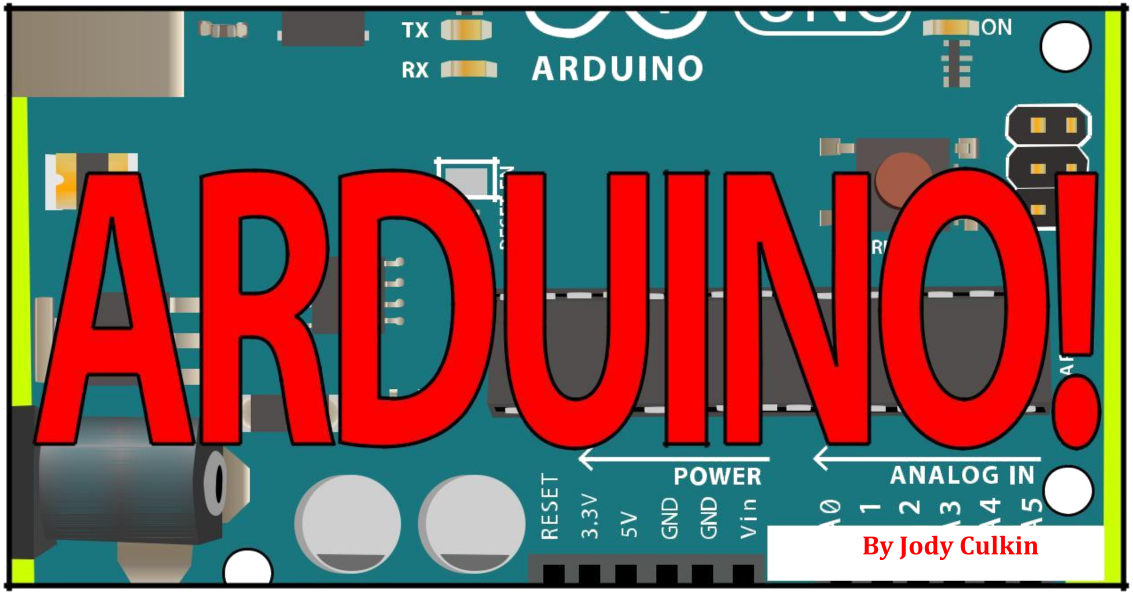


તે ઓપન સોર્સ ઇલેક્ટ્રોનિક્સ પ્રોટોટાઇપિંગ પ્લેટફોર્મ છે.

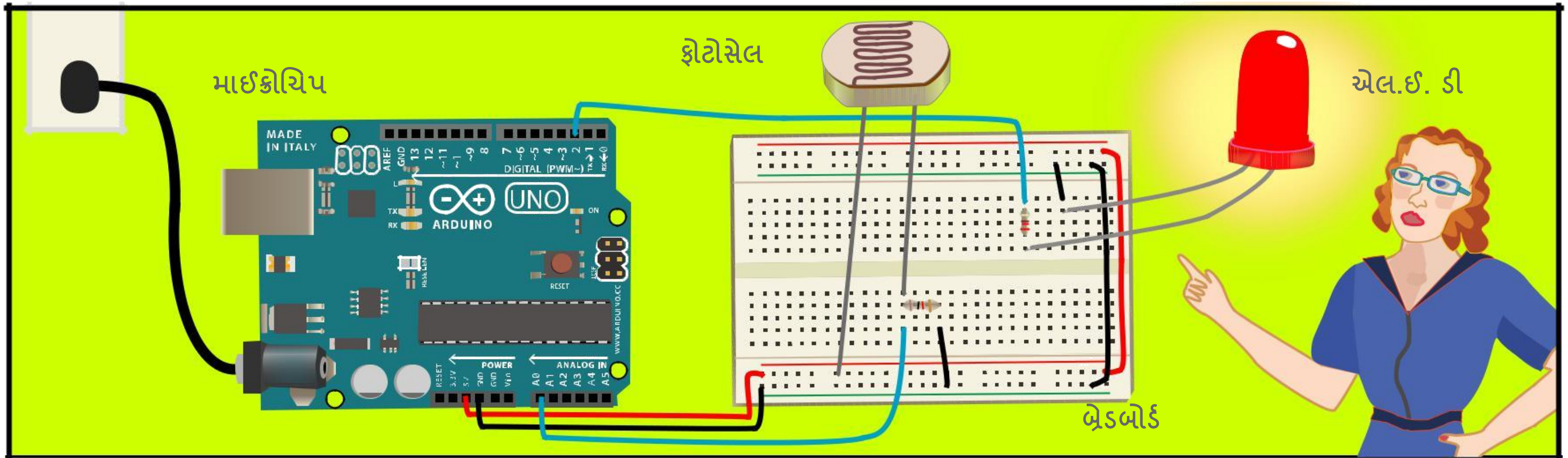
એનો અર્થ શું થાય?

