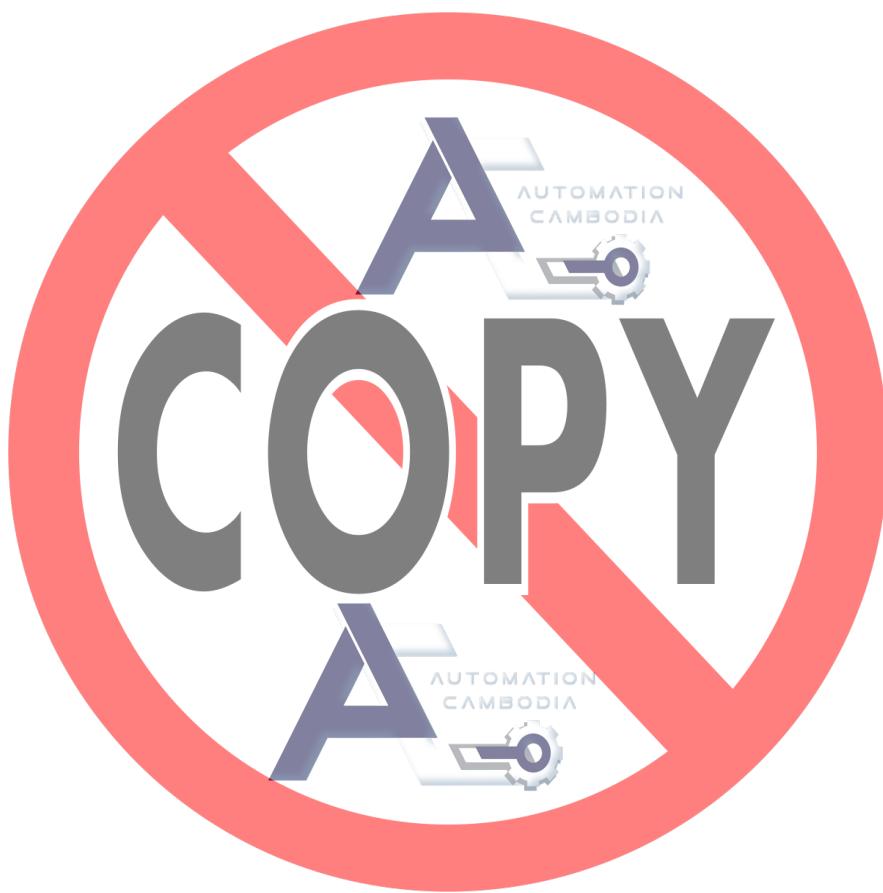


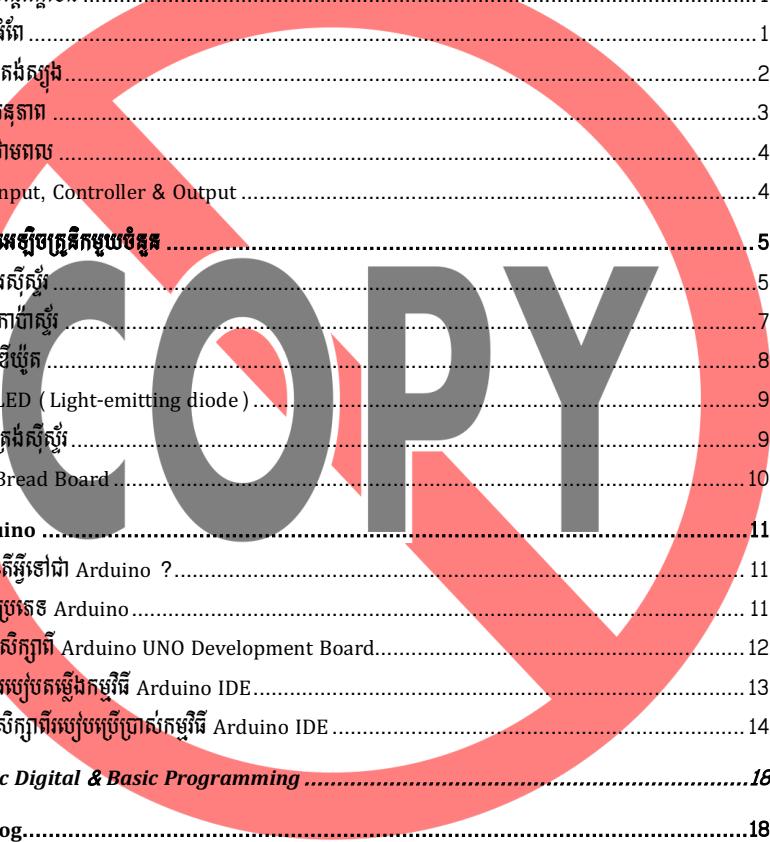
ពិនិត្យការងារ
និងអនុវត្តន៍



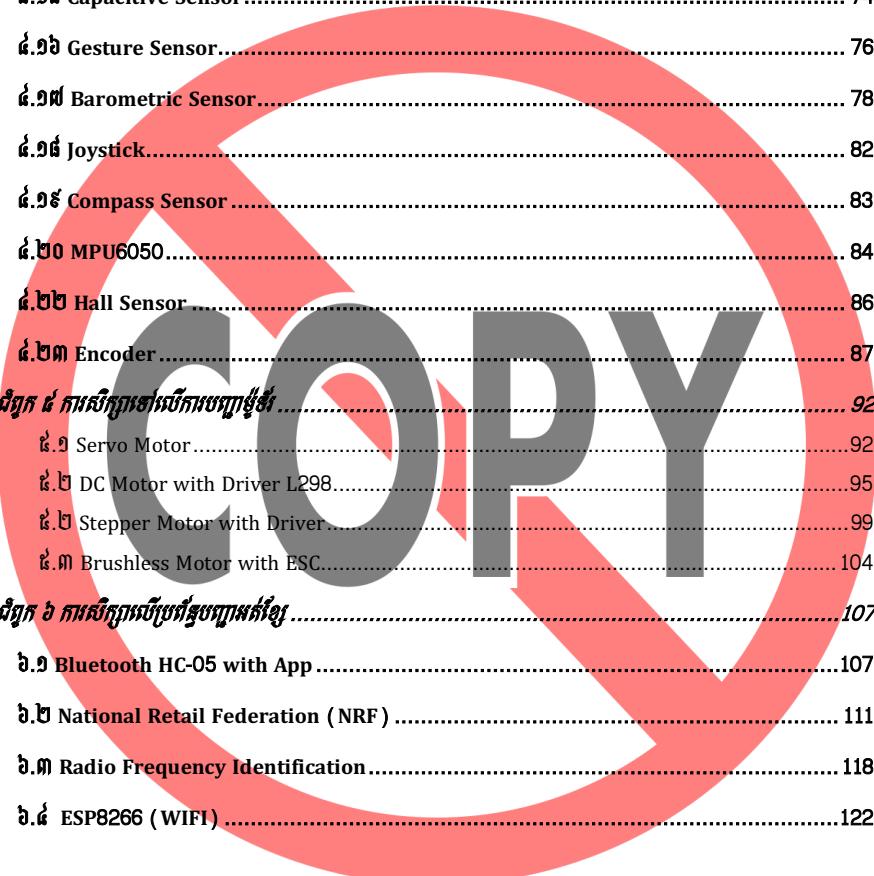
ពិនិត្យការងារ
និងអនុវត្តន៍

ទេសិកា

ជំរឿក ១ ទំនាក់ទំនង	1
១.១ កម្រិតដែលអាចប្រើប្រាស់បាន	1
១.១.១ ចេវនាអតិថិជន	1
១.១.២ អំពេជា	1
១.១.៣ តម្លៃស្ថាប់	2
១.១.៤ អនុវត្ត	3
១.១.៥ ចាយណ៍	4
១.១.៦ Input, Controller & Output	4
១.២ បិទ្ទូរបញ្ជីកម្មឈើទូទៅ	5
១.២.១ ធម៌ស្ថី	5
១.២.២ ការិយាល័យ	7
១.២.៣ ខិស្ស	8
១.២.៤ LED (Light-emitting diode)	9
១.២.៥ ត្រួតស្ថីស្ថី	9
១.២.៦ Bread Board	10
១.៣ Arduino	11
១.៣.១ តើខ្លួនគោរព Arduino ?	11
១.៣.២ ប្រព័ន្ធគ្មាន Arduino	11
១.៣.៣ សិក្សាទី Arduino UNO Development Board	12
១.៣.៤ របៀបការងារក្នុង Arduino IDE	13
១.៣.៥ សិក្សាទីរបៀបប្រើប្រាស់ក្នុង Arduino IDE	14
ជំរឿក ២ Basic Digital & Basic Programming	18
២.១ Analog	18
២.២ Digital	18
២.៣ Variable Type	19
២.៤ Further Syntax	20
២.៥ Arithmetic Operator	20
២.៦ Comparison Operator	20



၁.၁ Bitwise Operator.....	20
၁.၂ Boolean Operator	22
၁.၃ Boolean Operator	22
၂။၁ Programming နည်း Arduino	24
၂.၁ Blink LED.....	24
၂.၂ Digital Input	29
၂.၃ 7 Segment.....	30
၂.၄ 7 Segment with Max7219.....	34
၂.၅ LCD	36
၂.၆ OLED.....	41
၂.၇ LED Matrix.....	42
၃။၁ Sensor	45
၃.၁ IR REMOTE	45
၃.၂ LDR.....	47
၃.၃ Motion Sensor or PIR Sensor	48
၃.၄ Thermistor	50
၃.၅ Humidity Sensor	52
၃.၆ Ultrasonic Sensor.....	54
၃.၇ Gas Methane Sensor	57
၃.၈ Alcohol Sensor (not Complete)	58
၃.၉ Rain Sensor	59
၃.၁၀ Soil Moisture Sensor	61
၃.၁၁ Flame Sensor	62
၃.၁၂ Heart Beat Sensor.....	64
၃.၁၃ Reed Switch Sensor	66
၃.၁၄ Weight Sensor.....	67



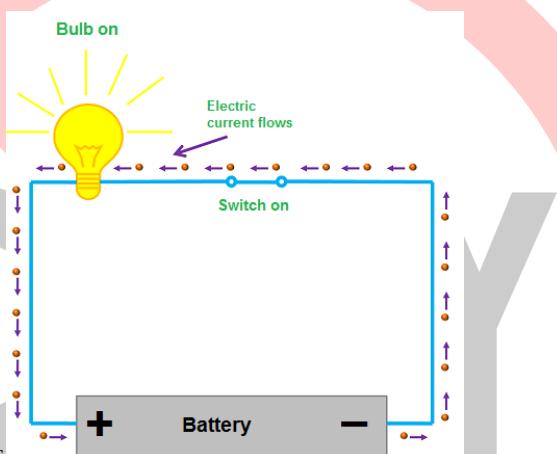
၄.၁၇ Capacitive Sensor	74
၄.၁၈ Gesture Sensor	76
၄.၁၉ Barometric Sensor	78
၄.၁၂ Joystick	82
၄.၁၅ Compass Sensor	83
၄.၂၀ MPU6050	84
၄.၂၁ Hall Sensor	86
၄.၂၃ Encoder	87
၅. ကျော် နှင့် စာမျက်နှာအသွေးပြုချက်များ	92
၅.၁ Servo Motor	92
၅.၂ DC Motor with Driver L298	95
၅.၃ Stepper Motor with Driver	99
၅.၄ Brushless Motor with ESC	104
၆. ကျော် ပါ စာမျက်နှာပို့ဆောင်ရွက်နည်း	107
၆.၁ Bluetooth HC-05 with App	107
၆.၂ National Retail Federation (NRF)	111
၆.၃ Radio Frequency Identification	118
၆.၄ ESP8266 (WIFI)	122

ចំណេះទីស្តីរួម ១ ម៉ាស៊ីនអតិថិជន

១.១ កម្រិតដំបូងអេឡិចត្រូនិក និងសៀវភៅ

១.១.១ ចំណេះទីស្តីរួម

យើងបានធ្វើប័ណ្ណហើយម៉ា នៅក្នុងណោហេ: ទៅដើរក ប្រាក់ កាលឃើមឱ្យម៉ោង មាស និងមាន ពាក្យាបានដែលទ្វាក់ គឺ សុខសិនកំមាននូវបិទាណចំនួនអេឡិចត្រូនិកដែលសំណើជាមួយ ចំនួន អេឡិចត្រូនិកទាំងអស់នេះត្រូវបានផ្តាស់ទិញយោស់ បើទៀតការណាស់ទិញបានដោយយ៉ាងណាក់ ដោយ កំណើនបានផ្តាស់ទិញទិន្នន័យប្រចាំនាង។



យោងទៅតាមឱ្យបានលើចំណេះទីស្តីរួមអតិថិជនដែលផ្តល់សំណើជាមួយសៀវភៅ ដោយផ្តល់អតិថិជន និង ប្រភពតាមធមល (Battery)។ លក្ខខណ្ឌដែលអាចចូលអេឡិចត្រូនិកដោយប្រាកាប់បាន ឬប្រាក់អេឡិចត្រូនិកទាំងអស់នេះសំណើជាមួយ បុគ្គលិោននៃប្រភពតាមធមល តួនករណីទាំងអស់នេះត្រូវបានគេហោចាល់បញ្ចូនដោយប្រើប្រាស់ការងារដែលមួយនឹងការងារដែលមួយនឹងការងារ។ ដោយការងារដែលមួយនឹងការងារ កំណើនបាននូវការងារដែលមួយនឹងការងារដែលមួយនឹងការងារ។

១.១.២ អំពេល

នៅពេលដែលបន្ទុកអតិថិជនត្រូវបានគេវាស់ ដោយប្រើប្រាស់ខ្សោតគុណ (Coulomb) បើយ ហិមាណនឹងលំហូរបន្ទុកអតិថិជននេះ: គឺត្រូវបានគេកំណត់ដោយផ្តល់បន្ទុកអតិថិជន (អេឡិចត្រូនិក) និងផ្តល់បន្ទុកអតិថិជន (Ampere)។ បើយកើតឡើង ទទួលបានចរន្តមួយអំពេល ប្រសិនបើមានហិមាណនឹងអេឡិចត្រូនិកដែលមួយគុណ។

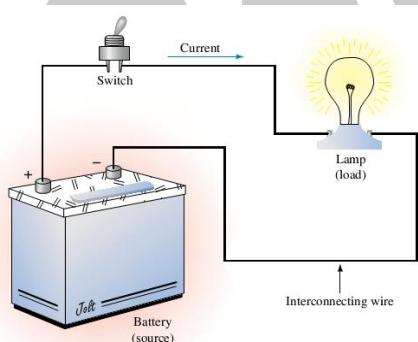
ពេលមួយវិទានី ផ្តល់ការតែងចាំណុចដែលគោរពមេរោគ។ និមួយធម្មតានៃចរណី ១ ដែលបានបង្ហាញជាទម្រង់គណនិភ័យ៖

$$I = \frac{Q}{t} \text{ (ampere, A)}$$

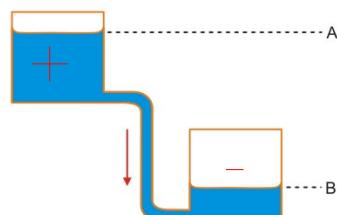
- I: ចរណីអតិថិជន (A)
- Q: គិតធាតុតូច (C)
- t: ពាក្យេ:ពេល ប្រកបដុងពេលសរបត្វឯងមួយវិទាន (s)

៩.១.៣ ពង់ស្បែង

នៅពេលដែលបង្ហាញអតិថិជនីត្រូវបានធ្វើឡើងថីចំណុចរបស់ក្នុងមួយ ហើយទៅបន្ថែមមួយរាយដី ចំណុចមួយដែលទៀតនៅក្នុងដែរ នៅក្នុងបណ្តុាលឈរកើតមាន តិប់សង្គច្ច័ន់ស្បែង បុក់កង់ស្បែង រាយខ្លួនខ្លួន និងខ្លួនខ្លួន ពង់ស្បែង បុត្រិសង្គច្ច័ន់ស្បែង នានាជីសដោចអូលច្បាស់ណាស់ ដែលវានិងធ្វើឡើងពីបន្ទុកអើងមាន (-) ទៅបន្ទុកកើងមាន (+) ។

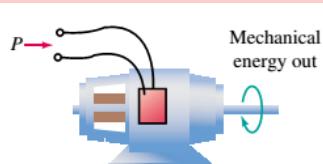
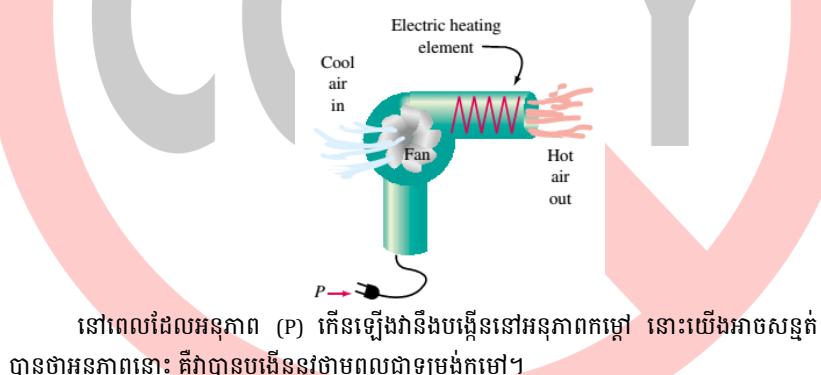
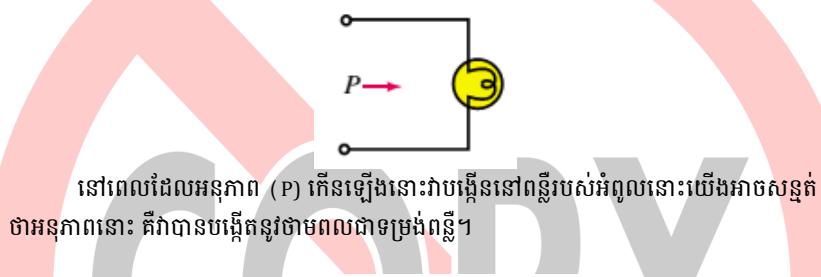


យើងសាកកពីពាក្យេដែលបំបុរាណទីក ជាមួយគារីកនឹងហ្មូរកើតឡើងខ្លួនទៅទីតាំងទាម។ ដូច្នានេះដោយ ចរណីអតិថិជនីវាដែងទៅបុរាណ បុចលែកពីគោលដែលមានបន្ទុកកើងមាន ទៅកាន់គោលដែលមានបន្ទុកអើងមាន។



១.១.៥ អនុកាត

ក្រឹមហិត្តរបៀប្រាស់មួយចំនួននៅក្នុងផ្ទៃដូចទាំងអស់ត្រាតូចជាតុ ឬទីកកកក ឬសុវត្ថិភាព និងទីកត្រិកតាក់ និងឱ្យបញ្ចូលជាមួយនឹងក្រឹមអតិសនិមួយចំនួនដែលធ្វើឡើកបានកំណត់តម្លៃដោយកម្រិតអនុកាតបស់វា ដែលយើងហេតា ក៏ (W) ។ ចំណែកការបិត្តរាជតិសនិមួយចំនួនឡើកបុចជាមួកទាំងឡាយ ឬមួកទាំងឡាយ ឬសុវត្ថិភាព និងបិត្តរាជតិសនិមួយចំនួនឡើកបុចជាមួកពាក្យាំងសេះ (Horsepower) (ក្នុងម៉ាក ១hp = 746W)



នៅពេលដែលអនុកាត (P) គឺនៅឡើងនោះ ការបង្កើតនៅអនុកាតមេរានិកមួកទាំងឡាយ (ការចបង្កើននៅលើវិនិមួកទាំងឡាយ ឬកម្លាំងកម្លានអូសរបស់មួកទាំងឡាយ)

បន្ទាប់ពីយើងបានមិនការបក្សរាយរូបចានបីខាងលើរួចរាល់ យើងរាជសន្យាតំងារអនុកាត គឺនាន់តាក់ទំនងយ៉ាងនិតិស្សិតជាមួយ និងថាមពល។ អនុកាតប្រើបានគេកំណត់ដោយនិមិត្តិសញ្ញាអក្សរ P ដែលមានរូបមន្ត្រាំង៖

$$P = \frac{W}{t}$$

ដែល W ជាទម្ចប់ (Work or Energy) ដែលត្រូវបានការដោយខ្លាតស្ថិត (J) ហើយ t គឺជានយោបាយដែលដោយប្រើខ្លាតជានាទី (s)។

ខ្លាតកាត់ អាជត្រូវបានចំនួនដោយការប្រើប្រាស់ខ្លាតស្ថិតក្នុងមួយវិនាទី (J/s) មានន័យថា $1W = 1J/s$ ។

៩.៩.៥ ចាមពល

នៅក្នុងដែកប្រាក់ថ្មីរបស់យើងស្មោះ យើងតែងតែឱ្យប្រទេសវិញការប្រើប្រាស់ខ្លាតស្ថិតនៃប្រាក់ប្រាក់អតិថិជនដែលបានប្រើប្រាស់ខ្លាតការប្រើប្រាស់ចាមពលអតិថិជននៅក្នុងផ្ទះសម្រាប់បំភូអំពូលត្រូវឱ្យនៅក្នុងដែកប្រាក់ ប្រើប្រាស់ត្រូវឱ្យដោយអតិថិជន មានវិធានបានប្រាក់ប្រាក់អតិថិជននៃការប្រើប្រាស់ខ្លាតស្ថិតក្នុងមួយវិនាទីក្នុងមួយពាន់ម៉ោង ដើម្បីដើរការប្រាក់ប្រាក់បានកំណត់ដោយនិមិត្តសញ្ញា W ដែលបង្ហាញចាមពលប្រចាំគីឡូវិក្សោះ

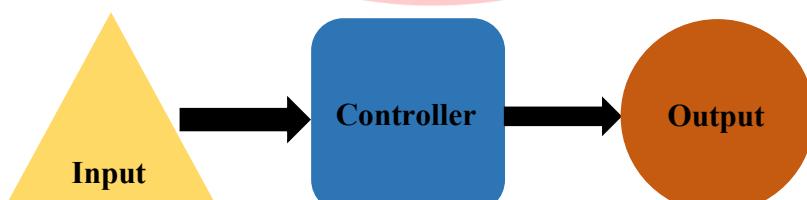
$$W = Pt$$

- P : ជាអនុភាពគិតជាដែល
- t : នៅពេលដែល t ការដោយប្រើប្រាស់ខ្លាតជានាទី s នៅចាមពល (W) ត្រូវបានកំណត់ជាពិនិត្យស្ថិត (J)
- t : នៅពេលដែល t ការដោយប្រើប្រាស់ខ្លាតជានាទី h នៅចាមពល (W) ត្រូវបានកំណត់ជាពិនិត្យស្ថិត Wh វិញ

ឧទ្ធស័ំ : ប្រសិនបើយើងប្រើប្រាស់អំពូលដែលស្ថិតចាមពល $100W$ នឹងមួយម៉ោង មានន័យថា $W=Pt=(100W)(1h)=100Wh=0.1KWh$ ។

៩.៩.៦ Input, Controller & Output

ជាមួយដើម្បីទ្វាក់ពេលវេលាប្រចាំថ្ងៃគឺការប្រាក់ប្រាក់ដែលត្រូវបានការប្រាក់ប្រាក់ដោយអតិថិជន ដើម្បីកសំខាន់របៀបដែលបានប្រើប្រាស់ខ្លាត។ Input, Controller និង Output



- Input ជាធិន្ទំយុទ្ធផលនឹងត្រូវបានដោយប្រើប្រាស់ខ្លាត។ Controller ដែលមានជូនជាបុរាណ, Sensor និងទឹកដែលត្រូវបានដោយប្រើប្រាស់ខ្លាត។

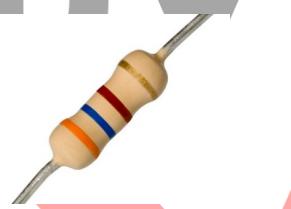
- Controller មានគុណភាពជាមួកចាំយកទិន្នន័យពីផ្ទុកខាង Input រួចធ្វើការវិភាគ បន្ទាប់មកបញ្ជា
ទៅផ្តើក Output ដែលនៅត្រូវនៅមានរូបធានា IC, PLC ជាដើម
- Output មានគុណភាពជើសរើសពីផ្ទុកខាង Controller ដែលបានបញ្ជាមក ដែលនៅ
ត្រូវនៅមានរូបធានា អំពូល មួយទៅជាដើម

១.២ បរិក្សាអនឡូត្រូនិកមួយចំណុះ

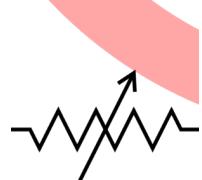
១.២.១ នសិក្សា

នសិក្សា ជាប្រភេទអនឡូត្រូនិកមួយប្រភេទដែលបានប្រើប្រាស់តាមនសិក្សាដែលមិនកំណត់លំហោរ
របស់ចរន្តអគ្គិសនីនៅកាត់។ នសិក្សាបានបង្កើចការណីប្រភេទ គឺនសិក្សាមានតម្លៃប្រចាំថ្ងៃ និងនសិក្សា
មានតម្លៃប្រចាំសប្តាហើរ។

- នសិក្សាអាមេរិកមួយចំណុះ



- នសិក្សាអាមេរិកមួយចំណុះ (Symbol មានប្រើប្រាស់ប្រភេទ)



$$\text{រូបមន្តុទឹក} \therefore I = \frac{V}{R}$$

- I: ចរន្តអគ្គិសនីគិតជា (A)
- V: តម្លៃស្បែកគិតជា (V)
- R: នសិក្សាដែល (ORM)

រូបមន្តតណាងសីស្សដែលជាស៊ីវិទ្យាអំពី សំណង់បច្ចេកទេស

$$R = R_1 + R_2 + R_3$$

$$V = V_1 + V_2 + V_3$$

$$I = I_1 = I_2 = I_3$$

រូបមន្តតណាងសីស្សដែលជាសំណង់បច្ចេកទេស

$$\frac{1}{R} = \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3}$$

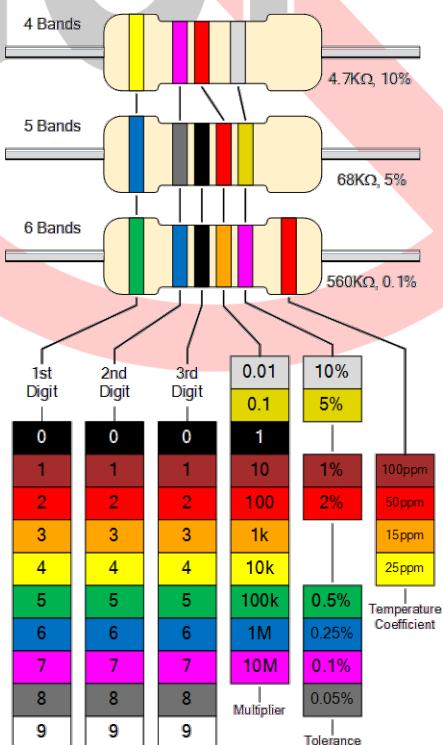
$$V = V_1 = V_2 = V_3$$

$$I = I_1 + I_2 + I_3$$

រូបមន្តតណាងកម្មវិធាន

$$P = V \times I$$

- របៀបមើលតម្លៃសីស្សដែលត្រួតពិនិត្យតម្លៃខ្លួន (Resistor Fixed Value)



១.២.២ កាត់ស្ទា

កាត់ស្ទា ជាក្រឹងអេឡិចត្រូនិកមួយប្រភេទដែលមានសមត្ថភាពធ្លីក និងធ្លីរចាមពល អតិសនី។ កាត់ស្ទា បូកដែលមានលក្ខណៈស្ទាត់ត្រឹមការងារបន្ទាន់លាយៗ: ពីរាក់ស្របត្រា និង មានសុលម្បូរសុំខ្លួចក្រិចនៅថ្ងៃឆ្នាំ: ទាំងពីរភាពៗ។ កាត់ស្ទា៖ មានពីរប្រភេទ គឺ កាត់ស្ទា៖ មានបូល កាត់ស្ទា៖ អគតុល និងកាត់ស្ទា៖ មានតម្លៃប្រហែល។

- កាត់ស្ទា៖ មានបូល



- កាត់ស្ទា៖ ត្រូវបូល



- កាត់ស្ទា៖ មានតម្លៃប្រហែល



រូបមន្ត្រឡេទេ ៖

$$C = \epsilon \frac{A}{d}$$

- C: តម្លៃកាត់ស្ថើរគិតជាគ F
- A : ផ្ទៃវីរបន្ទះលោហ៍គិតជា m^2
- d: ប្រវែងគំលាកពីបន្ទះលោហ៍គិតជា m
- ε : តម្លៃដែនខិះអេឡិចត្រូនិក

រូបមន្ត្រគណនាករយៈពេលធ្វើក និងរយៈពេលឡើង ៖

$$\tau = R \cdot C$$

- τ : រយៈពេលធ្វើក និងឡើងកាត់ស្ថើរ
- R: តម្លៃហសិស្សនី (orm)
- C: តម្លៃកាត់ស្ថើរ

គណនាបង្កាត់កាត់ស្ថើរជាលើសទី ៖

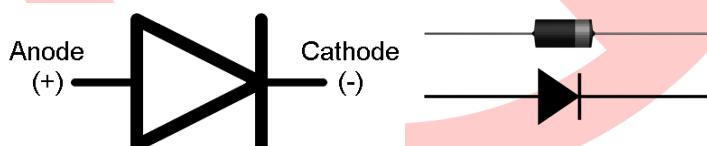
$$\frac{1}{C} = \frac{1}{C_1} + \frac{1}{C_2} + \frac{1}{C_3}$$

គណនាបង្កាត់កាត់ស្ថើរជាលើខ្លួន ៖

$$C = C_1 + C_2 + C_3$$

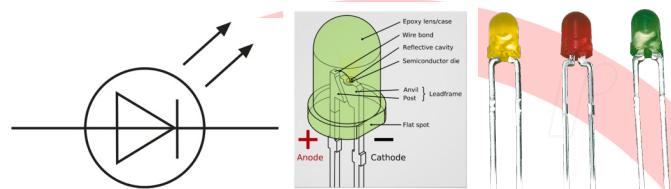
១.២.៣ ឌីយីត

ឌីយីត ជាប្រភេទអេឡិចត្រូនិកតាក់កណ្តាលបច្ចេកដែលបានបង្កើតឡើងដោយប្រព័ន្ធឌីយីតិស (ទិសដៅតែមួយ បុងកទិស)។



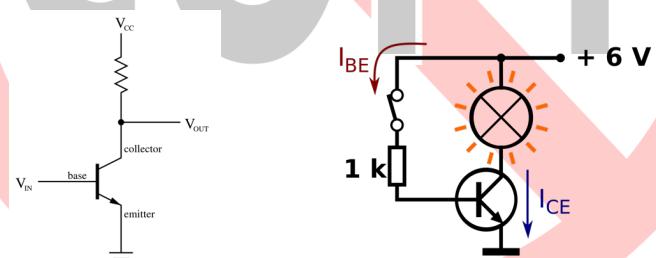
១.២.៤ LED (Light-emitting diode)

LED ជាក្រឹងអេឡិចត្រូនិកតាក់កណ្តាលចម្លងមួយប្រភេទ ដែលរានឱងបញ្ចប់ពន្លឺនូវរោល ដែលមានចំនួនអគ្គិសនិត្យនៃកាត់វា។

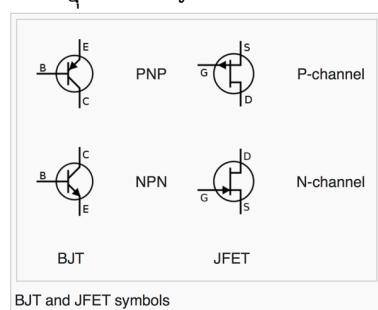


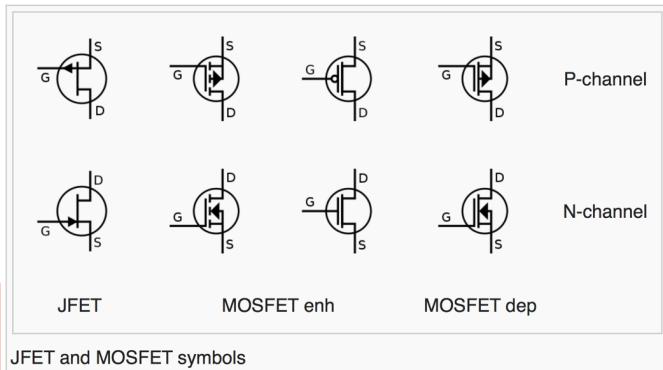
១.២.៥ ត្រង់សុស្សវា

ត្រង់សុស្សវា ជាក្រឹងអេឡិចត្រូនិកតាក់កណ្តាលចម្លង ដែលរានឱងប្រាស់វាជាមួយកណ្តើកសុឆ្នាំលំ បុចាកុកតាក់អេឡិចត្រូនិកជាដីមោ។ ត្រង់សុស្សវាពានបែងចែកជាបីស្ថានីយដាច់ឆ្នាំ ថ្មីនៃក្រុងបញ្ហាប់សៀវភៅនិងក្រុងឆ្នាំ (External Circuit) ដែលនៅក្នុងនោះមានផូចធានាណីង B (Base) និង C (Collector) និងផូចធានារឿង E (Emitter) ដែលការិករារនៃតែលដែលមានកម្រិតបរិមាណចរន្ត អគ្គិសនិដែលនឹងផ្តល់ឲ្យនីង B (Base) ត្រូវបានតាមការរបស់ប្រភេទត្រង់សុស្សវា នោះវានឹងងើឱ ឲ្យនីង C (Collector) និងនីង E (Emitter) ជាប់ឆ្នាំ។

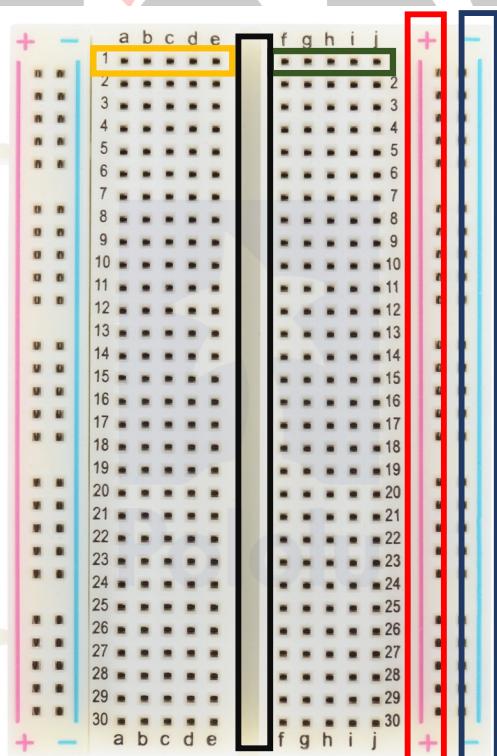


ត្រង់សុស្សវាទានបង្កើតឡើងជាក្រឹងប្រភេទដែលរានក្នុងទី និងតម្រូវការតាមការងារ ដែលយើងត្រូវប្រាស់ ដែលនៅក្នុងនោះមានផូចធានា ៖





៩.២.១ Bread Board



- ចតុកាលកណ្តាលក្រហម : ចំណុចដោតនៅក្នុងចតុកាលកណ្តាលក្រហមជាប់ត្រាទាំងអស់

- ចតុការណាគណ៍ខ្សោះ : ចំណុចដោតនៅក្នុងចតុការណាគណ៍រួចរាល់ជាទាំងអស់បុំន្តូចំណុចដោតនៅក្នុងចតុការណាគណ៍ខ្សោះ មិនជាប់ជាមួយចំណុចដោតនៅក្នុងចតុការណាគណ៍ក្រហមនៅទេ
- ចតុការណាគណ៍ទីក្រុង : ចំណុចដោតនៅក្នុងចតុការណាគណ៍ទីក្រុងជាប់ឆ្នាំ ទាំងអស់ (ជាប់មួយឆ្នាំ)
- ចតុការណាគណ៍បែងចែង : ចំណុចដោតនៅក្នុងចតុការណាគណ៍បែងចែងជាប់ឆ្នាំ អស់ (ជាប់មួយឆ្នាំ) បុំន្តូចំណុចដោតនៅក្នុងចតុការណាគណ៍បែងចែង មិនជាប់ឆ្នាំទេនឹងចំណុចដោតនៅក្នុងចតុការណាគណ៍ទីក្រុងឡើយ
- ចតុការណាគណ៍ខ្សោះ : ជាកំនែងខណ្ឌត្រូវចំណុចដោតចតុការណាគណ៍បែងចែង និងចំណុចដោតណណ៍ទីក្រុង

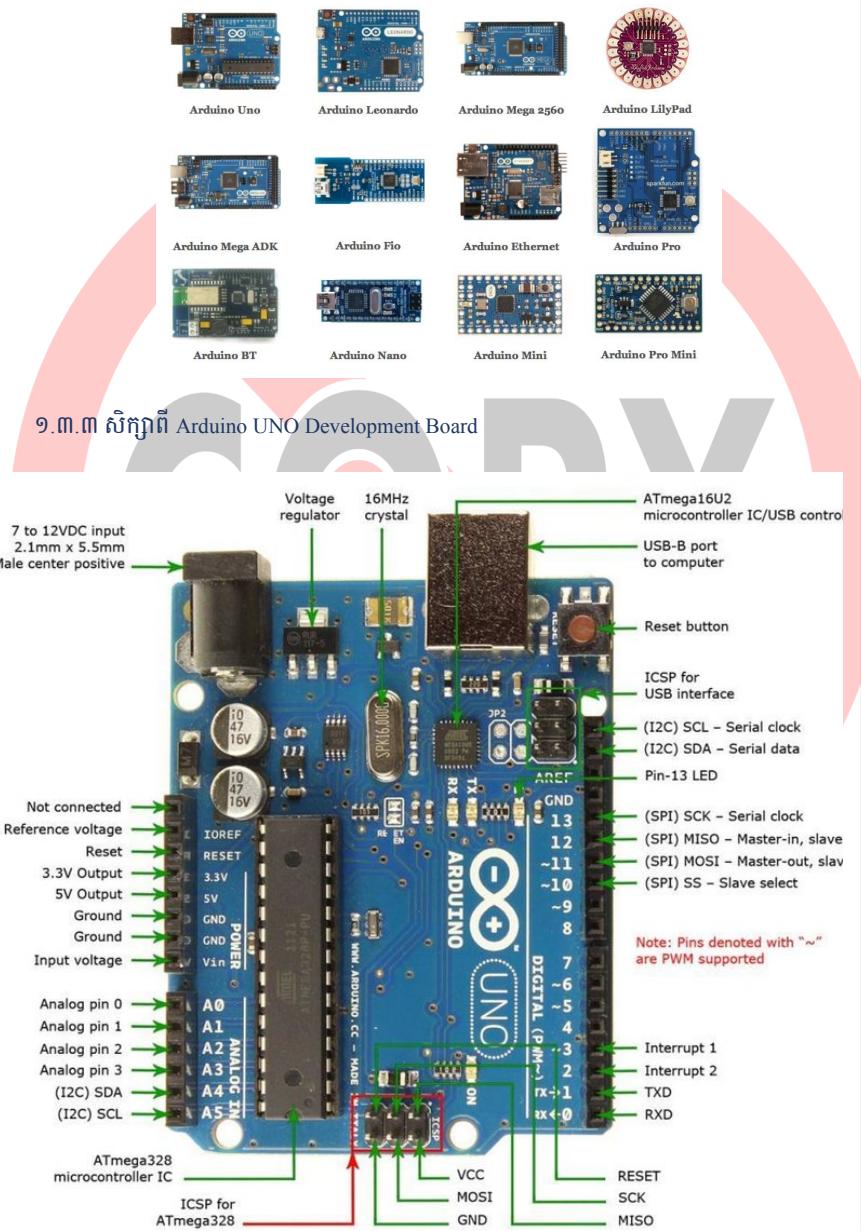
៩.៣ Arduino

៩.៣.១ តើវិញ្ញាន Arduino ?

