ອເປີດ led.

Delay (1000): ຢຸດການທຳງານເປັນເວລາ

1 ວິນາທີ (1000 ມິວລິວິນາທີ ).

1.15 ການທຳງານຂອງວົງຈອນ:

1. ເມື່ອໂປຣເເກຣມເລີ່ມທຳງານ Arduino ຈະສົ່ງສັນຍານ HIGH ໄປທີ່ Pin 13 ເຮັດໃຫ້ led ຕິດ
2. ຫຼັງຈາກນັ້ນໂປຣເເກຣມຈະຢຸດທຳງານເປັນເວລາ 1 ວິນາທີ(ຕາມຄຳສັ່ງ delay (1000))
3. ຈາກນັ້ນ Arduino ຈະສົ່ງສັນຍານ low HIGH ໄປທີ່ Pin 13 ເຮັດໃຫ້ led ດັບ
4. ໂປຣເເກຣມຈະຢຸດທຳງານອີກ 1 ວິນາທີ
5. ກະບວນການນີ້ຈະວົນລູບໄປເລື່ອຍໆ

ເຮັດໃຫ້ led ກະພິບ.

ຕົວຢ່າງ Link in Tinkercad:

https://www.tinkercad.com/things/cvyP9fFIw9a-1-experiment-blink?sharecode=IAkYurLgWOFjNl5SCZ4O\_cCq7BtQ1Yri2w\_S-UU-GEU

A qr code with a white background

AI-generated content may be incorrect.

ບົດທີ3 Switch

1. Switch ເເມ່ນຫຍັງ:

A circuit board with wires

AI-generated content may be incorrect.

ເປັນສ່ວນປະກອບອິເລັກທຣໍນິກທີ່ໃຊ້ໃນການຄວມຄຸມວົງຈອນໄຟຟ້າ ໂດຍການເປີດຫຼືປິດວົງຈອນເພື່ອໃຫ້ກະເເສໄຟຟ້າໄຫຼຫຼືຢຸດໄຫຼ

Switch ເປັນອຸປະກອນພື້ນຖານທີ່ໃຊ້ໃນໂຄງການອິເລັກທຣໍນິກຫຼາຍປະເທດເເລະສາມາດນຳມາໃຊ້ໃນ tinkercad ເພື່ອຈຳລອງການທຳງານຂອງວົງຈອນໄດ້

1.1 ປະເພດຂອງ Switch:

1. Push Button Switch

ສະວິດ ທີ່ທຳງານເມື່ອກົດປຸ່ມເເລະຈະກັບສູ່ສະຖານະເດີມເມື່ອປ່ອຍ

ມັກໃຊ້ໃນໂຄງການທີ່ຕ້ອງການຄວບຄຸມຊົ່ວຄາວ ເຊັ່ນ ການເປີດ/ປິດ led ຊົ່ວຄາວ

2. Toggle Switch

* ສະວິດທີ່ສາມາດລ໋ອກສະຖານະໄດ້

ເຊັ່ນເປີດຫຼືປິດ

- ເໝາະສຳຫຼັບການຄວບຄຸມທີ່ຕ້ອງການໃຫ້ຄົງສະຖານະໄວ້ຈົນກວ່າຈະສະລັບອີກຄັ້ງ

3. Slide Switch

ສະວິດທີ່ທຳງານໂດຍການເລື່ອນສະວິດໄປມາເພື່ອເປີດຫຼືປິດວົງຈອນ.

1.3 ການໃຊ້ສະວິດໃນ Tinkercad

ສະວິດ Tinkercad ສາມາດນຳມາໃຊ້ໃນວົງຈອນເພື່ອຄວບຄຸມການທຳງານຂອງອຸປະກອນອື່ນໆ

ເຊັ່ນ led ,ມໍເຕີ,ຫຼືເຊັນເຊີ ຕໍ່ໄປນີ້ເປັນຕົວຢ່າງການໃຊ້ງານສະວິດໃນວົງຈອນ Arduino:

1.31 ສ່ວນປະກອບທີ່ໃຊ້:

Arduino (ເຊັ່ນ Arduino Uno)

ສະວິດ (Push Button)

LED

ຕົວຕ້ານທານ(Resistor)

ສາຍເຊື່ອມຕໍ່

1.32 ການຕໍ່ວົງຈອນ:

ຕໍ່ຂາໜື່ງຂອງສະວິດເຂົ້າກັບ Digital Pin ຂອງ

Arduino (ເຊັ່ນ Pin 2)

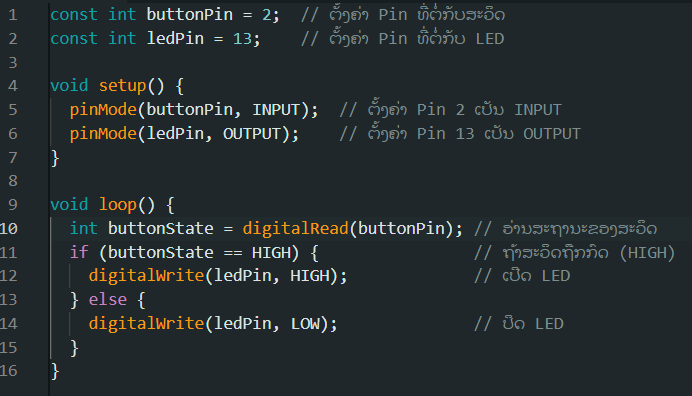
ຕໍ່ຂາອີກເບື້ອງຂອງສະວິດເຂົ້າກັບ

Ground (GND)

ຕໍ່ LED ກັບ Digital Pin ອື່ນ (ເຊັ່ນ Pin 13) ຜ່ານຕົວຕ້ານທານ

ຕໍ່ Ground ຂອງ LED ເຂົ້າກັບ Ground ຂອງ Arduino

1.4 ຂຽນໂປຣເເກຣມ:



1.5 ການທຳງານ:

ເມື່ອກົດສະວິດ Arduino ຈະອ່ານສັນຍານຈາກສະວິດເເລະສົ່ງສັນຍານHIGH ໄປທີ່ LED ເຮັດໃຫ້ LED ຕິດ.