ઓપન સોર્સ - "સંસાધનો જેનો ઉપયોગ, પુનઃવિતરિત અથવા મફતમાં ફરીથી લખી શકાય છે"

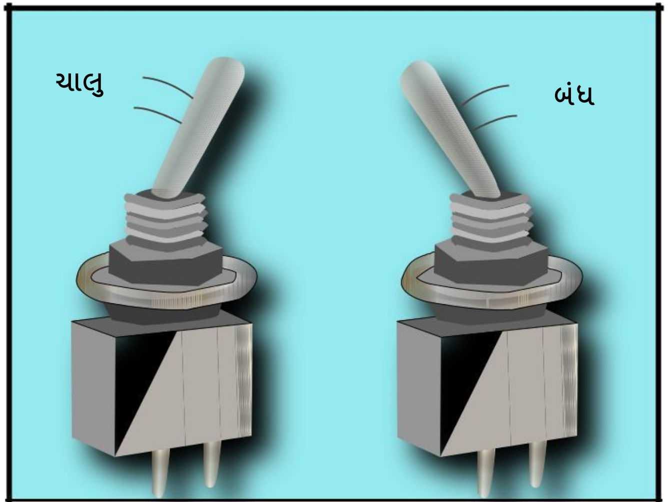
ઈલેક્ટ્રોનિક્સ - "ટેકનોલોજી જે વિવિધ માધ્યમો દ્વારા ઇલેક્ટ્રોનની નિયંત્રિત ગતિનો ઉપયોગ કરે છે."

પ્રોટોટાઇપ - "એક મૂળ સ્વરૂપ જે અન્ય વસ્તુઓ માટે આધાર અથવા ધોરણ તરીકે સેવા આપી શકે છે."

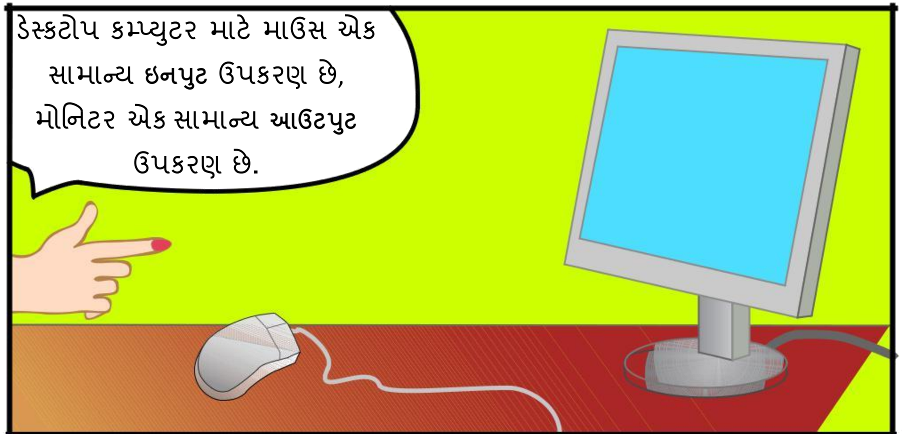
પ્લેટફોર્મ - "સોફ્ટવેર ફ્રેમવર્ક સાથેનું હાર્ડવેર આર્કિટેક્ચર જેના પર અન્ય સોફ્ટવેર ચાલી શકે છે."