Arduino ឬ Microcontroller ដែលគេបានឈានឈិកឡើងជាលក្ខណៈ: Open source ទាំងអ្នក Hardware និងផ្នែក Software ដែលគេឱយមយករាយបែបប្រាស់ជាមួយនឹងការគ្រប់គ្រង ឬផ្តិចត្រូវឱ្យការ ឧទាហរណ៍ជាក់ស្តីងួចធាន ម៉ាស៊ីនដែលដំណើរការដោយស្វ័យប្រវត្តិ និងរូបុពជានិមិត្ត Arduino ឬ Development Board ដែលមានសារ: សំខាន់សម្រាប់អ្នកដែលមានបំណងចង់សិក្សាស្មានយល់បង្កើមទៅលើការសរស់រារ Program ក្នុងក្រុងការសរស់រារ program ។ ចំណាត់ការការណា program របស់រារបានដោយភ្លាម់ C++ និង C និង C# ។ នៅក្នុងការសរស់រារ program បញ្ចូលទៅក្នុង Arduino Board យើងប្រើការ Arduino IDE ដែលយើងអាចទាញយកបាននៅក្នុង Website នូវការរបស់ Arduino និងបែបប្រាស់ប្រភេទ USB តាមប្រភេទ របស់ Arduino Board ។ ចំពោះ Arduino IDE យើងអាចប្រើប្រាស់នៅលើកុំពុំទៅកាន ប្រើប្រាស់នៅលើទូរសព្ទដូចនេះ Android កំពុង។

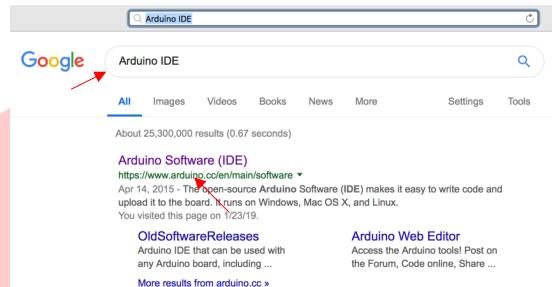
៩.៣.២ ប្រភេទ Arduino

Arduino ជ្រើនបានឈានឈិកឡើងមានប្រើប្រាស់ដែលយើងអាចប្រើប្រាស់នឹងបានតាមទំហំការងាររបស់យើងបាន ដែលនៅក្នុងនៅទេ: មានដូចជា Arduino RS232, Arduino Diecimila, Arduino Due, Arduino Uno, Arduino Uno SMD, Arduino Leonardo, Arduino Pro, Arduino Mega, Arduino Nano, Arduino LilyPad, Arduino DUE និងមានជាប្រើប្រាស់ប្រភេទទៅត្រូវបាន ប្រភេទ Arduino ទាំងអស់ខ្ពស់នាមទៅលើឈើប្រពិបត្តិការ ទំហំអង្គធិនុយ និងឃើង (Pin) ដែលគ្រប់ប្រើប្រាស់ជាមួយផ្នែក Input និងផ្នែក Output ។



១.៣.៣ របៀបតម្លៃកម្មវិធី Arduino IDE

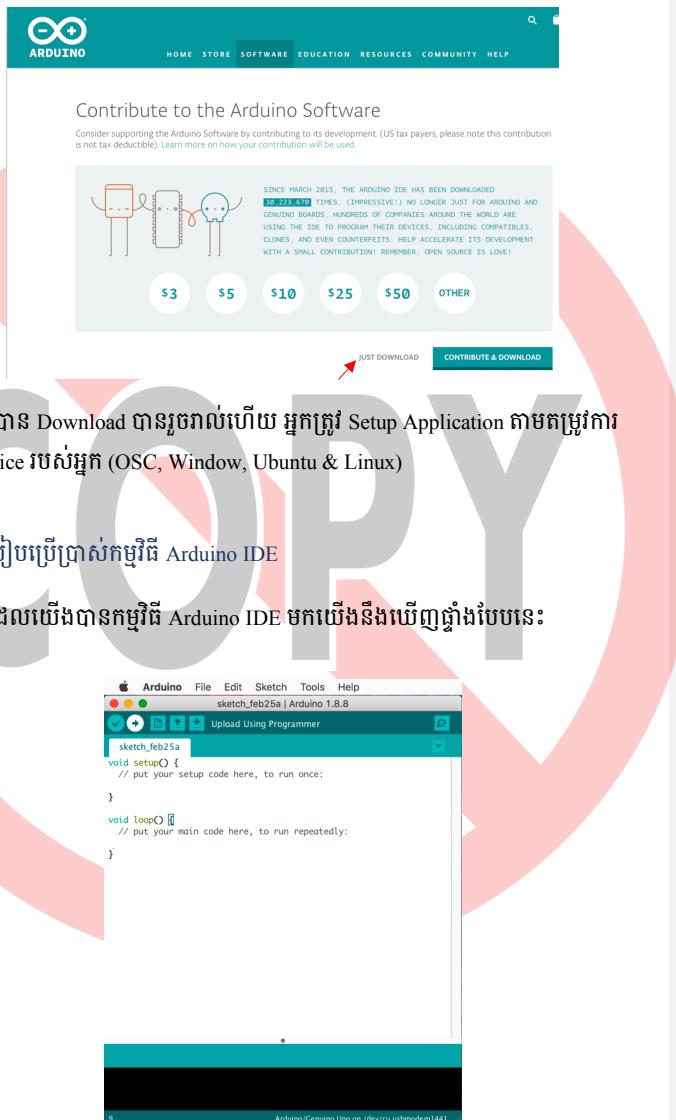
- ជំហានដំបូងចុលទៅកាន់ Google រួចរាល់ថា Arduino IDE



- បន្ទាប់មកទូទាត់ទូទាត់ Arduino Software (IDE) រួចរាល់សម្រាប់ក្នុង Application ដែលត្រូវជាមួយ Laptop ឬ PC របស់អ្នក



- បន្ទាប់មកទូទាត់ទូទាត់ Just Download វានឹងទាយយ៉ាង Application នៅ៖មកជាក់នៅក្នុង Device របស់អ្នក



- បន្ទាប់ពីការ Download តាមរយៈការចូលរួមគ្នា Setup Application តាមកម្រវត្ថុរារបស Device របស់អ្នក (OSC, Window, Ubuntu & Linux)

៩.៣.៤ សិក្សាតីរបៀបបង្កើតកម្មវិធី Arduino IDE

- នៅពេលដឹងទីការណ៍កម្មវិធី Arduino IDE មកយើងនឹងយើលដ្ឋានខាងក្រោម៖

```

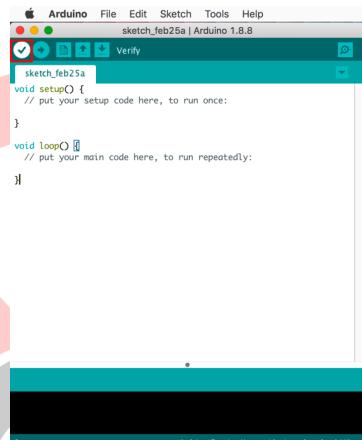
Arduino File Edit Sketch Tools Help
sketch_feb25a | Arduino 1.8.8
Upload Using Programmer
sketch.feb25a
void setup() {
  // put your setup code here, to run once:
}

void loop() {
  // put your main code here, to run repeatedly:
}

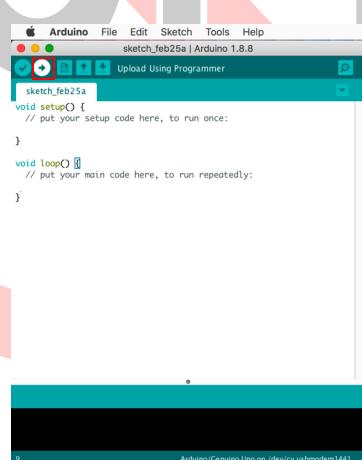
```

Arduino/Genuino Uno on /dev/cu.usbmodem1411

- ចូចប៉ុណ្ណោះ Verify ដើម្បីបញ្ជាក់ថានឹកការសរសេរ Program របស់យើងមានកំហុសផ្តល់បច្ចេកទេសត្រួចណា បូអន្ត

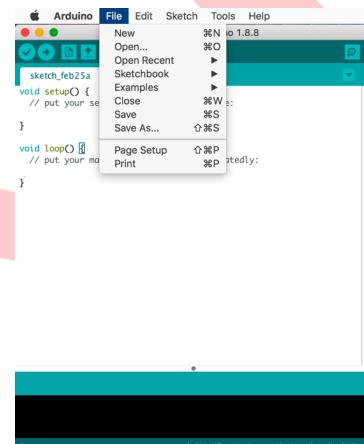


- ចូចប៉ុណ្ណោះ Upload ដើម្បីបញ្ចូន Program ដែលយើងបានសរសេរទៅកាន់ Arduino Board

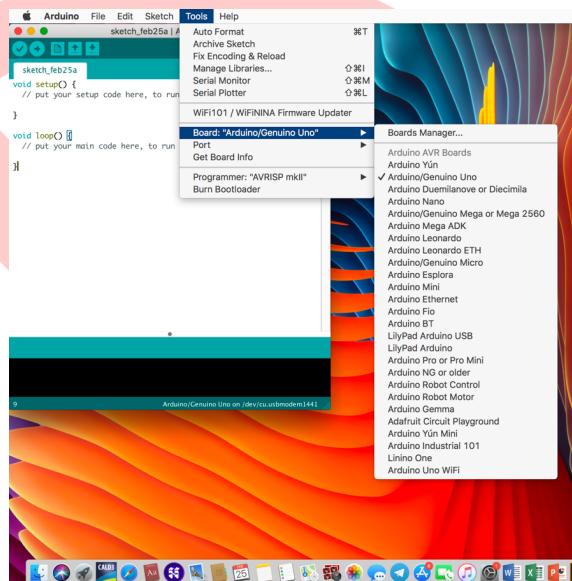


- ចូចលើប៉ុណ្ណោះ Serial Monitor សម្រាប់បើកមិនទទួលខ្លួនយើងបាន Read បុកឱ្យ Print (ដែលឈានលូបត្រាតែយើងកំណត់នៅក្នុង Program)

- នៅលើដែលដែលមួកចុចចូល File យើងនឹងយើងមួយខាងសំខាន់ៗមួយចំនួនដូចជា
 - New: ស្របតាមឯកសាររបស់របស់គ្រប់គ្រង់
 - Open: ស្របតាមឯកសាររបស់របស់គ្រប់គ្រង់
 - Open Recent: ស្របតាមឯកសាររបស់របស់គ្រប់គ្រង់ដែលយើងទីប៉ុណ្ណោះ
 - Example: ស្របតាមឯកសារយុទ្ធសាស្ត្រយុទ្ធសាស្ត្រ
 - Close: ស្របតាមឯកសារយុទ្ធសាស្ត្រ
 - Save ឬ Save As: ប្រើស្របតាមឯកសារយុទ្ធសាស្ត្រ
 - Page Setup: ស្របតាមឯកសារយុទ្ធសាស្ត្រ



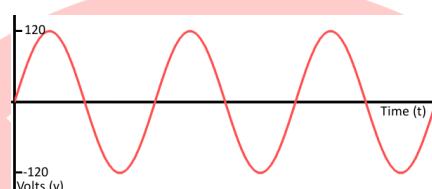
- នៅពេលដែលយើងចូចចូល Tool អ្នកគ្រប់ដីនឹងមុខងារធីសំខាន់ផ្តើមជាតា ៖
 - Board: សម្រាប់ប្រើប្រាស់ដីសប្តាហ៍ Arduino Development Board ដែលអ្នកនឹងត្រួររបស់ប្រាស់ជាមួយ Arduino IDE
 - Port: សម្រាប់ប្រើប្រាស់ដីស Port ដែលអ្នកកំណើនប្រើប្រាស់ Arduino Development Board ជាមួយ PC ឬ Laptop



ទំព័រ ២ Basic Digital & Basic Programming

២.១ Analog

Analog ជាសិក្សាល់ដែលគេប្រើប្រាស់សម្រាប់បញ្ជីទិន្នន័យ ដែលនៅក្នុងការប្រើប្រាស់ជាមួយ Arduino Development Board តម្លៃរហស់សិក្សាល់ Analog មានចែងចាំពី 0 ដល់ 1023 (10Bit)។



នៅក្នុងការប្រើប្រាស់ Arduino Development Board ជាមួយ Sensor ប្រភេទ Analog គឺនឹងមានប្រើប្រាស់ជាការកំណត់តម្លៃនៅក្នុងចែងចាំណាយអម្ចាស់ដើម្បីត្រួវមានប្រតិបត្តិការណាយអម្ចាស់ឡើង។ ឧបាទរណ៍ដូចជា LDR ដែលជាប្រភេទ Sensor មានតម្លៃប្រចាំឆ្នាំអាសយដ្ឋាននៅក្នុងក្រុងក្រោមតុល្យ។ គឺតែងតែកំណត់តម្លៃ Analog ដែលទទួលបានពីខាងក្រោម LDR ដើម្បីទៅប្រព័ន្ធបាន (Controller) ទៅលើ Output Device ណាយអម្ចាស់ដូចជាពេលយប់ ធ្វើឲ្យអំពុលភ្លើជាដើម។

២.២ Digital

Digital ជាសិក្សាល់ដែលគេប្រើប្រាស់សម្រាប់បញ្ជីទិន្នន័យអូចត្រាទីនីង Analog ដែរ ប៉ុន្តែ តម្លៃរហស់សិក្សាល់ Digital មានទៅពីរបុរុណុំ គឺ High (1) និង Low (0)។



នៅក្នុងការប្រើប្រាស់ Arduino Development Board ជាមួយ Sensor ប្រភេទ Digital គឺតែងតែកំណត់តម្លៃខ្លួនបែប ពិត (High :1) បូចិនពិត (Low :0) នៅពេលដែលបញ្ចូនឈាម ពិត (High:1) ឬប្រើប្រាស់ Output Device ណាយអម្ចាស់ដែលការពើឱ្យអម្ចាស់ ជាក់ស្តីដូចជា Motion Sensor ឬ PIR Sensor ឬប្រភេទ Digital Sensor ដែលបានសកម្មភាពមុនស្សែ ស្វែ បុក្រឹមូយៗដែលមានចលនាទាមការកំណត់ពីការកំណត់អាជាហរណ៍ពេលមានមុនស្សែដើម្បីកាត់ Motion Sensor ដើម្បីការងារ Controller និងទទួលបានទិន្នន័យពីខាង Sensor ឬចបញ្ជាញអំពុលភ្លើជាដើម។

៤.៣ Variable Type

វាតាការកំណត់ប្រភេទរបស់អចេរ ដើម្បីការកំណត់នៃវិធីទូទៅប្រភេទអចេរនេះទាំង ព័ត៌មានកំណត់អចេរនេះវាទានសារ៖ សំខាន់លាងស៊ែរ ដើម្បីយើងត្រូវដឹងទុក ច្បាស់ថា តើគេត្រូវឱ្យអចេរដើម្បីបានកំណត់នេះយើងយកទៅប្រើប្រាស់នៅក្នុងលក្ខខណ្ឌបែបណា ហើយនិងជាយើងកំណត់តើអចេរខ្លួនតើលក្ខខណ្ឌដើម្បីយើងនឹងយកមកប្រើប្រាស់ វិធីទូទៅប្រភេទ តាមដែលដឹងទុក នៃលក្ខខណ្ឌសរសរ។

ប្រភេទ តម្លៃបូត្រូវ	Bytes	តម្លៃកំណត់	ពន្យល់
int	2	-32768 ទៅ 32767	ប្រើប្រាស់ជាមួយចំនួនលេខគត់ ចំនួនវិធីទាន ចំនួនអវិធីទាន
unsigned int	2	0 ទៅ 65535	ខុសត្រាតី int ត្រូវបានប្រើប្រាស់បានពេលចំនួនវិធីទាន
long	4	-2147483648 ទៅ 2147483647	ផ្ទុចត្រូវឱ្យ int តែខាងក្រោមថា long ចំនួនទៅដល់ខ្លះ ពាន់បាន
unsigned long	4	0 ទៅ 4294967295	ខុសត្រាតី long ត្រូវបានប្រើប្រាស់បានពេលចំនួនវិធីទាន
float	4	-3.4028235E + 38 ទៅ 3.4028235E + 38	ប្រើប្រាស់ជាមួយអចេរដើម្បីលក្ខខណ្ឌតម្លៃបូត្រូវ
double	4	-3.4028235E + 38 ទៅ 3.4028235E + 38	ផ្ទុចត្រូវឱ្យ float ដើម្បីខុសត្រាតីកំណត់តម្លៃលេខរបស់ double បានក្រួចតូលរៀងនិងក្រួចចំនួន
char	1	-128 ទៅ 127	ប្រើប្រាស់ជាមួយក្នុងក្រួចចំនួន
boolean	1	1 : ពិត 0 : មិនពិត	ប្រើប្រាស់ជាមួយតម្លៃកិត្តិក និងតម្លៃមិនគិត

- ចំពោះអញ្ញតិត្តិកនៃតម្លៃទៅ ហើយស្ថិតិនៅក្នុងប្រភេទអញ្ញតិត្តិកនេះហើយក្នុងប្រភេទ Keyword មួយនៅពីមួយ តើ Const ។ ឧទាហរណ៍ Const int, Const float,.....

៣.៤ Further Syntax

ប្រភេទ Further Syntax	មុខងារ
; (ចុចក្បាស)	ប្រើប្រាស់នៅពេលដំឡើងបញ្ជី Statement
ឃ (ត្រូវបាន)	សម្រាប់ប្រើប្រាស់នៅពេលដំឡើងបើកអនុគមន៍ និង ចិត្តអនុគមន៍
// (Comment មួយឯង)	សម្រាប់បិទកុងក្នុងជូនណាមួយ បើកទុកសរស់រៀបចំណាំនៅក្នុង Program
/* */ (Comment ចាប់ពីនូវឯង)	ឧស្សារី // ក្នុងថាគារប្រើប្រាស់សម្រាប់បិទប្រើប្រាស់ជូន
#define	ប្រើប្រាស់សម្រាប់ការប្រកាសនិមិត្តិសញ្ញា
#include	ការទាញយក Library មកប្រើប្រាស់នៅក្នុង Sketch ដើម្បីដាក់បង្កើតសាស្តរ

៣.៥ Arithmetic Operator

+, -, *, /, % ទាំងអស់នេះជា Arithmetic Operator ដើម្បីប្រើប្រាស់សម្រាប់ធ្វើប្រាកាណវិធី បួន ឬក ឬក គុណុក ឬកនិង ចែកយកសំណល់។

៣.៦ Comparison Operator

ប្រភេទ Comparison Operator	ការប្រើប្រាស់
==	ប្រើប្រាស់សម្រាប់ប្រើប្រាស់បញ្ជីព័ត៌មានថីរស្សីគ្នា
!=	ប្រើប្រាស់សម្រាប់ប្រើប្រាស់បញ្ជីព័ត៌មានថីរស្សីគ្នា
<	ប្រើប្រាស់សម្រាប់ប្រើប្រាស់បញ្ជីព័ត៌មានថីរស្សីគ្នា
>	ប្រើប្រាស់សម្រាប់ប្រើប្រាស់បញ្ជីព័ត៌មានថីរស្សីគ្នា
<=	ប្រើប្រាស់សម្រាប់ប្រើប្រាស់បញ្ជីព័ត៌មានថីរស្សីគ្នា បុស្សីរ
>=	ប្រើប្រាស់សម្រាប់ប្រើប្រាស់បញ្ជីព័ត៌មានថីរស្សីគ្នា បុស្សីរ

៣.៧ Bitwise Operator

អនុគមន៍កញ្ចឹម។ AND, OR, NOT, XOR, Shift Left, Shift Right ដើម្បីទាំងអស់នេះវានឹងកើតមានឡើងនៅពេលប្រកែតំបន់បុគ្គលិក។

- AND GATE

$A \text{ AND } B = A \cdot B$

A	B	A.B
0	0	0
0	1	0
1	0	0
1	1	1

- OR GATE

$A \text{ OR } B = A + B$

A	B	A.B
c	0	0
0	1	1
1	0	1
1	1	1

- NOT GATE

$\text{NOT}(A) = \bar{A}$

A	\bar{A}
0	1
1	0

- XOR GATE

$A \text{ XOR } B = A \oplus B$

A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

Arduino Symbol

AND	&
OR	
NOT	~

XOR	\wedge
Shift Left	$<<$
Shift Right	$>>$

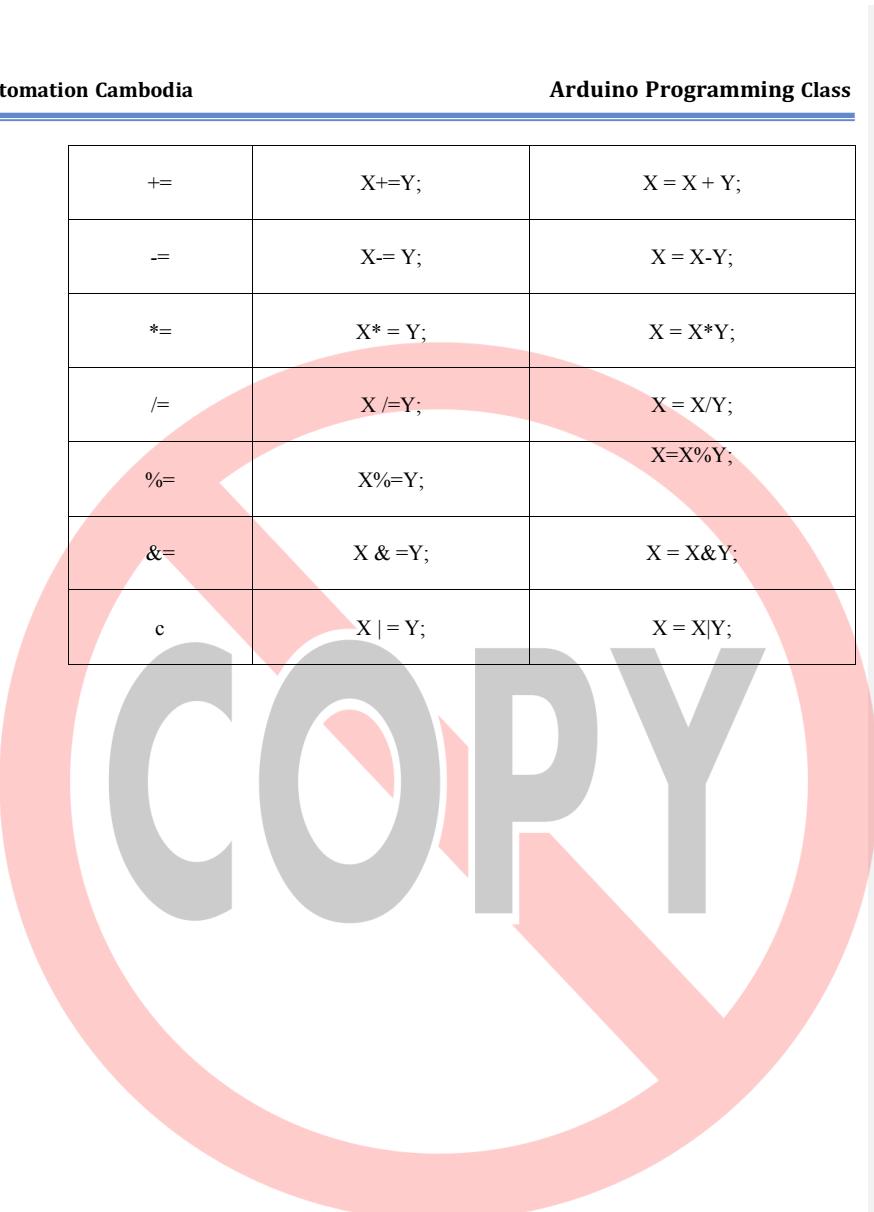
៤.៤ Boolean Operator

Arduino Symbol	ការប្រើប្រាស់
AND	$\&\&$ ធ្វើការប្រើប្រាស់បញ្ចូលតុលាតី បុរីនិងជាន់នឹង បើលក្ខខណ្ឌទាំងអស់ពីត គឺពីត នៅបីលក្ខខណ្ឌណាមួយមិនពិត នោះលក្ខខណ្ឌទាំងអស់តី មិនពិត
OR	$\ $ ធ្វើការប្រើប្រាស់បញ្ចូលតុលាតី បុរីនិងជាន់នឹង បើលក្ខខណ្ឌណាមួយពិត គឺពិត នៅបីលក្ខខណ្ឌទាំងអស់តែមិនពិត នោះលក្ខខណ្ឌទាំងអស់តី មិនពិត
NOT	លក្ខខណ្ឌនេះពិត ឬ៖ត្រូវកត្តិថ្លែងប្រើបង្កើតដោយបង្កើតឯកជាមួយនឹងប្រព័ន្ធបង្កើតឯកជាមួយនឹងប្រព័ន្ធ

៥.៤ Boolean Operator

Arduino Symbol	ឧបាទរណ៍	លទ្ធផល
++	X++;	ឬក X ដោយកត្តិថ្លែង 1 នៃលទ្ធផលទូលាតានតី យកតម្លៃចាស់មុននឹងឬក
	++X;	ឬក X ដោយកត្តិថ្លែង 1 នៃលទ្ធផលទូលាតានតី យកតម្លៃដែលបានឬកហើយ
--	X--;	ដីក X ដោយកត្តិថ្លែង 1 នៃលទ្ធផលទូលាតានតី យកតម្លៃចាស់មុននឹងដីក
	--X;	ដីក X ដោយកត្តិថ្លែង 1 នៃលទ្ធផលទូលាតានតី យកតម្លៃដែលបានដីកហើយ

<code>+=</code>	<code>X+=Y;</code>	<code>X = X + Y;</code>
<code>-=</code>	<code>X-= Y;</code>	<code>X = X-Y;</code>
<code>*=</code>	<code>X*= Y;</code>	<code>X = X*Y;</code>
<code>/=</code>	<code>X /=Y;</code>	<code>X = X/Y;</code>
<code>%=</code>	<code>X% =Y;</code>	<code>X=X%Y;</code>
<code>&=</code>	<code>X & =Y;</code>	<code>X = X&Y;</code>
<code> </code>	<code>X = Y;</code>	<code>X = X Y;</code>



COPY

វំរូន ៣ Programming ទៅលើ Arduino

៣.៩ Blink LED

- ការរំភោតសៀវភៅ

តម្រូវការគ្រប់បង្គែត :Bread-Board, LED, Resistor 220Ω, Arduino Uno និងខ្លួនភ្លើង

- ផ្តល់

```

1. int led =8;
2.
3. void setup() {
4.   pinMode(led,OUTPUT);
5. }
6.
7. void loop() {
8.   digitalWrite(led,HIGH);
9.   delay(1000);
10.  digitalWrite(led,LOW);
11.  delay(1000);
12. }
```

ក្នុងខាងលើនេះធ្វើឡើងអំពុលភ្លើយ៖ពេល 10 វិនាទី រួចរាល់ទៅវិញ (ភ្លើលតែម្រាត)

Syntax & Parameter:

`pinMode(pin,mode)`

- pin: ជាលេខដឹងដែលប្រើប្រាស់ជាមួយ Arduino Board
- mode: INPUT, OUTPUT, INPUT_PULLUP & INPUT_PULLDOWN

`digitalWrite(pin,value)`

- pin: ជាលេខដឹង Output ដែលបានគ្រប់ជាមួយ Arduino
- value: ជាដែនល្អែង Digital ដែលប្រើបញ្ជាញទៅពិត (HIGH: 1) ឬមិនពិត (LOW: 0)

`delay(ms)`

- ms: វិយោបេលពិតជាមីលវិនាទី

`delayMicroseconds (us)`

- us: វិយោបេលពិតជាមីលវិនាទី

a. `if (conditional) and ==,!=,<,>`

ឯកត្រានគប្រើប្រាស់នៅក្នុងលក្ខខណ្ឌដែលបានការប្រៀបដែលបានពិត ឬមិនពិត។

Syntax: `if (variable < 20)`

```
{  
    //do something here  
}
```

Example :

```
if (x < 20)  
{  
    digitalWrite(led,HIGH);  
}
```

```
if (x < 20) digitalWrite(led,HIGH);
```

```
if (x < 20) { digitalWrite(led,HIGH);}
```

ការសរសេរទាំងបីរបៀបនេះអ្នកអាចប្រើសិនិសការសរសេរតាមរបៀបមួយណាក៏។

b. `if/else`

ឯកត្រានគប្រើប្រាស់ជាមួយ `if` ដែលវាងដីរការតាមនៅពេលដែលប្រើប្រាស់នៅក្នុង `if` មិនពិត នៅការនៅក្នុង `if` ហើយចូលទៅដីរការក្នុងនៅក្នុង `else` វិញ។

Syntax:

```
if (variable > 20)
{
    //do something here;
}
else if (variable == 100)
{
    //do something here;
}
else
{
    //do something here;
}
```

c. For Statement

រាជការនៃមការងារមួយដែលដំណើរការដែលលាស់**ប្រព័ន្ធសិតនៅក្នុងចំនួនកំណត់ជាក់លាក់មួយ។**

Syntax:

```
For (initialization; condition; increment)
{
    // do something here
}
```

Example:

```
for (int x = 0; x <= 50; x++) {
    println(x);
}
```

```
for(int x = 0; x < 100; x++) {
    println(x); // prints 0 to 99
}
```

The diagram shows the structure of a for loop. It starts with 'for(' followed by a set of parentheses containing three parts: 'int x = 0', 'x < 100', and 'x++'. Inside the parentheses, arrows point from left to right, labeled 'declare variable (optional)', 'initialize', 'test', and 'increment or decrement' respectively. Below the loop, the code 'println(x); // prints 0 to 99' is shown.

d. Switch / case Statements

ដំណើរការនេះ **Switch / case statement** រាជការប្រព័ន្ធដ្មីន **if** ដែលខ្ពស់ត្រង់ថាយកព័ត៌មានអមពេមួយ ទៅប្រព័ន្ធដ្មីនព័ត៌មានដែលបានកំណត់ជាប្រព័ន្ធ នៅក្នុងករណីដែលព័ត៌មានអមពេមាន៖ មានព័ត៌មានដ្ឋានទៅនឹងកំណត់ជាប្រព័ន្ធទាំងឡាយ ក្នុងករណីប្រព័ន្ធដ្មីនរាជការក្នុងចំណួននេះ។ នៅពេលដែលគោរពក្នុងករណីប្រព័ន្ធដ្មីន ចូលរួមការក្នុងចំណួននេះ គឺជាបញ្ជាផ្ទៃក្នុងការប្រព័ន្ធដ្មីន។

Syntax:

```
Switch (x)
{
    case value:
        // do something here
    break;
```

```
case value:  
    //do something here  
    break;
```

```
case value:  
    //do something here  
    break;
```

```
default:  
    //do something here  
    break;  
}
```

Parameter:

x: ជាអាយុយដែលត្រូវបានអចេកនៃការពារក្នុងក្រុមហ៊ុយ
value: តម្លៃដែលបានគ្រប់គ្រងដោយក្រុមហ៊ុយ

Syntax:

```
Switch (x)  
{  
    case 1:  
        // do something here when x =1  
        break;  
    case 2:  
        //do something here when x = 2  
        break;  
    case 3:  
        //do something here when x = 3  
        break;  
  
    default:  
        //do something here when nothing else matches  
        break;  
}
```

e. while loops

while loops ជាការកំណត់តម្លៃខណ្ឌមួយដែលនឹងធ្វើការដោយក្រុមហ៊ុយ នៅពេល
ដែល Expression បែស់ភាពិត នៅពេលដែល Expression បែស់ភាពិតនឹងបញ្ជូរថី
ដំណឹកការនោះភ្លាម។

Syntax:

```
while (expression)  
{  
    //do something here  
}
```

Parameters:

expression: នៅក្នុងករណីដែល expression expression ពិតាគនោះវានឹងធ្វើប្រកិច្ច
ការដែលខ្លួន នៅក្នុងករណីដែល expression មិនពិត នោះវានឹងបញ្ចប់ប្រកិច្ចការក្រោង
ចំណុចនោះ។

Example:

```
x = 0;
while (x < 100)
{
    //do something repetitive 100 times
    x++;
}
```

f. **do-while**

ដំណើរការស្រប់នូវគ្មាននៃ **while loops** នៅរ ខ្ពស់គ្រប់ចាប់ផ្តើមការរួចរាល់ហើយ
ទីមក្សាស្របតាមការបញ្ជូនការក្រោងចំណុចនោះ។

Syntax:

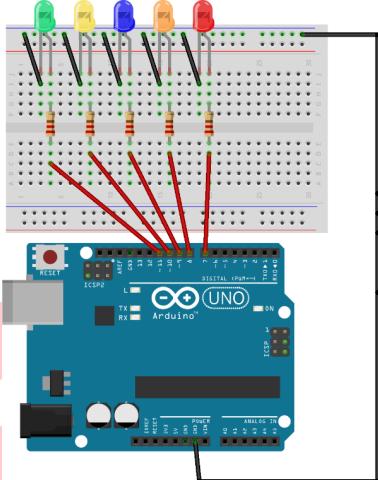
```
do
{
    //do something here
}
while (condition);
```

Example:

```
int x = 0;
do
{
    delay(50);
    //do something here
}
while (x < 100);
```

អនុវត្តន៍ : ប្រើសិសមអំពុល LED 5 ត្រាប់មកត្រាប់ជាមួយ Arduino UNO ដែលត្រូវដោក់ជាថ្មី។ អ្នកត្រូវ
ព្យាយាប់ :

- អំពុលទី 1 ទៅដើមទី 7 របស់ Arduino
- អំពុលទី 2 ទៅដើមទី 8 របស់ Arduino
- អំពុលទី 3 ទៅដើមទី 9 របស់ Arduino
- អំពុលទី 4 ទៅដើមទី 10 របស់ Arduino
- អំពុលទី 5 ទៅដើមទី 11 របស់ Arduino

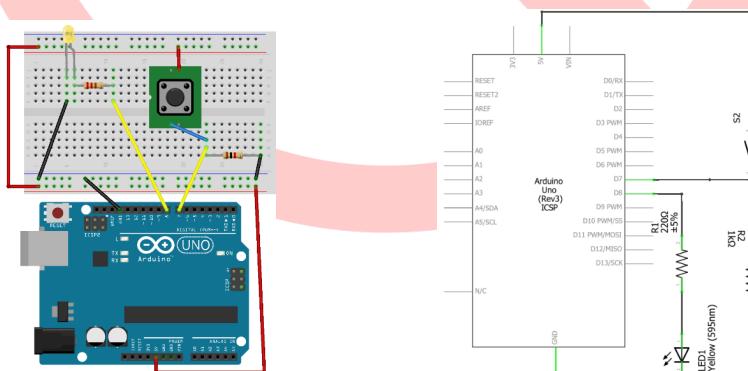


បន្ទាប់មកទី៥ តម្លៃកត្រូវសរសរុជប្រអប់ទាំង 5 នៅខាងលើប្រព័ន្ធឌាតិ ដើម្បី ស្នើរួច ពី ស្នើមក ធ្វើងវិញ ដើម្បីការដែលនៅក្នុង loop មួយ។

៣.២ Digital Input

Digital Input ជារបៀបដឹងប្រព័ន្ធឌាតិ ទិន្នន័យប្រព័ន្ធគឺ Digital តាមរូប: ដើម្បី Digital របស់ក្នុង

ការរក្សាយ



ក្នុង

```

1. int led=8;
2. int button=7;
3.
4. void setup()
5. {
6.   pinMode(led,OUTPUT);
7.   pinMode(button,INPUT);
8. }
9.
10.void loop()
11.{}
12. if(digitalRead(button) == HIGH)
13. {
14.   digitalWrite(led,HIGH);
15. }
16.

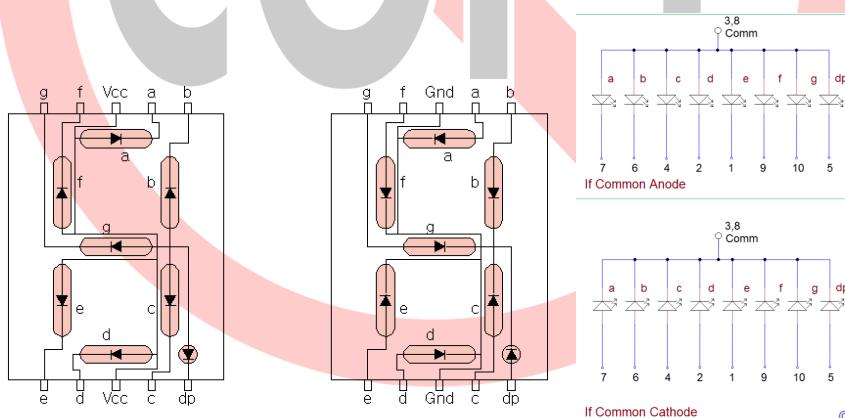
```

Syntax:

- **digitalRead(pin):** pin លេខដឹង Input ដែលត្រូវប់ជាមួយ Arduino
- **digitalWrite(pin):** pin លេខដឹង Output ដែលបានត្រូវប់ជាមួយ Arduino
- **pinMode(pin, mode) :** pin លេខដឹងដែលបានត្រូវប់ជាមួយ Arduino ។ mode ប្រកាសជា Input ឬ Output

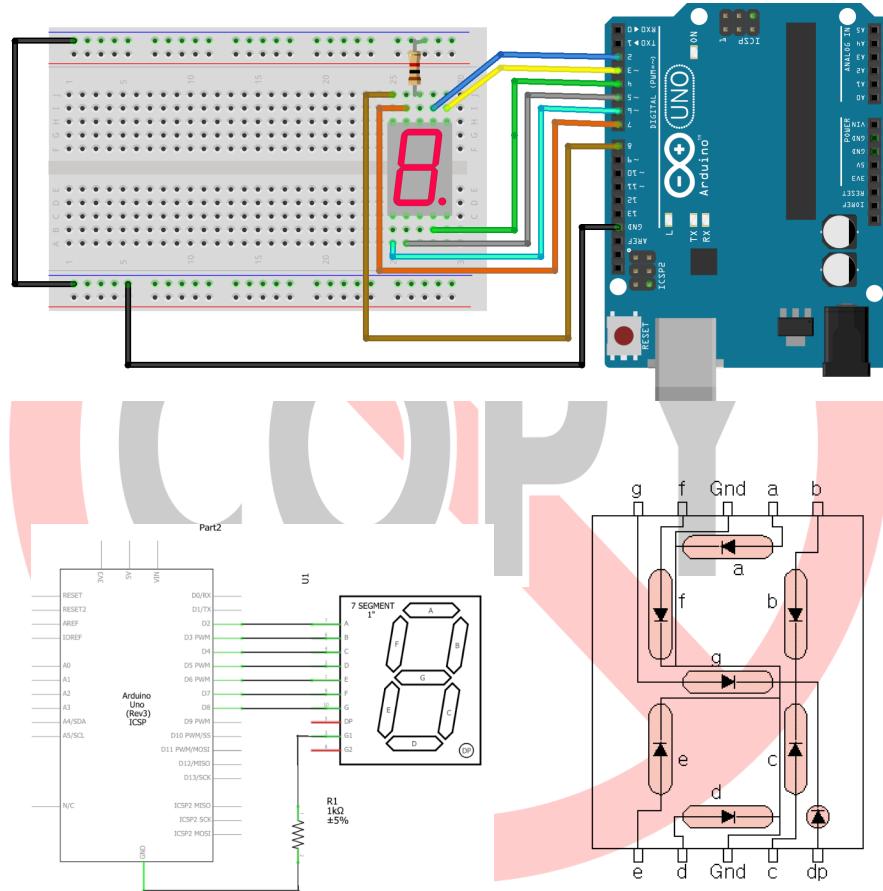
៣.៣ 7 Segment

ធាតុក្រើសអេឡិចត្រូនិកមួយប្រភេទដែលបានបង្កើតឡើងដោយប្រើប្រាស់អំពុល LED ចំនួន 7 ត្រាប់ដាច់ពីគ្នា ដែលបានបង្កើតឡើងដោយប្រភេទដែលនៅក្នុងនោះមានដូចជា ៖



7 segment ជាមួយតែលក្ខណៈ ដែលអនុញ្ញាតឡើង 7 បញ្ហាលិតាថែន ដើម្បីបង្ហាញពាណិជ្ជកម្មដី មួយ បុក្រោះតែមួយលក្ខណៈ ដែលការបង្ហាញទាំងអស់នេះវាការស្របយទៅបែកបានតែតុលាងប៉ុណ្ណោះ តាមរាយការណ៍ នឹងបង្ហាញបញ្ហាលិតាថែន ដែលត្រូវបានបង្ហាញឡើង នៅក្នុងពេលវេលាដែលត្រូវបានបង្ហាញ។

រាយការណ៍



ចុង

```

1. #define segA 2
2. #define segB 3
3. #define segC 4
4. #define segD 5
5. #define segE 6
6. #define segF 7
7. #define segG 8
8.
9. void setup()
10.{}
11. for(int i=2; i<9; i++)
12. {
13.   pinMode(i,OUTPUT);
14. }
15.}
16.
17.void loop()
18.{}
19. digitalWrite(segA,HIGH);
20. digitalWrite(segB,HIGH);
21. digitalWrite(segC,HIGH);
22. digitalWrite(segD,HIGH);
23. digitalWrite(segE,HIGH);
24. digitalWrite(segF,HIGH);
25. digitalWrite(segG,LOW);
26.}
```

ក្នុងនេះនឹងធ្វើឡើ 7 Segment បង្ហាញលេខ 0

អនុវត្តន៍ : ចូរអ្នកសរស់រក្សាទួរ 7 Segment
បង្ហាញលេខ 1 ដល់លេខ 9

ដោយសារតែការសរស់លេខបានកែតាំងជាបីត្រិប្បាស់កញ្ចាំងបាយទាំងស្រុង ហើយវិធីសាងស្តែននេះវាពើឡើកដូចសៀវភៅដែលយើងមិនដឹងពីថ្មី នៅក្នុងក្រុងក្រាម 7 Segment បង្ហាញតីលេខ 0 ទៅដឹងទៀត ៩។ ដូច្នេះគេមានវិធីសាងស្តែនមួយបែបឡើតែដែលការងារត្រួតពិនិត្យការសរស់រក្សាទួរ សរស់លេខខ្លួន ហើយយើងមានមូលដ្ឋាននៃក្រុងការបំបកប្រព័ន្ធគោល 2 គោល 10 គោល 16 (Binary Number, Decimal Number and Hexadecimal)

Example in Arduino Program

```
byte Z = B00000011 //Binary Number of Z = 3
byte Z = 3 //decimal Number of Z = 3
byte Z = 0x03
```

Decimal	Binary	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

អង្គភាសា:

Decimal	Binary	Hexadecimal
0 =	B0 0 0 0	= 0x00
1 =	B0 0 0 0	= 0x01
2 =	B0 0 0 0	= 0x02
3 =	B0 0 0 0	= 0x03
4 =	B0 0 0 0	= 0x04
5 =	B.....	= 0x....
6 =	B.....	= 0x....
7 =	B.....	= 0x....
8 =	B.....	= 0x....
9 =	B.....	= 0x....
10 =	B.....	= 0x....
11 =	B.....	= 0x....
12 =	B.....	= 0x....
13 =	B.....	= 0x....
14 =	B.....	= 0x....
15 =	B.....	= 0x....
80 =	B.....	= 0x....
	354 = B.....	
	123 = B.....	