ຕົວຢ່າງ Link in Tinkercad:

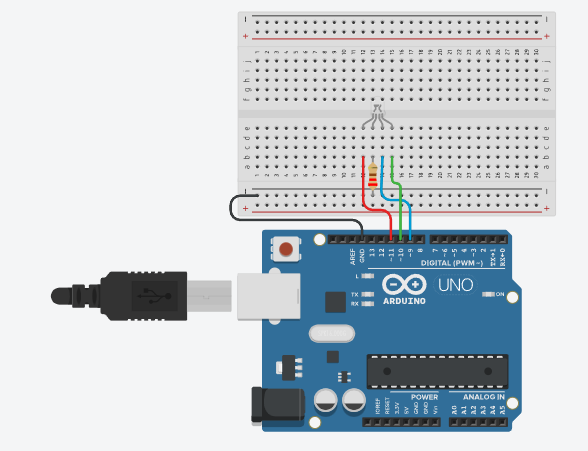
https://www.tinkercad.com/things/dMgCvs3qKHW-3-rgb-or-led-with-serial?sharecode=YVUB\_XQ-frLhMYtVyexzW3WWZ7HG-SA0G4nDgigX9xQ

A qr code with a white background

AI-generated content may be incorrect.

ບົດທີ4 RGB

1. RGB or LED with Serial ເເມ່ນຫຍັງ:



ທີ່ສາມາດມີເເສງໄດ້ຫຼາຍສີ ໂດຍປະສົມສີເເດງ (Red), ຂຽວ (Green), ເເລະສີຟ້າ (Blue)

ເຂົ້າດ້ວຍກັນ ເຊິ່ງເປັນຫຼັກການດຽວກັບລະບົບສີ RGB ທີ່ໃຊ້ໃນຈໍພາບຫຼືອຸປະກອນສະເເດງຜົນຕ່າງໆ RGB LED ມີຂາເຊື່ອມຕໍ່ 4 ຂາ ໄດ້ເເກ່ ຂາ (Anode) ຫຼື ຂາ (Cathode) ຮ່ວມເເລະຂາສຳຫຼັບເເຕ່ລະສີ (Red, Green, Blue) ໂດຍການຄວບຄຸມຄວາມສະຫວ່າງຂອງເເຕ່ລະສີຈະເຮັດໃຫ້ໄດ້ສີທີ່ເເຕກຕ່າງກັນອອກໄປ

Serial Communication

ຄືການສື່ສານຂໍ້ມູນລະຫວ່າງອຸປະກອນອິເລັກທຣໍນິກຜ່ານ (Serial Port) ເຊິ່ງໃນທີ່ນີ້ໝາຍເຖິງການສົ່ງຂໍ້ມູນຈາກຄອມພິວເຕີຫຼືMicrocontroller (ເຊັ່ນ Arduino) ໄປທີ່ RGB LEDເພື່ອຄວບຄຸມການສະເເດງຜົນສີຕ່າງໆ

1.1 RGB or LED with Serial ການທຳງານ:

ການທຳງານຂອງ RGB LED with Serial ໃນ

**Tinkercad** ເປັນຂະບວນການທີ່ໃຊ້

Microcontroller (ເຊັ່ນ Arduino) ຄວບຄຸມການສະເເດງຜົນຂອງ RGB LED ຜ່ານການສື່ສານເເບບ (Serial Communication) ໂດຍຜູ້ໃຊ້ສາມາດສົ່ງຄຳສັ່ງຈາກຄອມພິວເຕີໄປທີ່Arduino

ເພື່ອປ່ຽນສີຂອງ RGB LED ໄດ້ຕາມຕ້ອງການ

1.11ຕໍ່ວົງຈອນໃນ Tinkercad:

* ຂາ (Anode) ຕໍ່ກັບຂາ 5V ຫຼື GND

(ຂື້ນກັບປະເພດຂອງ RGB LED)

* ຂາສີເເດງ (Red) ຕໍ່ກັບຂາ PWM ຂອງ Arduino (ເຊັ່ນ ຂາ 9)
* ຂາສີຂຽວ (Green) ຕໍ່ກັບຂາ PWM ຂອງ Arduino (ເຊັ່ນ ຂາ10)
* ຂາສີຟ້າ (Blue) ຕໍ່ກັບຂາ PWM ຂອງArduino (ເຊັ່ນ ຂາ11)
* ຕົວຕ້ານທານ (Resistor) ທີ່ຂາເເຕ່ລະສີຂອງ