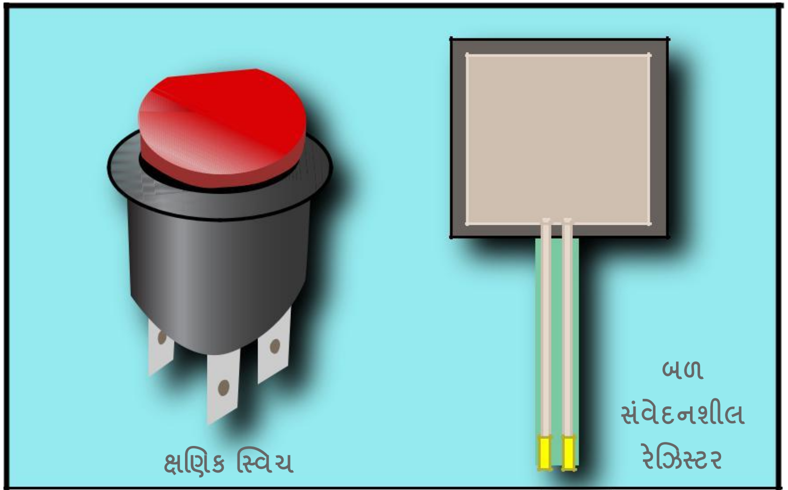
આર્ડુઇનો એક **માઇક્રોચિપ** ધરાવે છે, જે ખૂબ જ નાનું કમ્પ્યુટર છે જેને તમે પ્રોગ્રામ કરી શકો છો. તમે તેને સેન્સર જોડી શકો છો જે પરિસ્થિતિઓને માપી શકે છે (જેમ કે રૂમમાં કેટલો પ્રકાશ છે). તે નિયંત્રિત કરી શકે છે કે અન્ય વસ્તુઓ તે પરિસ્થિતિઓ પર કેવી પ્રતિક્રિયા આપે છે (રૂમ અંધારું થાય છે, એલઇડી ચાલુ થાય છે).



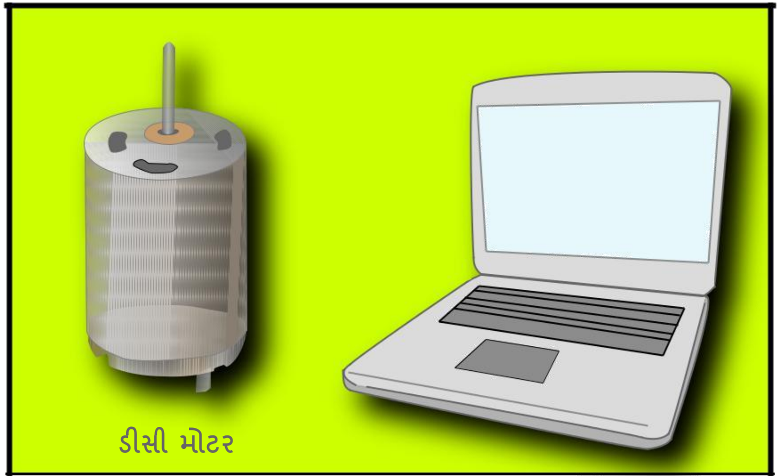
અથવા જ્યારે સ્વીચ દબાવવામાં આવે ત્યારે તે પ્રતિક્રિયા કરી શકે છે.



માઇક્રોકન્ટ્રોલર્સ કોઈપણ કમ્પ્યુટરની જેમ **ઇનપુટ્સ** અને **આઉટપુટનો** ઉપયોગ કરે છે. ઇનપુટ્સ વપરાશકર્તા અથવા પર્યાવરણમાંથી માહિતી મેળવે છે જ્યારે આઉટપુટ કેપ્ચર કરવામાં આવેલી માહિતી સાથે કંઈક કરે છે.