7 segment :

Number	g	f	e	d	c	b	a	Hex
0	0	1	1	1	1	1	1	0x3F
1	0	0	0	0	0	1	0	0x06
2	0	1	0	1	1	0	1	0x5B
3	0	1	0	0	1	1	1	0x4F
4	0	1	1	0	0	1	1	0x66
5	0	1	1	0	1	1	0	0x6D
6	0	1	1	1	1	1	0	0x7D
7	0	0	0	0	0	1	1	0x07
8	0	1	1	1	1	1	1	0x7F
9	0	1	1	0	0	1	1	0x67

បន្ទាប់ពីយើងចានសិក្សាតីការបំបែកប្រព័ន្ធគោលទាំងបីខាងលើមួយមក យើងត្រូវរៀនប្រើប្រាស់ Array នៅក្នុងការសរសររកឃើមឱឡើត។ Array Type ជាការប្រឈមលក្ខណៈអចេរដារាជិននៅក្នុងកន្លែងទៅមួយ ដើម្បីអចេរទាំងនេះគឺអាចបង្ការការប្រើប្រាស់បានដោយដឹងថាគាត់ បុសនៃសរស់អចេរនៅលើ។

ចូល:

```

1. int digit[10] = {0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x67
   };
2.
3. void setup()
4. {
5.   for (int x=2; x<9 ; x++)
6.   {
7.     pinMode(x,OUTPUT);
8.   }
9. }
10.
11.void loop()
12.{ 
13.   for (int i=2; i<9 ; i++)
14.   {
15.     digitalWrite(i,bitRead(digit[3], i-2));
16.   }
17. }
```

Syntax:

bitRead(x,n)
 x: លេខដែលត្រូវទាយយក n bit ណាមួយ
 n: លំដាប់ bit ដើម្បីរកនា

អនុវត្ត:

ចូលសរសេរកូដឱ្យ 7 Segment បង្ហាញ
 លេខពី 0 ដល់ 9 ដោយប្រើប្រាស់
 bitRead & array ។

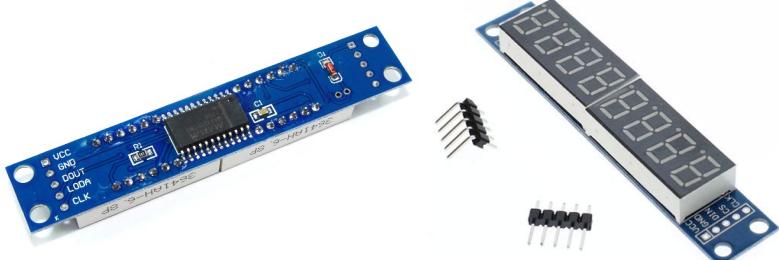


```

int myIns[6];
int myPins[] = {2, 4, 8, 3, 6};
int mySenVals[6] = {2, 4, -8, 3, 2}; //mySenVals[2] contain -8
char message[6] = "hello"
```

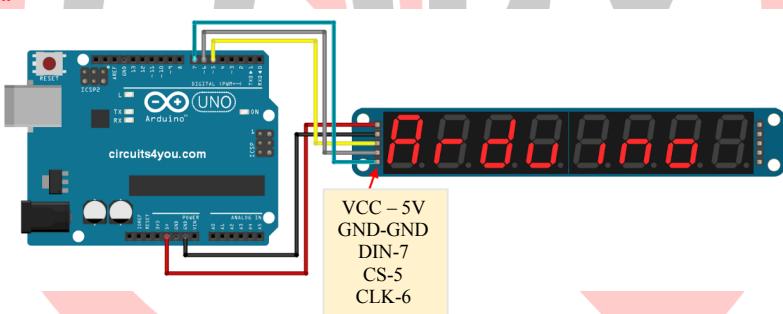
រូប 7 Segment with Max7219

ផ្ទុកដែនរួមឱ្យបានសិក្សាអីឡូដីការប្រើប្រាស់ 7 segment មួយរួមឱ្យក្រោរករត្សាប់ខ្លួន ហើយត្រូវបានគ្រប់ដើម្បីនាំនឹង Arduino។ សារក្រុមហ៊ុនបានបង្កើតយើងត្រូវការប្រើប្រាស់ 7 Segment ចំនួន 5 គ្រប់ដើម្បីរួមឱ្យបានសម្រាប់ផ្ទុកការត្រូវប់ លើសពីនេះទៅទៀតយើងនឹងប្រើប្រាស់បញ្ហាមួយគឺ ទូទៅខាងក្រោម។ ផ្ទុកការប្រើប្រាស់ 7 Segment មួយគ្រប់ត្រូវបានគ្រប់ជាប្រព័ន្ធផ្លូវការប្រើប្រាស់ 7 Segment ដើម្បីធ្វើការប្រាក់ប្រាក់។ ផ្ទុកការប្រើប្រាស់ 7 Segment មួយគ្រប់ត្រូវបានគ្រប់ជាប្រព័ន្ធផ្លូវការប្រើប្រាស់ 7 Segment ដើម្បីធ្វើការប្រាក់ប្រាក់។ ក្នុងការប្រើប្រាស់ 7 Segment មួយគ្រប់ត្រូវបានគ្រប់ជាប្រព័ន្ធផ្លូវការប្រើប្រាស់ 7 Segment ។



ដើម្បីបង្កើត 7Segment With Max7219 ប្រភេទនេះបានយើងត្រូវការបង្កើត Library មួយដែលមានលក្ខណៈ LedControl ដែលរាជ្យ Library មួយដែលគេបានរៀបចំឡើងសម្រាប់សម្រួលដែលការសរសេរក្នុងជាមួយ 7 Segment with Max7219 ប្រភេទនេះ។ ចុចចូល Link នេះដើម្បីទាញយក Library LedControl : <https://github.com/wayoda/LedControl>

ការរៀបចំ:



រូប:

```

1. #include "LedControl.h"
2.
3. LedControl lc=LedControl(7,6,5,1); // DIN , CLK CS
4.
5. void setup()
6. {
7.   lc.shutdown(0,false);
8.   lc.setIntensity(0,8);
9.   lc.clearDisplay(0);
10. }
11.
12.void loop()

```

```

13.{  

14.  for (int x = 0 ; x <=99 ; x++)  

15. {  

16.   lc.setDigit(0,0,x%10,false);  

17.   lc.setDigit(0,1,x/10,false);  

18.   delay(1000);  

19. }  

20.}

```

ក្នុងនេះនឹងធ្វើឲ្យ 7 Segment រាប់ពី 0 ដល់ 99

អនុវត្ត 1 : ចូលសរស់រាប់ពី 0 ដល់ 99 រួចរាល់ចំណាំយករាយពី 99 មក 0 វិញ្ញាបី

អនុវត្ត 2 : ចូលសរស់រាប់ដែលទាន់ដំណើរការរួចរាល់នឹងលូបចងចរណ៍។ អ្នកត្រូវរារើសនឹងអំពុល LED ចំនួន 3 ត្រាប់ដែលមានពណ៌ងឱ្យចាត់ ពណ៌ខ្លោន លើស និងក្រហម។

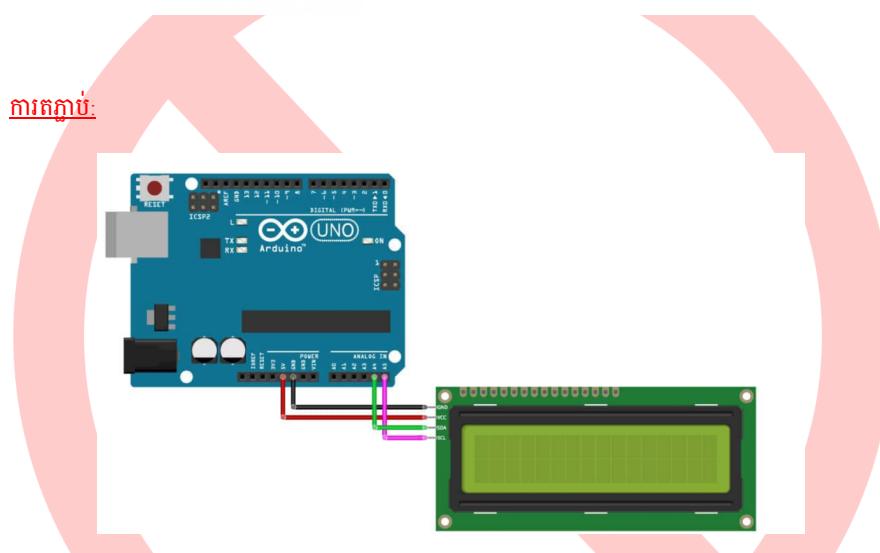
សក្ខាន់ :

- អ្នកត្រូវធ្វើឲ្យ 7 Segment រាប់ចំណាំយករាយពី 60 មក 0 នៅក្នុងពេលវេលាដែនោះអ្នកត្រូវធ្វើឲ្យអំពុល LED ពណ៌ក្រោមនៃខ្លួន។
- នៅពេលដែល 7 Segment រាប់ចំណាំយករាយនឹងលី 0 នៅក្នុងលក្ខខណ្ឌទី 1 ហើយ អ្នកត្រូវធ្វើឲ្យ 7 Segment រាប់មួននៅខ្លួនទី 2 មក 0 វិញ្ញាបី ដែលក្នុងពេលវេលាដែនោះត្រូវធ្វើឲ្យអំពុលពណ៌ក្រោមនៃខ្លួន។
- នៅពេលដែល 7 Segment រាប់ចំណាំយករាយមានលី 0 នៅក្នុងលក្ខខណ្ឌទី 2 ហើយ អ្នកត្រូវធ្វើឲ្យ 7 Segment រាប់ចំណាំយករាយមួននៅខ្លួនទី 3 តាប់ពី 80 មកនីល់ 0 ដែលក្នុងពេលវេលាដែនោះអ្នកត្រូវធ្វើឲ្យអំពុលពណ៌ក្រហម។

បញ្ជាក់ : អំពុល LED ទាំងបីមិនភាពត្រួតពេលវេលាដែលមួយបានទេ

៣.៥ LCD

LCD (Liquid Crystal Display) ជាប្រព័ន្ធផ្លូវការត្រូវការប្រព័ន្ធទៅដែលគេបានយករាយទៅប្រព័ន្ធសម្រាប់បង្ហាញរបស់គេ។ Controller នៃ LCD មានសមត្ថភាពបង្ហាញតម្លៃលេខ និងអក្សរឡាតាំងនៃក្រុងវត្ថុរបស់គេ។ ដែលទាន់ការបង្ហាញនៃក្រុងគ្រប់គ្រង 7 Segment នៅក្នុងការបង្ហាញតាមអក្សរឡាតាំង។ LCD ដែលយើងនឹងលើកមកសិក្សានៅប្រព័ន្ធ LCD 16 x 2 វាអាចបង្ហាញទិន្នន័យចំនួន 2 ឯកជាតិតាតា ដែលនៅក្នុងមួយរាយទាន់ចំនួន Column ចំនួន 16 ។ ដោយសារការបង្ហាញនៃក្រុងគ្រប់គ្រង នូវការបង្ហាញក្នុងការទាយយក Library មួយយកមកប្រើប្រាស់នៅ <https://www.makerguides.com/tutorials/character-i2c-lcd-arduino/>



I2C LCD Connections

I2C LCD	Arduino
GND	GND
VCC	5V
SDA	A4
SCL	A5

ចិត្ត: (សម្រាប់ Check Address ដើម្បី Protocol I2C បានកត្តាប់ អ្នកមិនចាំបាច់ប្រើប្រាស់កុងនេះក៏បាន)

1. #include <Wire.h>

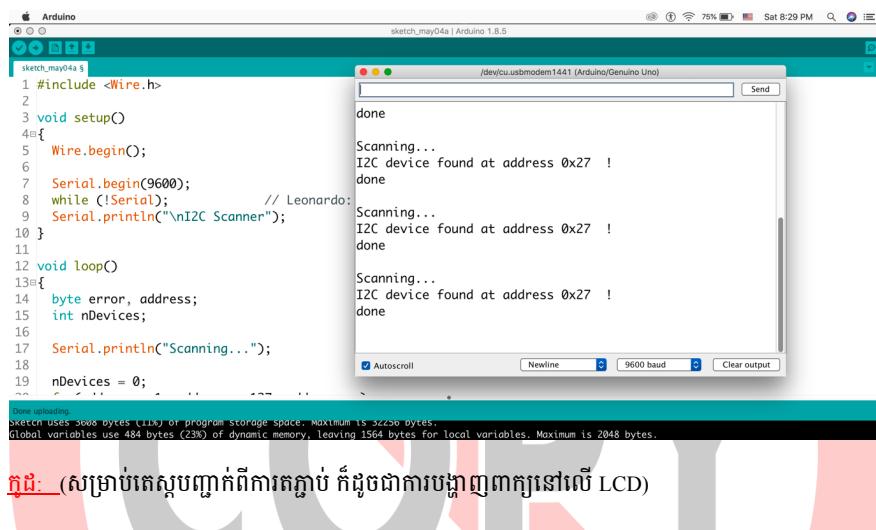
```
2.  
3. void setup()  
4. {  
5.   Wire.begin();  
6.  
7.   Serial.begin(9600);  
8.   while (!Serial);  
9.   Serial.println("\nI2C Scanner");  
10.}  
11.  
12.void loop()  
13.{  
14.   byte error, address;  
15.   int nDevices;  
16.  
17.   Serial.println("Scanning...");  
18.  
19.   nDevices = 0;  
20.   for(address = 1; address < 127; address++)  
21.   {  
22.     // The i2c_scanner uses the return value of  
23.     // the Write.endTransmission to see if  
24.     // a device did acknowledge to the address.  
25.     Wire.beginTransmission(address);  
26.     error = Wire.endTransmission();  
27.  
28.     if (error == 0)  
29.     {  
30.       Serial.print("I2C device found at address 0x");  
31.       if (address<16)  
32.         Serial.print("0");  
33.       Serial.print(address,HEX);  
34.       Serial.println(" !");  
35.  
36.       nDevices++;  
37.     }  
38.     else if (error==4)  
39.     {  
40.       Serial.print("Unknown error at address 0x");  
41.       if (address<16)  
42.         Serial.print("0");  
43.       Serial.println(address,HEX);  
44.     }
```

```

45. }
46. if (nDevices == 0)
47.   Serial.println("No I2C devices found\n");
48. else
49.   Serial.println("done\n");
50.
51. delay(5000);      // wait 5 seconds for next scan
52.

```

ទិន្នន័យ Upload ក្នុងខាងលើមិនមែនបានបង្ហាញលទ្ធផល នៅក្នុង Serial Monitor នៅពេលដែលបង្ហាញ Address របស់ I2C



```

1. #include <Wire.h>
2. #include <LiquidCrystal_I2C.h>
3.
4. LiquidCrystal_I2C lcd(0x27,16,2);
5. void setup()
6. {
7.
8. lcd.init();
9. lcd.backlight();
10.}
11.void loop()
12.{
13.
14.lcd.setCursor(3,0);
15.lcd.print("Automation");

```

```

16.lcd.setCursor(4,1);
17.lcd.print("Cambodia");
18.}

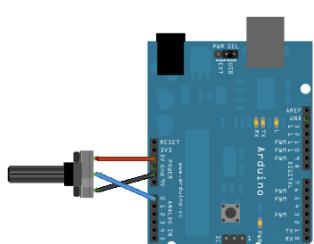
```

មុខងារមួយចំនួនដែលត្រូវកែយល់ដើម្បី :

- [LiquidCrystal\(\)](#)
- [begin\(\)](#)
- [clear\(\)](#)
- [home\(\)](#)
- [setCursor\(\)](#)
- [write\(\)](#)
- [print\(\)](#)
- [cursor\(\)](#)
- [noCursor\(\)](#)
- [blink\(\)](#)
- [noBlink\(\)](#)
- [display\(\)](#)
- [noDisplay\(\)](#)
- [scrollDisplayLeft\(\)](#)
- [scrollDisplayRight\(\)](#)
- [autoscroll\(\)](#)
- [noAutoscroll\(\)](#)
- [leftToRight\(\)](#)
- [rightToLeft\(\)](#)
- [createChar\(\)](#)

សម្រាប់ព័ត៌មានលម្អិតភាពចូលទៅកាន់ Link នេះ : <https://www.arduino.cc/en/Reference/LiquidCrystal>

អនុវត្តន៍ : អ្នកត្រូវតែត្រូវប៉ាប់ Potentiometer ជាមួយ Arduino ។ អ្នកត្រូវសរសោរដូចមួយ ដែលភាពឈ្មោះយើង នៃមុខងារកំណែលេខដែលទទួលបានគាយរយៈ Analog Read ហើយកំណែលេខដែលប្រប្រើបានទាំងអស់នេះ នឹងត្រូវបង្កាញនៅលើអេក្រង់ LCD ។



៣.៦ OLED

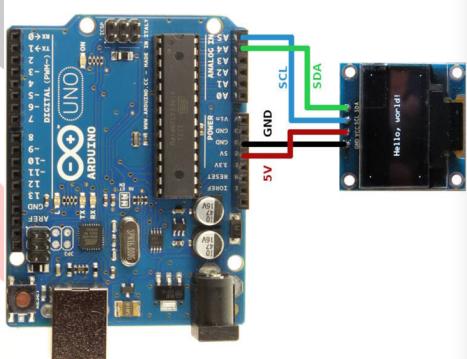
OLED ជាដំឡើង LCD ខ្លួនចម្លាយ ប្រើប្រាស់សម្រាប់ទុកបង្ហាញដូចខាន់ខាន់ Liquid Crystal ដើម្បី បុំណុំ ភាគមានលក្ខណៈពិសេសប្រើប្រាស់ជាមុន Liquid Crystal ។



នឹងរួមប្រើប្រាស់ប្រភេទ OLED នេះបាន អ្នកត្រូវ Download Library ខាងក្រោមពីដែលនៅក្នុងនៅៗ មានដូចជា :

https://github.com/adafruit/Adafruit_SSD1306 និង <https://github.com/adafruit/Adafruit-GFX-Library>

ការរក្សាស្តី:



កូដ:

```

1. #include <Wire.h>
2. #include <Adafruit_SSD1306.h>
3. #include <Adafruit_GFX.h>
4. #define OLED_ADDR 0x3C
5.
6. Adafruit SSD1306 display(-1);
7. int data = 0 ;
8. void setup()
9. {

```

```

10. display.begin(SSD1306_SWITCHCAPVCC, OLED_ADDR);
11. display.clearDisplay();
12. display.display();
13. Serial.begin(9600);
14. }
15.
16.void loop()
17. {
18. display.clearDisplay();
19. display.setTextSize(1);
20. display.setTextColor(WHITE);
21. display.setCursor(30,0);
22. display.print("Automation");
23.
24. display.setTextSize(1);
25. display.setTextColor(WHITE);
26. display.setCursor(35,10);
27. display.print("Cambodia");
28. display.display();
29. }

```

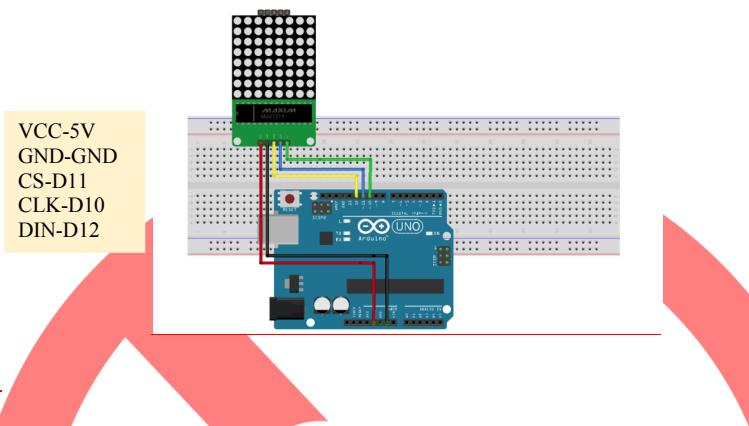
ក្រុងខាងលើនេះបង្ហាញពាក្យ Automatation Cambodia នៅលើ OLED

អនុវត្តន៍ : អ្នកត្រូវតភ្តាប់ OLED មួយចូលទៅកាន់ Arduino និងតភ្តាប់ Variable មួយចូលទៅកាន់ Arduino បន្ទាប់មកទៀតមួកត្រូវយកតែផ្លូវ Analog ដែលបាន Read ចំណូនឯ Variable មកបង្ហាញនៅលើ អេក្រង់ OLED។

៣.៤ LED Matrix

LED Matrix ជាសំរី LED ដែលគេបានតម្លៃបន្ថែមទាំងនេះបង្កើរយ ជាតុរុណយ និងជូរដែក។ Matrix ដែលយើងនឹងយកមកសិក្សានេះជាប្រភេទ Matrix 8x8 មាននំយចា 8 ជូរដែក និង 8 ជូរឈរ។ Matrix គេនឹងយកវាទៅប្រព័ន្ធសាមួយគ្នាដោយប្រើប្រាស់ការបញ្ចូនការពី Arduino ដែលបានបង្ហាញពាក្យ និងបង្ហាញពាក្យ។

ការរក្សាប់:



រូប៖

```

1. #include "LedControl.h"
2.
3. int DIN=12;
4. int CS=11;
5. int CLK=10;
6.
7. byte A[8] = {0x00,0x3C,0X42,0X42,0X7E,0X42,0X42,0X42};
8. byte B[8] = {0x7C,0x42,0x42,0x7C,0x7C,0x42,0x42,0x7C};
9.
10.LedControl lc=LedControl(DIN,CLK,CS,0);
11.
12.void setup()
13.{}
14.
15. lc.shutdown(0,false);
16. lc.setIntensity(0,5);
17. lc.clearDisplay(0);
18.}
19.
20.void loop()
21.{}
22. printByte(A);
23. delay(1000);
24. printByte(B);
25. delay(1000);
26.}
27.
28.void printByte (byte character [])
29.{}

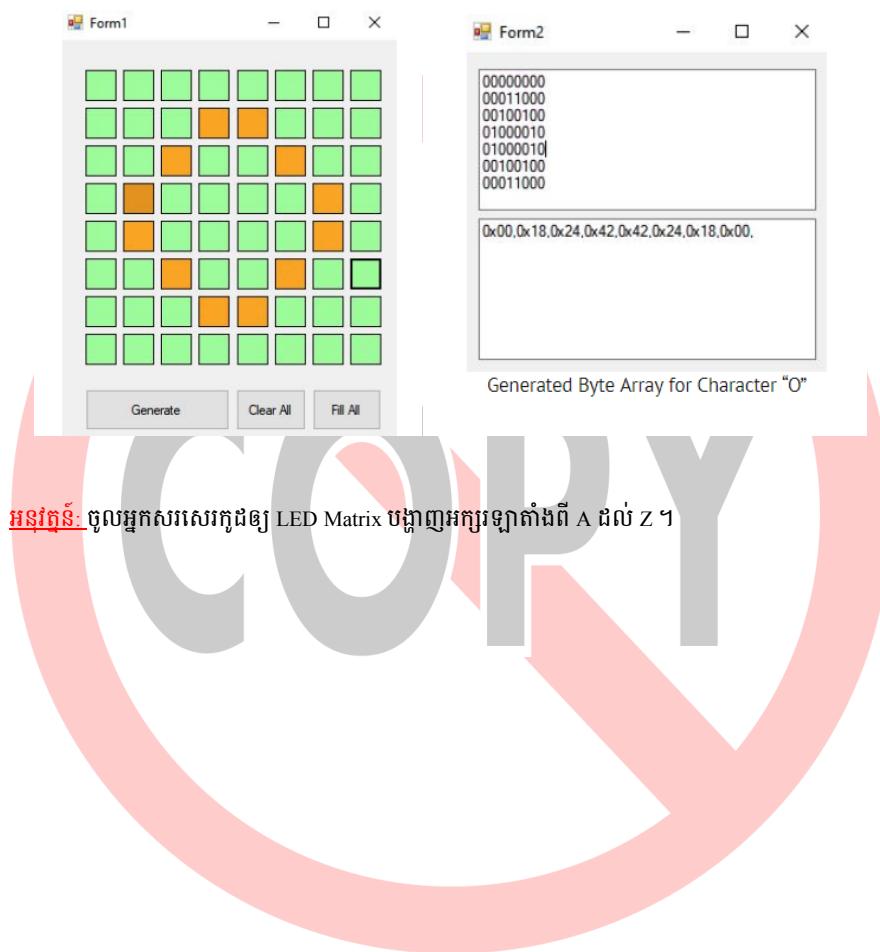
```

```

30. int i=0;
31. for(i=0;i<8;i++)
32. {
33. lc.setRow(0,i,character[i]);
34. }
35.

```

របៀបកំណត់ចំនួនធានា Array ដើម្បីបង្ហាញពាកសមិនយូរ



ឧំពុក & Sensor

៤.៩ IR REMOTE

អ្នកអាចទិញ Remote ដែលមាននៅលក់លើទីផ្សារមកប្រើប្រាស់បាននៅក្នុងតម្លៃខ្លួន។ យើងអាចប្រើប្រាស់ IR Receiver ជាអ្នកទទួលទិន្នន័យ (ក្នុង) ដីឡើងខាង Remote។ នៅក្នុងតម្លៃខ្លួននេះអ្នកនឹងអាចយកបិជ្ជីតីរបស់បច្ចុប្បន្ន ដូចជា Button ដែលមាននៅលើ Remote ។ ហើយអាចយក Remote របស់អ្នកទៅបានគ្រប់គ្រង់ដោយប្រើប្រាស់ក្រុមហ៊ុយទិន្នន័យនានា ដូចជា LED មូនក្នុងចាប់ដើម។

ដើម្បីប្រើប្រាស់ IR Remote អ្នកនឹងចូរការណែនាំការពាយក្រុងលម្អិតក្នុងការប្រើប្រាស់ Link នេះ ដើម្បីទាញយក Library មកប្រើប្រាស់: <https://www.arduinolibraries.info/libraries/i-remote>



ការកែត្រា:

- Signal Pin Connect to D11 Arduino Pin
- GND Pin Connect to GND Arduino Pin
- VCC Pin Connect to 5V Arduino Pin

រូប:

នៅពេលដែលអ្នក Upload Program ឱងក្រាយចូលទៅក្នុង Arduino Board រួចរាល់ហើយ អ្នកត្រូវបើក Serial Monitor ឲ្យចូលរួចរាល់លទ្ធផលនៃការទទួលទិន្នន័យនៅលើ IR Remote នៅក្នុងនឹងយើង។ នៅពេលនេះអ្នកនឹងដើរតម្លៃខ្លួននៃការទទួលទិន្នន័យនៅលើ IR Remote។

```

1. #include <IRremote.h>
2. int RECV_PIN = 11;
3. IRrecv irrecv(RECV_PIN);
4. decode_results results;
5. void setup()
6. {
7.   Serial.begin(9600);
8.   Serial.println("Enabling IRin");
9.   irrecv.enableIRIn();

```

```

COM5 (Arduino/Genuino Mega or Mega 2560)
16728765
16716015
2534850111
16718055
16754775
16769055
16754775
16769055

```

```

10. Serial.println("Enabled IRin");
11. pinMode(8,OUTPUT);
12. }
13.void loop()
14.{
15. if(irrecv.decode(&results)) {
16. Serial.println(results.value);
17. irrecv.resume();
18. }
19. delay(100);
20. }
```

ក្នុងនេះយើងបានទទួលខ្លួនពីការបញ្ចូនការណ៍ដែលត្រូវការការពារស្ថិតិយោគ

- បញ្ចូនការណ៍ដែលបានទទួលខ្លួនពីការបញ្ចូនការណ៍ដែលត្រូវការការពារស្ថិតិយោគ

```

1. #include <IRremote.h>
2. int RECV_PIN = 11;
3. IRrecv irrecv(RECV_PIN);
4. decode_results results;
5.
6. void setup()
7. {
8. Serial.begin(9600);
9. irrecv.enableIRIn();
10. }
11.void loop() {
12. if(irrecv.decode(&results)) {
13. Serial.println(results.value);
14. irrecv.resume();
15. if(results.value == 16724175)
16. {
17. Serial.println("1");
18. }
19. else if(results.value == 16718055)
20. {
21. Serial.println("2");
22. }
23. }
24. delay(100);
25. }
```

```

COM5 (Arduino/Genuino Mega or Mega 2560)
|
16724175
1
16718055
2
```

- គូរបញ្ជាក់ថានេះនឹងរួមអ្នកកំភាពយក Remote ដែលបានប្រើប្រាស់ជាមួយ IR នេះទៅបញ្ចប់គ្នានៅក្នុងអគ្គិសនីនៅក្នុងផ្ទះ (AC 220V) ដើម្បីដោរ ដូចជា អំពុល ក្នុង ជាន់ ជាថីជាមួយនូវក្រុងក្រាល់តែបែន្ទំម។ ក្នុងនេះមានក្រុងការប្រើប្រាស់ប្រព័ន្ធដែលត្រូវបានក្រុងការប្រើប្រាស់ដោយប្រាក់ប្រាក់។

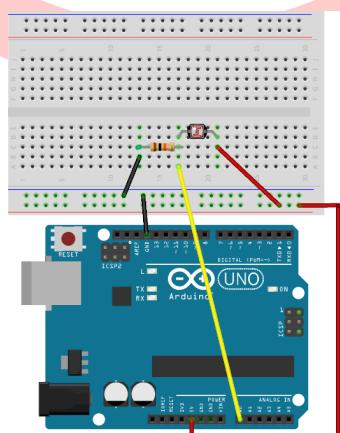
- អនុវត្ត :** អ្នកត្រូវប្រើប្រាស់នឹង Remote ធ្វើសំខាន់ខ្ពស់លើលម្អិត ឬទៅប្រើប្រាស់លើ LED ។
- បីក្នុងលេខ 1 : ចុចលើកទីមួយដើម្បីអំពុលទី ១ នឹង ចុចលើកទីពីរ ដើម្បីអំពុលទី ១ រលកវិញ្ញា
 - បីក្នុងលេខ 2 : ចុចលើកទីមួយដើម្បីអំពុលទី ២ នឹង ចុចលើកទីពីរ ដើម្បីអំពុលទី ២ រលកវិញ្ញា
 - បីក្នុង Power : ចុចលើកទីមួយដើម្បីអំពុលទាំងពីរ ឬ ចុចលើកទីពីរដើម្បីអំពុលទាំងពីរលកវិញ្ញា

៤.២ LDR

Light Dependent Resistor ជាប្រព័ន្ធ Sensor ម៉ឺងដែលបានការបែងច្រែបង្ហាញកាមរយៈកម្រិតពន្លឹង។ គោរចម្លកវាទោសកម្មិតពន្លឹងនៅក្នុងបន្ទប់ បុក្រោះជាគ្រោះដែលបានបង្ហាញស្ម័គ្រីតិច តែបែងច្រៀងបែងច្រៀងបន្ទប់អំពុលភ្លើង។



ការរោងចក្រវិទ្យា



ចូល:

នៅពេលដែលអ្នក Upload កូដខាងក្រោមនេះចូលទៅតួអង់ Arduino Board វិញ ហើយ អ្នកត្រូវបើកមឺន តម្លៃប្រចាំថ្ងៃ (Analog) ដែលទទួលបានពីផ្ទៃខាងក្រោម LDR។

```

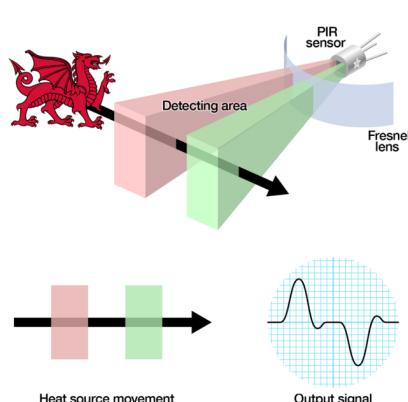
1. int val;
2.
3. void setup()
4. {
5.   Serial.begin(9600);
6.   pinMode(A0,INPUT);
7. }
8.
9. void loop()
10.{           226
11.   val = analogRead(A0); 209
12.   Serial.println(val); 226
13. }
```

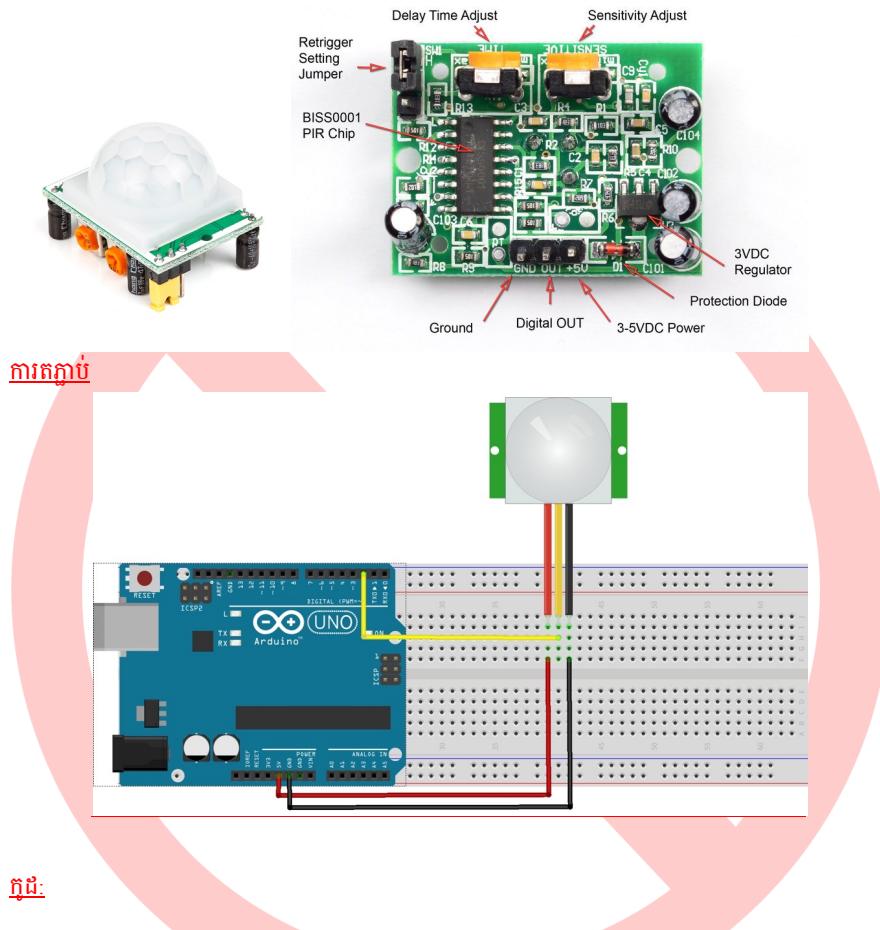
COM5 (Arduino/Genuino Mega or Mega 2560)	
226	
209	
226	
208	
227	
210	
228	
212	
226	
213	
226	

អនុវត្ត: ចូលសរសរុណិជ្ជាយក្សារដែលការដាក់យក្សាយប្រភពិត្ថម្ម។ នៅពេលដែលអ្នកយកដោទាញចុះដូច និង LDR កំណើមានទន្ល់ចូលទៅកាន់ LDR (ប្រើបង្ហើបង្ហើបានបាយបំផុត) ត្រូវធ្វើឲ្យមំពូល LED ត្រូវបានបង្ហើបាន ដែលអ្នកយកត្រឹម Flash Smart Phone ទៅបង្ហើបាននៅលើ LDR ប្រើបាយការ LDR នៅក្នុងមួយដែលមានពន្លឹក ត្រូវបែកត្រាន់ (ប្រើបង្ហើបានបាយបំផុត ត្រូវបានបង្ហើបាន) អ្នកត្រូវចូលរួមរាល់ LED ដែលក្នុងនោរបាលតែទៅវិញ។

៤.៣ Motion Sensor or PIR Sensor

Motion Sensor ឬ Sensor មួយប្រភេទនេះបានសមត្ថភាពចាប់សកម្មភាពរបស់មនុស្ស សម្រាប់បាន។





```

1. int inputPin = 2;
2. int val = 0;
3.
4. void setup()
5. {
6.   pinMode(inputPin, INPUT);
7.   Serial.begin(9600);
8. }
9.
10.void loop()
11.{ 
12.   val = digitalRead(inputPin);

```

```

13.
14. if (val == HIGH)
15. {
16.   Serial.println("Motion detected!");
17. }
18.
19. else
20. {
21.   Serial.println("Motion End");
22. }
23.
24.

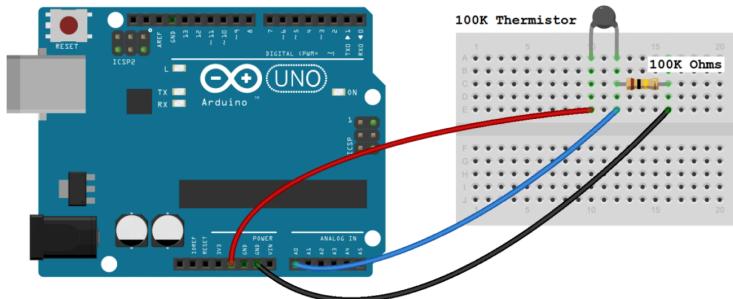
```