RGB LED ເພື່ອປ້ອງກັນບໍ່ໃຫ້ LED ເສຍຫາຍ

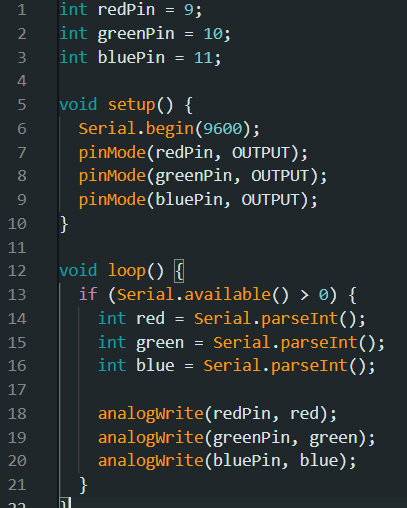
ຈາກກະເເສເກີນ

1.12 ຂຽນໂປຣເເກຣມໃນ Arduino:

ໂປຣເເກຣມຈະຮັບຄ່າສີ RGB ຈາກ Serial Monitor ເເລະສົ່ງສັນຍານ PWM ໄປຍັງຂາເເຕ່

ລະສີຂອງ RGB LED ເພື່ອຄວບຄຸມຄວາມສະຫວ່າງຂອງເເຕ່ລະສີ

ຕົວຢ່າງ Code:



1.13 ທົດສອບໃນ Tinkercad:

* ອັບໂຫຼດ Code ໄປທີ່ Arduino ໃນ Tinkercad
* ເປີດ Serial Monitor ເເລະປ້ອນຄ່າສີ

RGB ໃນຮູບເເບບ R G B

(ເຊັ່ນ 255 0 ສຳຫຼັບສີເເດງ)

* RGB LED ຈະປ່ຽນສີຕາມຄ່າທີ່ປ້ອນເຂົ້າໄປ

1.14 ຕົວຢ່າງການປ້ອນຄ່າໃນ Serial Monitor:

ສີເເດງ: 255 0 0

ສີຂຽວ: 0 255 0

ສີນໍ້າຟ້າ: 0 0 255

ສີເຫຼືອງ: 255 255 0

ສີມ່ວງ: 255 0 255

ສີຟ້າ: 0 255 255

ສີຂາວ: 255 255 255

ສະຫຼຸບການທຳງານ:

1 ຜູ້ໃຊ້ປ້ອນຄ່າສີ RGB ຜ່ານ Serial Monitor

2 Arduino ຮັບຄ່າສີເເລະສົ່ງສັນຍານ PWM ໄປທີ່ຂາເເຕ່ລະສີຂອງ RGB LED

3 RGB LED ປ່ຽນສີຕາມຄ່າທີ່ປ້ອນເຂົ້າໄປ.

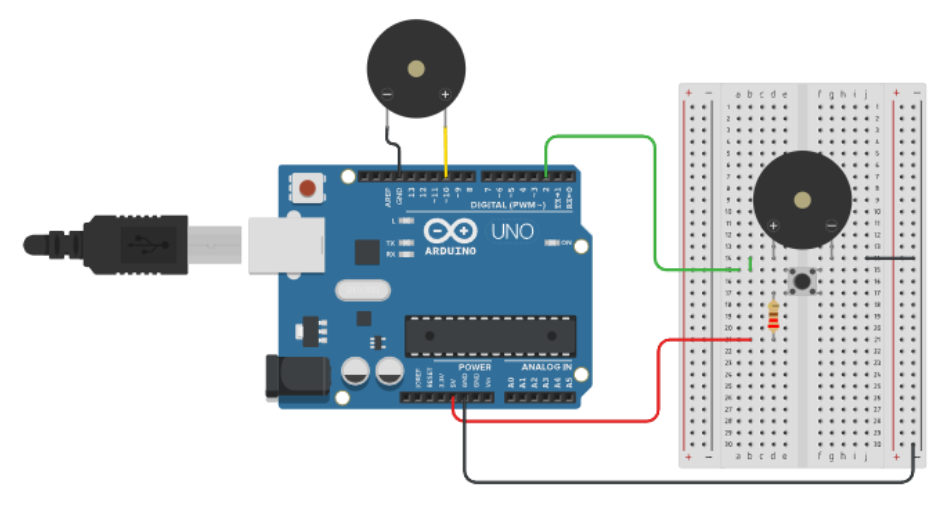
ຕົວຢ່າງ Link in Tinkercad:

https://www.tinkercad.com/things/dMgCvs3qKHW-3-rgb-or-led-with-serial?sharecode=YVUB\_XQ-frLhMYtVyexzW3WWZ7HG-SA0G4nDgigX9xQ



ບົດທີ 5 Passive Buzzer and Active Buzzer

1. Passive Buzzer and Active Buzzer ເເມ່ນຫຍັງ:



ເປັນອຸປະກອນສ້າງສຽງທີ່ໃຊ້ໃນວົງຈອນອິເລັກທຣໍນິກ

ໂດຍທັ້ງສອງປະເພດມີຫຼັກການທຳງານເເລະຄຸນສົມບັດທີ່ເເຕກຕ່າງກັນ ດັ່ງນີ້:

1.1 Passive Buzzer (ບັດເຊີເເບບເເພซีฟ)

ຫຼັກການທຳງານ: Passive Buzzer ບໍ່ມີວົງຈອນກຳເນີດສຍງພາຍໃນ ຕ້ອງໃຊ້ສັນຍານໄຟຟ້າພາຍນອກ(ເຊັ່ນ ສັນຍານ PWM) ເພື່ອຄວບຄຸມຄວາມຖີ່ເເລະຮູບເເບບຂອງສຽງ