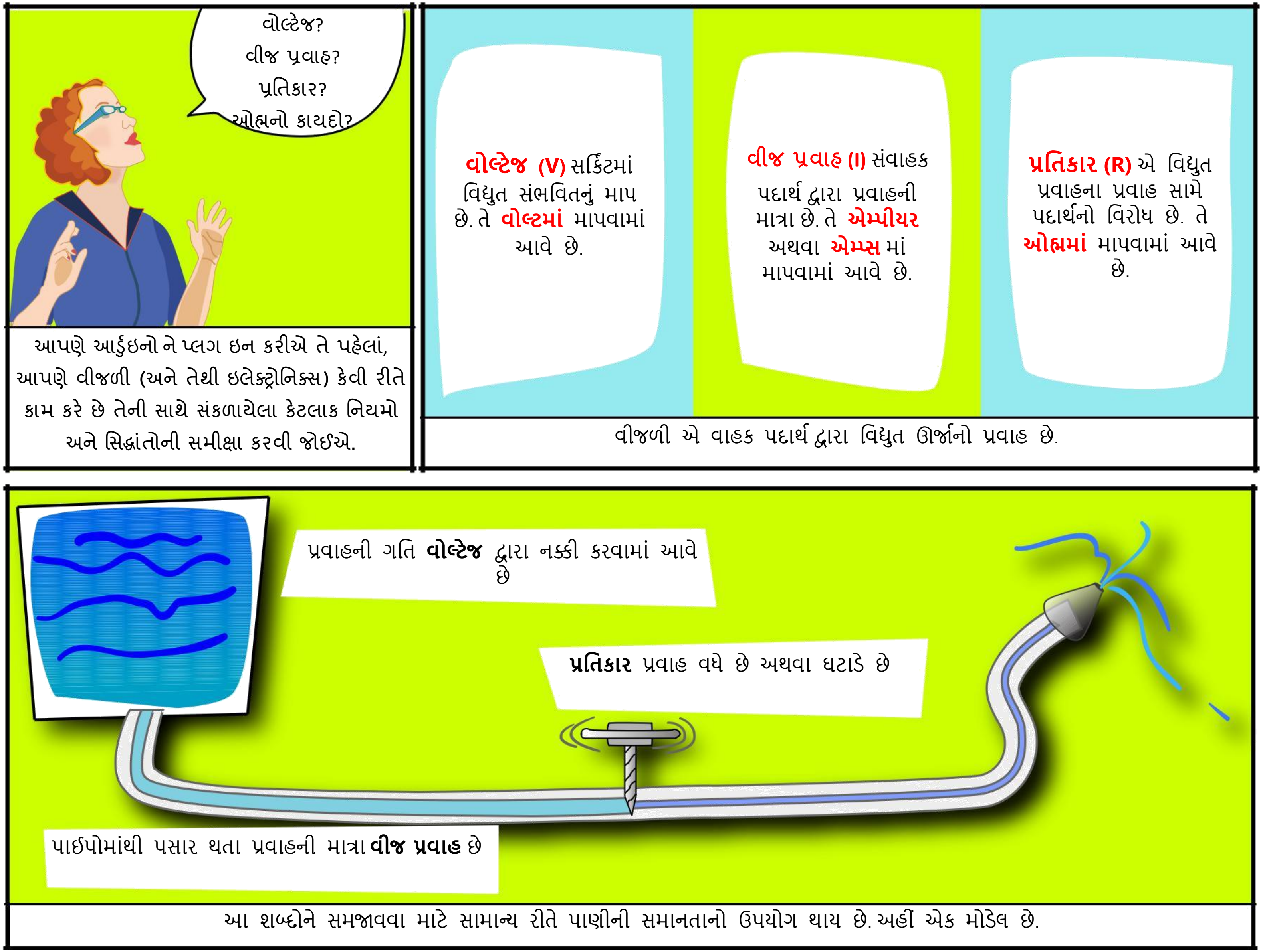
આર્ડુઇનો માં ઇનપુટ સ્વીચ અથવા સેન્સર હોઈ શકે છે.