អនកចូល: នៅពេលដែលមានសកម្មភាពនៅមុខ PIR ឬ Motion Sensor ដឹកប្រើបាន Buzzer ណួនសរឡង និង
អំពុល LED តិច (Alarm System) ។

៤.៤ Thermistor

Thermistor ជាអស្សីស្តុងមួយដែលមានតម្លៃប្រយោលខ្ពស់នៅពាណិជ្ជកម្មរបស់វា វាប្រើបានទៅ
ប្រើប្រាស់សម្រាប់វាស់សិក្សាសាធារណៈនៅក្នុងកបិតខ្ពស់ដែលចត្រូន៍ពី -90°C ដល់ 300°C ។ Thermistor មាន
ភាពខសត្វាតិ Resistance Temperature Detectors (RTDs) ដោយសរាយតម្លៃប្រើប្រាស់ខាងក្រោមនៅជាព្យាយក
ផ្សំពីសកម្មភាព ឬ Polymer ទណ្ឌ:ដែលបានតម្លៃ RTDs ធ្វើពីលេហ៍:សុទ្ធភ័យការងារសំគាល់រហូតដល់ការ
អង្គរ។ ផ្សេងៗនៅក្នុងចំណុចនេះ:យើងនឹងធ្វើការសិក្សាតីការរាស់សិក្សាសាធារណៈដោយប្រើប្រាស់ Thermistor ។

ការកែត្រា:



មុននេះយើងដឹងថារបៀបរួចទូទាត់សំណង់ទីតាំងរបស់ក្នុងការបង្កើតមិនមែនលាច់ខ្លួនទេ
ពាន់ទី Thermistor (Analog) ទ្វាត់ជាក់ម៉ោងក្រោរសិរី

$$\frac{1}{T} = \frac{1}{T_0} + \frac{1}{\beta} \ln \left(\frac{R}{R_0} \right)$$

- T សីគូណុភាពដែលត្រូវរាស់គិតជា K
- To សីគូណុភាពនៅបន្ទាប់ពីសោចន់ 25°C = 298.15K
- β ចំនួននៃ NTC 3950 មានន័យថា β = 3950
- R និងស្ថាដែនីគូណុភាពកំពុងរាស់
- R₀ និងស្ថាដែនីគូណុភាព 25°C = 100Kohm

រូប:

```

1. const float T0 = 298.15;
2. const float R0 = 10000;
3. const float belta = 3950;
4. const float r = 100000;
5.
6. float val;
7. float R,vol,T;
8.
9. void setup()
10.{}
11. Serial.begin(9600);
12.{}
13.
14.void loop()
```

```

15.{  

16. val = analogRead(A0);  

17. vol = (val * 5) / 1023;  

18. R = (vol * r) / (5 - vol);  

19. T = (T0 * belta) / (belta + (T0 * log(R/R0)));  

20. Serial.println(T);  

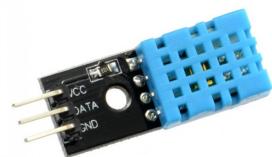
21.}

```

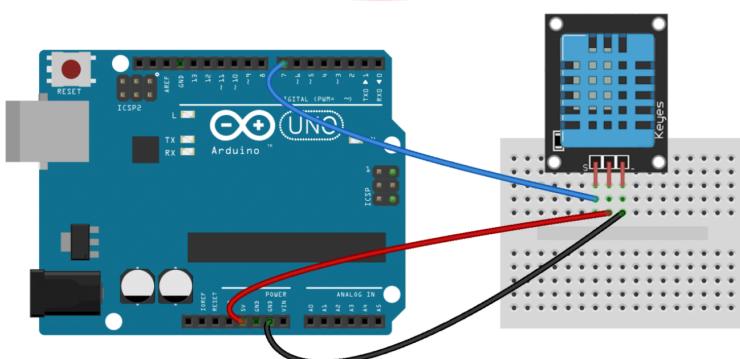
អនុវត្ត: ចូលទាយយកតម្លៃដែលអ្នកទទួលបានពី Thermistor (Temperature) មកបង្ហាញនៅលើ 7Segment 8Digit) °

៤.៥ Humidity Sensor

Humidity Sensor ជា Sensor មួយប្រភេទដែលមានសមត្ថភាពរាយក្រឹងសំណង់សំណង់ក្នុងការបាន
ដែលត្រួតពិនិត្យមួយករាយដែលបានក្រោមការរក្សាទុកដាក់ការសាងគ្មាន ការរក្សាទុកដាក់ក្នុងការបាន
នៅក្នុងផ្ទះ កំដួងជាប់ការ ស្ថិតិថ្មីជាដី។



ការតាម្ភៈ:



នឹងប្រើប្រាស់ប្រភេទ Sensor នៃបានយើងត្រូវចូលទៅទាញយក Library ពីរខាងក្រោមដែលនៅក្នុងនោះ
មានដូចជា :

<https://github.com/adafruit/DHT-sensor-library> និង https://github.com/adafruit/Adafruit_Sensor

- កូដ:

```

1. #include "DHT.h"
2. #define DHTPIN 7
3. #define DHTTYPE DHT22
4. DHT dht(DHTPIN, DHTTYPE);
5.
6. void setup() {
7.   Serial.begin(9600);
8.   Serial.println(F("DHTxx test!"));
9.
10. dht.begin();
11. }
12.
13.void loop()
14.{
15.
16. delay(2000);
17.
18. float h = dht.readHumidity();
19.
20. float t = dht.readTemperature();
21.
22. float f = dht.readTemperature(true);
23.
24.
25. if (isnan(h) || isnan(t) || isnan(f)) {
26.   Serial.println(F("Failed to read from DHT sensor!"));
27.   return;
28. }
29.
30.
31. float hif = dht.computeHeatIndex(f, h);
32.
33. float hic = dht.computeHeatIndex(t, h, false);
34.
35. Serial.print(F("Humidity: "));
36. Serial.print(h);

```

```

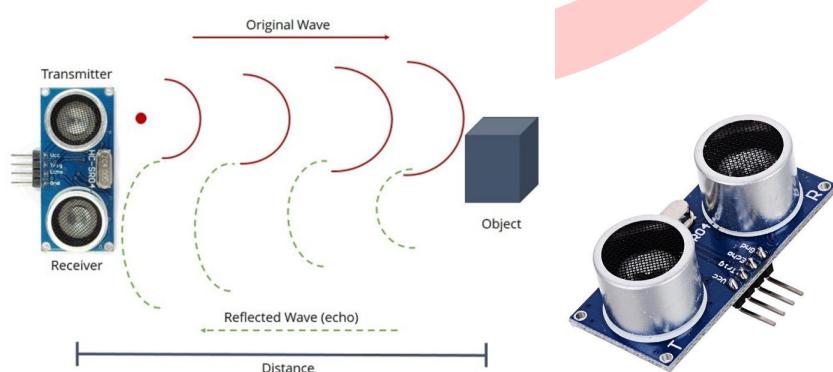
37. Serial.print(F("% Temperature: "));
38. Serial.print(t);
39. Serial.print(F("°C "));
40. Serial.print(f);
41. Serial.print(F("°F Heat index: "));
42. Serial.print(hic);
43. Serial.print(F("°C "));
44. Serial.print(hif);
45. Serial.println(F("°F"));
46.

```

៤.៦ Ultrasonic Sensor

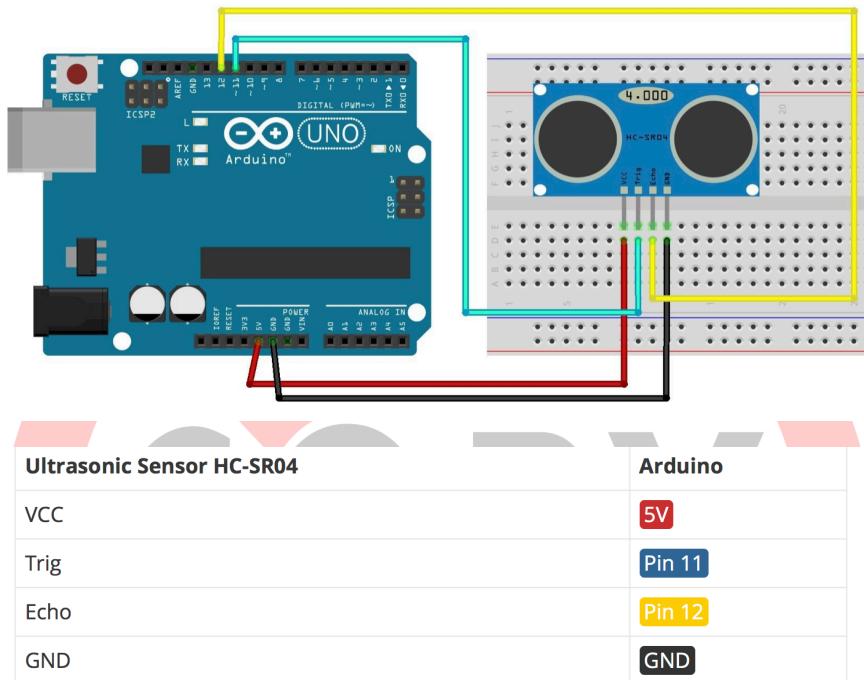
Ultrasonic Sensor ឬ Sensor មួយប្រភេទដែលមានសមត្ថភាពអាចវារកស់ច្បាយបាន ដោយប្រើប្រាស់ Sonar ដើម្បីកំណត់ថាមាយរូបឯកច្បាប់ទៅនឹងស្តីប្រច័ំរឿង។ បីសិនជាយើងយកវាទៅប្រើប្រាស់សម្រាប់ការសំចាយបាយក្នុងដែលមិនទាន់ធ្វើបានស្ថិត ឧបាទរកឯកច្បាប់ដែលបានរួចរាល់ ឬវិនិច្ឆ័យដែលយើងទទួលបានមិនភាស់លាស់ ពេលខ្លះវាចោរឲច្បឹងនិងយើងទទួលបាន Error ដើម្បីមាន។ ខាងក្រោមជាគ័ត៌មានលម្អិតរបស់ Ultrasonic Sensor

- Power Supply : +-5VDC
- Quiescent Current : <2mA
- Working Current : 15mA
- Effectual Angle : <15°
- Ranging Distance : 2cm - 400cm / 1" - 13ft
- Resolution : 0.3cm
- Measuring Angle : 30 Degree
- Trigger Input Pulse width :10us
- Dimension : 45mm x20mm x15mm



ដើម្បីប្រើប្រាស់ Sensor ប្រភេទនេះបានអ្នកត្រូវទាញយក Libray មួយតាមកំណែ Link នេះចាន់ :
<https://www.arduinolibraries.info/libraries/ultrasonic>

ការគ្រប់



រឿង: (ក្នុងជោគជ័យ Library)

```

1. int trigPin = 11;
2. int echoPin = 12;
3. long duration, cm, inches;
4.
5. void setup()
6. {
7.   Serial.begin (9600);
8.   pinMode(trigPin, OUTPUT);
9.   pinMode(echoPin, INPUT);
10. }
11.
12.void loop() {
13.
```

```

14. digitalWrite(trigPin, LOW);
15. delayMicroseconds(5);
16. digitalWrite(trigPin, HIGH);
17. delayMicroseconds(10);
18. digitalWrite(trigPin, LOW);
19.
20. pinMode(echoPin, INPUT);
21. duration = pulseIn(echoPin, HIGH);
22. cm = (duration*0.034/2);
23.
24. Serial.print(cm);
25. Serial.print("cm");
26. Serial.println();
27.
28. delay(250);
29.

```

រូប៖ (ក្នុងដំឡើងត្រូវការយក Library មករបៀបណា)

```

1. #include <Ultrasonic.h>
2.
3. Ultrasonic ultrasonic(11, 12); // trig and echo
4. int distance;
5.
6. void setup()
7. {
8.   Serial.begin(9600);
9. }
10.
11.void loop() {
12.   distance = ultrasonic.read();
13.   Serial.print("Distance in CM: ");
14.   Serial.println(distance);
15.   delay(1000);
16. }

```

អនវគ្គេះ អ្នកត្រូវសរសរុកឯណុយដំឡើងត្រូវបង្ហាញនៅលើ Serial Monitor ដើម្បីរួចរាល់ Ultrasonic Sensor
ដើម្បីធ្វើត្រូវិតមុនការសំប្តិជាអ្នក។

៤.៧ Gas Methane Sensor

Gas Methane Sensor ឬ Sensor មួយប្រភេទដែលមានសមត្ថភាពរាយចាប់យកពីរឿង និងចាំបាយហ្មតុសដែលមាននៅក្នុងខ្សោយ។ Sensor ប្រភេទនេះអាចធន់បាននៅក្នុងស៊ីគណុភាពចន្លោះពី -10 ដល់ 50 °C



ក្នុង:

```

1. #define Sensor A0
2. int data;
3.
4. void setup()
5. {
6.   Serial.begin(9600);
7.   pinMode(Sensor,INPUT);
8. }
9.
10.void loop()
11.{ 
12.   data = analogRead(Sensor);
13.   Serial.println(data);
14. }
```

អនុវត្ត: អ្នកត្រូវតភ្ជាប់ Sensor ពីប្រភេទផ្សេងៗគ្នា តើ PIR Sensor (Digital Sensor) និង Gas Methane Sensor (Analog Sensor)។ នៅពេលដែលមានសកម្មភាពនៅខាងមុន PIR Sensor អ្នកត្រូវធ្វើឲ្យអំពុលមួយឆ្លើយ (Lamp 220V AC)។ ចំណែកជាគម្រោង Gas Methane Sensor វិញ្ញាននៅពេលដែលមានផ្សេងៗ និងចំហាយបញ្ហាយអ្នកត្រូវរឹងឲ្យ Buzzer បង្កើតរាយដែលបានបានការបារិត (Alarm) ។

៤. ៤ Alcohol Sensor (not Complete)

Alcohol Sensor ជាអ្នកត្រូវប្រភេទដែលមានសមត្ថភាពរាយសំកម្រិតជាតិភាគុលុយខាងក្រោមជាតិមានលម្អិតអំពី Sensor c

Features

- 5V operation
- Simple to use
- LEDs for output and power
- Output sensitivity adjustable
- Analog output 0V to 5V
- Digital output 0V or 5V
- Low Cost
- Fast Response
- Stable and Long Life
- Good Sensitivity to Alcohol Gas
- Both Digital and Analog Outputs
- On-board LED Indicator



Technical Data

- Concentration : 0.05 mg/L – 10 mg/L Alcohol
- Operating Voltage : 5V ±0.1
- Current Consumption : 150mA
- Operation Temperature : -10°C – 70°C

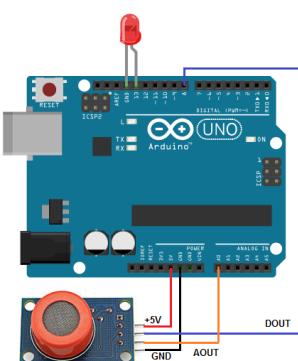
Pin Out

- VCC – Input Power Supply
- GND – Supply Ground
- DO – Digital Output
- AO – Analog Output

Applications

- Vehicle Alcohol Detector
- Portable Alcohol Detector

ការគ្រប់



ក្នុង:

```
1. float sensor_volt;
```

```

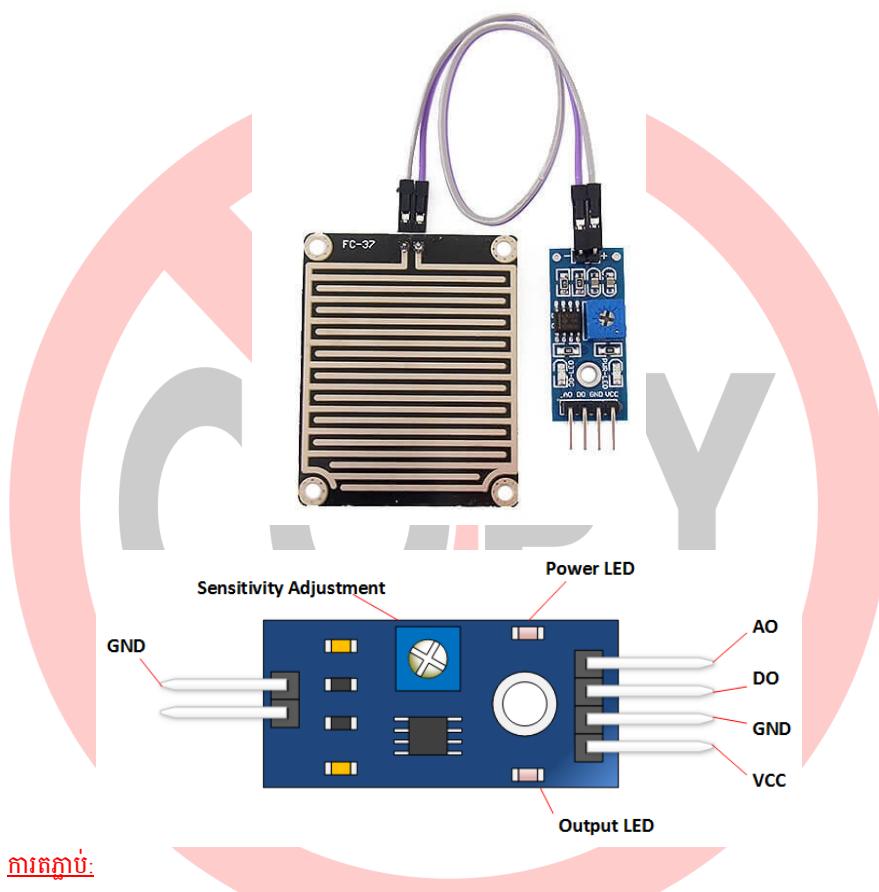
2. float RS_gas;
3. float ratio;
4. float BAC;
5.
6. void setup()
7. {
8.   Serial.begin(9600);
9. }
10.
11.void loop()
12.{ 
13. int sensorValue = analogRead(A0); c
14. sensor_volt=(float)sensorValue/1024*5.0;
15. RS_gas = (5.0-sensor_volt)/sensor_volt;
16. ratio = RS_gas/R0;
17. BAC = 0.1896*ratio^2 - 8.6178*ratio/10 + 1.0792
18. Serial.print("BAC = ");
19. Serial.println(BAC*0.0001);
20. Serial.print("\n\n");
21. delay(1000);
22.}

```

អនវត្ថុ ចូលទាញយកទិន្នន័យដែលទទួលបានពីផ្ទើកខាង Sensor មកបង្ហាញនៅលើ 7Segment 8Digit ។ នៅពេលដែលធ្វើការរាយសំគិតជាតិភាពល កេលដែលត្រូវដែលអ្នកទទួលបានដូចត្រូវបានរាយដែលអ្នកបានកំណត់ អ្នកត្រូវធ្វើឲ្យ Buzzer ឯនិតសម្រេចឡើង។

៤.៤ Rain Sensor

Rain Sensor ជា Sensor មួយប្រភេទដែលមានសមត្ថភាពរាយដូចត្រូវបានរាយនៅពេលដែលមានតំណាក់ទីក។ តែម្នាក់ដែលយើងនឹងទទួលបានពីផ្ទើកខាង Sensor នៅវាដាតពីផ្ទើនៃការបំមែបម្រូលកំម្មៈ រហូតដល់ស្ថិស្អោ ដែលត្រូវបានរាយស្ថិស្អោនឹងមានតម្លៃប្រមាណ 100K ohm នៅពេលដែល Sensor មានសំណើមហើយនៅពេល Sensor ត្រូវបានរាយស្ថិស្អោនឹងមានតម្លៃប្រមាណ 1M ohm។

ការរំភ្លោះ:

- Connect A0 - Pin A0 Arduino
- Connect GND - Pin GND Arduino
- Connect VCC - 5V Arduino (**5V for Analog and 3.3V for Digital**)

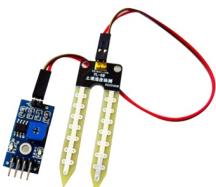
Pin, Control, or Indicator	Description
VCC	+5 Volts Power Source
GND	Ground or negative power source
DO	Digital Output. Goes low when moisture exceeds set threshold.
A0	Analog Output - Zero to five volts. The lower the voltage, the greater the moisture.
Power LED	Indicates that power is applied
Output LED	Illuminates when moisture has exceeded threshold set by sensitivity adjustment.
Sensitivity Adjustment	Clockwise is more sensitive. Counterclockwise is less sensitive.

ចូល:

```
1. const int sensorMin = 0;
2. const int sensorMax = 1023;
3.
4. void setup()
5. {
6.   Serial.begin(9600);
7. }
8.
9. void loop()
10.{ 
11.int sensorReading = analogRead(A0);
12.int range = map(sensorReading, sensorMin, sensorMax, 0, 3);
13.switch(range)
14.{ 
15. case 0:
16.   Serial.println("Flood");
17. break;
18. case 1:
19.   Serial.println("Rain Warning");
20. break;
21. case 2:
22.   Serial.println("Not Raining");
23. break;
24. }
25. delay(1);
26.}
27.
28.
29.
```

៤.៩០ Soil Moisture Sensor

Soil Moisture Sensor ជា Sensor មួយប្រភេទដែលមានសមត្ថភាពរាល់កម្រិតសំណែនក្នុងដី។ វាបានគេប្រើបាយក្នុងការធ្វើរាយការណ៍ដំណឹងការកំណត់តម្លៃក្នុងដី។ នៅពេលដែលយើងដឹងថាគម្រិតសំណែនក្នុងដីមិនត្រឹមត្រូវឡើងទេ យើងអាចធ្វើការកំណត់តម្លៃក្នុងដីដោយប្រើបាយការកំណត់តម្លៃក្នុងដី។



របៀប:

- Connect A0 - Pin A0 Arduino
- Connect GND - Pin GND Arduino
- Connect VCC - 5V Arduino (**5V for Analog and 3.3V for Digital**)

កូដ:

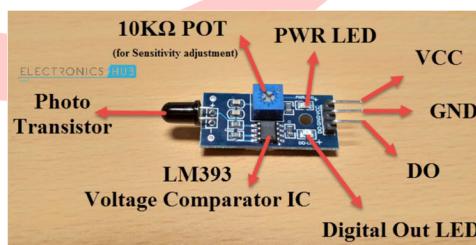
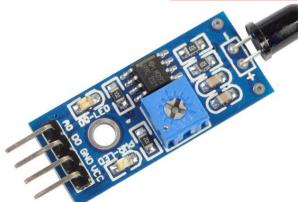
```

1. void setup()
2. {
3.   Serial.begin(9600);
4. }
5. void loop()
6. {
7.   int sensorValue = analogRead(A0);
8.   Serial.println(sensorValue);
9.   delay(1);
10.}

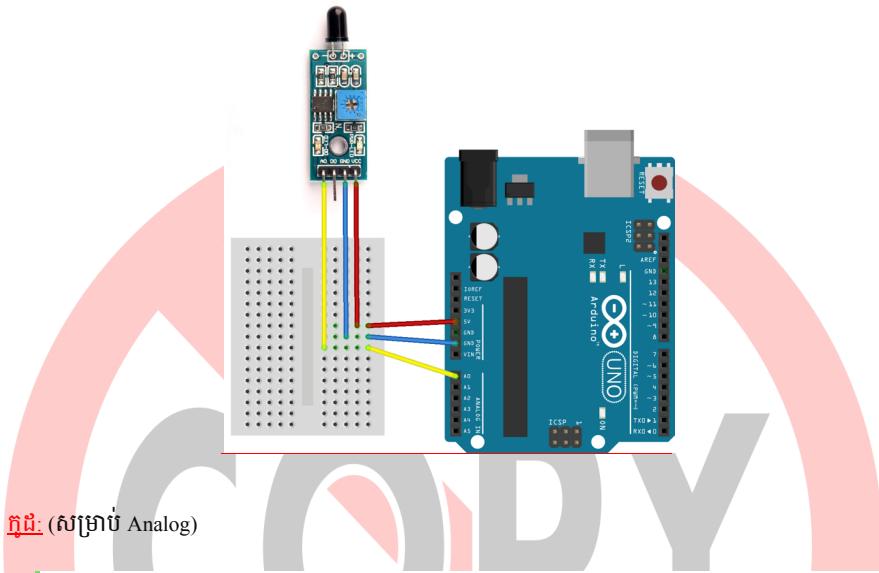
```

៤.៩៩ Flame Sensor

Flame Sensor ជា Sensor មួយប្រភេទដែលមានសមត្ថភាពអាច Detect ពន្លឹងគីឡូចារ៉ាន។



ការអភិវឌ្ឍ:



```

1. #define flame A0
2. int data=0;
3.
4. void setup() {
5.   pinMode(flame,INPUT);
6.   Serial.begin(9600);
7.
8. }
9.
10.void loop() {
11.   data = analogRead(flame);
12. }
```

ចុះ: (សម្រាប់ Digital)

```

1. const int flamePin = 11;
2. int Flame = HIGH;
3.
4. void setup()
5. {
```

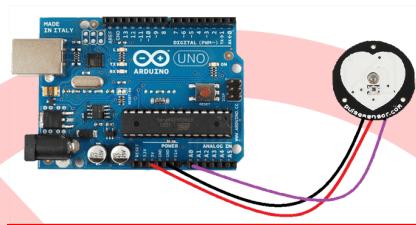
```

6. pinMode(flamePin, INPUT);
7. Serial.begin(9600);
8. }
9.
10.void loop()
11. {
12. Flame = digitalRead(flamePin);
13. if (Flame== LOW)
14. {
15. Serial.println("Fire!!!");
16. digitalWrite(buzzerPin, HIGH);
17. }
18. else
19. {
20. Serial.println("No worries");
21. digitalWrite(buzzerPin, LOW);
22. }
23. }
```

៤.១២ Heart Beat Sensor

Heart Beat Sensor ជា Sensor មួយប្រភេទដែលមានសមត្ថភាពរាយការសង្ឃភាពបែងចាន។ យោងទទួលបានការអនុវត្តន៍យ៉ាងតាមពេលវេលាបែងចានសិក្សាល្អាច្រើន។ ចំណោះអ្នកដែលមានរាយការបានបែងចាន 18 ឆ្នាំទៅ ឡើងកម្រិតសង្ឃភាពបែងចាន 60 ទៅ 100 beats per minute (bpm) ។ ការបែងចាននេះសូមរាយការលើសមត្ថភាពរបស់មនុស្សការងារ។ សម្រាប់ក្នុងវដែលមានរាយការបែងចាន 6 ដល់ 15 ឆ្នាំ កម្រិតសង្ឃភាពបែងចាននេះគឺ 70 ទៅ 100 beats per minute (bpm) ។



ការរំភ្លាស់:

- GND To GND Arduino Pin
- VCC to 5V Arduino Pin
- Data to A0 Arduino Pin

នឹងបង្កើតប្រព័ន្ធសម្រាប់បង្កើតប្រព័ន្ធដែលបានបង្កើតឡើង យើងអាចមានវិធីពីរយ៉ាង យើងអាចប្រព័ន្ធសម្រាប់ Library ក៏ពាន មិនប្រើក៏ពាន ប៉ុន្មោះសម្រាប់ការអនុវត្តន៍ទៅក្នុងចំណុចនេះ យើងបង្កើតប្រព័ន្ធសម្រាប់ Library មួយដើម្បីចិន្ទយដល់ការសរស់រោគ ដូចខាងក្រោម និងក្នុងកិច្ចការបង្កើតប្រព័ន្ធ។

Program ដូចខាងក្រោមគឺជាព្យាយក Library មួយតាមតម្លៃរបស់ការសរស់រោគ

<https://github.com/WorldFamousElectronics/PulseSensorPlayground>

កូដ: (Code ខាងក្រោមនេះអាចបង្ហាញបាននៅលើ Serial Monitor ប៉ុន្មោះអ្នកបានចិត្តយកទិន្នន័យដែលទទួលបានពី Sensor មកបង្ហាញនៅលើ OLED ឬ LCD)

```

1. #define USE_ARDUINO_INTERRUPTS true
2. #include <PulseSensorPlayground.h>
3.
4. const int PulseWire = 0;
5. const int LED13 = 13;
6. int Threshold = 550;
7. PulseSensorPlayground pulseSensor;
8.
9. void setup() {
10.
11.Serial.begin(9600);
12.pulseSensor.analogInput(PulseWire);
13.pulseSensor.blinkOnPulse(LED13);
14.pulseSensor.setThreshold(Threshold);
15.
16.if (pulseSensor.begin())
17.{
18.Serial.println("We created a pulseSensor Object !");
19.}
20.}
```

```

21.
22.void loop() {
23.
24.int myBPM = pulseSensor.getBeatsPerMinute();
25.if (pulseSensor.sawStartOfBeat())
26. {
27.Serial.println("♥ A HeartBeat Happened ! ");
28.Serial.print("BPM: ");
29.Serial.println(myBPM);
30.}
31.delay(20);
32.}
```

៤.៩ ឯក Reed Switch Sensor

Read Switch Sensor ឬ Sensor មួយប្រភេទដែលមានការបែបបង្ហាញនៅពេលដែលមានម៉ោងកន្លែងនិងរក្សា



ការគ្រប់:

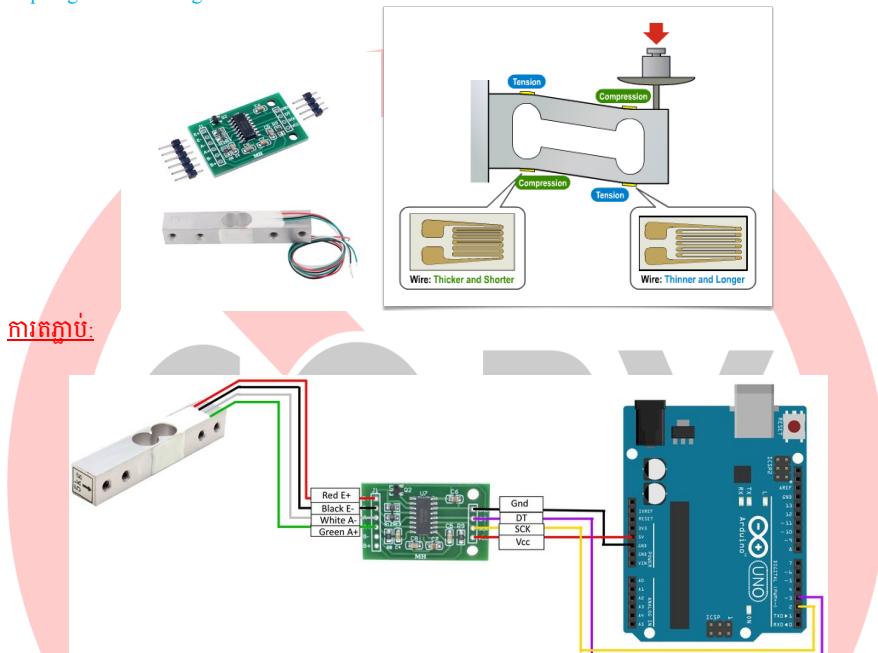
- VCC Connect 5V Arduino
- GND Connect to GND Arduino
- Do Connect to Pin2 Arduino

```

1. void setup()
2. {
3. Serial.begin(9600);
4. }
5.
6. void loop() {
7. byte data = digitalRead(2);
8. Serial.println(data);
9. }
```

៤.១៤ Weight Sensor

Weight Sensor ជាប្រភេទ Sensor មួយដែលមានសមត្ថភាពរាយទឹនទម្រន៍បាន។ ចំណោះ: Sensor ដែលយើងឱងលើកមកកសិក្សានេះជា Sensor ដែលអាចទឹនទម្រន៍បំផុតប្រមាណ 5Kg។ នឹងធ្វើប្រើប្រាស់ប្រភេទ Sensor នេះបានដោយការយករាយក្នុងមួកក្រុវ៉ែតាល្អយក Library មួយមកប្រើប្រាស់ គឺ: <https://github.com/bogde/HX711>



នឹងធ្វើប្រើប្រាស់ប្រភេទ Sensor នេះបានដោយការយករាយក្នុងការបង្កើតប្រព័ន្ធដែលសារការប្រើប្រាស់ប្រភេទ Sensor នឹងទម្រន៍ ដូចខាងក្រោមនេះ:

ជំណាក់កាលទី 1 មួកក្រុវ៉ែតាល្អយករាយក្នុងការបង្កើតប្រព័ន្ធ (Calibrate) និងការគ្រប់បញ្ជីការ (Calibrate (បញ្ជីការ Sensor នឹងយករាយក្នុងការបង្កើតប្រព័ន្ធ))

រូប:

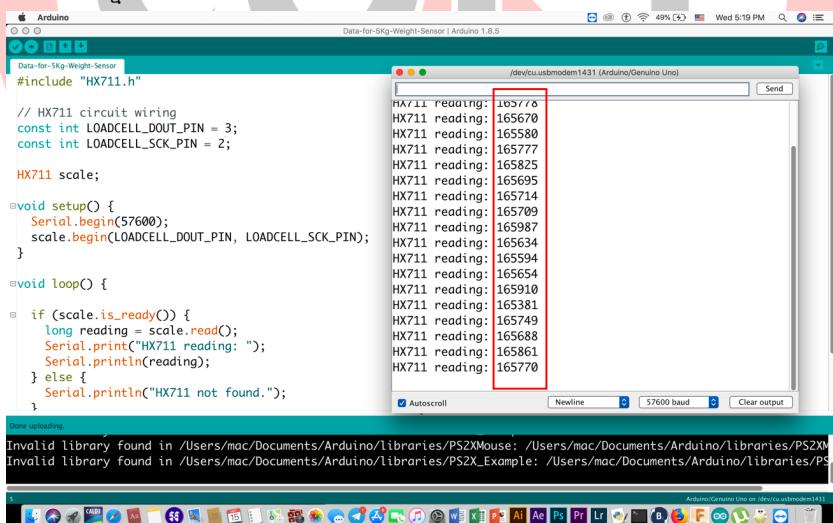
1. #include "HX711.h"
- 2.
3. **const int** LOADCELL_DOUT_PIN = 3;
4. **const int** LOADCELL_SCK_PIN = 2;
- 5.
6. HX711 scale;
- 7.

```

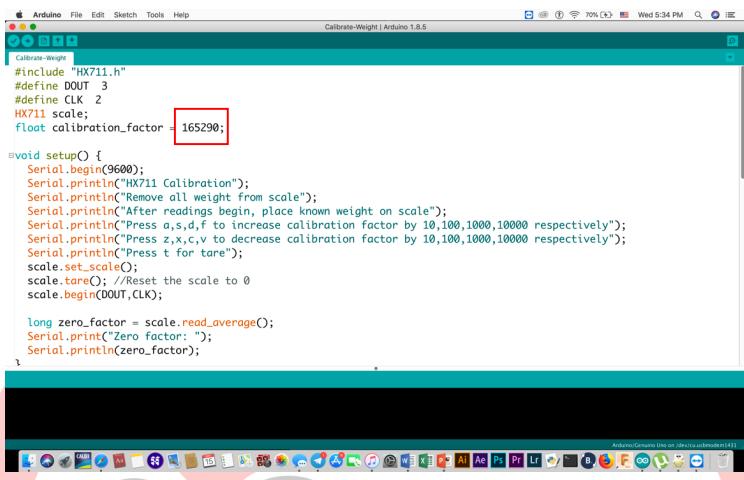
8. void setup() {
9.   Serial.begin(57600);
10.  scale.begin(LOADCELL_DOUT_PIN, LOADCELL_SCK_PIN);
11. }
12.
13.void loop() {
14.
15. if (scale.is_ready()) {
16.   long reading = scale.read();
17.   Serial.print("HX711 reading: ");
18.   Serial.println(reading);
19. } else {
20.   Serial.println("HX711 not found.");
21. }
22.
23. delay(1000);
24.
25. }

```

- បន្ទាប់ពីអ្នកបញ្ចូលក្នុងចរណ៍ អ្នកត្រូវបើក Serial Monitor នៅអ្នកនឹងយើងឱ្យបន្ថែមដែលទូលាលបានពីផ្តើមការងារ Sensor



- នៅពេលដែលអ្នកទូលាលបានក្នុងដែលទូលាលបានពីផ្តើមការងារ Sensor តាមរយៈ Serial Monitor ហើយ អ្នកត្រូវយកតម្លៃនេះមកជាកំណើននៅក្នុង Program ខាងក្រោម



```

#include "HX711.h"
#define DOUT 3
#define CLK 2
HX711 scale;
float calibration_factor = 165290;

void setup() {
  Serial.begin(9600);
  Serial.println("HX711 Calibration");
  Serial.println("Remove all weight from scale");
  Serial.println("After readings begin, place known weight on scale");
  Serial.println("Press a,s,d,f to increase calibration factor by 10,100,1000,10000 respectively");
  Serial.println("Press z,x,c,v to decrease calibration factor by 10,100,1000,10000 respectively");
  Serial.println("Press t for tare");
  scale.set_scale();
  scale.tare(); //Reset the scale to 0
  scale.begin(DOUT,CLK);

  long zero_factor = scale.read_average();
  Serial.print("Zero factor: ");
  Serial.println(zero_factor);
}

```

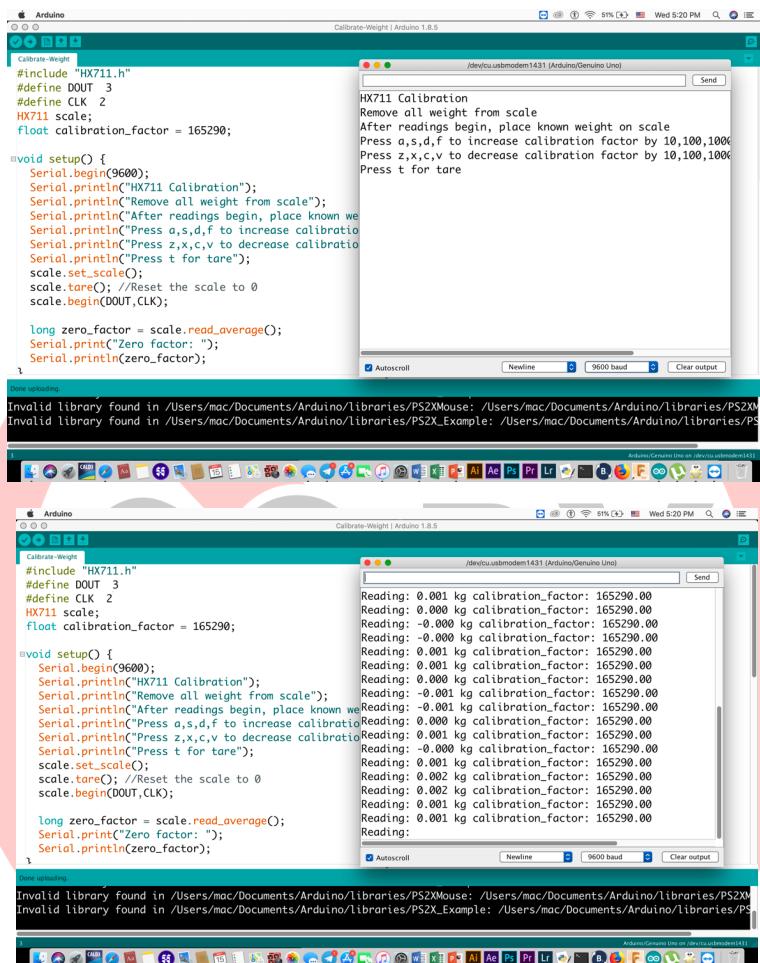
រូប:

1. #include "HX711.h"
 2. #define DOUT 3
 3. #define CLK 2
 4. HX711 scale;
 5. float calibration_factor = 165290;
 6.
 7. void setup() {
 8. Serial.begin(9600);
 9. Serial.println("HX711 Calibration");
 10. Serial.println("Remove all weight from scale");
 11. Serial.println("After readings begin, place known weight on scale");
 12. Serial.println("Press a,s,d,f to increase calibration factor by 10,100,1000,
 ,10000 respectively");
 13. Serial.println("Press z,x,c,v to decrease calibration factor by 10,100,100
 0,10000 respectively");
 14. Serial.println("Press t for tare");
 15. scale.set_scale();
 16. scale.tare(); //Reset the scale to 0
 17. scale.begin(DOUT,CLK);
 18.
 19. long zero_factor = scale.read_average();
 20. Serial.print("Zero factor: ");
 21. Serial.println(zero_factor);
 22. }

```

23.
24.void loop() {
25.
26. scale.set_scale(calibration_factor);
27.
28. Serial.print("Reading: ");
29. Serial.print(scale.get_units(), 3);
30. Serial.print(" kg");
31. Serial.print(" calibration_factor: ");
32. Serial.print(calibration_factor);
33. Serial.println();
34.
35. if(Serial.available())
36. {
37.     char temp = Serial.read();
38.     if(temp == '+' || temp == 'a')
39.         calibration_factor += 10;
40.     else if(temp == '-' || temp == 'z')
41.         calibration_factor -= 10;
42.     else if(temp == 's')
43.         calibration_factor += 100;
44.     else if(temp == 'x')
45.         calibration_factor -= 100;
46.     else if(temp == 'd')
47.         calibration_factor += 1000;
48.     else if(temp == 'c')
49.         calibration_factor -= 1000;
50.     else if(temp == 'f')
51.         calibration_factor += 10000;
52.     else if(temp == 'v')
53.         calibration_factor -= 10000;
54.     else if(temp == 't')
55.         scale.tare();
56. }
57. }
```

- បន្ទាប់ពីបញ្ចប់លទ្ធផល Program ឲ្យចាកល់ហើយ អ្នកត្រូវបើក Serial Monitor ឲ្យចាកយបញ្ហាលអក្សរ t > Enter



The screenshot shows the Arduino IDE interface. On the left, the code for 'Calibrate-Weight' is displayed:

```
#include "HX711.h"
#define DOUT 3
#define CLK 2
HX711 scale;
float calibration_factor = 165290;

void setup() {
  Serial.begin(9600);
  Serial.println("HX711 Calibration");
  Serial.println("Remove all weight from scale");
  Serial.println("After readings begin, place known weight on scale");
  Serial.println("Press a,s,d,f to increase calibration factor by 10,100,1000");
  Serial.println("Press z,x,c,v to decrease calibration factor by 10,100,1000");
  Serial.println("Press t for tare");
  scale.set_scale();
  scale.tare(); //Reset the scale to 0
  scale.begin(DOUT,CLK);

  long zero_factor = scale.read_average();
  Serial.print("Zero factor: ");
  Serial.println(zero_factor);
}

long zero_factor = scale.read_average();
Serial.print("Zero factor: ");
Serial.println(zero_factor);
```

On the right, the Serial Monitor window shows the calibration process:

HX711 Calibration
Remove all weight from scale
After readings begin, place known weight on scale
Press a,s,d,f to increase calibration factor by 10,100,1000
Press z,x,c,v to decrease calibration factor by 10,100,1000
Press t for tare

Reading: 0.001 kg calibration_factor: 165290.00
Reading: 0.000 kg calibration_factor: 165290.00
Reading: -0.000 kg calibration_factor: 165290.00
Reading: -0.000 kg calibration_factor: 165290.00
Reading: 0.001 kg calibration_factor: 165290.00
Reading: 0.001 kg calibration_factor: 165290.00
Reading: 0.000 kg calibration_factor: 165290.00
Reading: -0.001 kg calibration_factor: 165290.00
Reading: -0.001 kg calibration_factor: 165290.00
Reading: 0.000 kg calibration_factor: 165290.00
Reading: 0.001 kg calibration_factor: 165290.00
Reading: 0.001 kg calibration_factor: 165290.00
Reading: 0.002 kg calibration_factor: 165290.00
Reading: 0.002 kg calibration_factor: 165290.00
Reading: 0.001 kg calibration_factor: 165290.00
Reading: 0.001 kg calibration_factor: 165290.00
Reading:

បន្ទាប់មកត្រួតពិនិត្យកម្លាំងស្ថិតលក្ខណៈទៅលើ Weight Sensor ដែលមិនត្រូវតែជាមុន រហូតដែលត្រូវត្រួតពិនិត្យកម្លាំងស្ថិតលក្ខណៈទៅលើ Calibrate ការធ្វើនេះមិនត្រូវបានបញ្ចប់ឡើងទៀត តាមតម្រូវការ Calibrate នឹងបានយកអំពីកម្លាំងស្ថិតលក្ខណៈទៅលើ Serial Monitor មិនត្រូវបានបញ្ចប់ឡើងទៀត (ការលើកចាត់ខ្លួនគឺជាប្រព័ន្ធដែលបានបញ្ចប់ឡើងទៀត) ដូចជាបានយកត្រួតពិនិត្យកម្លាំងស្ថិតលក្ខណៈទៅលើ Serial Monitor មិនត្រូវបានបញ្ចប់ឡើងទៀត 0.5Kg មកជាកំណត់តែប៉ុណ្ណោះ និងត្រូវបានយកត្រួតពិនិត្យកម្លាំងស្ថិតលក្ខណៈទៅលើ Serial Monitor មិនត្រូវបានបញ្ចប់ឡើងទៀត 0.5Kg តែ ទៅនិងយកត្រួតពិនិត្យកម្លាំងស្ថិតលក្ខណៈទៅលើ Serial Monitor មិនត្រូវបានបញ្ចប់ឡើងទៀត 0.79Kg ដូចជាអ្នកត្រូវធ្វើការ Calibrate

```

Arduino
HX711_basic_example | Arduino 1.8.5
Calibrate_Weight
{
    char temp = Serial.read();
    if(temp == '+' || temp == 'a')
        calibration_factor += 10;
    else if(temp == '-' || temp == 'z')
        calibration_factor -= 10;
    else if(temp == 's')
        calibration_factor += 100;
    else if(temp == 'x')
        calibration_factor -= 100;
    else if(temp == 'd')
        calibration_factor += 1000;
    else if(temp == 'c')
        calibration_factor -= 1000;
    else if(temp == 'f')
        calibration_factor += 10000;
    else if(temp == 'v')
        calibration_factor -= 10000;
    else if(temp == 't')
        scale.tare();
}
}

void setup()
{
    Serial.begin(9600);
}

void loop()
{
    if(Serial.available() > 0)
        Serial.write(calibration_factor);
}

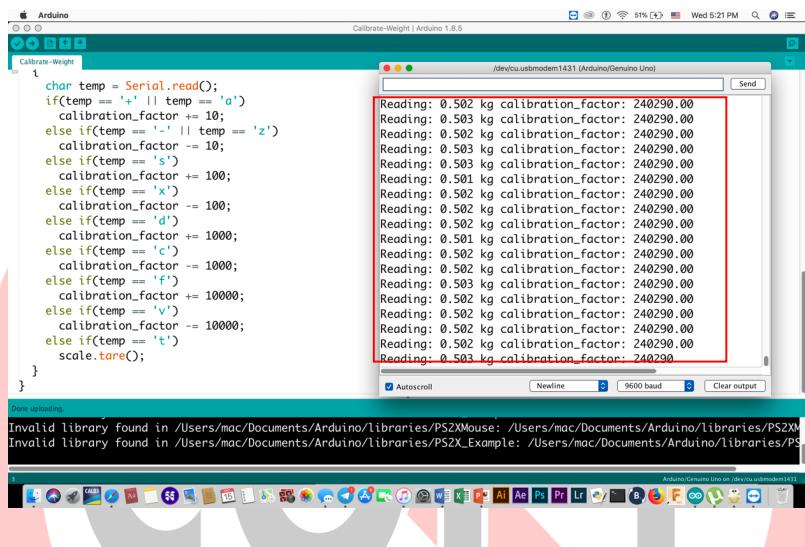
```

Reading: 0.789 kg calibration_factor: 165290.00
 Reading: 0.790 kg calibration_factor: 165290.00
 Reading: 0.790 kg calibration_factor: 165290.00
 Reading: 0.790 kg calibration_factor: 165290.00
 Reading: 0.789 kg calibration_factor: 165290.00
 Reading: 0.790 kg calibration_factor: 165290.00
 Reading: 0.790 kg calibration_factor: 165290.00
 Reading: 0.788 kg calibration_factor: 165290.00
 Reading: 0.789 kg calibration_factor: 165290.00
 Reading:
 Autoscroll Newline 9600 baud Clear output

- ដូច្នេះដើម្បីធ្វើការកែតម្លៃ ប្រឈម Calibrate តិច្ឆេទ Kg នៃលទ្ធផលបានពីក្នុងខាង Sensor បានត្រឹមត្រូវអ្នកត្រួតពិនិត្យការកែតម្លៃ ដែលបានកំណត់ឡើង Program ខាងលើគឺ
- រាយនិមិត្តិសញ្ញា + ប្រអក្សរ a នៅក្នុង Serial Monitor ចូចចូល Enter ដើម្បីបូកតិច្ឆេទ Calibrate ចាស់នឹង 10
 - រាយនិមិត្តិសញ្ញា - ប្រអក្សរ z នៅក្នុង Serial Monitor ចូចចូល Enter ដើម្បីដូកតិច្ឆេទ Calibrate ចាស់នឹង 10
 - រាយអក្សរ s នៅក្នុង Serial Monitor ចូចចូល Enter ដើម្បីបូកតិច្ឆេទ Calibrate ចាស់នឹង 100
 - រាយអក្សរ x នៅក្នុង Serial Monitor ចូចចូល Enter ដើម្បីដូកតិច្ឆេទ Calibrate ចាស់នឹង 100
 - រាយអក្សរ d នៅក្នុង Serial Monitor ចូចចូល Enter ដើម្បីបូកតិច្ឆេទ Calibrate ចាស់នឹង 1000
 - រាយអក្សរ c នៅក្នុង Serial Monitor ចូចចូល Enter ដើម្បីបូកតិច្ឆេទ Calibrate ចាស់នឹង 1000
 - រាយអក្សរ f នៅក្នុង Serial Monitor ចូចចូល Enter ដើម្បីបូកតិច្ឆេទ Calibrate ចាស់នឹង 10000
 - រាយអក្សរ v នៅក្នុង Serial Monitor ចូចចូល Enter ដើម្បីបូកតិច្ឆេទ Calibrate ចាស់នឹង 10000
 - រាយអក្សរ t នៅក្នុង Serial Monitor ចូចចូល Enter ដើម្បីបូកតិច្ឆេទ Kg ដើម្បីលទ្ធផលបានពីក្នុង Sensor Reset មក 0

បញ្ហាក់ :

- ដើម្បីចូរតិច្ឆេទដើម្បីក្នុង អ្នកត្រួតពិនិត្យ Calibrate
- ដើម្បីចូរតិច្ឆេទដើម្បីលើ អ្នកត្រួតពិនិត្យ Calibrate
- បញ្ជាប់ពីការបូក ដូកតិច្ឆេទ Calibrate តាមរយៈ Serial Monitor ប្រចាំការអ្នកនឹងទទួលបានពីក្នុងដែលត្រឹមត្រូវម្មាយ (ភ្លែងដែលជាកំណើន 0.5Kg) ដូច្នេះអ្នកត្រួតពិនិត្យដែលបាន Calibrate ត្រូវការលំមកដាក់បញ្ចាលនៅក្នុង Program ខាងក្រោម



រូប៖ (Display With Serial Monitor and LCD)

```

1. #include "HX711.h"
2. #include <Wire.h>
3. #include <LiquidCrystal_I2C.h>
4. LiquidCrystal_I2C lcd(0x27,16,2);
5. #define DOUT 3
6. #define CLK 2
7. HX711 scale;
8. long data;
9.
10. float calibration_factor = 165290;
11.
12. void setup() {
13.   Serial.begin(9600);
14.   Serial.println("Press T to tare");
15.   scale.set_scale(240290); // Line 15
16.   scale.tare();
17.   scale.begin(DOUT,CLK);
18.   lcd.init();
19.   lcd.backlight();
20. }
21.
22. void loop() {

```

```

23. Serial.print("Weight: ");
24. Serial.print(scale.get_units());
25. Serial.println(" kg");
26. lcd.setCursor(0,0);
27. lcd.print("Weight = ");
28. lcd.print(scale.get_units());
29. lcd.print("Kg");
30. lcd.print("    ");
31. if(Serial.available())
32. {
33.     char temp = Serial.read();
34.     if(temp == 't' || temp == 'T')
35.         scale.tare();
36. }
37.

```

នៅពេលអ្នកបញ្ចូល Program ឲចាកល់ហើយ អ្នកត្រូវបើក Serial Monitor ឲចាប់យករួម T or t > Enter ដើម្បី
គារចិត្តទមន់បាន។ អ្នកត្រូវរាយអក្សរ T or t > Enter នៅក្នុង Serial Monitor សិនដើម្បី Reset ទិន្ន័យទៅ ០
មុននឹងជាក់ត្រូវឱ្យ។ បើសិនជាអ្នកជាក់ត្រូវហើយបាន Reset ទិន្ន័យទៅ ០ នោះវានឹងធ្វើឡើងទម្លៃវត្ថុ
បែសអ្នកមិនត្រឹមត្រូវទេ។

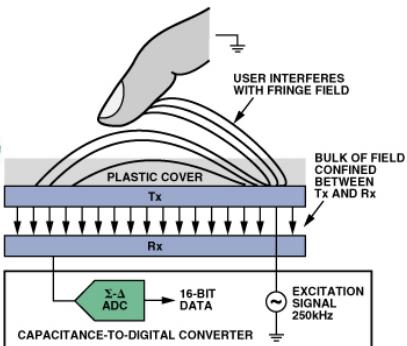
ហត្ថភាគ៖ កល់ទិន្ន័យ Calibrate (ពីផ្ទុមាន បុរាណិផ្ទុមាន) បុរិយាយការអនុវត្តន៍របស់អ្នក នៅខាងលើនេះទៅ
គារបិចចាតាននៅទៅ ៩ ត្រូវបានចំប្រើប្រាស់បានការអត្ថិត្តិក Mechanic (ពាប់ Sensor ..) និងប្រភេទ
Sensor និងដោរា

អ្នកអាចទាមរយៈ Program ទាំងបីការាយ៖ Link នៃក្រាមទេចាន់ :

<https://drive.google.com/drive/folders/17DBiSfJ0ZzYBhWtNnJv3QjtsSR7gq9Ck?usp=sharing>

៤. ១៤ Capacitive Sensor

Capacitive Sensor ឲ្យប្រើបិចចាតា Button ដែរ ឬទេនកាត់ដើរឯងទិន្នន័យទិន្នន័យចំណួនដោយរាយ
នឹងការបែបម្រូលបិមាណដែនឡើងត្រូវមានចាប់ពី ៣ និងចាប់ពី ៧ ដែលបានរាយក្នុងការបង្កើត។



ការរំភ្លោះ:

- VCC to 5V Arduino
- GND to GND Arduino
- OUT1 to pin2 Arduino
- OUT2 to pin3 Arduino
- OUT3 to pin4 Arduino
- OUT4 to pin5 Arduino

កូដ:

```

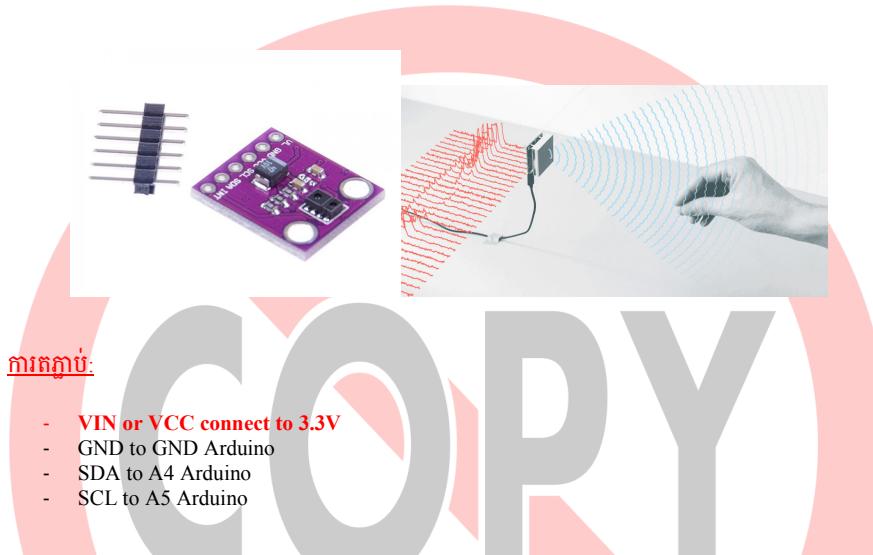
1. #define Button1 2
2. #define Button2 3
3. #define Button3 4
4. #define Button4 5
5. void setup()
6. {
7.     for (int pin=2; pin<=5; pin++) pinMode(pin,INPUT);
8.     Serial.begin(9600);
9. }
10.
11.void loop()
12.{
13.    if (digitalRead(Button1) == HIGH) Serial.println("Button-1");
14.    else if (digitalRead(Button2) == HIGH) Serial.println("Button-2");
15.    else if (digitalRead(Button3) == HIGH) Serial.println("Button-3");
16.    else if (digitalRead(Button4) == HIGH) Serial.println("Button-4");
17.    else Serial.println("None");
18.}

```

៤.១៦ Gesture Sensor

Gesture Sensor ជា Sensor បុយប្រភេទដែលមានសមត្ថភាពអាចធ្វើការបានស្ថានសកម្មភាពបស់មនុស្សបានដោយរឿងប្រព័ន្ធបច្ចេកវិទ្យាសាស្ត្រកំណុំទៀត ឬជាមួយធានានៃ Mathematical Algorithms។ នឹងមីនីរបីប្រព័ន្ធប្រភេទ Sensor នេះបានដោងជាប្រព័ន្ធ Download Library ខាងក្រោមនេះ :

https://github.com/sparkfun/SparkFun_APDS-9960_Sensor_Arduino_Library



```

1. #include <Wire.h>
2. #include <SparkFun_APDS9960.h>
3.
4. #define APDS9960_INT 2
5.
6. SparkFun_APDS9960 apds = SparkFun_APDS9960();
7. int isr_flag = 0;
8.
9. void setup() {
10.
11. pinMode(APDS9960_INT, INPUT);
12.
13. Serial.begin(9600);
14. Serial.println();
15. Serial.println(F("-----"));
16. Serial.println(F("SparkFun APDS-9960 - GestureTest")));
17. Serial.println(F("-----"));
18.

```

Formatted: Right: -25.4 mm

Deleted: 2

```
19. attachInterrupt(0, interruptRoutine, FALLING);
20.
21. if ( apds.init() ) {
22.   Serial.println(F("APDS-9960 initialization complete"));
23. } else {
24.   Serial.println(F("Something went wrong during APDS-9960 init!"));
25. }
26.
27. // Start running the APDS-9960 gesture sensor engine
28. if ( apds.enableGestureSensor(true) ) {
29.   Serial.println(F("Gesture sensor is now running"));
30. } else {
31.   Serial.println(F("Something went wrong during gesture sensor init!"));
32. }
33.
34. void loop() {
35.   if( isr_flag == 1 ) {
36.     detachInterrupt(0);
37.     handleGesture();
38.     isr_flag = 0;
39.     attachInterrupt(0, interruptRoutine, FALLING);
40.   }
41. }
42. void interruptRoutine() {
43.   isr_flag = 1;
44. }
45. void handleGesture() {
46.   if ( apds.isGestureAvailable() ) {
47.     switch ( apds.readGesture() ) {
48.       case DIR_UP:
49.         Serial.println("UP");
50.         break;
51.       case DIR_DOWN:
52.         Serial.println("DOWN");
53.         break;
54.       case DIR_LEFT:
55.         Serial.println("LEFT");
56.         break;
57.       case DIR_RIGHT:
58.         Serial.println("RIGHT");
59.         break;
60.       case DIR_NEAR:
61.         Serial.println("NEAR");
```

```

62.     break;
63. case DIR_FAR:
64.     Serial.println("FAR");
65.     break;
66. default:
67.     Serial.println("NONE");
68. }
69. }
70. }
```

៤.១៧ Barometric Sensor

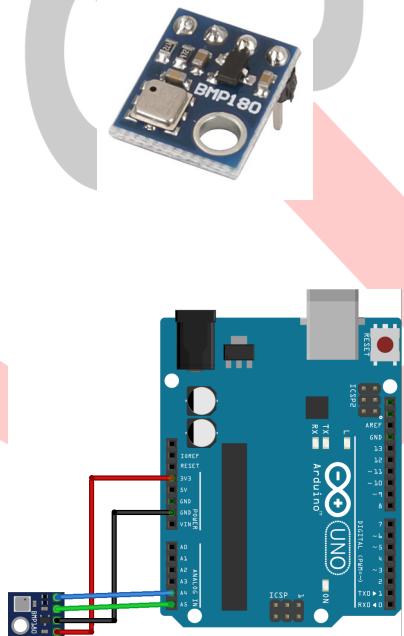
Barometric Sensor មួយប្រភេទដែលអាចក្លាត់អាកាសតាតុ រកយើងឲ្យយោះក្នុងសំនួរទិន្នន័យ។ ឥឡូវនេះជាដំឡើងស្ថិតិយោះបានចូលរួមនៅក្នុងការប្រើប្រាស់បច្ចេកទេសបំផុតសម្រាប់ការប្រើប្រាស់នៅក្នុងស្ថិយោះការប្រើប្រាស់បច្ចេកទេសបំផុតសម្រាប់ការប្រើប្រាស់បច្ចេកទេសបំផុត។ នឹងមិនបានប្រើប្រាស់បច្ចេកទេសបំផុតបានដូចខាងក្រោម។

<https://www.arduinolibraries.info/libraries/sparkfun-t5403-barometric-sensor-library>

or

https://github.com/LowPowerLab/SFE_BMP180

រាយការណ៍



- VCC connect to 3.3V
- GND connect to GND

- SDA connect to A4
- SCL connect to A5

ផ្នែក:

```
1. #include <SFE_BMP180.h>
2. #include <Wire.h>
3. #define ALTITUDE 1655.0
4.
5. SFE_BMP180 pressure;
6.
7. void setup()
8. {
9.   Serial.begin(9600);
10.  Serial.println("REBOOT");
11.
12. if (pressure.begin())
13.  Serial.println("BMP180 init success");
14. else
15. {
16.
17.  Serial.println("BMP180 init fail\n\n");
18.  while(1); // Pause forever.
19. }
20.
21.
22.void loop()
23.{
24. char status;
25. double T,P,p0,a;
26.
27. Serial.println();
28. Serial.print("provided altitude: ");
29. Serial.print(ALTITUDE,0);
30. Serial.print(" meters, ");
31. Serial.print(ALTITUDE*3.28084,0);
32. Serial.println(" feet");
33.
34.
35. status = pressure.startTemperature();
36. if (status != 0)
37. {
38.  delay(status);
```

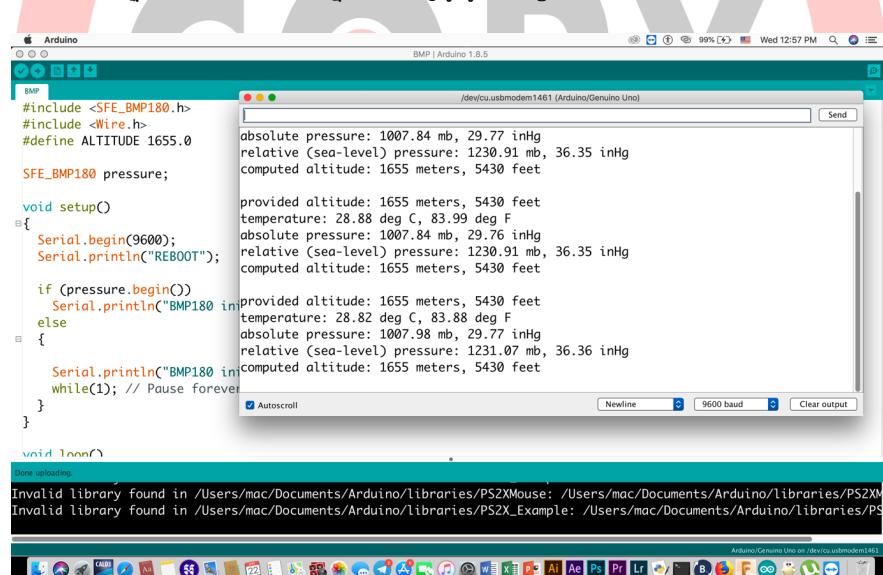
```
39.  
40. status = pressure.getTemperature(T);  
41. if (status != 0)  
42. {  
43.  
44.   Serial.print("temperature: ");  
45.   Serial.print(T,2);  
46.   Serial.print(" deg C, ");  
47.   Serial.print((9.0/5.0)*T+32.0,2);  
48.   Serial.println(" deg F");  
49.  
50. status = pressure.startPressure(3);  
51. if (status != 0)  
52. {  
53.   delay(status);  
54.  
55.  
56.   status = pressure.getPressure(P,T);  
57.   if (status != 0)  
58.   {  
59.     // Print out the measurement:  
60.     Serial.print("absolute pressure: ");  
61.     Serial.print(P,2);  
62.     Serial.print(" mb, ");  
63.     Serial.print(P*0.0295333727,2);  
64.     Serial.println(" inHg");  
65.  
66.     p0 = pressure.sealevel(P,ALTITUDE); // we're at 1655 meters (Bou  
lder, CO)  
67.     Serial.print("relative (sea-level) pressure: ");  
68.     Serial.print(p0,2);  
69.     Serial.print(" mb, ");  
70.     Serial.print(p0*0.0295333727,2);  
71.     Serial.println(" inHg");  
72.  
73.     a = pressure.altitude(P,p0);  
74.     Serial.print("computed altitude: ");  
75.     Serial.print(a,0);  
76.     Serial.print(" meters, ");  
77.     Serial.print(a*3.28084,0);  
78.     Serial.println(" feet");  
79.   }  
80. else Serial.println("error retrieving pressure measurement\n");
```

```

81.    }
82.    else Serial.println("error starting pressure measurement\n");
83.    }
84.    else Serial.println("error retrieving temperature measurement\n");
85.    }
86.    else Serial.println("error starting temperature measurement\n");
87.
88.    delay(5000); // 
89.

```

នេះពេលមួយបែក Serial Monitor អ្នកនឹងយើងដូចខាងក្រោម :



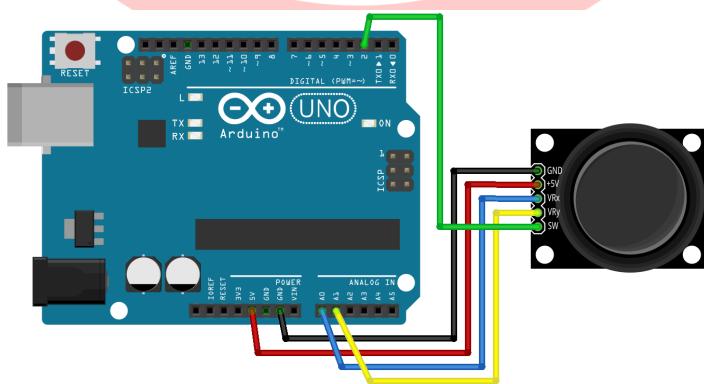
Unit	1 hPa =
Pascal	100 Pa
Newton's per square meter	100 N/m
Atmosphere	0.000986923 atm
Bar	0.001 bar
Millibar	1 mbar
Millimeters Mercury	0.750063755 mmHg
Torr	0.750061683 torr
Pounds per square inch	0.014503774 psi

៤.១៤ Joystick

Joystick ជាអ្នកដំឡើងដែលបានប្រើប្រាស់ដើម្បីបញ្ចូនការងាររបស់ខ្លួន។



ការរៀបចំ:



សិទ្ធិ:

```

1. #define X A0
2. #define Y A1
3. #define SW 2
4. void setup() {
5.   Serial.begin(9600);
6.
7. }
8. void loop() {
9.   // put your main code here, to run repeatedly:
10.  int Data_X = analogRead(X);
11.  int Data_Y = analogRead(Y);
12.  int Data_SW = digitalRead(SW);
13.  Serial.println(Data_X);
14.  Serial.println(Data_Y);
15.  Serial.println(Data_SW);
16.  delay(200);
17.
18. }
```

៤.៩ គម្រោង Sensor

Compass Sensor ជាអ្នកប្រភេទដែលអាចវាសកម្មភាពដែនម៉ារ៉ែទិចនៅក្នុង 3 Axis។ ចំពោះ Sensor ប្រភេទនេះគឺមែនមួយករណីប្រចាំឆ្នាំដែលបានប្រើប្រាស់ប្រភេទឱ្យប្រតិស្ថានរកចិត្តនៃលាភមួយ (Navigation System)។ នឹងធ្វើប្រចាំឆ្នាំប្រភេទ Sensor នេះបានអ្នកប្រព័ន្ធពាយក Library មួយមកប្រើប្រាស់ :

<https://github.com/jarzebski/Arduino-HMC5883L>

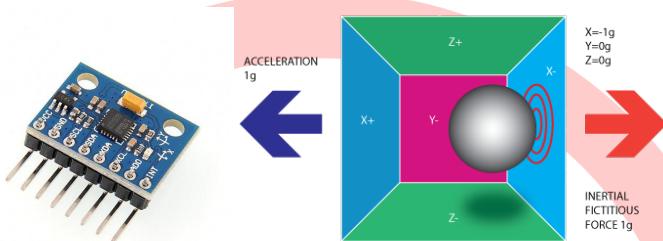
ការរៀបចំ:

- VCC Connect to 5V Arduino
- GND Connect to GND Arduino
- SDA Connect to A4 Arduino
- SCL Connect to A5 Arduino

ក្នុង:

៤.២០ MPU6050

ជាប្រភេទ Sensor មួយប្រភេទដែលមានសមត្ថភាពភាពធ្វើការរាស់កម្រិតលម្អិតតាមទិន្នន័យ X Y Z នានា ពាណីខ្លួនគ្នាដែលបានប្រើប្រាស់ជាមួយប្រភេទកញ្ចប់លម្អិត ខាងក្រោមនេះជាផ្លូវការ Balancing ដ៏លើម៉ា ដើម្បីប្រើប្រាស់ Sensor ប្រភេទនេះបានអ្នកគ្រប់ទាញយក Library ពីរបៀបប្រើប្រាស់ : <https://github.com/jarzebski/Arduino-MPU6050> and <https://github.com/ElectronicCats/mpu6050>



ការគ្រប់:

- VCC Connect to 5V Arduino
- GND Connect to GND Arduino
- SDA Connect to A4 Arduino
- SCL Connect to A5 Arduino

ក្នុង:

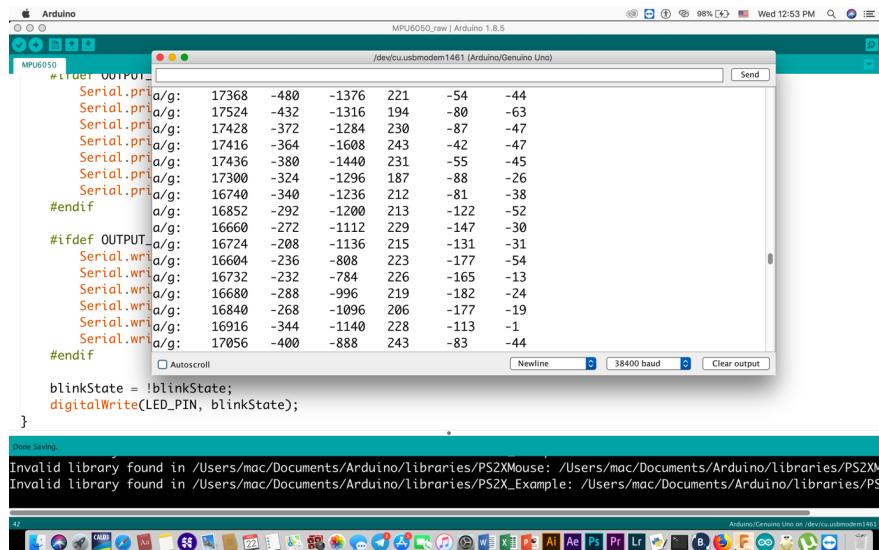
```

1. #include "I2Cdev.h"
2. #include "MPU6050.h"
3.
4. #if I2CDEV_IMPLEMENTATION == I2CDEV_ARDUINO_WIRE
5.     #include "Wire.h"
6. #endif
7.
8. MPU6050 accelgyro;
9. int16_t ax, ay, az;
10.int16_t gx, gy, gz;
11.
12.#define OUTPUT_READABLE_ACCELGYRO
13.
14.#define LED_PIN 13
15.bool blinkState = false;
16.
17.void setup() {
18.

```

```
19. #if I2CDEV_IMPLEMENTATION == I2CDEV_ARDUINO_WIRE
20.     Wire.begin();
21. #elif I2CDEV_IMPLEMENTATION == I2CDEV_BUILTIN_FASTW
22.     Fastwire::setup(400, true);
23. #endif
24. Serial.begin(38400);
25. Serial.println("Initializing I2C devices...");
26. accelgyro.initialize();
27.
28. Serial.println("Testing device connections...");
29. Serial.println(accelgyro.testConnection() ? "MPU6050 connection successful" : "MPU6050 connection failed");
30. pinMode(LED_PIN, OUTPUT);
31. }
32.
33.void loop() {
34.     accelgyro.getMotion6(&ax, &ay, &az, &gx, &gy, &gz);
35.
36. #ifdef OUTPUT_READABLE_ACCELGYRO
37.     Serial.print("a/g:\t");
38.     Serial.print(ax); Serial.print("\t");
39.     Serial.print/ay); Serial.print("\t");
40.     Serial.print(az); Serial.print("\t");
41.     Serial.print(gx); Serial.print("\t");
42.     Serial.print(gy); Serial.print("\t");
43.     Serial.println(gz);
44. #endif
45.
46. #ifdef OUTPUT_BINARY_ACCELGYRO
47.     Serial.write((uint8_t)(ax >> 8)); Serial.write((uint8_t)(ax & 0xFF));
48.     Serial.write((uint8_t)(ay >> 8)); Serial.write((uint8_t)(ay & 0xFF));
49.     Serial.write((uint8_t)(az >> 8)); Serial.write((uint8_t)(az & 0xFF));
50.     Serial.write((uint8_t)(gx >> 8)); Serial.write((uint8_t)(gx & 0xFF));
51.
52.     Serial.write((uint8_t)(gy >> 8)); Serial.write((uint8_t)(gy & 0xFF));
53. #endif
54.
55.     blinkState = !blinkState;
56.     digitalWrite(LED_PIN, blinkState);
57. }
```

នៅពេលដែលអ្នកបើក Serial Monitor អ្នកនឹងយើងឲ្យចូលទៅមានព្រម :



```
#include <Wire.h>
#include <MPU6050.h>

MPU6050 mpu;
Lcd1602 lcd;

void setup() {
    mpu.initialize();
    lcd.begin(16, 2);
}

void loop() {
    float roll = mpu.getRoll();
    float pitch = mpu.getPitch();
    float yaw = mpu.getYaw();

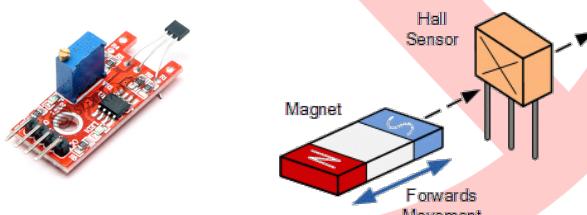
    lcd.print("Roll: ");
    lcd.print(roll);
    lcd.print(" Pitch: ");
    lcd.print(pitch);
    lcd.print(" Yaw: ");
    lcd.print(yaw);

    delay(1000);
}
```

Done Saving
Invalid library found in /Users/mac/Documents/Arduino/libraries/PS2Mouse: /Users/mac/Documents/Arduino/libraries/PS2X
Invalid library found in /Users/mac/Documents/Arduino/libraries/PS2X_Example: /Users/mac/Documents/Arduino/libraries/PS2X

៤.២២ Hall Sensor

Hall Sensor ជា Sensor មួយប្រភេទដែលមានការបំប្លែងប្រឈមតម្លៃតាមកម្រិតដែនម៉ាញ្ញីទិន្នន័យ ដែលវាតាន់បងបែកតម្លៃបសនៃដែនម៉ាញ្ញីចាប់បុណ្យសំណាល់លូ។



ការគ្រប់:

- VCC Connect to 5V Arduino
- GND Connect to GND Arduino
- A0 Connect to A0 Arduino

កូដ:

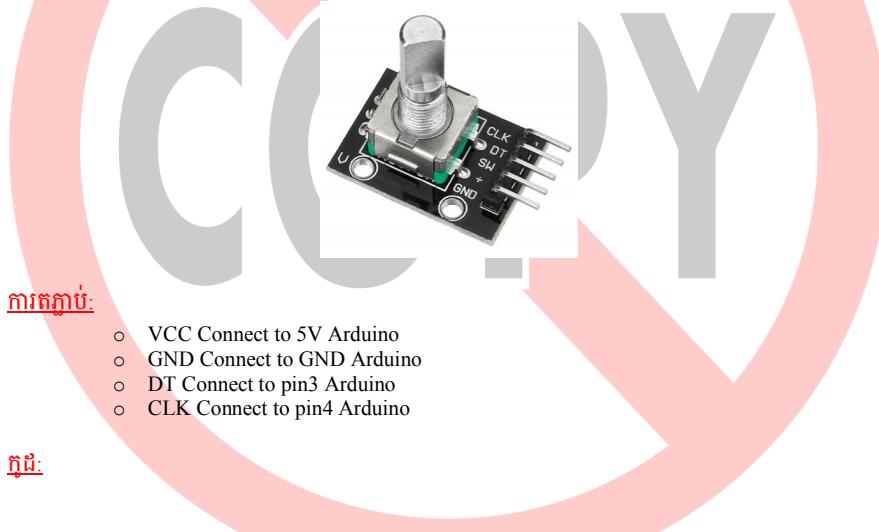
1. int hallSensorPin = A0;
2. int state = 0;
- 3.

```

4. void setup()
5. {
6.   Serial.begin(9600);
7. }
8. void loop(){
9.
10. state = analogRead(hallSensorPin);
11. Serial.println(state);
12.}
```

៤.២៣ Encoder

Rotary Encoder ជាប្រភេទ Sensor មួយដែលរកចកំណត់ចំណុចប្រចាំង Position ឬរាយ បើកកំណត់ចំនួនប្រស់ខ្លួន។ តើវានៅពីអ្វីដែលទទួលបានពី Encoder នាទិសនឹង ដែលរកចកាត់ថ្មី Analog ឬកូដ Digital ។ ជាពីរគ្រប់គ្រង់ដែលបានប្រើប្រាស់ជាមួយមួយទៅ (Servo Motor, DC Motor & Stepper Motor)។



```

1. #define outputA 3
2. #define outputB 4
3. int counter = 0;
4. int aState;
5. int aLastState;
6. void setup() {
7.   pinMode (outputA,INPUT);
8.   pinMode (outputB,INPUT);
```

```

9.
10. Serial.begin (9600);
11. // Reads the initial state of the outputA
12. aLastState = digitalRead(outputA);
13.}
14.void loop() {
15. aState = digitalRead(outputA); // Reads the "current" state of the output
A
16. // If the previous and the current state of the outputA are different, that
means a Pulse has occured
17. if (aState != aLastState){
18. // If the outputB state is different to the outputA state, that means the e
ncoder is rotating clockwise
19. if (digitalRead(outputB) != aState) {
20. counter++;
21. } else {
22. counter--;
23. }
24. Serial.print("Position: ");
25. Serial.println(counter);
26. }
27. aLastState = ; // Updates the previous state of the outputA with the curr
ent state
28.}

```

- ការអនុវត្តន៍ ៩** អ្នកគ្រែទីនឹងប្រព័ន្ធសាន់មួយដែលអ្នកគ្រែទីនឹងប្រព័ន្ធឌីជីថទិនមួយចំនួនដូចជា៖
- Arduino UNO (Nano or Mega) x1
 - LCD with IC2 Module x1
 - Smoke Sensor x1
 - Humidity Sensor x1
 - Flame Sensor
 - IR receiver with Remote x1
 - relay with lamp 220V AC x1
 - Rain sensor x1
 - buzzer x3
 - jumper