ສຽງ: ສາມາດສ້າງສຽງໄດ້ຫຼາກຫຼາຍຄວາມຖີ່ເເລະຮູບເເບບ ຂື້ນຢູ່ກັບສັນຍານທີ່ປ້ອນເຂົ້າ

ການໃຊ້ງານ: ຕ້ອງໃຊ້ Microcontroller ຫຼືວົງຈອນສ້າງສັນຍານເພື່ອຄວບຄຸມສຽງ

ຕົວຢ່າງການໃຊ້ງານ: ໃຊ້ໃນອຸປະກອນທີ່ຕ້ອງການສຽງຫຼາກຫຼາຍເຊັ່ນດົນຕີອິເລັກທຣໍນິກ, ລະບົບເເຈ້ງເຕືອນທີ່ຊັບຊ້ອນ

1.2 Active Buzzer (ບັດເຊີເເອກທີບ)

- ຫຼັກການທຳງານ: Active Buzzer ມີວົງຈອນກຳເນີດສຽງພາຍໃນຕົວ ເຮັດໃຫ້ສາມາດສ້າງສຽງໄດ້ທັນທີເມື່ອຈ່າຍໄຟໃຫ້ (ມັກເປັນໄຟ DC) ໂດຍບໍ່ຕ້ອງໃຊ້ສັນຍານພາຍນອກ

* ສຽງ: ມັກສ້າງສຽງດຽວ(ເຊັ່ນ "beep") ທີ່ຄວາມຖີ່ຄົງທີ່
* ການໃຊ້ງານ: ໃຊ້ງານງ່າຍພຽງຈ່າຍໄຟກໍຈະເກີດສຽງທັນທີ
* ຕົວຢ່າງການໃຊ້ງານ: ໃຊ້ໃນອຸປະກອນເເຈ້ງເຕືອນງ່າຍໆເຊັ່ນ ໂມງປຸກ, ອຸປະກອນເເຈ້ງເຕືອນ
* ສະຫຼຸບຄວາມເເຕກຕ່າງ
* Active Buzzer: ໃຊ້ງານງ່າຍເເຕ່ສ້າງສຽງໄດ້ຈຳກັດ
* Passive Buzzer: ຕ້ອງໃຊ້ສັນຍານຄວບຄຸມເເຕ່ສ້າງສຽງໄດ້ຫຼາກຫຼາຍ

ການເລືອກໃຊ້ຂື້ນຢູ່ກັບຄວາມຕ້ອງການຂອງໂຄງການຫາກຕ້ອງການສຽງງ່າຍໆໃຫ້ເລືອກ Active Buzzer

ເເຕ່ຫາກຕ້ອງການຄວບຄຸມສຽງໄດ້ລະອຽດໃຫ້ເລືອກ Passive Buzzer.

1. ຕົວຢ່າງ Link in Tinkercad:

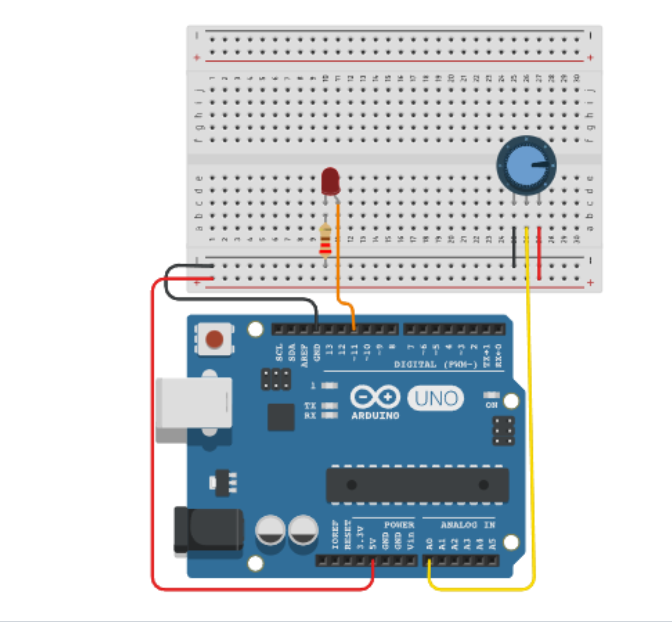
https://www.tinkercad.com/things/9oIOJaIejpU-4-passive-buzzer-and-active-buzzer?sharecode=2Jto0pc2lrtL-1mG7yl6ffl26tMx8TCiFNagIGG099Q

A qr code with black squares

AI-generated content may be incorrect.

ບົດທີ 6 Potentiometer

1. Potentiometer ເເມ່ນຫຍັງ:



ເປັນອຸປະກອນອິເລັກທຣໍນິກຊະນິດໜື່ງທີ່ເຮັດໜ້າທີ່ເປັນຕົວຄວບຄຸມຫຼືປັບຄ່າຄວາມຕ້ານທານໄຟຟ້າໄດ້ໂດຍມັກເອີ້ນສັ້ນໆວ່າ "Pot" ຫຼື "ຕົວປັບຄ່າ" ມີລັກສະນະເປັນອຸປະກອນ 3 ຂາ ເເລະສາມາດປັບຄ່າຄວາມຕ້ານທານລະຫວ່າງຂາສອງຂາໄດ້ດ້ວຍການໝຸນຫຼືເລື່ອນຕົວຄວບຄຸມ

* ສ່ວນປະກອບເເລະໂຄງສ້າງຂອງ Potentiometer

1.1 ຕົວຖັງເເລະເເກນໝຸນ: ມີເເກນໝຸນຫຼືສະໄລສຳຫຼັບປັບຄ່າ

1.2 ຂົ້ວຕໍ່ ມີ 3 ຂາໄດ້ເເກ່:

- ຂາທີ່ 1 ເເລະຂາທີ່ 3: ເປັນຂາສຸດຂອງຄວາມຕ້ານທານ(ຄ່າຄວາມຕ້ານທານລວມ)

* ຂາທີ່ 2 ເປັນຂາສຳຜັດທີ່ເລື່ອນໄດ້ (Wiper) ເຊິ່ງຈະປ່ຽນຄ່າຄວາມຕ້ານທານເມື່ອໝຸນຫຼືເລື່ອນ
  1. ເເຖບຄວາມຕ້ານທານ: ພາຍໃນມີເເຖບຄວາມຕ້ານທານ (Resistive Strip)

ທີ່ເຮັດຈາກວັດສະດຸເຊັ່ນຄາບອນຫຼືລວດພັນ

2.1 ຫຼັກການທຳງານ:

* ເມື່ອໝຸນເເກນຫຼືເລື່ອນຕົວຄວບຄຸມ ຂາສຳຜັດ (Wiper) ຈະເຄື່ອນໄປຕາມເເຖບຄວາມຕ້ານທານ
* ຄ່າຄວາມຕ້ານທານລະຫວ່ງາຂາສຳຜັດ (ຂາທີ່ 2) ເເລະຂາອື່ນໆ ຈະປ່ຽນເເປງຕາມຕຳເເໜ່ງຂອງ Wiper
* ຖ້າ Wiper ຢູ່ໃກ້ຂາທີ່ 1 ຄ່າຄວາມຕ້ານທານລະຫວ່າງຂາທີ່ 1 ເເລະຂາທີ່ 2 ຈະນ້ອຍລົງ
* ຖ້າ Wiper ຢູ່ໃກ້ຂາທີ່ 3 ขาที่ ຄ່າຄວາມຕ້ານທານລະຫວ່າງ 1 ເເລະຂາທີ່ 2 ຈະເພີ່ມຂື້ນ

2.1 ປະເພດຂອງ Potentiometer:

1. Rotary Potentiometer: ປັບຄ່າດ້ວຍການໝຸນເເກນ (ມັກໃຊ້ໃນອຸປະກອນປັບສຽງ ເຊັ່ນ ວິທະຍຸ)

2. Linear Potentiometer: ປັບຄ່າດ້ວຍການເລື່ອນຕົວຄວບຄຸມເປັນເສັ້ນຕົງ (ມັກໃຊ້ໃນເຄື່ອງມືວັດ)

3. Digital Potentiometer: ໃຊ້ສັນຍານ digital ໃນການຄວບຄຸມຄ່າຄວາມຕ້ານທານ (ມັກໃຊ້ໃນວົງຈອນອິເລັກທຣໍນິກສະໄໝໃໝ່).

2.2 ການໃຊ້ງານ Potentiometer:

1. ປັບລະດັບສຽງ: ໃນເຄື່ອງສຽງ ເຊັ່ນ ວິທະຍຸ, ເຄື່ອງຂະຫຍາຍສຽງ
2. ຄວບຄຸມຄວາມສະຫວ່າງ: ໃນອຸປະກອນເເສງສະຫວ່າງ ເຊັ່ນ ໄຟປັບຄວາມສະຫວ່າງໄດ້
3. ຄວບຄຸມຄວາມໄວ: ໃນມໍເຕີຫຼືພັດລົມ
4. ເຊັນເຊີວັດຕຳເເໜ່ງ: ໃນລະບົບຄວບຄຸມຫຼືຫຸ່ນຍົນ
5. ປັບຄ່າໃນວົງຈອນອິເລັກທຣໍນິກ: ເຊັ່ນ ຕັ້ງຄ່າຄວາມໄວຂອງວົງຈອນ.

2.3ຕົວຢ່າງການຕໍ່ວົງຈອນ:

ຖ້າໃຊ້ Potentiometer ເປັນຕົວເເບ່ງເເຮງດັນ

(Voltage Divider):

ຂາທີ່ 1 ຕໍ່ກັບເເຫ່ງຈ່າຍໄຟ (VCC)

ຂາທີ່ 2 ຕໍ່ກັບກຣາວ (GND)

ຂາທີ່ 3 ຈະໃຫ້ເເຮງດັນອອກທີ່ປ່ຽນເເປງໄດ້ຕາມຕຳເເໜ່ງ

Wiper.

* 1. ສະຫຼຸບ:

Potentiometer ເປັນອຸປະກອນທີ່ໃຊ້ປັບຄ່າຄວາມຕ້ານທານໄດ້ຢ່າງຕໍ່ເນື່ອງ ມີທັ້ງແບບໝຸນເເລະເເບບເລື່ອນ ໃຊ້ງານໃນວົງຈອນອິເລັກທຣໍນິກເເລະອຸປະກອນຕ່າງໆເພື່ອຄວບຄຸມຫຼືປັບປ່ຽນຄ່າ ເຊັ່ນ ລະດັບສຽງ, ຄວາມສະຫວ່າງ, ຫຼືຕຳເເໜ່ງ.