કોઈપણ વસ્તુ અમે ચાલુ અને બંધ કરવા અને નિયંત્રણ કરવા માંગીએ છીએ તે આઉટપુટ હોઈ શકે છે. તે મોટર અથવા કમ્પ્યુટર પણ હોઈ શકે છે.

ઇનપુટ્સ અને આઉટપુટ **ડિજિટલ** અથવા **એનાલોગ** હોઈ શકે છે. ડિજિટલ માહિતી દ્વિસંગી છે - તે ક્યાં તો સાચું અથવા ખોટું હશે. એનાલોગ માહિતી સતત છે, તે મૂલ્યોની શ્રેણીની હોઈ શકે છે.

સ્વીચ એ ડિજિટલ ઇનપુટ છે; સેન્સર એ એનાલોગ ઇનપુટ છે. એનાલોગ સેન્સરની શ્રેણી તેના ડિજિટલ ડેટામાં રૂપાંતર દ્વારા સીમિત છે.


વોલ્ટેજ?
વીજ પ્રવાહ?
પ્રતિકાર?
ઓહ્મનો કાયદો?
આપણે આર્ડુઇનો ને પ્લગ ઇન કરીએ તે પહેલાં, આપણે વીજળી (અને તેથી ઇલેક્ટ્રોનિક્સ) કેવી રીતે કામ કરે છે તેની સાથે સંકળાયેલા કેટલાક નિયમો અને સિદ્ધાંતોની સમીક્ષા કરવી જોઈએ.
વોલ્ટેજ (V) સર્કિટમાં વિદ્યુત સંભવિતનું માપ છે. તે **વોલ્ટમાં** માપવામાં આવે છે.
વીજ પ્રવાહ (I) સંવાહક પદાર્થ દ્વારા પ્રવાહની માત્રા છે. તે **એમ્પીયર** અથવા **એમ્પ્સ** માં માપવામાં આવે છે.
પ્રતિકાર (R) એ વિદ્યુત પ્રવાહના પ્રવાહ સામે પદાર્થનો વિરોધ છે. તે **ઓહ્મમાં** માપવામાં આવે છે.
વીજળી એ વાહક પદાર્થ દ્વારા વિદ્યુત ઊર્જાનો પ્રવાહ છે.
પ્રવાહની ગતિ **વોલ્ટેજ** દ્વારા નક્કી કરવામાં આવે છે
પ્રતિકાર પ્રવાહ વધે છે અથવા ઘટાડે છે
પાઇપોમાંથી પસાર થતા પ્રવાહની માત્રા **વીજ પ્રવાહ** છે
આ શબ્દોને સમજાવવા માટે સામાન્ય રીતે પાણીની સમાનતાનો ઉપયોગ થાય છે. અહીં એક મોડેલ છે.

ઓહ્મનો કાયદો

વીજ પ્રવાહ = વોલ્ટેજ / પ્રતિકાર

(I= V/R)

અથવા

પ્રતિકાર = વોલ્ટેજ / વીજ પ્રવાહ प्रतिरोध

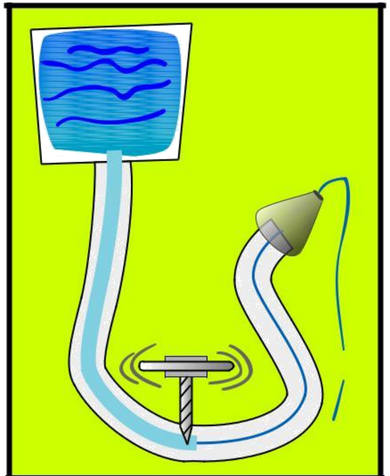
(R = V/I)