ដែនកំណត់: អ្នកគ្រែទាញរាយកិនិមួយដែលទទួលបានពី Humidity Sensor (Temperature and Humidity) មក បង្ហាញនៅលើ LCD 16x2 ។ នៅពេលដែលមានស្ម័ន បុរីចំហាយហ្មាល់ចូលមកការ៖ Smoke Sensor អ្នកគ្រែទី ឲ្យ Buzzer បន្ទីជាសម្រេង។ បន្ទាប់មកទៀតនៅពេលដែលមានដំណើរការនៅលើ Rain Sensor (ប្រើបង្កើចធាន មានពេលភ្លោះ) អ្នកគ្រែទី ឲ្យ Buzzer មួយទៀតបន្ទីជាសម្រេង (Buzzer ខុសត្រា សម្រេងកំខុសត្រា) ។ ចំណោក ឬ Flame Sensor វិញ នៅពេលដែលមានទន្លឹក្រឹង (អតិភីយ) អ្នកគ្រែទី ឲ្យ Buzzer មួយទៀតបន្ទីសម្រេង (Buzzer ខុសត្រា សម្រេងកំខុសត្រា) ។ អ្នកគ្រែទីប្រាស់ LDR ដើម្បីកំណត់តម្លៃដែលរករារ LED មួយដែលតែលយប់អំ

ពុល LED ភី (កន្លឹមធាក់ស្ថុងនៅក្នុងបន្ទប់) នៅពេលដែលព្រឹកអ្នកត្រូវឡើងអំពុល LED រលកតិច (យក ទន្លឹមធីនឹង Flash មកបញ្ចាំងធាក់ LDR)។ លក្ខខណ្ឌចុងក្រាយអ្នកត្រូវយក Remote ដែលមានធានាល័ងខ្ពស់ ត្រូវឡើងការបិទបិទអំពុល 220V AC មួយ (ចុចលើកទី ១ ហើយ ចុចលើកទី ២ បិទ) ។

បញ្ជាក់ : មុខងារទាំងអស់ត្រូវដំណើរការឱ្យមកការ និងបែងចែកសម្រាប់ Buzzer ឲ្យបានច្បាស់នៅពេលមុខងារ ណាមួយដែលការចូលសរុបការយល់ច្រល់។

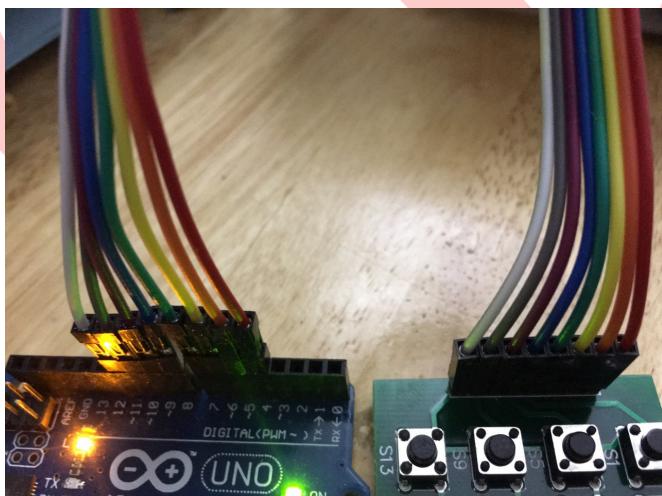
ការអនុវត្តន៍ ២ អ្នកត្រូវកែម្រួងប្រព័ន្ធភាក់សោរជាយប្រើប្រាស់ Keypad ដើម្បីរួមបញ្ចូល Password

- Arduino UNO (Nano or Mega) x1
- Keypad x1
- led
- buzzer
- LCD with I2C module x1
- jumper

ដែនកំណត់ : អ្នកត្រូវកំណត់លេខ 6789 ជាលេខសំងារនៅក្នុងការដោះស្រាយ ។ នៅពេលដែលអ្នករាយបញ្ហាលេខសម្ងាត់នានត្រឹមត្រូវ អ្នកត្រូវឡើងអំពុល LED Blink រយៈពេល ៥ វិនាទី។ នៅក្នុងករណីដែលអ្នករាយបញ្ហាលេខសម្ងាត់ខ្លួន ៣ នឹងអ្នកត្រូវឡើង Buzzer នូនឹសម្រាប់ឡើង។ Buzzer នឹងបណ្តូរប៉ុន្មោះដែលការនៅពេលដែលអ្នករាយបញ្ហាផ្លូវការ។

ការរួមប្រាស់ Keypad :

ការគ្រប់:



ដើម្បីរួមប្រាស់ Keypad បានដោយដាយស្ថុលអ្នកត្រូវទាយយក Library មួយខាងក្រោមនេះ :

<https://github.com/Chris--A/Keypad>

ចូល:

```

1. #include <Keypad.h>
2.
3. const byte ROWS = 4;
4. const byte COLS = 4;
5.
6. char hexaKeys[ROWS][COLS] = {
7.     {'0','1','2','3'},
8.     {'4','5','6','7'},
9.     {'8','9','A','B'},
10.    {'C','D','E','F'}
11.};
12.
13. byte rowPins[ROWS] = { 7,6,5,4 };
14. byte colPins[COLS] = { 11,10,9,8,};
15.
16.
17. Keypad customKeypad = Keypad( makeKeymap(hexaKeys), rowPins, colPins, R
    OWS, COLS);
18.
19. void setup(){
20.     Serial.begin(9600);
21. }
22.
23. void loop(){
24.     char customKey = customKeypad.getKey();
25.
26.     if (customKey){
27.         Serial.println(customKey);
28.     }
29. }
```

ដើម្បីងាយស្រួលទៅក្នុងការកំណត់លក្ខខណ្ឌលេខកុដសម្ងាត់អ្នកភាពទាយយក Library Password មួយមកប្រើប្រាស់ : <https://playground.arduino.cc/Code/Password/>

ការរក្សាទុក ៣ អ្នកត្រូវក្លើងប្រព័ន្ធសុវត្ថិភាពមួយ ដែលអ្នកត្រូវការក្រឹងឡាចិចត្រូនិកមួយចំនួនដូចជា៖

- Arduino UNO (Nano or Mega) x1
- Ultrasonic (អ្នកភាពប្រើប្រាស់ Sensor ធ្វើឱ្យកំណត់ផ្ទើផ្លូវ Motion Sensor ឬ Laser with LDR) x1
- Keypad x1
- Buzzer x1
- led
- Relay with lamp 220V AC x1
- LCD with I2C module x1
- jumper

ដែនកំណត់ : អ្នកត្រូវកំណត់លេខ 1234 ដានខាងលើដែនដើរការរបស់ Ultrasonic នានានីយោចនៅលេខ 1234 នៅលើ Keypad ដែលនឹងបង្ហាញនៅលើ LCD ត្រូវបានត្រូវបានបង្ហាញនៅលើ LCD និងចូលរួមតុល LED ត្រូវយោបល់ 5 វិនាទី បុំនែនក្នុង

ករណីដែលកាយបញ្ហាលេខសំងាត់ខ្ពស់ទូន 3 ដីន មួកត្រូវធ្វើឲ្យ Buzzer បន្ថីជាសម្បែង និងធ្វើឲ្យ អំពុល 220V AC ត្រូវ។ ឱ Buzzer នឹង អំពុលបណ្តុះបណ្តុះដំណើរការទាំងឡាយៗក្នុងការត្រួតពិនិត្យការងារ ឱចិនបញ្ហាលេខសំងាត់ទូន នៅពេលមាននរណាម្នាក់ដែរកាលមុខ Ultrasonic ដែលមិនបានចិត្តបញ្ហាលេខសំងាត់មួកត្រូវធ្វើឲ្យ Buzzer និងអំពុល 220V AC ចាប់ធ្វើឲ្យដំណើរការ ការណើឲ្យបណ្តុះបណ្តុះដំណើរការទាំងឡាយៗ នៅក្នុងករណីដែលមានមួកត្រូវរណាម្នាក់ចិត្តបញ្ហាលេខសំងាត់បានត្រឹមត្រូវ។ លើសពីនេះទៅទៀតមួកត្រូវមានមឺនយកការងារស្ថិតនៃបញ្ហាលេខសំងាត់។ កល់ការរាយបញ្ហាលេខសំងាត់ បុក់ឆ្លាស់បញ្ហាលេខសំងាត់ត្រូវបង្ហាញនៅលើ LCD ។

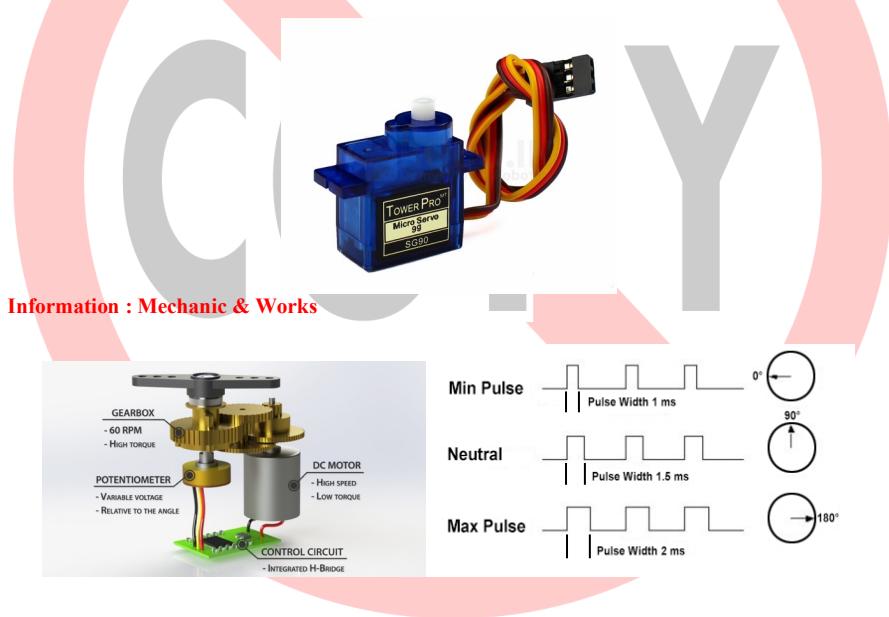
បញ្ជាក់ : ប្រព័ន្ធដំណើរការមួយនេះមិនមានសរុបតាម 100% នោះទេ (ត្រូវបានចាមពលផ្តើមត្រូវបង្ហាញនៅលើ LCD និងត្រូវការងារដំឡើងដោយស្ថិតត្រូវបង្ហាញនៅលើ LCD នៅពេលដំឡើង Restart /ON/OFF Microcontroller ត្រូវបង្ហាញនៅលើ LCD នៅពេលដំឡើង EEPROM.....)



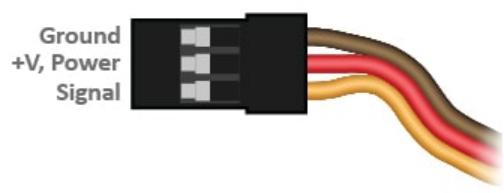
ចំណែក និង ការសម្រាប់ប្រើប្រាស់ការបញ្ចប់ផ្ទាល់ខ្លួន

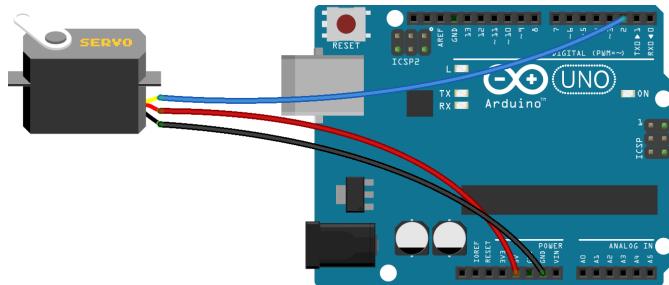
៤.៩ Servo Motor

Servo Motor ជាការប្រើប្រាស់ដែលត្រូវការដាយការកំណត់មុខ្លួយ: មុខ្លួយ Servo Motor មាន 2 ប្រភេទដីជី ប្រភេទ DC Motor និងប្រភេទ AC Motor។ ចំពោះ Servo Motor ខ្លះវាបាត់ប្រើបាស់បានត្រឹមចំនួន 180 Degree បុំពីរ Servo Motor ខ្លះវាបាត់ប្រើបាស់បានចំនួន 360 Degree។ នៅក្នុងការសិក្សាយើង សូមអ្នកើសថា Servo Motor ប្រភេទ DC និងការបាត់ប្រើបាស់បានចំនួន 180 Degree ។ Servo Motor មិនដឹងថាមីនិង DC មួយគឺត្រូវបានប្រើប្រាស់។ ដូច្នេះបើសិនជាការ Arduino IDE របស់អ្នកមិនមាន Library របស់ Servo Motor នោះទេ អ្នកត្រូវចូលទៅ Download Link មួយនេះ : <https://github.com/arduino-libraries/Servo>



ការប្រើប្រាស់



រូប៖

```

1. #include <Servo.h>
2. Servo myservo;
3. int pos = 0;
4. void setup()
5. {
6.   myservo.attach(2);
7. }
8. void loop() {
9.   for (pos = 0; pos <= 180; pos += 1) {
10.     myservo.write(pos);
11.     delay(15);
12.   }
13.   for (pos = 180; pos >= 0; pos -= 1) {
14.     myservo.write(pos);
15.     delay(15);
16.   }
17. }
```

អនុវត្តន៍: អ្នកត្រូវបង្កើតលិខិត MPU6050 ដូចមានប្រព័ន្ធសម្រាប់ Servo Motor ។ ចូរសរសែរ Program មួយនៃពេលវិជ្ជោយ
អ្នកត្រូវបង្កើតលិខិត MPU6050 តាមទិន្នន័យ Y អ្នកត្រូវបង្កើតលិខិត Servo Motor លើកការណាមបន្ទាប់ប្រព័ន្ធដែលត្រូវ Rotate
នៅលើ Y

ការពេញរូប៖

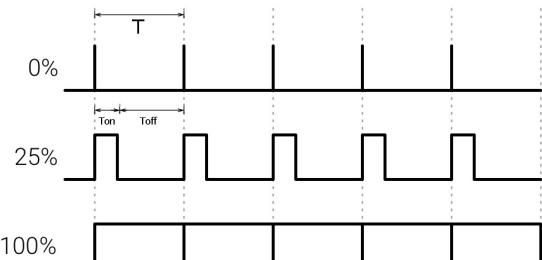
- **Servo Motor**
 - Pin Out Connect to Pin2 Arduino
 - GND Connect to GND Arduino
 - VCC Connect to 5V Arduino
- **MPU6050**
 - SDA Connect to A4 Arduino
 - SCL Connect to A5 Arduino

- GND Connect to GND Arduino
- VCC Connect to 5V Arduino

រូប:

```
1. #include <Wire.h>
2. #include <MPU6050.h>
3. #include <Servo.h>
4. Servo sg90;
5. int servo_pin = 2;
6. MPU6050 sensor ;
7. int16_t ax, ay, az ;
8. int16_t gx, gy, gz ;
9.
10.void setup ()
11.{  
12.sg90.attach ( servo_pin );
13.Wire.begin ( );
14.Serial.begin ( 9600);
15.Serial.setTimeout(100);
16.Serial.println ( "Initializing the sensor" );
17.sensor.initialize ( );
18.Serial.println (sensor.testConnection ( ) ? "Successfully Connected" : "Co  
nnnection failed");
19.delay (1000);
20.Serial.println ( "Taking Values from the sensor" );
21.delay (1000);
22.}  
23.void loop ()
24.{  
25.sensor.getMotion6 (&ax, &ay, &az, &gx, &gy, &gz);
26.Serial.print ("ay = ");
27.Serial.println (ay);
28/ay = map (ay, 17000, -17000, 180, 0) ;
29.Serial.println (ay);
30.sg90.write (ay);
31.}
```


❖ Pulse Width Modulation



ការអេបត្រ: (វិធីទី ១)

- Connect OUTPUT A to DC Motor
- Connect +12V Power to 5V Arduino (For DC Motor 5V)
- Connect Power GND to GND Arduino
- Connect IN1 to Pin2 Arduino (Arduino Digital Pin, PWM Pin or Analog Pin)
- Connect IN2 to Pin3 Arduino (PWM Pin)

រូប:

```

1. #define speeds 3 // 3 5 6 9 10 11
2. #define directions 2
3.
4. void setup() {
5.   pinMode(speeds,OUTPUT);
6.   pinMode(directions,OUTPUT);
7. }
8.
9. void loop() {
10.  digitalWrite(directions,1);
11.  analogWrite(speeds,150);
12.
13.}

```

បញ្ជាក់ : នៅពេលដែលអ្នកប្រើប្រាស់សៀវភៅបែបនេះក្នុងការបញ្ចូមទៅ អ្នកត្រូវចងចាំថាពេលដែលដែលអ្នកនឹងបានការបញ្ចូមទៅ អ្នកត្រូវចងចាំថាទៅពេលដែលអ្នកនឹងការបញ្ចូមទៅ

Direction 1 ឬ HIGH លើក្នុង Full Speed តើ ០ ។ នៅក្នុងក្រណីដែលអ្នកកំណត់ Direction ០ ឬ LOW លើក្នុង Full Speed តើ ២៥៥ ។

ការអេបត្រ: (វិធីទី ២)

- Connect OUTPUT A to DC Motor
- Connect +12V Power to 5V Arduino (For DC Motor 5V)

- Connect Power GND to GND Arduino
- Connect 5V to 5V Pin Arduino
- Connect EN1 to Pin 3 Arduino (**PWM Pin**)
- Connect IN1 to Pin2 Arduino (**Arduino Digital Pin, PWM Pin or Analog Pin**)
- Connect IN2 to Pin4 Arduino (**Arduino Digital Pin, PWM Pin or Analog Pin**)

រូប:

```

14.#define speeds 3 // 3 5 6 9 10 11
15.#define directions_1 2
16.#define directions_2 4
17.
18.void setup() {
19. pinMode(speeds,OUTPUT);
20. pinMode(directions_1,OUTPUT);
21. pinMode(directions_2,OUTPUT);
22. }
23.
24.void loop() {
25. digitalWrite(directions_1,1);
26. digitalWrite(directions_2,0);
27. analogWrite(speeds,150);
28.
29. }
```

អនឡាត់ខ្លួន ប្រើប្រាស់ Joystick (ទិន្នន័យក្បឹង X) ដាក់សម្រាប់ប្រើប្រាស់ជាមុនរបស់មុន (ប្រើប្រាស់ដីទី 2)។

ការពេញលេញ:

- DC Motor with Driver L298 module
 - Connect OUTPUT A to DC Motor
 - Connect +12V Power to 5V Arduino (For DC Motor 5V)
 - Connect 5V to 5V Pin Arduino
 - Connect Power GND to GND Arduino
 - Connect EN1 to Pin 3 Arduino (**PWM Pin**)
 - Connect IN1 to Pin2 Arduino (**Arduino Digital Pin, PWM Pin or Analog Pin**)
 - Connect IN2 to Pin4 Arduino (**Arduino Digital Pin, PWM Pin or Analog Pin**)
- JoyStick
 - Connect GND to GND Arduino
 - Connect VCC to 5V Arduino
 - Connect VRX or X to A0 Arduino

```
1. #define PinMotor A0
2. #define ControlMotor1 2
3. #define ControlMotor2 4
4. #define PinSpeed 3
5.
6. int ValuePinMotor;
7. int NewValuePinMotor;
8.
9. int SpeedValue1;
10.int SpeedValue2;
11.int SpeedValueRight;
12.int SpeedValueLeft;
13.
14.void setup() {
15.
16. pinMode(ControlMotor1 ,OUTPUT);
17. pinMode(ControlMotor2 ,OUTPUT);
18. pinMode(PinSpeed ,OUTPUT);
19.
20. Serial.begin(9600);
21.
22.}
23.void loop() {
24.
25. ValuePinMotor = analogRead(PinMotor);
26. NewValuePinMotor = map(ValuePinMotor, 0, 1023, 0, 255);
27.
28. if( NewValuePinMotor < 150){
29.
30. digitalWrite( ControlMotor1 ,HIGH);
31. digitalWrite( ControlMotor2 ,LOW);
32.
33. SpeedValue1 = map(NewValuePinMotor, 0, 140, 255, 0);
34.
35. analogWrite( PinSpeed , SpeedValue1);
36. }
37. else if( NewValuePinMotor > 120){
38. digitalWrite( ControlMotor1 ,LOW);
39. digitalWrite( ControlMotor2 ,HIGH);
40.
41. SpeedValue1 = map(NewValuePinMotor, 140, 255, 0, 255);
42.
43. analogWrite( PinSpeed , SpeedValue1);
```

```

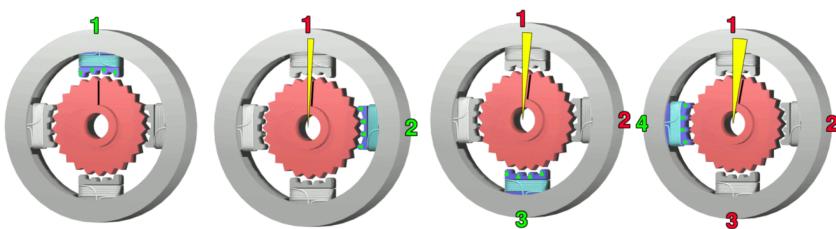
44. }
45. else{
46. digitalWrite( ControlMotor1 ,LOW);
47. digitalWrite( ControlMotor2 ,LOW);
48. SpeedValue1 = 0;
49.
50. analogWrite( PinSpeed , SpeedValue1);
51. }
52.
53. Serial.println(NewValuePinMotor);
54. delay(200);
55.
56.}

```

FIG Stepper Motor with Driver



Stepper Motor ជាប្រភេទ DC មួនរដល់ធ្វើការបញ្ចាស់ទីតាត់ហានុចំពោះ។ ធានមូតា
នៅក្នុងមួនរដល់មួយមានបណ្តុះបុងចាប់ពីរៀងទៅ ដែលបានរៀបចំជាប្រមិជលគេឡើងថា
Phases ដាយមួននឹងធ្វើការផ្តាស់ទីមួយជំហាននៅក្នុងមួយ Phase ។ ខាងក្រោមនឹងបង្ហាញពីនំណើរ
ការរបស់ Stepper Motor នៃលជ្ជប្រភេទមួន Unipolar មាន 4 Phase



ជាទុកដានបែងចែកប្រភេទមួយនៃប្រភេទគ្រឿងប្រភេទគ្រឿង Bipolar motor និង Unipolar motor។

ចំពោះការប្រើប្រាស់មួយទៅប្រភេទនេះដើរ យើងបានទទួលសម្រាតិតារច្រើនរួចរាល់ជាមួយគ្រឿងប្រភេទមួយនៃប្រភេទគ្រឿងគ្រឿងការកំណត់ទៅលើចំណុច។

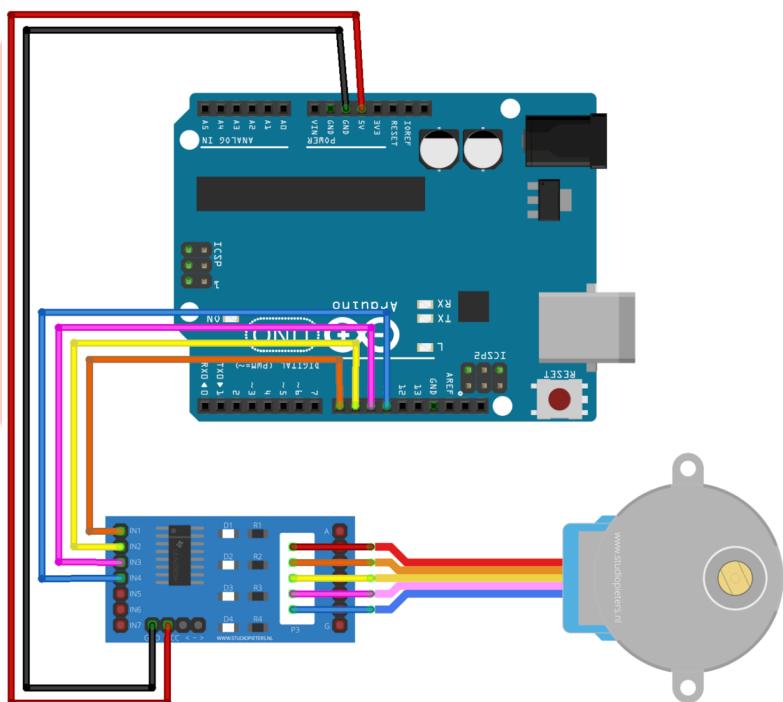
- Positioning: មួយទៅ Stepper ដែលបានប្រើបាស់បានលើការកំណត់ទៅលើចំណុច។ ជាមួយគ្រឿងប្រភេទមួយនៃប្រភេទគ្រឿងគ្រឿងការកំណត់ទៅលើចំណុច។ មួយទៅ Stepper ដែលបានប្រើបាស់បានលើការកំណត់ទៅលើចំណុច។ ជាមួយគ្រឿងប្រភេទមួយនៃប្រភេទគ្រឿងគ្រឿងការកំណត់ទៅលើចំណុច។ ជាមួយគ្រឿងប្រភេទមួយនៃប្រភេទគ្រឿងគ្រឿងការកំណត់ទៅលើចំណុច។
- Speed Control: ចំពោះការកំណត់ទៅលើប្រើប្រាស់មួយទៅ Stepper វាបានប្រើបាស់បានលើការកំណត់ទៅលើចំណុច។ ជាមួយគ្រឿងប្រភេទមួយនៃប្រភេទគ្រឿងគ្រឿងការកំណត់ទៅលើចំណុច។ ជាមួយគ្រឿងប្រភេទមួយនៃប្រភេទគ្រឿងគ្រឿងការកំណត់ទៅលើចំណុច។ ជាមួយគ្រឿងប្រភេទមួយនៃប្រភេទគ្រឿងគ្រឿងការកំណត់ទៅលើចំណុច។
- Low Speed Torque: ជាមួយគ្រឿងប្រភេទមួយទៅ DC នូវពេលដែលបានប្រើបាស់បានលើការកំណត់ទៅលើចំណុច។ ប៉ុន្មានចំពោះមួយទៅ Stepper នៅពេលដែលបានប្រើបាស់បានលើការកំណត់ទៅលើចំណុច។ ជាមួយគ្រឿងប្រភេទមួយនៃប្រភេទគ្រឿងគ្រឿងការកំណត់ទៅលើចំណុច។ ជាមួយគ្រឿងប្រភេទមួយនៃប្រភេទគ្រឿងគ្រឿងការកំណត់ទៅលើចំណុច។ ជាមួយគ្រឿងប្រភេទមួយនៃប្រភេទគ្រឿងគ្រឿងការកំណត់ទៅលើចំណុច។

ទីផ្សារនៃគ្មានសម្រាតិតិបន្ទិចបន្ទុចដែលយុទ្ធមានរួចរាល់ជាមួយគ្រឿងប្រភេទមួយនៃប្រភេទគ្រឿងគ្រឿងការកំណត់ទៅលើចំណុច។

- Low Efficiency ចំពោះបន្ទុកអតិសនិត្តដែលមួយទៅ Stepper motor ត្រូវការក្នុងពេលដែលបានប្រើបាស់បានលើការកំណត់ទៅលើចំណុច។ ជាមួយគ្រឿងប្រភេទមួយនៃប្រភេទគ្រឿងគ្រឿងការកំណត់ទៅលើចំណុច។ ជាមួយគ្រឿងប្រភេទមួយនៃប្រភេទគ្រឿងគ្រឿងការកំណត់ទៅលើចំណុច។ ជាមួយគ្រឿងប្រភេទមួយនៃប្រភេទគ្រឿងគ្រឿងការកំណត់ទៅលើចំណុច។ ជាមួយគ្រឿងប្រភេទមួយនៃប្រភេទគ្រឿងគ្រឿងការកំណត់ទៅលើចំណុច។ ជាមួយគ្រឿងប្រភេទមួយនៃប្រភេទគ្រឿងគ្រឿងការកំណត់ទៅលើចំណុច។
- Limited High Speed Torque ជាមួយគ្រឿងប្រភេទមួយទៅ Stepper នៅពេលដែលបានប្រើបាស់បានលើការកំណត់ទៅលើចំណុច។ នៅពេលបានប្រើបាស់បានលើការកំណត់ទៅលើចំណុច។ នៅពេលបានប្រើបាស់បានលើការកំណត់ទៅលើចំណុច។ នៅពេលបានប្រើបាស់បានលើការកំណត់ទៅលើចំណុច។ នៅពេលបានប្រើបាស់បានលើការកំណត់ទៅលើចំណុច។
- No Feedback វាអាចិនបានប្រើបាស់បានលើការកំណត់ទៅលើចំណុច។ នៅពេលបានប្រើបាស់បានលើការកំណត់ទៅលើចំណុច។ នៅពេលបានប្រើបាស់បានលើការកំណត់ទៅលើចំណុច។ នៅពេលបានប្រើបាស់បានលើការកំណត់ទៅលើចំណុច។ នៅពេលបានប្រើបាស់បានលើការកំណត់ទៅលើចំណុច។

Stepper ដែលយើងដើរការប្រើបាស់មួយទៅ Model 28-BYJ48 Stepper motor។

ការរំភ្លោះ:



រូប:

```

1. #include <Stepper.h>
2. #define STEPS 32
3. Stepper stepper(STEPS, 8, 10, 9, 11);
4. int val = 0;
5. void setup() {
6.   Serial.begin(9600);
7.   stepper.setSpeed(200);
  
```

```

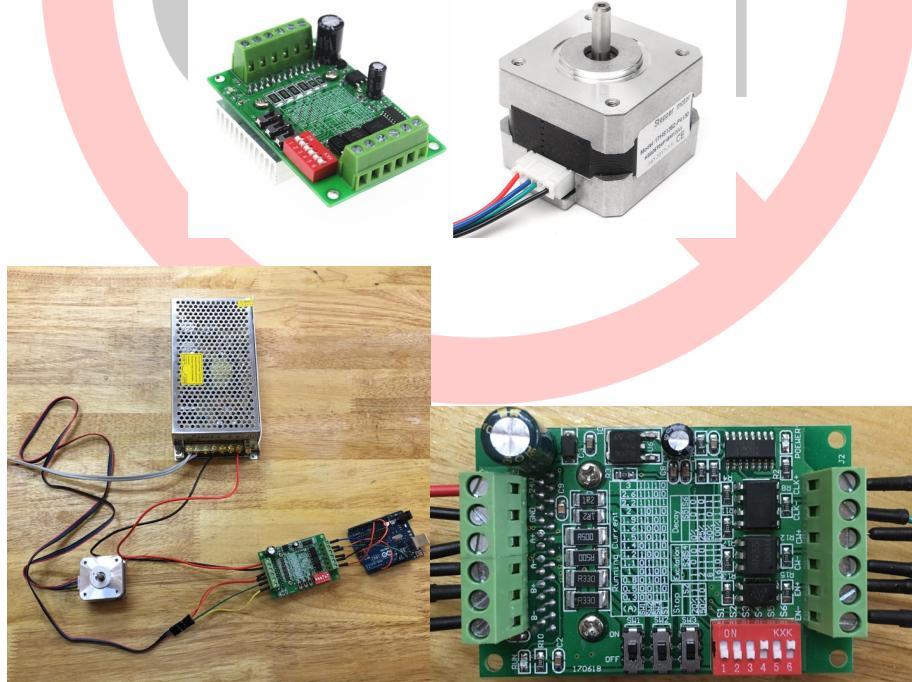
8. }
9. void loop() {
10. if (Serial.available()>0)
11. {
12. val = Serial.parseInt();
13. stepper.step(val);
14. Serial.println(val);
15. }
16.}

```

Note : Speed សម្រាប់ Stepper Motor ប្រភេទនេះបានដែល 1200

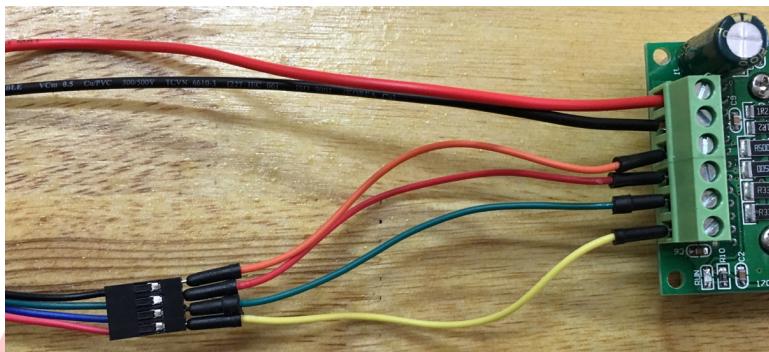
- Speed កាន់តុកដំ Torque ចូលរកាន់ត្រឡប់
- Speed កាន់តុកពិរិប Torque ចូលរកាន់ត្រឡប់

ការសិក្សា Stepper Motor ជាមួយ Driver ដែលគោរពមកធ្វើជាមួយ CNC Machine ឬ 3D printer



ការអភិវឌ្ឍ:

- Driver with Motor
 - Connect 24V to 12V Power Supply
 - Connect Ground to Ground Power Supply



- Driver with Controller
 - Connect CLK- CW- EN- to GND Arduino
 - Connect CLK+ to Pin5 Arduino
 - Connect CW+ to Pin2 Arduino
 - Connect EN+ to Pin8 Arduino

រូប:

```

1. const int stepPin = 5;
2. const int dirPin = 2;
3. const int enPin = 8;
4. void setup() {
5.   pinMode(stepPin,OUTPUT);
6.   pinMode(dirPin,OUTPUT);
7.
8.   pinMode(enPin,OUTPUT);
9.   digitalWrite(enPin,LOW);
10.
11. }
12.void loop() {
13.
14.   digitalWrite(dirPin,HIGH);
15.   for(int x = 0; x < 800; x++) {
16.     digitalWrite(stepPin,HIGH);
17.     delayMicroseconds(500);

```

```

18. digitalWrite(stepPin,LOW);
19. delayMicroseconds(500);
20. }
21. delay(1000);
22.
23. digitalWrite(dirPin,LOW);
24. for(int x = 0; x < 800; x++) {
25.   digitalWrite(stepPin,HIGH);
26.   delayMicroseconds(500);
27.   digitalWrite(stepPin,LOW);
28.   delayMicroseconds(500);
29. }
30. delay(1000);
31.
32.}

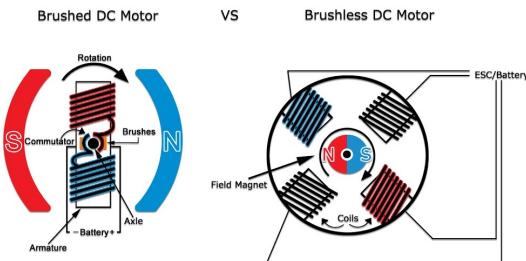
```

៤.៣ Brushless Motor with ESC

Brushless Motor ជាប្រភេទមួកមួយដែលមានខ្លួនត្រូវការងារឡើង វគ្គការ សម្រាប់មួកទៅ DC ឬមួក Servo និង Stepper Motor ពេលដែលរាយការការពិន្ទុ បង្កើតស្តីពី Brushless Motor ប្រភេទនេះ ពេលដែលរាយការការពិន្ទុការបង្កើតសំបកខាងក្រោមនៅទីផ្សារ។



កាលខ្លួនត្រូវការងារដែលការរាយការ DC Motor និង Brushless Motor



សម្រាប់ការបញ្ចូមទីរំបែនេះយើងត្រូវមាន Driver ឬយសម្រាប់ប្រើប្រាស់តាមយកនៃក្នុងការបញ្ចូលដឹកនាំរបស់មួយទៅ ដែលមានលក្ខណៈថា Electronic Speed Control (ESC) ។ នៅក្នុងការប្រើប្រាស់នេះ ESC មកប្រើប្រាស់យើងត្រូវដឹងទៀតបីមាណាពាយខ្លួនអគ្គិសនីដែល Brushless Motor ត្រូវការ ។ នៅក្នុងការសិក្សាយើងតាមរបៀបនេះអគ្គិសនីប្រមាណ 30A ។



ការគ្រប់:

- នៅក្នុងការគ្រប់អ្នកត្រូវព័ត៌មាន Power Supply 12V ទៅកាន់ ESC (Red = 12V & Black = GND)
- ព័ត៌មានចុះលក្ខណៈរបស់ ESC ទៅកាន់ខ្លួនបីរបស់មួយទៅ (Phase)
- ចំពោះខ្សោយ Control របស់ ESC
 - ខ្សោយណកក្នុងក្នុប់ទៅកាន់ GND របស់ Arduino
 - ខ្សោយណកក្នុងក្នុប់ទៅកាន់ 5V របស់ Arduino
 - ខ្សោយណកលើក្នុប់ទៅកាន់ Pin 8 របស់ Arduino

រូប:

1. #include <Servo.h>
- 2.
3. Servo esc;

```
4. void setup()
5.
6. {
7. esc.attach(8);
8. esc.writeMicroseconds(1000);
9. Serial.begin(9600);
10.Serial.setTimeout(100);
11.
12.}
13.void loop()
14.{
15.
16. if(Serial.available()>0)
17. {
18. String txt = Serial.readString();
19. int data = txt.toInt();
20. Serial.println(data);
21. esc.writeMicroseconds(data);
22.
23. }
24.}
```

អនុវត្តន៍ : ប្រើប្រាស់ Joystick ធ្វើកំណត់លេខូនរបស់ Brushless Motor ។

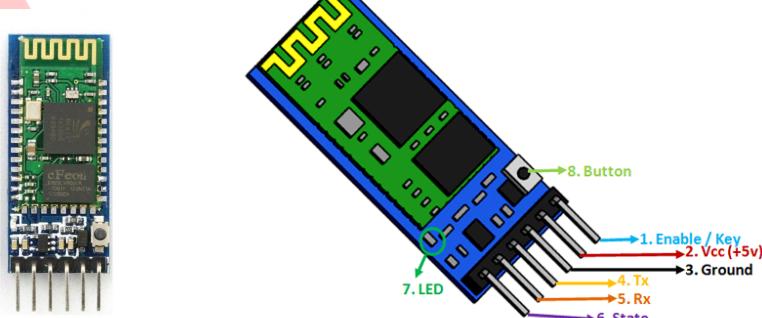
ចំណុះ នៃ ការសិក្សាថ្មីប្រព័ន្ធបញ្ជាក់ខ្លួន

៤.៩ Bluetooth HC-05 with App

Bluetooth HC-05 ជាប្រភេទប្រព័ន្ធអេឡិចត្រូនិក Low Energy មួយដែលយើងអាចប្រើប្រាស់ជាការទំនាក់ទំនងនៃការខ្សោយបានដើម្បី Microcontroller មួយគ្នានៅក្នុង Microcontroller មួយទៀត ឬ Microcontroller ទៅកាន់ Smart Phone និងទៅកាន់ Computer ក៏ពីរ។ Bluetooth HC-05 ជាប្រភេទ wireless Communication Device ដែលធ្វើការបញ្ចូល ប្លងទូលទិន្នន័យនៃក្នុង Protocol UART។នៅក្នុងការសិក្សា យើងនឹងលើកកម្មការបញ្ចាត់ Application នៃក្នុង Smart Phone Android ទៅកាន់ Microcontroller ដែលយើងនឹងធ្វើការបញ្ចាត់ RC car kit មួយ។

- ចូរបញ្ចាក់ដោយ** ចំពោះប្រភេទ Bluetooth Low Energy ដែលនេះវាតិតាតិបាកដីណើការជាមួយនឹងប្រព័ន្ធប្រតិបត្តិការ IOS (iPhone or MAC) ឬ ម៉ាស៊ីនឯឈាយទៀតចំពោះ Application មួយចំនួនដែលអាចប្រើប្រាស់ជាមួយ Bluetooth HC-05 ក៏ដូចជា Arduino មិនសូវទៅនៅលើ App Store នៅទៅ ដូចដែលបានរក្សាទុកការពេលបញ្ចាផ្ទៃចំណុចនេះអ្នកត្រូវមាន Smart Phone ដែលមានប្រព័ន្ធប្រតិបត្តិការ Android ។

ការរក្សាប់



Specification :