ຕົວຢ່າງ Link in tinkercad:

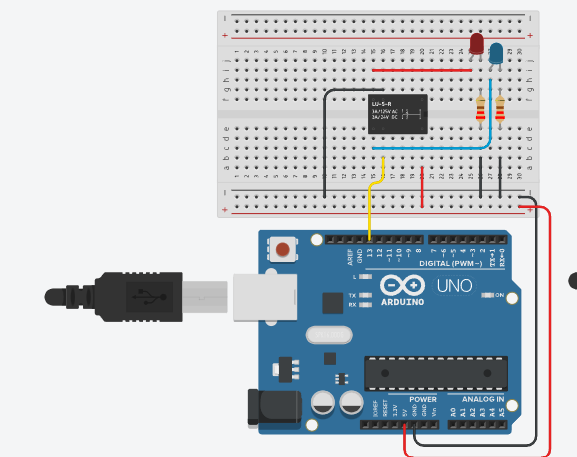
https://www.tinkercad.com/things/hb9Z5NWaxIr-5-potentiometer-volume?sharecode=KqPM7CuKWvKno6xAgQZkPywSxzLAscl4U\_4F-26q5RU

A qr code on a white background

AI-generated content may be incorrect.

ບົດທີ 7 Relay with 12 V

1. Relay with 12 V ເເມ່ນຫຍັງ



ເປັນອຸປະກອນສະວິດໄຟຟ້າທີ່ໃຊ້ສັນຍານໄຟຟ້ານ້ອຍເພື່ອຄວບຄຸມວົງຈອນໄຟຟ້າທີ່ມີກຳລັງສູງກວ່າ ໂດຍທົ່ວໄປ

Relay ປະກອບດ້ວຍມ້ວນເເມ່ເຫຼັກໄຟຟ້າເເລະຊຸດໜ້າສຳຜັດ ເມື່ອຈ່າຍໄຟໃຫ້ມ້ວນ ຈະເກີດສະນາມເເມ່ເຫຼັກ

ທີ່ດືງໜ້າສຳຜັດໃຫ້ເປີດຫຼືປິດເຮັດໃຫ້ສາມາດຄວບຄຸມອຸປະກອນອື່ນໆໄດ້.

1. ວົງຈອນ Relay 12 V ໃນ tinkercad

ໃນ tinkercad ເຈົ້່າສາມາດສ້າງວົງຈອນ

Relay 12 V ໄດ້ໂດຍໃຊ້ Components

ຕ່າງໆເຊັ່ນ:

1 Relay 12 V: ອຸປະກອນ Relay ທິ່ທຳງານທີ່ເເຮງດັນ 12 V

2 ເເຫຼ່ງຈ່າຍໄຟ 12 V: ສຳຫຼັ້ບຈ່າຍໄຟໃຫ້ Relay

ສະວິດຫຼື Microcontroller:

1. ສຳຫຼັ້ບຄວບຄຸມ Relay
2. ໂຫຼດ: ເຊັ່ນ ຫຼອດໄຟຫຼືມໍເຕີ ທີ່ Relay

ຈະຄວບ ຄຸມ

1. ຂັ້ນຕອນການສ້າງໃນ **Tinkercad**

* ເພີ່ມ Components: ເລືອກ Relay 12 V,

ເເຫຼ່ງຈ່າຍໄຟ, ສະວິດເເະລໂຫຼດ

* ຕໍ່ວົງຈອນ: ຕໍ່ຂົ້ວມ້ວນ Relay ເຂົ້າກັບ

ເເຫຼ່ງຈ່າຍໄຟ 12 ເເລະສະວິດ ຕໍ່ໜ້າ

ສຳຜັດ Relay ເຂົ້າກັບໂຫຼດ

* ຈຳລອງວົງຈອນ: ໃຊ້ເຄື່ອງມືຈຳລອງເພື່ອທົດສອບການທຳງານຂອງ Relay

1. ຕົວຢ່າງການໃຊ້ງານ

- ຄວບຄຸມຫຼອດໄຟ: ໃຊ້ Relay ເປີດປິດຫຼອດໄຟ

- ຄວບຄຸມມໍເຕີ: ໃຊ້ Relay ຄວບຄຸມມໍເຕີ

5. ສະຫຼຸບ:

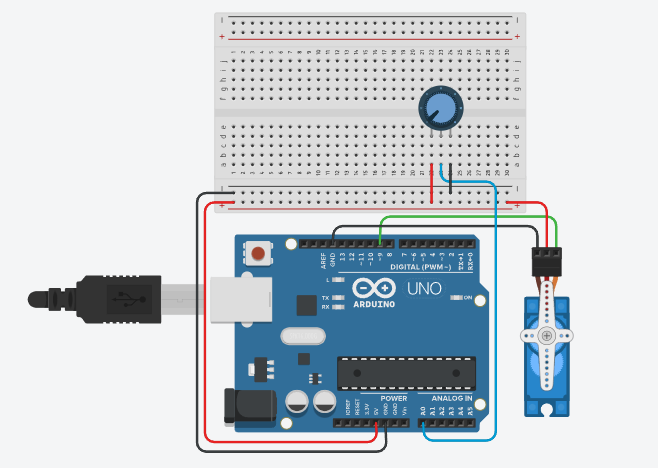
"Relay with 12 V in Tinkercad" ຄືການຈຳລອງ

ວົງຈອນ Relay 12 V ໃນ Tinkercad ເຊິ່ງຊ່ວຍໃຫ້

ເຮົາເຂົ້າໃຈການທຳງານຂອງ Relay ເເລະທົດສອບວົງຈອນກ່ອນສ້າງຈີງ

ຕົວຢ່າງ Link in tinkercad:

https://www.tinkercad.com/things/5kojtkV7Va7-6-sweep-and-knob-servo-motor-?sharecode=A0980kTiAqv9xDDiH55xKSmLm124xRyOTAMe6bWdnn8



ບົດທີ 8 servo

1. **Sweep and Knob Servo Motor:**

A circuit board with wires and a power cord

AI-generated content may be incorrect.