અથવા

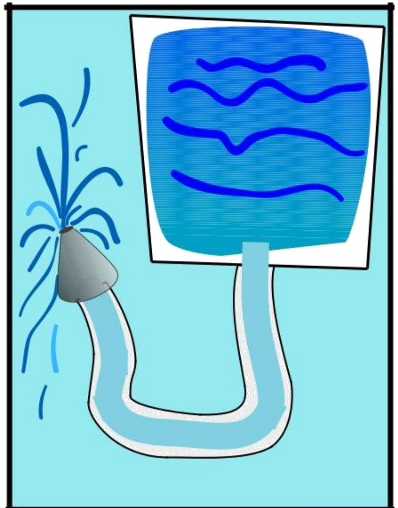
વોલ્ટેજ = પ્રતિકાર * વીજ પ્રવાહ वोल्टेज

(V = R*I)

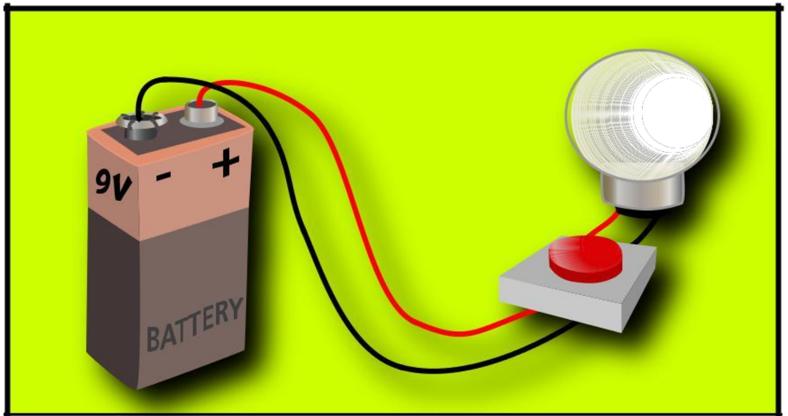
જર્મન ભૌતિકશાસ્ત્રી જ્યોર્જ ઓહ્મ દ્વારા શોધાયેલ વોલ્ટેજ, વીજ પ્રવાહ અને પ્રતિકાર વચ્ચેનો સંબંધ છે.



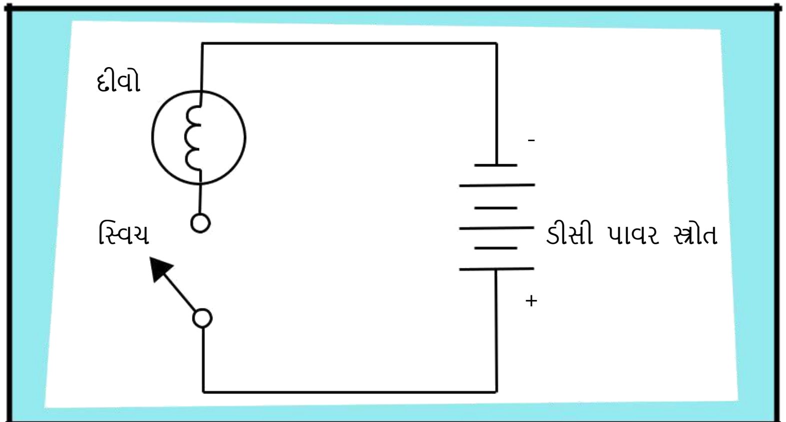
ઉદાહરણ તરીકે, પ્રતિકાર વધારો, ઓછો પ્રવાહ.



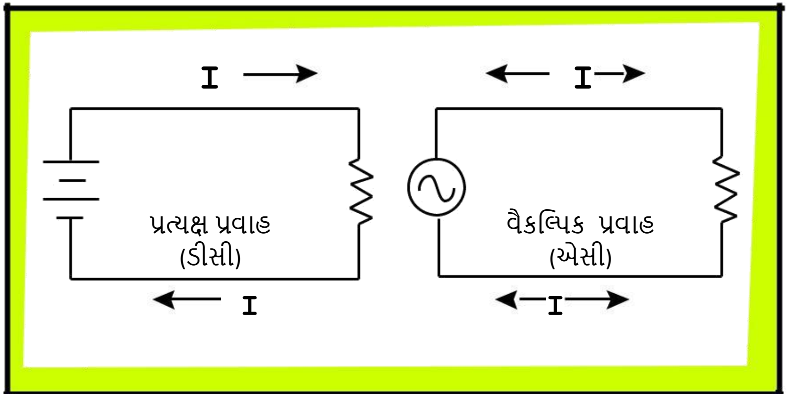
અથવા સંભવિત વધારો, વધુ પ્રવાહ.