1	Enable Key	/ This pin is used to toggle between Data Mode (set low) and AT command mode (set high). By default it is in Data mode
2	Vcc	Powers the module. Connect to +5V Supply voltage
3	Ground	Ground pin of module, connect to system ground.
4	TX Transmitter	- Transmits Serial Data. Everything received via Bluetooth will be given out by this pin as serial data.
5	RX Receiver	- Receive Serial Data. Every serial data given to this pin will be broadcasted via Bluetooth
6	State	The state pin is connected to on board LED, it can be used as a feedback to check if Bluetooth is working properly.
7	LED	Indicates the status of Module <ul style="list-style-type: none"> • Blink once in 2 sec: Module has entered Command Mode • Repeated Blinking: Waiting for connection in Data Mode • Blink twice in 1 sec: Connection successful in Data Mode
8	Button	Used to control the Key/Enable pin to toggle between Data and command Mode

Default Setting របស់ Bluetooth HC-05 :

- Bluetooth Name : HC-05
- Password : 1234 or 0000
- Communication : Slave
- Mode : Data Mode
- Mode Baud Rate : 9600,8,N,1
- Command Mode Baud Rate : 38400,8,N,1
- Firmware : LINVOR

ការកែត្រាប័ណ្ណ Bluetooth :

- Connect VCC to 5V Arduino
- Connect GND to GND Arduino
- Connect RX to Tx Arduino
- Connect Tx to Rx Arduino

បើសិនជាអ្នកប្រើប្រាស់ដឹងទី ០ និង ១ របស់ Arduino ពេល Upload Program អ្នកត្រូវដើរនឹង RX Arduino ចេញបៀវក្ខអុកដើម្បីចាំងរបស់ Arduino ត្រូវទៅ Device ណាមួយអ្នកមិនអាច Upload Program ចូលទៅក្នុង Arduino Development Board បាននោះទេ ។

ក្នុង (នៅពេលដែលអ្នក Upload Program ត្រូវបានបញ្ជូនទៅកាមពីរបាលខាងក្រោមជាមួយ)

```

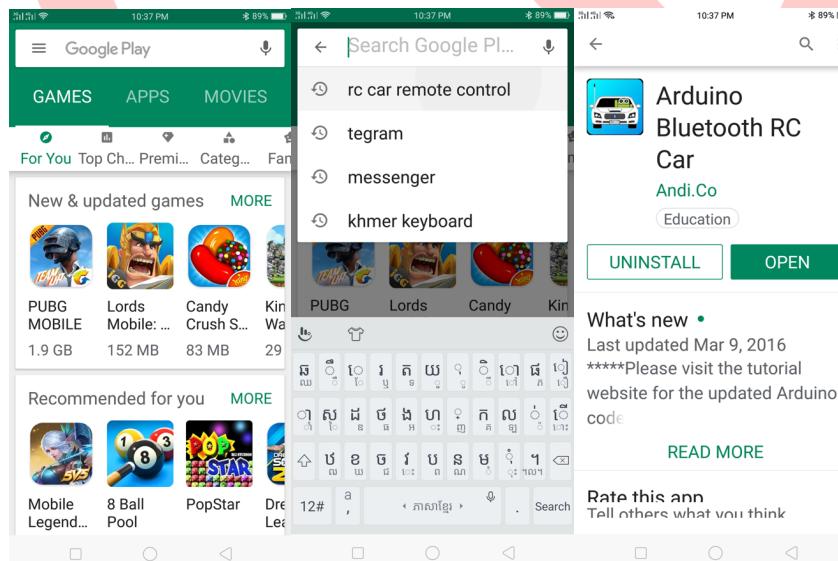
1. char command =0;
2. void setup() {
3.
4. Serial.begin(9600);
5. }
6.
7. void loop() {
8. if (Serial.available() > 0)
9. {
10. command = Serial.read();
11. }
12.}

```

នៅក្នុងការសិក្សានេះយើងនឹងដោនាំពីការធ្វើនៅលម្អិតបច្ចុប្បន្នដែលបានបង្កើតឡើង នៅក្នុង Smart Phone Android ដើម្បីលើកការណ៍ដៃបច្ចុប្បន្ននៃការប្រើប្រាស់ Arduino Bluetooth RC Car។

ជំហតទី 1

ចូលទៅកាន់ Play Store - Arduino Bluetooth RC Car -Install



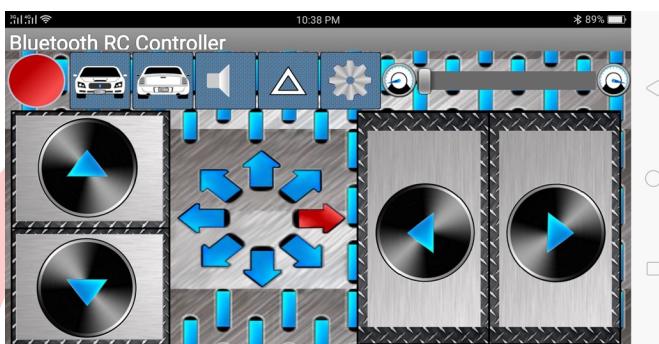
ជំហានទី 2

ពេលដែល Install Application នៃយុទ្ធសាស្ត្រការណ៍ Bluetooth Setting របស់អ្នកដើម្បីធ្វើការពត្យាប់ជាមួយ Bluetooth និងអ្នកជាមុនសិន។

Name : HC-05

Password : 1234 or 0000

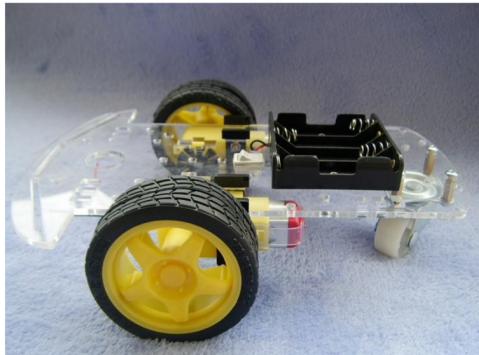
ជំហានទី 3 ចូចចូលទៅកាន់ Application ដើម្បីបញ្ជាផ្ទាល់ការពារឡើង និងចូចចូលទៅកាន់ Setting របស់ App ហើយ ចូចលើកក្បែរ Connect to Car



នៅពេលរួមចូលពណ៌ក្របេម ត្រូវស្វែងរកទិន្នន័យថាការពត្យាប់ពី Bluetooth ទៅកាន់ Application របស់អ្នកទទួលបានព័ត៌មានថាគារពារឡើង បន្ទាប់មកទៀតអ្នកត្រូវបើក Serial Monitor ហើយចូចបូកនឹងនៅលើ Application ដើម្បីមើលទិន្នន័យដើម្បីការទាញយកទិន្នន័យ។

គូរបញ្ជាក់នូវដំណឹង : តាមការពារស្ថិតិយវិធីរបស់អ្នក ចំពោះបូកនឹងខ្លួន ឬនិងភាពជំណើរការបានទេ រាយការណ៍មែនជាបញ្ហា Connection របស់អ្នកនៅទៅទៅ តែវាបាបញ្ហារាយការណ៍ Application និង Version របស់ Smart Phone របស់អ្នក។ បុំផ្តុំរាយការណ៍ទៅលើបូកនឹងដែលមិនស្ថិតិយវិធីរបស់អ្នក ឬបុំផ្តុំរាយការណ៍ទៅលើបូកនឹងដែលមិនស្ថិតិយវិធីរបស់អ្នក។

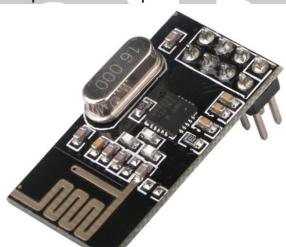
អនុវត្តន៍ : បើប្រាស់ Application ដើម្បីការបញ្ជាក់ទិន្នន័យ។



៤.២ National Retail Federation (NRF)

NRF ជាប្រភេទ Module សម្រាប់ធ្វើការទំនាក់ទំនងពីទីតាំងមួយទៅការទីតាំងមួយផ្លូវ ដោយធ្វើការទំនាក់ទំនងត្រាគារយកតែខ្សោយ (Wireless Communication) ប្រើប្រាស់ការបញ្ជីន Signal ដោយប្រើប្រាស់ Frequency។ គេនិយមយកវាទៅប្រើប្រាស់ជាមួយការបញ្ហានឹងច្បាយដូចជា ការបញ្ហានទិន្នន័យឬ Station ឬយោង Station ឬយ, Smart Home Automation ឬបុត្រិ RC Car, RC Boat, RC Plan និង Drone ជាដើម។

នៅក្នុងសិក្សាយើងបានរឿនសិស្សនៃ NRF Module ឬយដែលមានលេខា៖ថា NRF24L01 ជាប្រភេទ Module ឬយដែលពេញនិយមបំផុតនៅក្នុងប្រពេទសកម្មជាតុ។

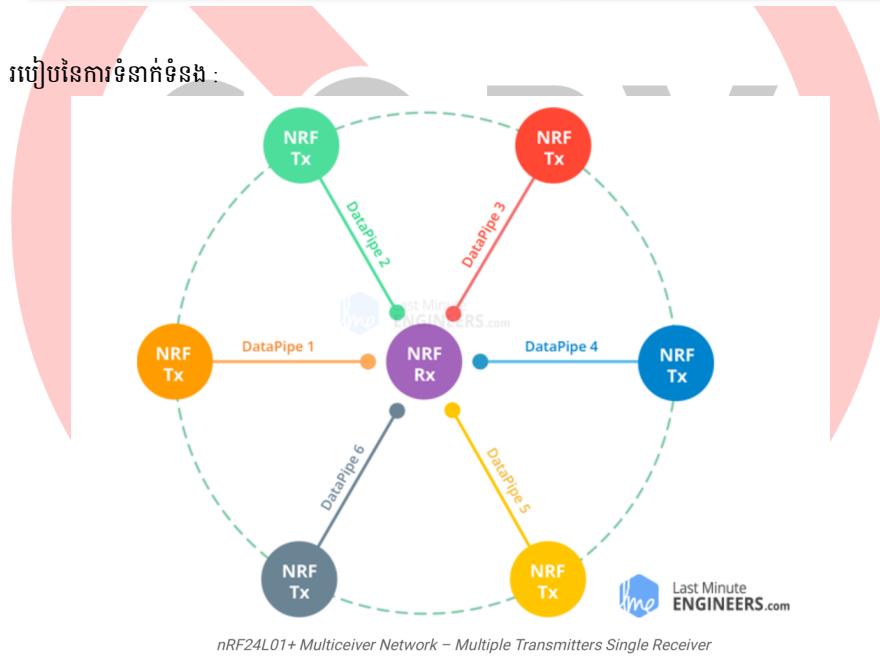


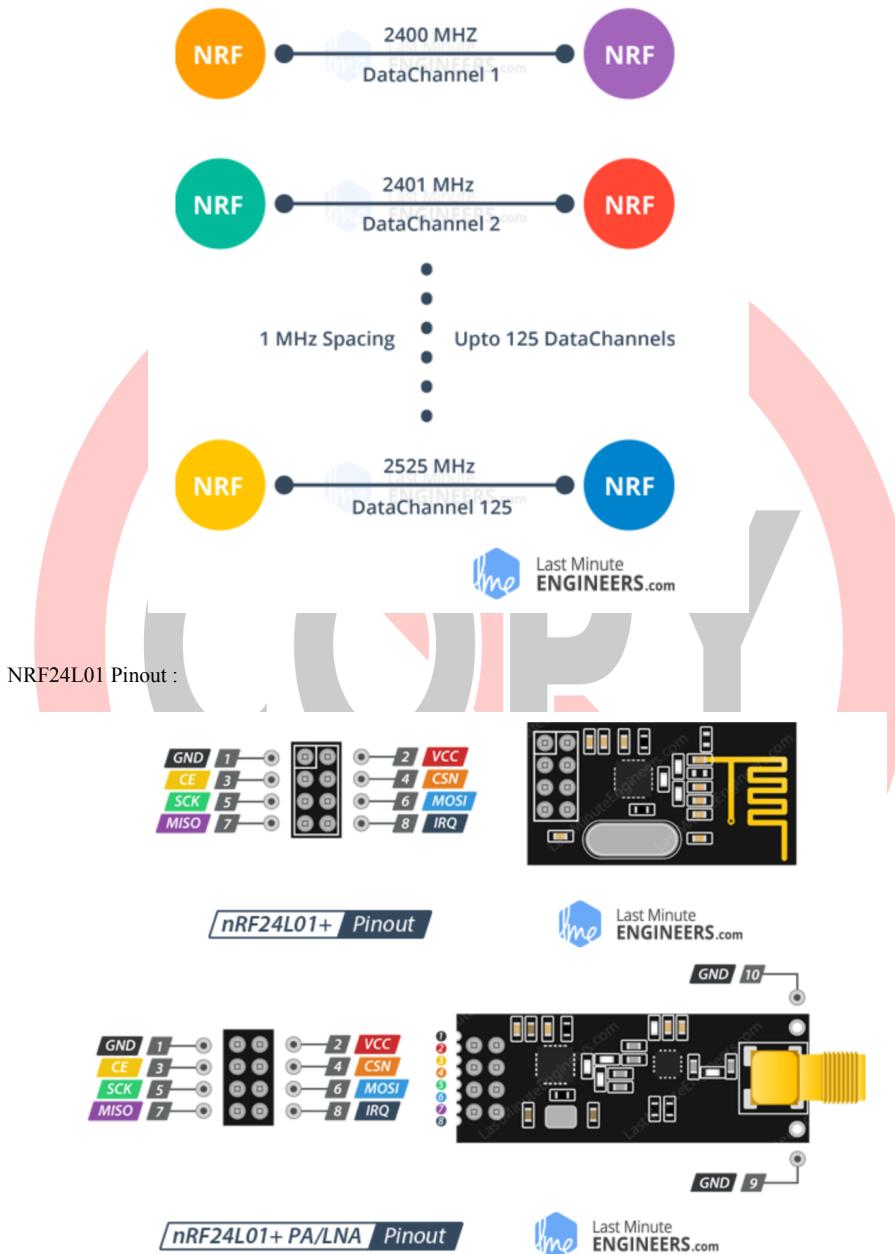
សម្រាប់អ្នកដែរចង់ឈ្មោះយល់បាននូវមិន NRF24L01 ប្រភេទនេះឡើងបាននូវការចូលចិត្តលើមិន Link ខាងក្រោមនេះបាន : <https://lastminuteengineers.com/datasheets/nrf24l01+2.4ghz-transceiver-datasheet.pdf>

នៅក្នុងការប្រើប្រាស់ Module ប្រភេទអ្នកត្រូវទាញយក Library ឬយមកប្រើប្រាស់តាមរយៈ Link ខាងក្រោមនេះ : <https://github.com/nRF24/RF24>

ណត់មានលម្អិតសំខាន់ៗ ឬយចំនួននឹង NRF24L01

Frequency Range	2.4 GHz ISM Band
Maximum Air Data Rate	2 Mb/s
Modulation Format	GFSK
Max. Output Power	0 dBm
Operating Supply Voltage	1.9 V to 3.6 V
Max. Operating Current	13.5mA
Min. Current(Standby Mode)	26µA
Logic Inputs	5V Tolerant
Communication Range	800+ m (line of sight)





ការអភិវឌ្ឍ:

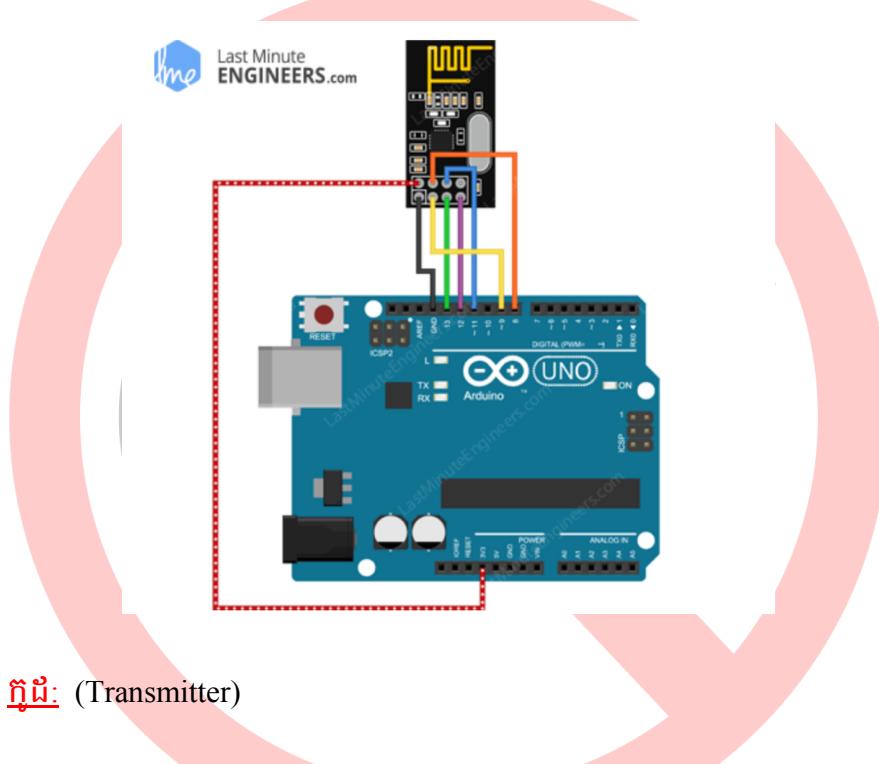
ការសិក្សាលើប្រព័ន្ធបញ្ជាក់ខ្សោយ

113

ផ្ទៃបង្កើងដោយ ហេម សុខិត្តិ

នៅក្នុងការគ្រប់ Module NRF មើលបានថែងទៅកាតិរតី Transmitter and Receiver ។ ស្ថិសសិសដើរ របស់អ្នកម្នាក់ ដើម្បីក្រោរការពារនា Transmitter អ្នកណាតា Receiver ។ សម្រាប់ការគ្រប់ជាមួយ Arduino ទាំង Transmitter និង Receiver គឺការគ្រប់ដូចត្រា។

ការគ្រប់ NRF24L01 ទាំង Transmitter និង Receiver :



ភ្លើង: (Transmitter)

```

1. //Include Libraries
2. #include <SPI.h>
3. #include <nRF24L01.h>
4. #include <RF24.h>
5.
6. //create an RF24 object
7. RF24 radio(9, 8); // CE, CSN
8.
9. //address through which two modules communicate.
10. const byte address[6] = "00001";
11.
12. void setup()

```

```

13.{  

14. radio.begin();  

15.  

16. //set the address  

17. radio.openWritingPipe(address);  

18.  

19. //Set module as transmitter  

20. radio.stopListening();  

21.  

22.void loop()  

23.{  

24. //Send message to receiver  

25. const char text[] = "Hello World";  

26. radio.write(&text, sizeof(text));  

27.  

28. delay(1000);  

29.}

```

ផ្តាំង: (Receiver)



```

1. //Include Libraries  

2. #include <SPI.h>  

3. #include <nRF24L01.h>  

4. #include <RF24.h>  

5.  

6. //create an RF24 object  

7. RF24 radio(9, 8); // CE, CSN  

8.  

9. //address through which two modules communicate.  

10.const byte address[6] = "00001";  

11.  

12.void setup()  

13.{  

14. while (!Serial);  

15. Serial.begin(9600);  

16.  

17. radio.begin();  

18.  

19. //set the address  

20. radio.openReadingPipe(0, address);

```

```

21.
22. //Set module as receiver
23. radio.startListening();
24.}
25.
26.void loop()
27.{  

28. //Read the data if available in buffer
29. if (radio.available())
30. {
31.   char text[32] = {0};
32.   radio.read(&text, sizeof(text));
33.   Serial.println(text);
34. }
35.}
```

ចំណាំ: Address : រាជធានីភ្នំពេញ ខេត្តភ្នំពេញ លីមូកស្រីស៊ូវ បានចំនួន 99999 ។ បែន្ទិនជាមួយកំណត់ Address នៃសំពាល់ Transmitter 00005 អ្នកប្រើប្រាស់យុទ្ធសាស្ត្រ និងខ្លួនគ្នា និង Receiver មក 00005 វិញ។

ក្នុងខាងក្រោមនេះគឺជាបញ្ជាក់ថា យើងបានប្រើប្រាស់ជាមួយ Joystick :

```

1. //Include Libraries
2. #include <SPI.h>
3. #include <nRF24L01.h>
4. #include <RF24.h>
5.
6. //create an RF24 object
7. RF24 radio(9, 8); // CE, CSN
8.
9. //address through which two modules communicate.
10. const byte address[6] = "00001";
11. int data =0;
12.
13.void setup()
14.{
15.   radio.begin();
16. //set the address
17.   radio.openWritingPipe(address);
18.   radio.setPALevel(RF24_PA_MIN);
```

```

19.
20. //Set module as transmitter
21. radio.stopListening();
22.}
23.void loop()
24.{
25.// radio.stopListening();
26. int potValue = analogRead(A0);
27. int angleValue = map(potValue, 0, 1023, 0, 255);
28. radio.write(&angleValue, sizeof(angleValue));
29.

```

ក្នុងខាងក្រោម Receiver នៅក្នុងការរបីត្រាស់ជីថយ Joystick :



```

1. //Include Libraries
2. #include <SPI.h>
3. #include <nRF24L01.h>
4. #include <RF24.h>
5.
6. //create an RF24 object
7. RF24 radio(9, 8); // CE, CSN
8.
9. //address through which two modules communicate.
10. const byte address[6] = "00001";
11.
12.
13.void setup()
14.{
15. while (!Serial);
16. Serial.begin(9600);
17. Serial.setTimeout(100);
18.
19. radio.begin();
20.
21. //set the address
22. radio.openReadingPipe(0, address);
23. radio.setPALevel(RF24_PA_MIN);
24.
25. //Set module as receiver
26. radio.startListening();
27.}

```

```

28.
29.void loop()
30.{
31.    radio.startListening();
32.    if (radio.available())
33.    {
34.        while (radio.available())
35.        {
36.            int data = 0;
37.            radio.read(&data, sizeof(data));
38.            Serial.println(data);
39.        }
40.    }
41.}

```

អនុវត្ត 1: ប្រើប្រាស់ Joystick ដើម្បីបញ្ចូលរូបស់ Brushless Motor

អនុវត្ត 2: ប្រើប្រាស់ Joystick ដើម្បីបញ្ចូនទ្វានបញ្ហា

៦.៣ Radio Frequency Identification

RFID ជាប្រព័ន្ធដែលមានរយៈពេលខ្លួនដែលត្រូវបានបញ្ជាក់ឡើងដោយប្រើប្រាស់ប្រព័ន្ធអេក្រិចប្រើប្រាស់ប្រព័ន្ធដែលមានរយៈពេលខ្លួនដែលត្រូវបានបញ្ជាក់ឡើង។ នៅក្នុងប្រព័ន្ធបានបញ្ជាក់ឡើងនៅក្នុងប្រព័ន្ធដែលមានរយៈពេលខ្លួនដែលត្រូវបានបញ្ជាក់ឡើង។

RFID ប្រើប្រាស់ដើម្បីបញ្ជាក់ឡើងដោយប្រើប្រាស់ប្រព័ន្ធដែលមានរយៈពេលខ្លួនដែលត្រូវបានបញ្ជាក់ឡើង។ ជាពេលវេលាយករាយការណ៍នៃការបញ្ជាក់ឡើងនៅក្នុងប្រព័ន្ធដែលមានរយៈពេលខ្លួនដែលត្រូវបានបញ្ជាក់ឡើង។

- Tag : ភាពជាប្រភេទ Card ឬ Keychain ជាន់ដែលនៅក្នុងប្រព័ន្ធដែលមានរយៈពេលខ្លួនដែលត្រូវបានបញ្ជាក់ឡើង។
- Reader : ជាប្រព័ន្ធដែលមានរយៈពេលខ្លួនដែលត្រូវបានបញ្ជាក់ឡើង។



Special Specification :

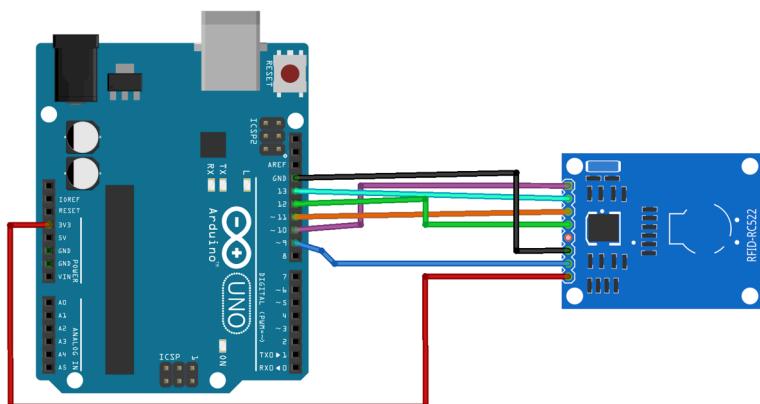
- Input Voltage : 3.3V
- Frequency : 13.56MHz

នៅក្នុងប្រព័ន្ធឌីជីថល យើងត្រូវការទាញយក Library មួយយកមកប្រព័ន្ធតែមទៀត: Link ខាងក្រោម:
<https://codeload.github.com/miguelbalboa/rfid/zip/master>

ការរំភោះ:

Pin	Wiring to Arduino Uno
SDA	Digital 10
SCK	Digital 13
MOSI	Digital 11
MISO	Digital 12
IRQ	unconnected
GND	GND
RST	Digital 9
3.3V	3.3V

Caution: You must power this device to 3.3V!



ក្រុង:

COPY

```

1. #include <SPI.h>
2. #include <MFRC522.h>
3.
4. #define SS_PIN 10
5. #define RST_PIN 9
6. MFRC522 mfrc522(SS_PIN, RST_PIN);
7. void setup()
8. {
9.   Serial.begin(9600);
10.  SPI.begin();
11.  mfrc522.PCD_Init();
12.  Serial.println("Approximate your card to the reader...");
13.  Serial.println();
14.
15. }
16. void loop()
17. {
18. // Look for new cards
19. if ( ! mfrc522.PICC_IsNewCardPresent())
20. {
21.   return;
22. }
23. // Select one of the cards
24. if ( ! mfrc522.PICC_ReadCardSerial())

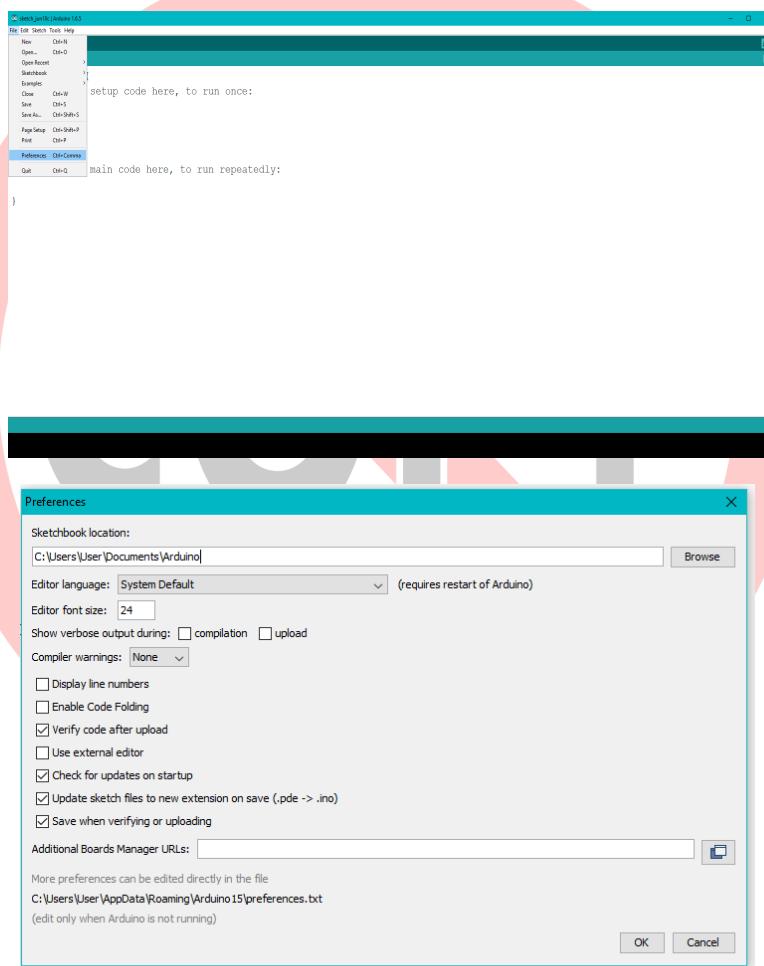
```

```
25. {
26.     return;
27. }
28. //Show UID on serial monitor
29. Serial.print("UID tag :");
30. String content= "";
31. byte letter;
32. for (byte i = 0; i < mfrc522.uid.size; i++)
33. {
34.     Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " ");
35.     Serial.print(mfrc522.uid.uidByte[i], HEX);
36.     content.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " "));
37.     content.concat(String(mfrc522.uid.uidByte[i], HEX));
38. }
39. Serial.println();
40. Serial.print("Message :");
41. content.toUpperCase();
42. if (content.substring(1) == "90 50 F7 A3") //change here the UID of the
   card/cards that you want to give access
43. {
44.     Serial.println("Authorized access");
45.     Serial.println();
46.     delay(3000);
47. }
48.
49. else {
50.     Serial.println(" Access denied");
51.     delay(3000);
52. }
53.}
```

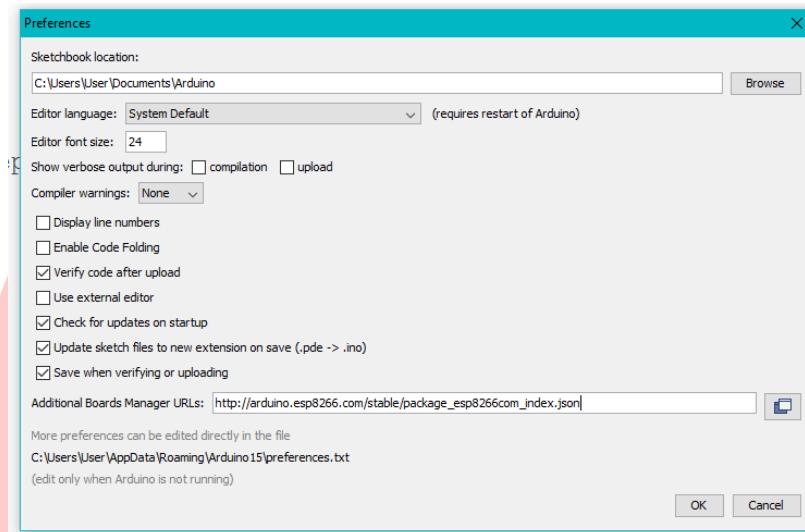
៦.៤ ESP8266 (WIFI)

មុននេះចូលដល់ការទន្លេពី ESP8266 អ្នកត្រូវធ្វើការតាមរបៀប Software មួយចំនួនតាមដំណាក់កាល ខាងក្រោមជាមុនសិន :

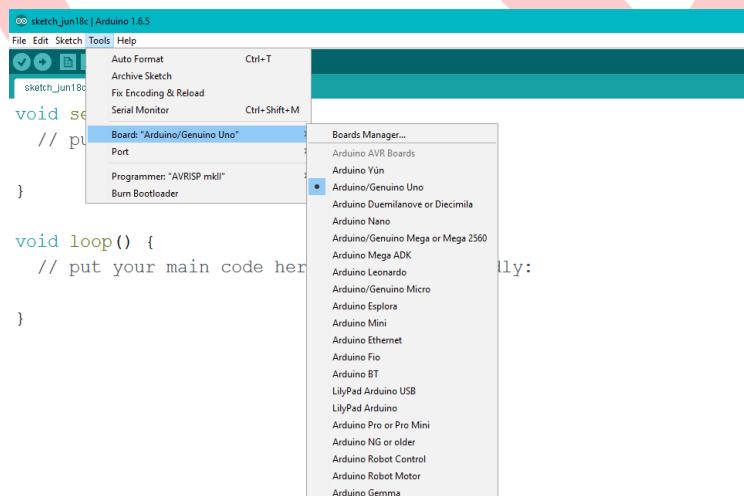
- ជំហាន 1 : ចូលទៅកាន់ Arduino IDE របស់អ្នកចូលទៅកាន់ File > Preference នៅខាងក្រោមនេះ យើងត្រូវចូលដល់ការទន្លេរបស់ក្រោម

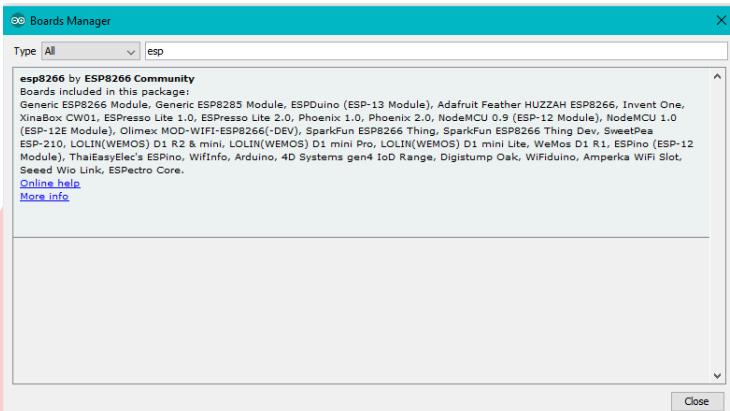


- ជំហាន 2 : https://arduino.esp8266.com/stable/package_esp8266com_index.json Copy Link នៃជាក់បញ្ចូលដូច្បែបខាងក្រោម ហើយ Click OK

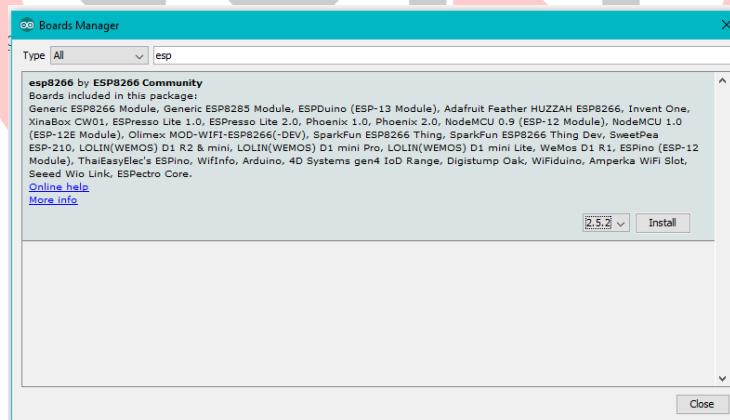


រូបភាព ៣ : ចូលទៅកាន់ Tool > Board > Boards Manager

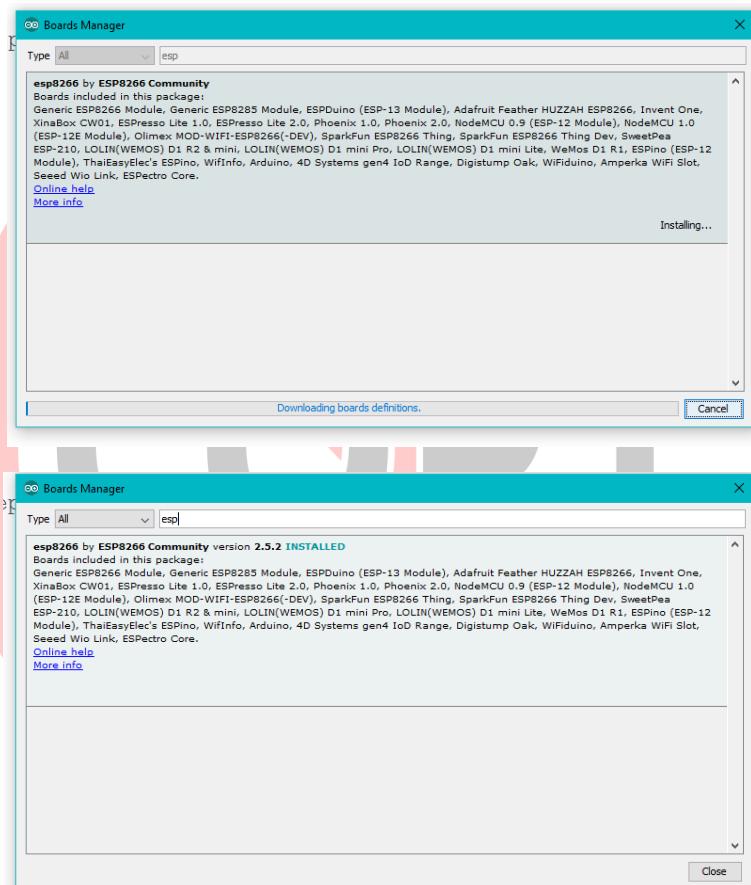




- ដំបាន 4 : កែតម្រូវក្នុងក្រុមហ៊ុន Board មួយដែលមាននេះថា ESP(សម្រាប់ Window) Node (សម្រាប់ Mac OSX)



- ដំបាន 5 : ចូល Install ផ្តើមទៅពាណិជ្ជកម្មរបស់ Board ពាណិជ្ជកម្មដោយធ្វើតីនៅក្នុងក្រុមហ៊ុនបន្ថែម



- ផែនទាន 6 : <https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers> ចុចនៅលើ Link នេះដើម្បីចូលទៅទាយយក Driver របស់ ESP8266

Automation Cambodia

Arduino Programming Class

<https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers>
APPLICATION NOTE Software

Legacy OS software and driver package download links and support information >

Download for Windows 10 Universal (v10.1.7)

Platform	Software	Release Notes
Windows 10 Universal	Download VCP (2.3 MB)	Download VCP Revision History

Download for Windows 7/8/8.1 (v6.7.6)

Platform	Software	Release Notes
Windows 7/8/8.1	Download VCP (5.3 MB) (Default)	Download VCP Revision History
Windows 7/8/8.1	Download VCP with Serial Enumeration (5.3 MB)	Download VCP Revision History

Download for Windows XP/Server 2003/Vista/7/8/8.1 (v6.7)

Platform	Software	Release Notes
Windows XP/Server 2003/Vista/7/8/8.1	Download VCP (3.66 MB)	Download VCP Revision History

Download for Windows 2K (v6.3a)



<https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers>
AN197: The Serial Communications Guide for the CP210x

Download Software

The CP210x Manufacturing DLL and Runtime DLL have been updated and must be used with v6.0 and later of the CP210x Windows VCP Driver. Application Note Software downloads affected are AN1445SW.zip, AN2055SW.zip and AN2235SW.zip. If you are using a 5.5 driver and need support, you can download archived Application Note Software.