**ເປັນຕົວຢ່າງການຄວບຄຸມເຊີໂວມໍເຕີ**

**(Servo Motor) ທີ່ມັກໃຊ້ໃນການຮຽນຮູ້**

**ເເລະການທົດລອງ ໂດຍສະເພາະກັບ Arduino**

**ເເລະໂປຣເເກຣມຈຳລອງເຊັ່ນ Tinkercad.**

2.ເຊີໂວມໍເຕີ **(Servo Motor)**

ເປັນມໍເຕີທີ່ສາມາດຄວບຄຸມຕຳເເໜ່ງການໝຸນໄດ້ຢ່າງເເມນຍຳ ໂດຍທົ່ວໄປສາມາດໝຸນໄດ້ປະມານ  0 ຫາ 180

ອົງສາ ມັນມັກໃຊ້ໃນງານທີ່ຕ້ອງການ ການເຄື່ອນໄຫວທີ່ເເມນຍຳ ເຊັ່ນ ຫຸ່ນຍົນ, ລະບົບຄວບຄຸມເເລະອື່ນໆ.

3. **Sweep Servo Motor**

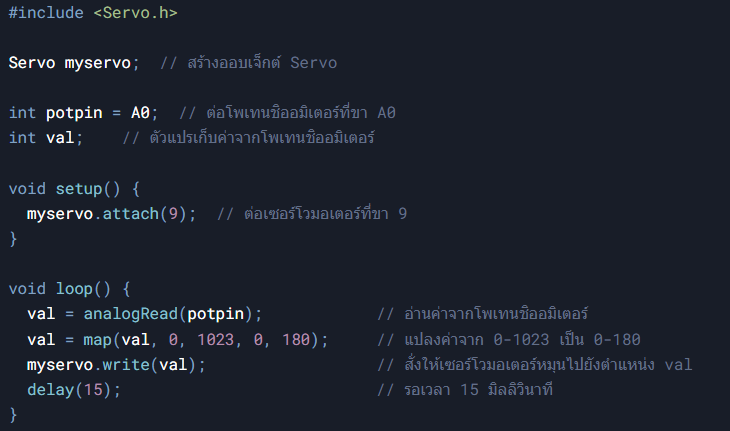
ໝາຍເຖິງການເຮັດໃຫ້ເຊີໂວມໍເຕີໝຸນໄປມາໃນມຸມທີ່

ກຳໜົດ ເຊັ່ນ ຈາກ 0 ຫາ 180 ອົງສາ ເເລະກັບມາທີ່

0 ອົງສາ ຕົວຢ່າງ Code Arduino ສຳຫຼັບການເຮັດ

Sweep:

ຕົວຢ່າງ Code:



4.ການໃຊ້ໃນ **Tinkercad**

ໃນ **Tinkercad ເຮົາສາມາດສ້າງວົງຈອນເເລະຂຽນ Code ເພື່ອຄວບຄຸມເຊີໂວມໍເຕີໄດ້ຢ່າງງ່າຍດາຍ:**

**1.ເພີ່ມ** Components: ເລືອກເຊີໂວມໍເຕີ, Arduino, ເເລະ Potentiometer (ສຳຫຼັບ Knob)

2.ຕໍ່ວົງຈອນ: ຕຕໍ່ເຊີໂວມໍເຕີເເລະ Potentiometer ເຂົ້າກັບ Arduino

3.ຂຽນ Code: ໃຊ້ຕົວຢ່າງ Code ດ້ານເທິງຫຼືຂຽນ Code ຂອງເຮົາເອງ

4.ຈຳລອງວົງຈອນ: ໃຊ້ເຄື່ອງມືຈຳລອງເພື່ອທົດສອບການທຳງານ.

5.ສະຫຼຸບ:

* **Sweep Servo Motor**: ເຮັດໃຫ້ເຊີໂວມໍເຕີໝຸນໄປມາໃນມຸມທີ່ກຳໜົດ
* **Knob Servo Motor**: ຄວບຄຸມເຊີໂວມໍເຕີດ້ວຍ Potentiometer.