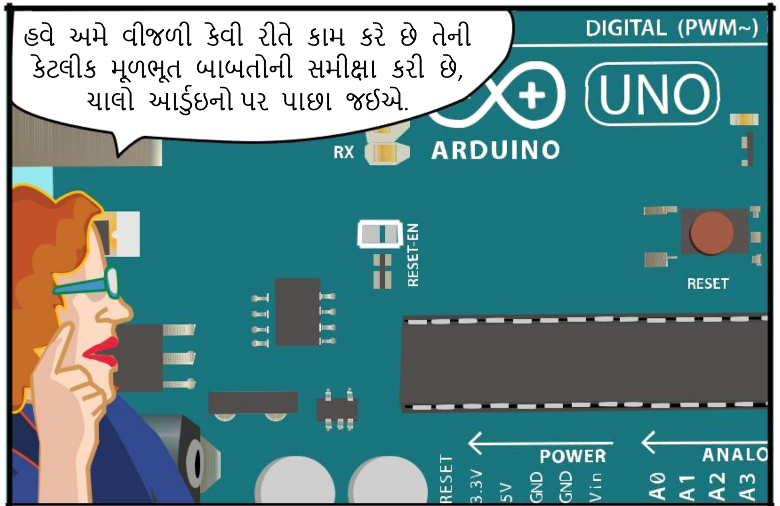
હવે ચાલો એક સરળ **સર્કિટ** જોઈએ. દરેક સર્કિટ એક બંધ લૂપ છે જેમાં ઊર્જા સ્રોત (બેટરી) અને લોડ (લેમ્પ) હોય છે. લોડ બેટરીની વિદ્યુત ઊર્જાને રૂપાંતરિત કરે છે અને તેનો ઉપયોગ કરે છે. આમાં એક સ્વીચ પણ છે.



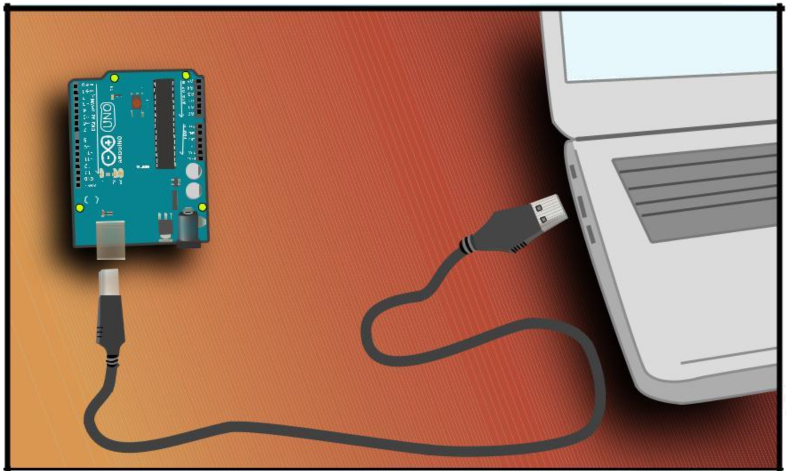
આ સમાન સર્કિટની **યોજનાકીય** છે (તે ઇલેક્ટ્રોનિક ઘટકો માટે પ્રતીકોનો ઉપયોગ કરીને સર્કિટનું પ્રતિનિધિત્વ કરે છે). જ્યારે સ્વીચ બંધ થાય છે, ત્યારે પાવર સ્રોતમાંથી વીજ પ્રવાહ વહે છે અને દીવો પ્રગટાવે છે.