Legacy OS software and driver

Download for Win

Platform	Software	Release Notes
Windows 10 Universal	Download VCP (2.3 MB)	Download VCP Revision History

Download for Windows 7/8/8.1 (v6.7.6)

Platform	Software	Release Notes
Windows 7/8/8.1	Download VCP (5.3 MB) (Default)	Download VCP Revision History
Windows 7/8/8.1	Download VCP with Serial Enumeration (5.3 MB)	Download VCP Revision History

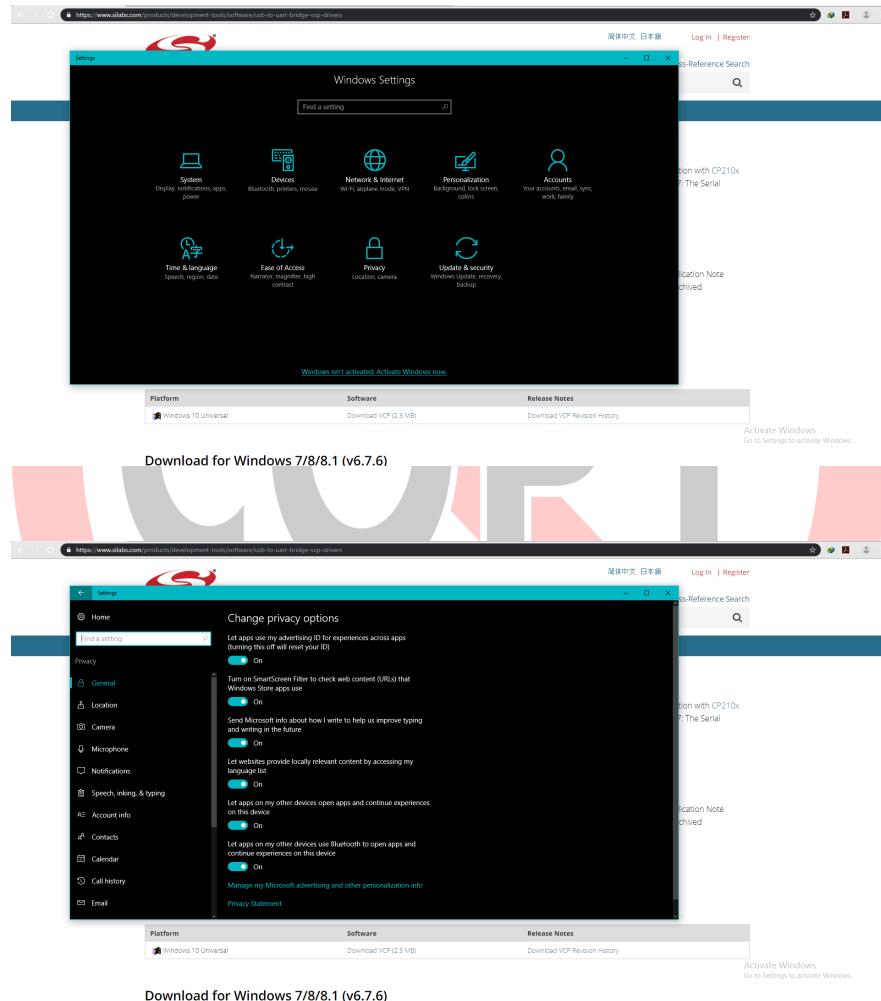
Download for Windows XP/Server 2003/Vista/7/8/8.1 (v6.7)

Platform	Software	Release Notes
Windows XP/Server 2003/Vista/7/8/8.1	Download VCP (3.66 MB)	Download VCP Revision History

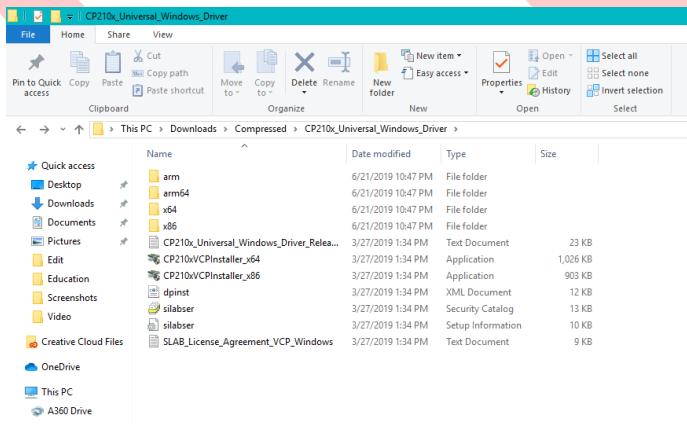
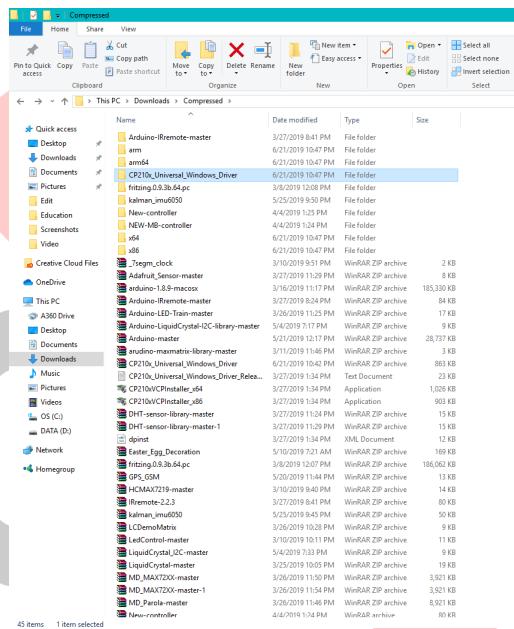
Automation Cambodia

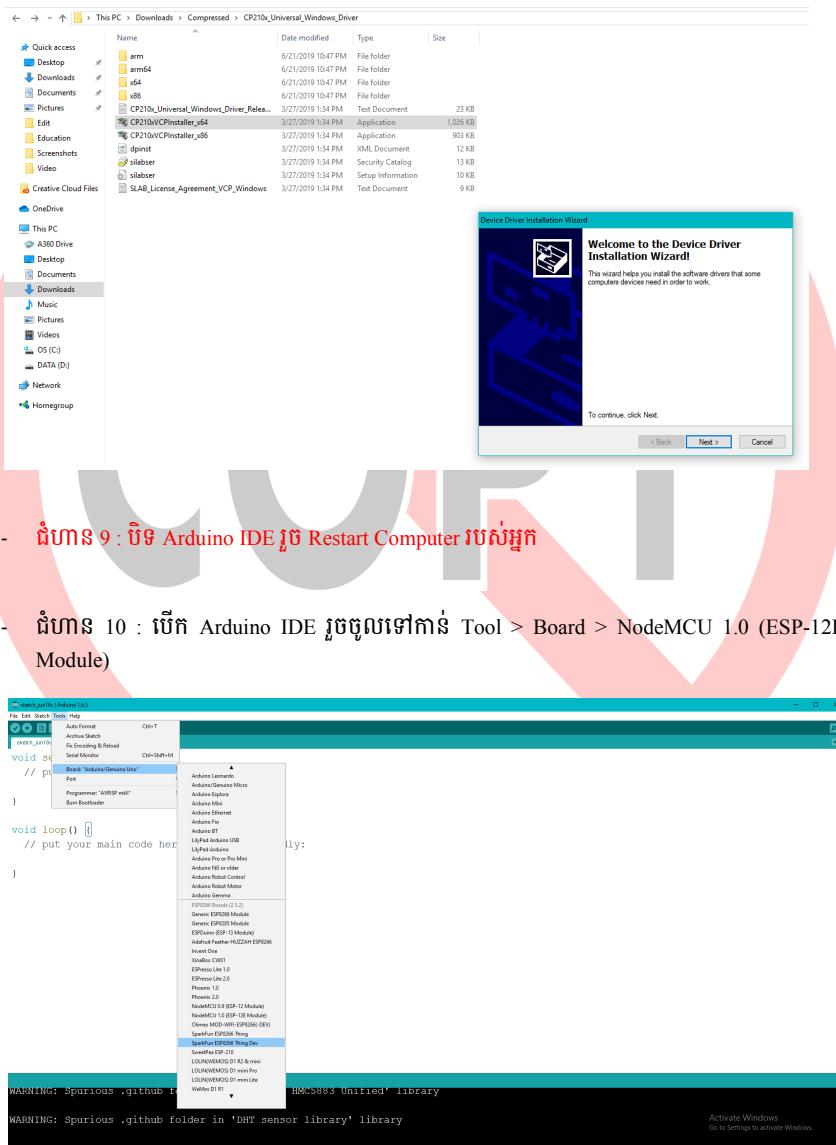
Arduino Programming Class

- ជំហាន 9 : ចូលទៅកាន់ Preference > Privacy > General អ្នកត្រូវ Allow ប្រកិត On រាល់មួយនាក់ដែលបានហាមយាត់ដោយ Window ទាំងអស់



- ជំហាន 8 : Extract File ដែលមានការ Download ឱច Install ចុចចូលទៅក្នុងដីម្បីធ្វើការ Install Driver (សម្រាប់ x64 ឬ x86 អាណ្យប់យេទ្ទិនី OS របស់ Laptop របស់អ្នកទាំងត្រូវបានប្រាស់)



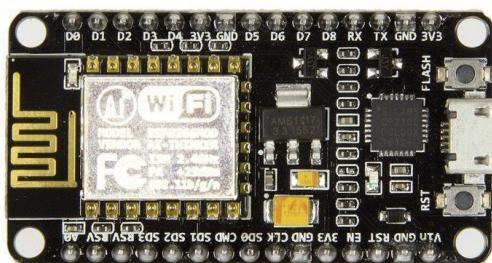


- ជំហាន 11 : ត្រូវ ESP8266 ចូលមកការកំណុចរបស់អ្នកដោយបើប្រពាស់ USB Android។ នឹងសិនជាអ្នកបានយើង Post របស់ ESP8266 ដែលបានគ្រប់មកកំណុចរបស់អ្នក នៅពាននៃយថាការ Install Software នៃសម្បាកទូទៅលបាតានដោតដីយ។



If you have something wrong with this install, please click link below for details and videos :
<https://www.youtube.com/watch?v=3gOKrMAz7WE&t=243s>

ESP8266 ជាប្រភេទ Development board WIFI មួយដែលមានព័ត៌មានសម្រាយ តែនិយមយកវាទៅបើប្រពាស់ នៅក្នុងបញ្ហាដោយបើប្រពាស់ Internet នៅក្នុងនឹងជានិម។



នៅក្នុងការបើប្រពាស់ WIFI Module បែបនេះ យើងខ្ញុំស្ម័ែលការណា App មួយដែលអាយស្រួលនៅក្នុងបើប្រពាស់ជាអ្នយ ESP8266 ដែលមានលេខាជាន Blynk ។

មុននឹងលាយនិងដល់ការ Install ទៅលើ App Blynk អ្នកត្រូវចូលទៅទាញរាយក Library មួយចំនួន សម្រាប់បើប្រពាស់ជាអ្នយ ESP8266 និង App Blynk ។ Link មាននៅខាងក្រោម :

<https://github.com/esp8266/Arduino> និង <https://github.com/blynkkk/blynk-library>

ចូល :

```

1. #define BLYNK_PRINT Serial
2.
3. #include <ESP8266WiFi.h>
4. #include <BlynkSimpleEsp8266.h>
5.
6. char auth[] = "YourAuthToken";
7.
8. char ssid[] = "YourNetworkName";
9. char pass[] = "YourPassword";
10.
11.void setup()
12.{  

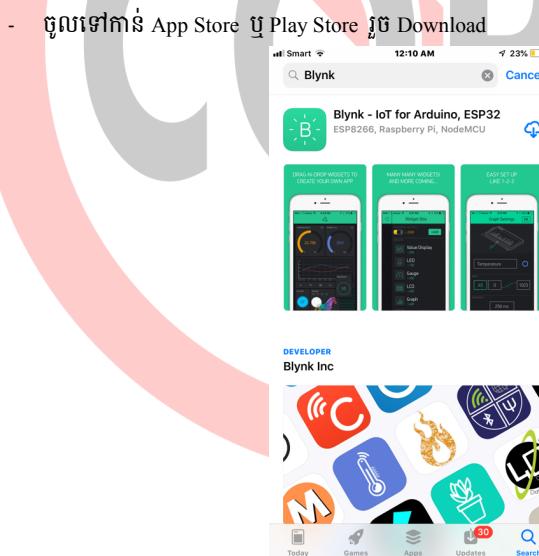
13. Serial.begin(9600);
14. Blynk.begin(auth, ssid, pass);
15.}  

16.
17.void loop()
18.{  

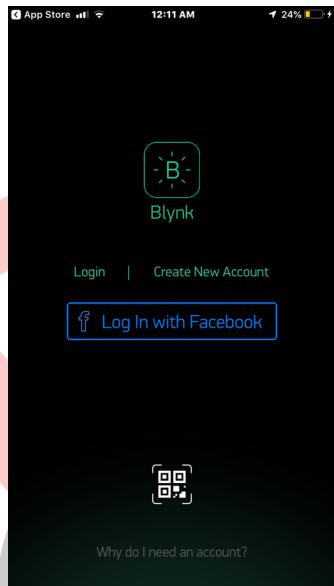
19. Blynk.run();
20.}

```

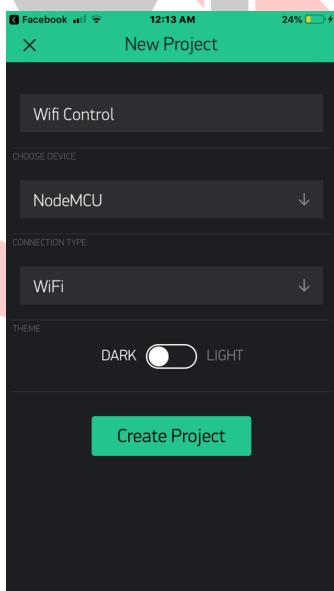
Copy Code ខាងលើទៅដំឡើង Arduino IDE ហើយអ្នក រួចអនុវត្តន៍តាមជំហានខាងក្រោម :



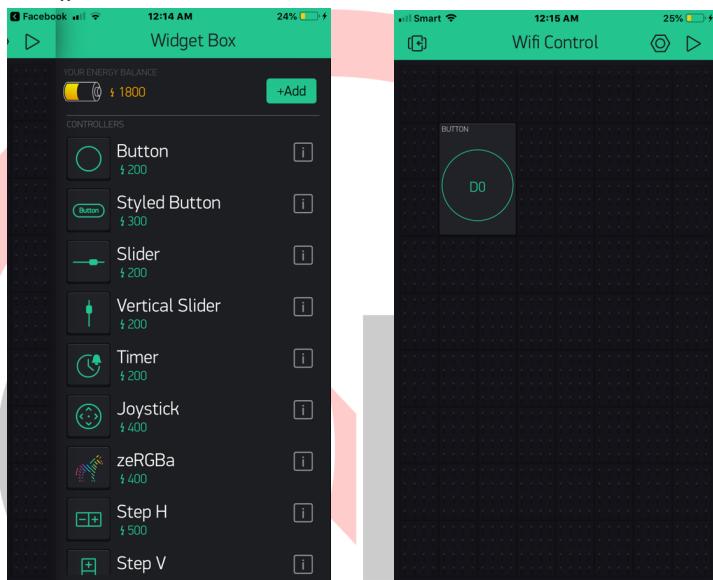
- ចូចចូលទៅកាន់ Blynk របស់ Register តាមដំឡើសម្បយណាក៏ពាន



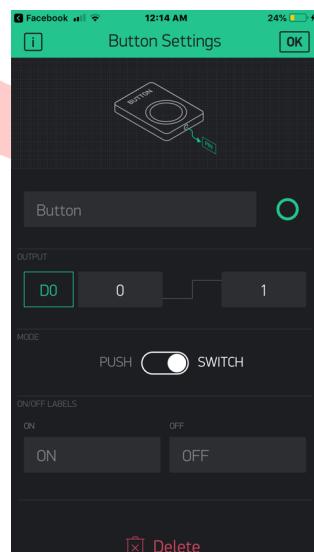
- ចូចចូលទៅកាន់ Blynk របស់ប្រើប្រាស់ដឹងថ្លែបខាងក្រោម :



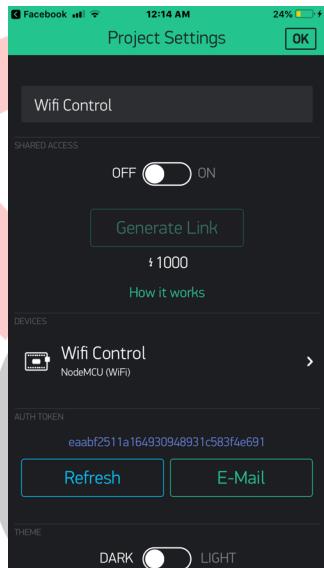
- Slide ពីខាងស្តាំកដ្ឋានអ្នកនឹងយើលដ្ឋានដូចខាងក្រោម ចុចលើ Button ដំបូងគេមួយសម្រាប់ធ្វើការតែស្ថាកល្បង



- ចុចទៅលើ Button ដើម្បីការណែនាំ រួចសាកកំណត់ដូចខាងក្រោមសម្រាប់ធ្វើការតែស្ថាកល្បង



- បន្ទាប់មកទៅតួចចូលទៅកាន់ Setting នូវ Copy Token ដើម្បីយកមកជាកំបញ្ឈូលនៅក្នុង Arduino IDE

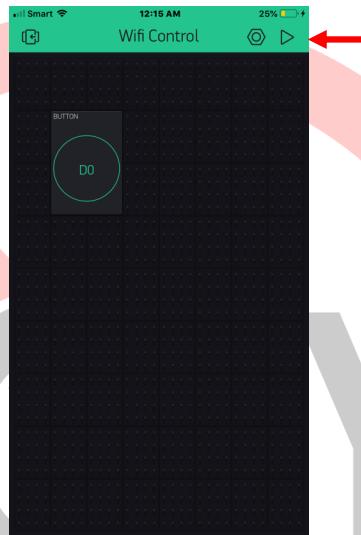


- ត្រួលប័ណ្ណការនៃ Arduino IDE ទិញ

```
char auth[] = "YourAuthToken";
char ssid[] = "YourNetworkName";
char pass[] = "YourPassword";
```

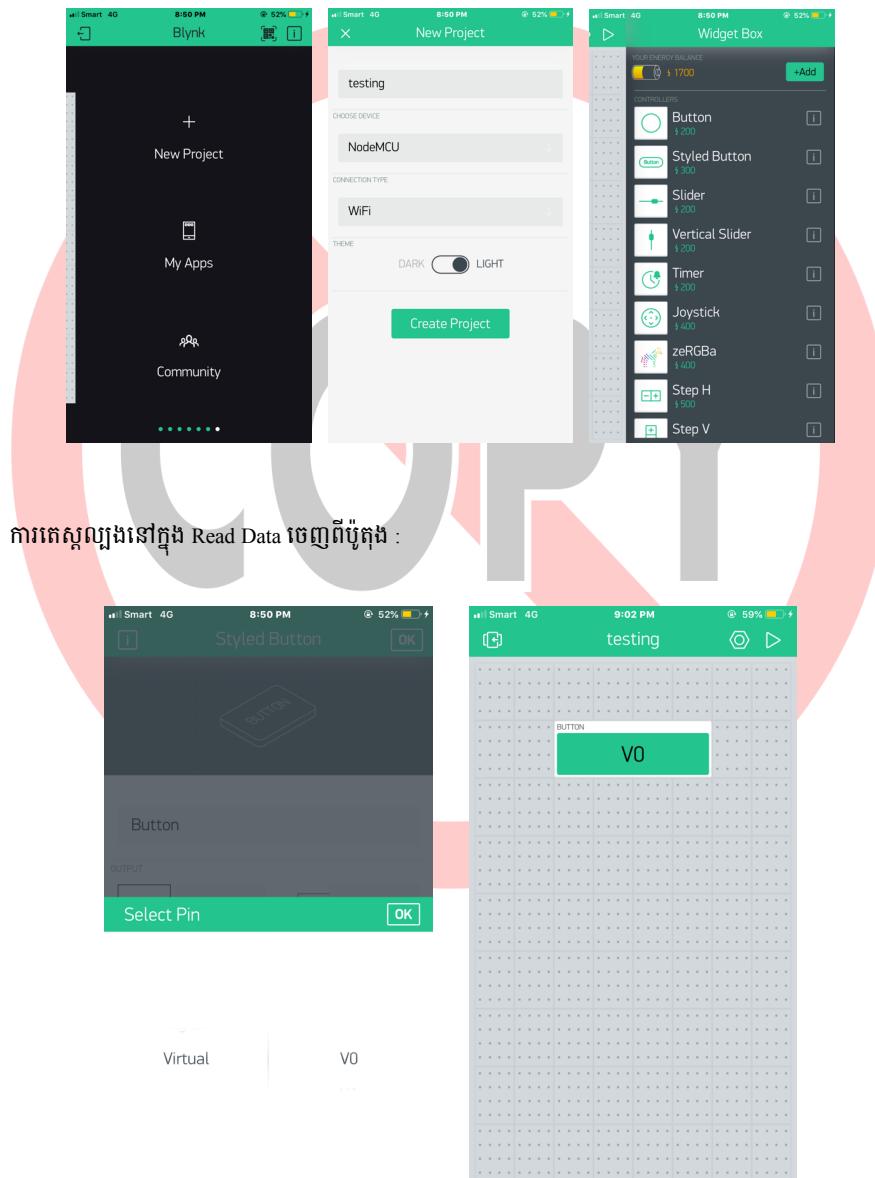
- YourAuthToken ជាកំបញ្ឈូល Token ដែលអ្នកបាន Copy ពីក្នុង App Blynk
- YourNetworkName : ឯកសារណ៍៖ WIFI របស់អ្នក (តម្លៃទូរគម្យ 100%) ឬរបីជាសារ Share Hotspot របស់ Smart Phone អ្នកទៅ
- YourPassword : ជាកំបញ្ឈូល Password WIFI របស់អ្នក

- Upload Program ចូលទៅកាន់ ESP8266
- ចូលទៅកាន់ Blynk វិញចូច Compile



បើសិនជាមួកប្រើប្រាស់ ESP8266 ដែលជា Board មិនមានទាន់ Driver របស់ CH340 ទេ នៅពេលដែលមួកបញ្ចប់ពី App Blynk ឡើងអាចមិន LED នៅលើ ESP8266 តានប្រាប់ LED នោះបានក្នុងបំពាយដើម្បីដឹង D0 យូចធាតុប្រព័ន្ធបានក្នុងករណីដែលមួកប្រើប្រាស់ ESP8266 ដែលជា Board ដំទាន់ទាន់ Driver របស់ CH340 អ្នកត្រូវធ្វើការក្នុងបំពាយដើម្បីដឹង D0 ដើម្បីធ្វើការតែស្ថាកល្បងនៅក្នុងការបញ្ចប់ App Blynk មក ESP8266 របស់អ្នកតាមរយៈ WIFI។

ការធ្វើអេស៊ីណាគលដ្ឋានការប្រើប្រាស់មុខងារមួយចំនួននៅក្នុង App Blynk

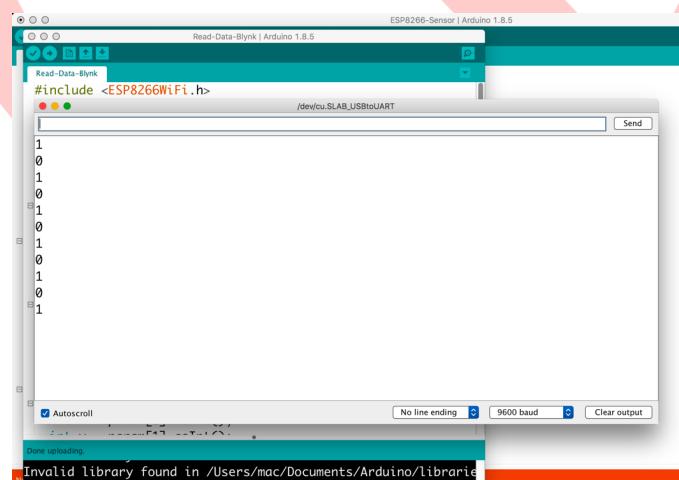


ចូល :

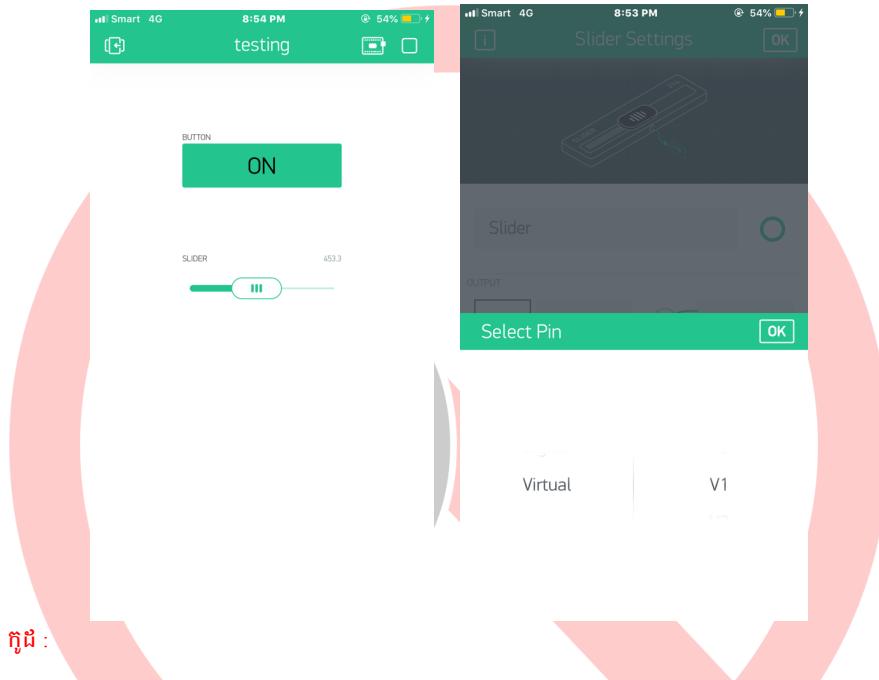
```

1. #include <ESP8266WiFi.h>
2. #include <BlynkSimpleEsp8266.h>
3. char auth[] = "22ca22c22e8d4d1dbcc4701437ada3ac";
4. char ssid[] = "Rith";
5. char pass[] = "rith017310121";
6.
7. BLYNK_WRITE(V0)
8. {
9.   int button = param.asInt();
10.  Serial.println(button);
11. }
12.void setup()
13.{
14. Serial.begin(9600);
15. Serial.begin(9600);
16. Blynk.begin(auth, ssid, pass);
17. }
18.void loop()
19.{
20. Blynk.run();
21. }
```

ទិន្នន័យ Serial Monitor បានបញ្ជាក់ថា App Blynk អ្នកនឹងយើងឱ្យបញ្ជាចោម :



ការទិន្នន័យនៃក្នុង Read Data ចិញ្ចី Slider :



តម្លៃ :

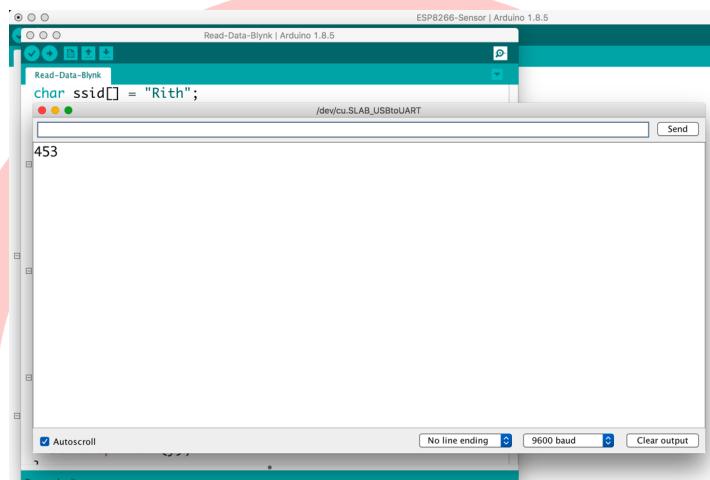
```

1. #include <ESP8266WiFi.h>
2. #include <BlynkSimpleEsp8266.h>
3. char auth[] = "22ca22c22e8d4d1dbcc4701437ada3ac";
4. char ssid[] = "Rith";
5. char pass[] = "rith017310121";
6.
7. BLYNK_WRITE(V0)
8. {
9.     int buttonn = param.asInt();
10.    Serial.println(buttonn);
11.}
12.BLYNK_WRITE(V1)
13.{ 
14.    int slide = param.asInt();
15.    Serial.println(slide);
16.}
17.void setup()
18.{ 
19.    Serial.begin(9600);

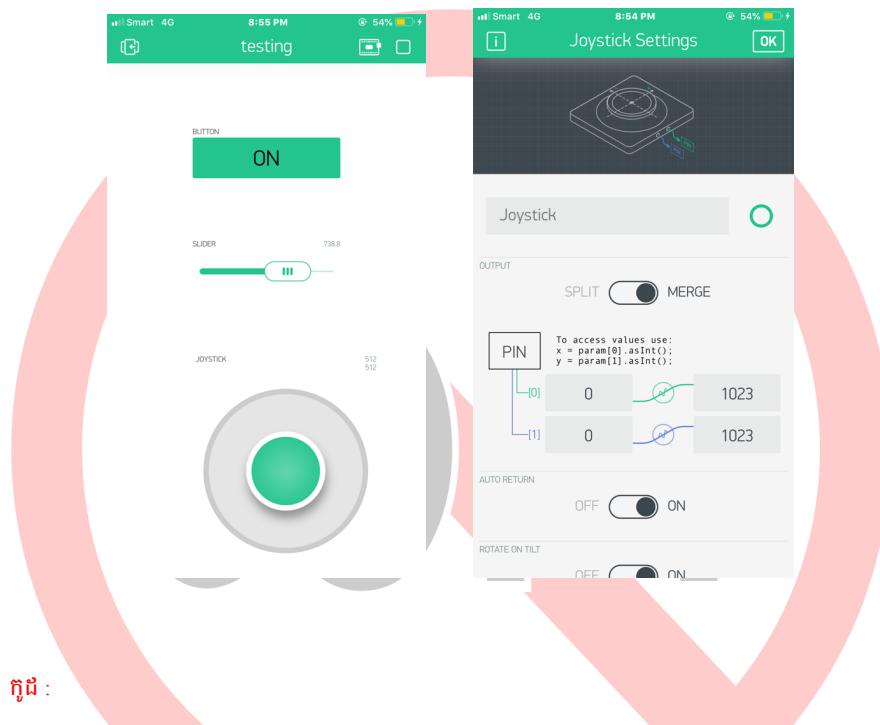
```

```
20. Serial.begin(9600);
21. Blynk.begin(auth, ssid, pass);
22.
23.void loop()
24.
25. Blynk.run();
26.
```

ទិន្នន័យ Serial Monitor បានបញ្ជាក់ថា App Blynk អ្នកនឹងយើងឱ្យចូលរួមទៅក្នុងការងារ។



ការទេស្ថិតលើរៀងបន្ទីយ JoyStick :



ក្នុង :

```

1. #include <ESP8266WiFi.h>
2. #include <BlynkSimpleEsp8266.h>
3. char auth[] = "22ca22c22e8d4d1dbcc4701437ada3ac";
4. char ssid[] = "Rith";
5. char pass[] = "rith017310121";
6.
7. BLYNK_WRITE(V0)
8. {
9.     int buttonn = param.toInt();
10.    Serial.println(buttonn);
11.}
12.BLYNK_WRITE(V1)
13.{ 
14.    int slide = param.toInt();
15.    Serial.println(slide);
16.}

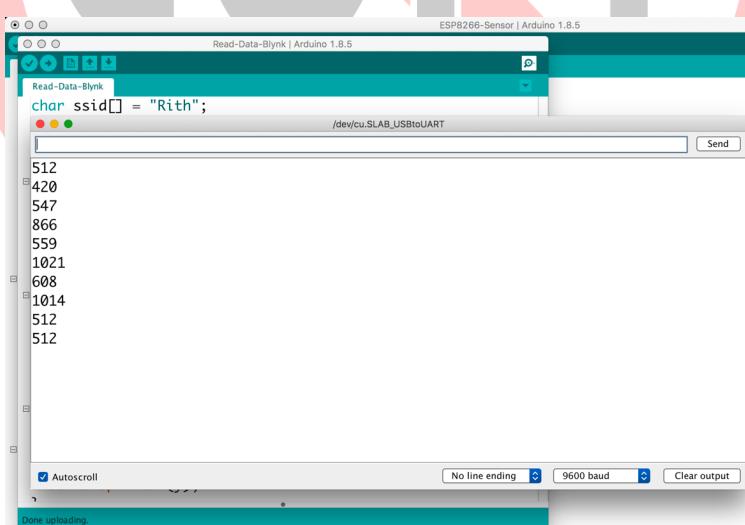
```

```

17.BLYNK_WRITE(V2)
18.{
19. int x = param[0].asInt();
20. int y = param[1].asInt();
21. Serial.println(x);
22. Serial.println(y);
23.}
24.void setup()
25.{
26. Serial.begin(9600);
27. Serial.begin(9600);
28. Blynk.begin(auth, ssid, pass);
29.}
30.void loop()
31.{
32. Blynk.run();
33.}

```

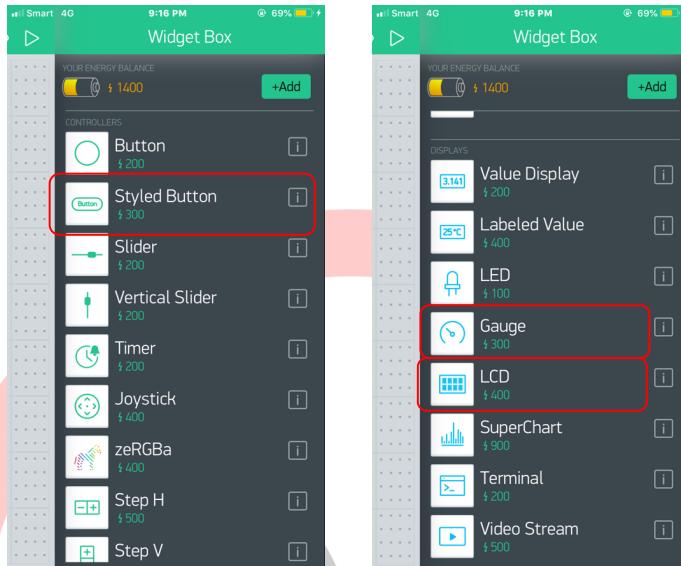
ពណ៌លខីក Serial Monitor បានចូលចិត្តពីលី App Blynk អ្នកនឹងដើរបញ្ជាច្រាម :



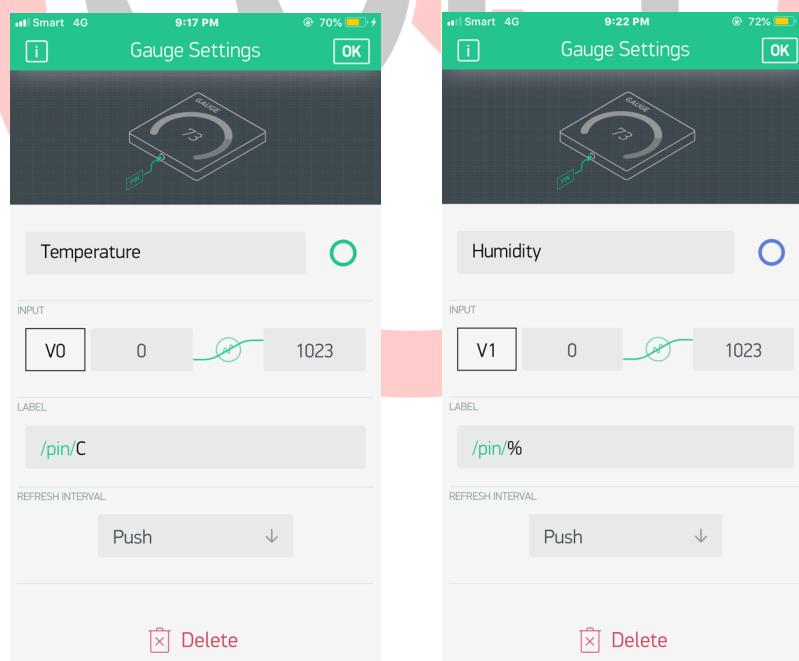
អនុវត្តន៍តម្រាង : អ្នកត្រួវទាញយកទិន្នន័យពី Humidity Sensor មកបង្ហាញនៅលើ Meter (Temperature & Humidity) នៅក្នុង App Blynk ។ បន្ទាប់មកទៀតអ្នកត្រូវកំណត់តម្លៃនៅលើ Rain Sensor នៅពេលដែលត្រួតព័ត៌មាន តាំងកាត់ទីកញ្ចប់អ្នកត្រូវសរសររុដដើម្បីឲ្យ LCD នៅក្នុង App Blynk បង្ហាញថា Raining នៅពេលដែលត្រួតព័ត៌មាន តាំងកាត់ទីកញ្ចប់អ្នកត្រូវធ្វើឲ្យ LCD នៅក្នុង App Blynk បង្ហាញថា None ។ ចំណុចចុងហ្មាយអ្នកត្រូវបើប្រាស់បូតុងពីនៅក្នុង App Blynk ដើម្បីបិទ ហើយ LED ពីរគ្រាប់ ។

បញ្ជាក់ បូតុងដែលអ្នកត្រូវបើប្រាស់ជាបូតុងដែលបើប្រាស់ដើម្បី Digital មិនមែន Virtual ទេ

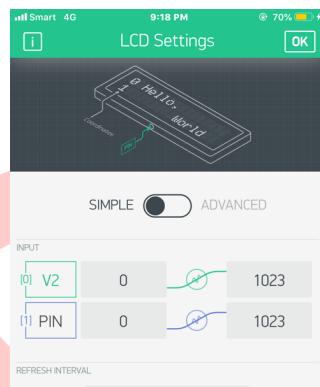




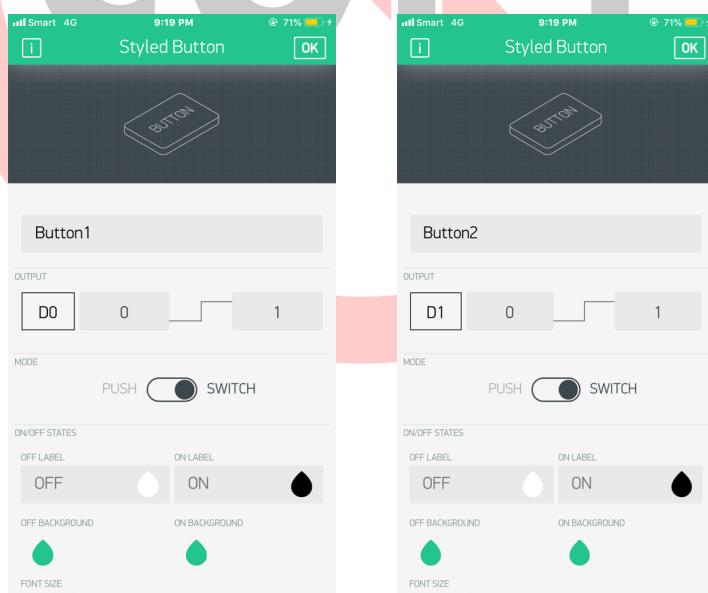
ការកំណត់នៅក្នុងប្រព័ន្ធដើម្បីរបស់ផើដឹងរបស់ Gauge or Meter (Temperature and Humidity)



ការកំណត់នៅក្នុងបច្ចុប្បន្នដើម្បីរបស់LCD



ការកំណត់នៅក្នុងបច្ចុប្បន្នដើម្បីរបស់Button



ចូល :

```

1. #include <ESP8266WiFi.h>
2. #include <BlynkSimpleEsp8266.h>
3. #include <SimpleTimer.h>
4. #include <DHT.h>
5. #define BLYNK_PRINT Serial
6. #define DHTTYPE DHT11
7. #define dht_dpin D2 //Pin2
8. DHT dht(dht_dpin, DHTTYPE);
9. SimpleTimer timer;
10.
11. char auth[] = "57c6b8ed83934d8f8cbe758d891a5167";
12. char ssid[] = "Huawei";
13. char pass[] = "monomy";
14. float t;
15. float h;
16. void Rain_Sensor()
17. {
18.     int data_Rain = analogRead(A0);
19.
20.     if (data_Rain < 700 )
21.     {
22.         Blynk.virtualWrite(V2, " Raining");
23.         Serial.println("Raining");
24.     }
25.     else
26.     {
27.         Blynk.virtualWrite(V2, " None");
28.         Serial.println("None");
29.     }
30. }
31. void Humidity_Sensor()
32. {
33.     float h = dht.readHumidity();
34.     float t = dht.readTemperature();
35.     if (isnan(h) || isnan(t)) {
36.         Serial.println("Failed to read from DHT sensor!");
37.         return;
38.     }
39.     Serial.println("Humidity and temperature\n\n");
40.     Serial.print("Current humidity = ");
41.     Serial.print(h);

```

```
42. Serial.print("% ");
43. Serial.print("temperature = ");
44. Serial.print(t);
45. Blynk.virtualWrite(V0, t);
46. Blynk.virtualWrite(V1, h);
47.
48.}
49.void setup()
50.{  
51. Serial.begin(9600);
52. Blynk.begin(auth, ssid, pass);
53. dht.begin();
54. timer.setInterval(2000, Humidity_Sensor);
55. timer.setInterval(1000L, Rain_Sensor);
56.}
57.void loop()
58.{  
59. Blynk.run();
60. timer.run();
61.}
```