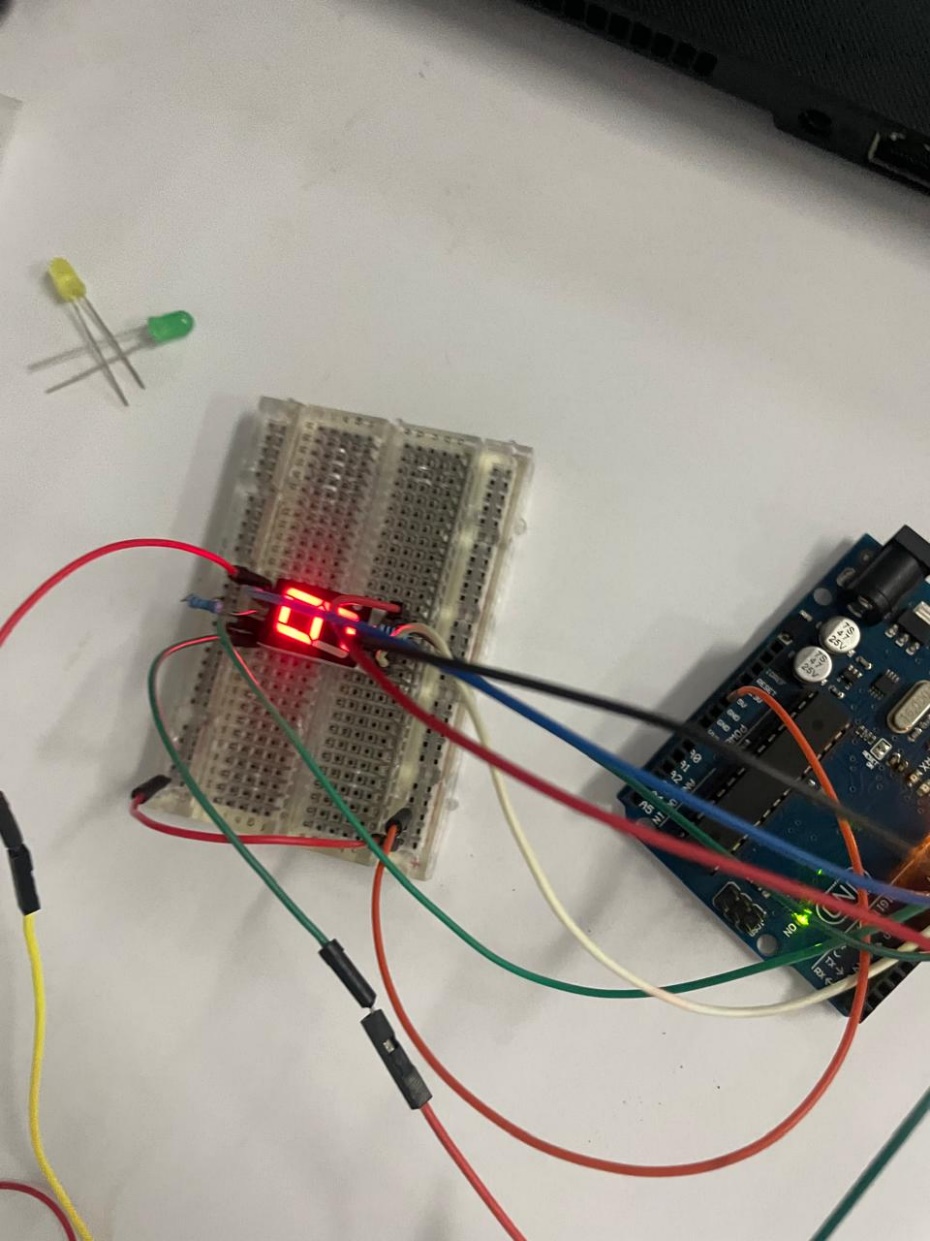
ຕົວຢ່າງ Link in tinkercad:

https://www.tinkercad.com/things/5kojtkV7Va7-6-sweep-and-knob-servo-motor-?sharecode=A0980kTiAqv9xDDiH55xKSmLm124xRyOTAMe6bWdnn8



ບົດທີ 9 seven segment

1. seven segment ເເມ່ນຫຍັງ:



ເປັນອຸປະກອນສະເເດງຜົນເເບບດິຈິຕຣໍທີ່ໃຊ້ສະແດງ

ຕົວເລກເເລະບາງຕົວອັກສອນ ໂດຍປະກອບດ້ວຍ

LED (Light Emitting Diode) 7 ສ່ວນທີ່ເອີ້ນວ່າ

"segment" ເຊິ່ງເເຕ່ສ່ວນສາມາດເປີດຫຼືປິດໄດ້ເພື່ອສ້າງຮູບຕົວເລກຫຼືອັກສອນ.

2.ສ່ວນປະກອບຂອງ **Seven Segment:**

**1/ Segment ທັ້ງ 7 ສ່ວນ**

* **Segment ເເຕ່ລະສ່ວນມີຊື່ເອີ້ນເປັນອັກສອນ**

**A ຫາ G**

* ເເຕ່ລະສ່ນວນສາມາດເປີດຫຼືປິດໄດ້ເພື່ອສ້າງຮູບຕົວເລກຫຼືອັກສອນ

2/ ຈຸດທົດນິຍົມ (Decimal Point):

* ບາງລຸ່ນມີຈຸດທົດນິຍົມ (DP) ເພື່ອສະເເດງຈຸດທົດນິຍົມໃນຕົວເລກ

3.ການສະເເດງຜົນ:

* ຕົວເລກ 0-9: ໂດຍການເປີດ - ປິດ Segment ທີ່ເໝາະສົມ
* ຕົວອັກສອນບາງຕົວ: ເຊັ່ນ A, B, C, D, E, F

4.ປະເພດຂອງ Seven Segment:

Common Anode: ຂົ້ວບວກຂອງ ທັ້ງໝົດເຊື່ອມຕໍ່ກັນ

Common Cathode: ຂົ້ວລົບຂອງ LED ທັ້ງໝົດເຊື່ອມຕໍ່ກັນ

5.ການໃຊ້ງານ

ເຄື່ອງຄິດເລກ

ໂມງດິຈິຕຣໍ

ເຄື່ອງວັດ

ປ້າຍສະເເດງຜົນ

6.ຂໍ້ຜິດ

* ສະເເດງຜົນໄດ້ຈຳກັດ

(ສະເພາະຕົວເລກເເລະຕົວອັກສອນບາງຕົວ)

* ຕ້ອງການການຄວບຄຸມຫຼາຍຂາ

7.ສະຫຼຸບ:

ເປັນອຸປະກອນສະເເດງຜົນທີ່ໃຊ້ກັນທົ່ວໄປໃນອຸປະກອນອິເລັກທຣໍເນື່ອງຈາກໃຊ້ງານງ່າຍເເລະລາຄາຖືກ.

ຕົວຢ່າງ Link in tinkercad:

https://www.tinkercad.com/things/cBtpbeDxncA-ldr-counter-using-arduino