ત્યાં બે સામાન્ય પ્રકારના સર્કિટ છે, **ડાયરેક્ટ પ્રવાહ** અને **વૈકલ્પિક પ્રવાહ**. ડીસી સર્કિટમાં, પ્રવાહ હંમેશા એક દિશામાં વહે છે. AC માં, પ્રવાહ નિયમિત ચક્રમાં વિરુદ્ધ દિશામાં વહે છે. અમે અહીં ફક્ત ડીસી સર્કિટ વિશે વાત કરીશું.



આર્ડુઇનો ને ચલાવવા માટે પાવરની જરૂર પડશે. તેને પ્રોગ્રામ કરવા માટે આપણે તેને કમ્પ્યુટર સાથે જોડવાની જરૂર પડશે.



યુએસબી કેબલ વડે આર્ડુઇનો ને કોમ્પ્યુટર સાથે જોડવાથી આપણને જરૂરી પાવર પુરવઠા થશે અને પ્રોગ્રામિંગ શરૂ કરવાની પરવાનગી મળશે.

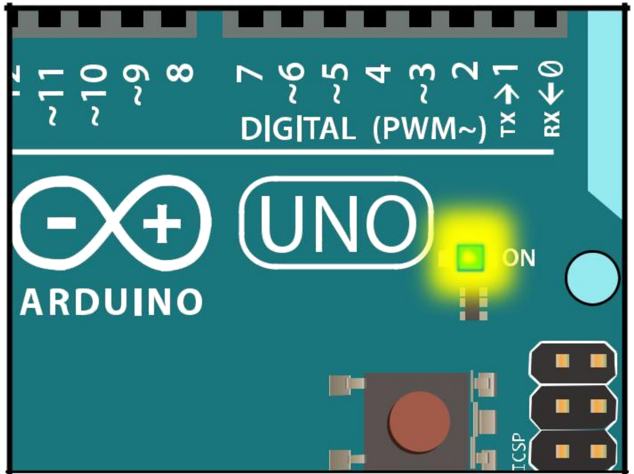
તમારે આર્ડુઇનો પ્રોગ્રામ કરવા માટે સોફ્ટવેર ડાઉનલોડ અને ઇન્સ્ટોલ કરવું પડશે. તે ઉપરના લિંક પરથી મફતમાં ઉપલબ્ધ છે. આર્ડુઇનો સોફ્ટવેર મેક ઓએસ x, વિન્ડોઝ અને લિનક્સ પ્લેટફોર્મ પર ચાલે છે.

મેક પર આર્ડુઇનો સોફ્ટવેર કેવી રીતે ઇન્સ્ટોલ કરવું તેની સૂચનાઓ માટે:
http://www.arduino.cc/en/Guide/MacOSX

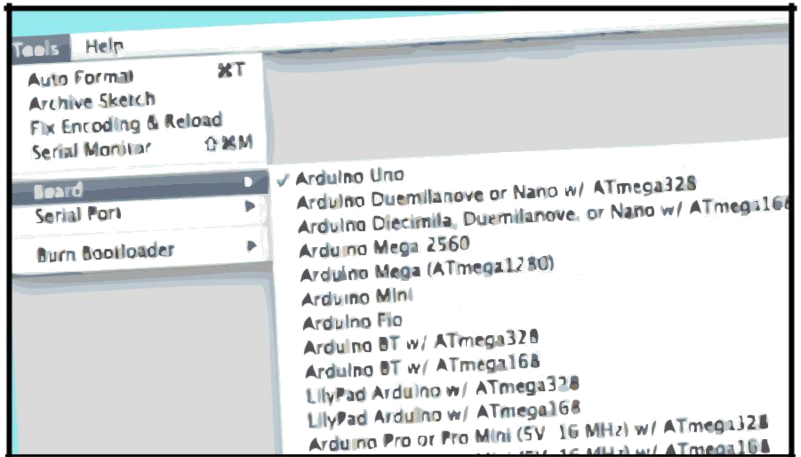
વિન્ડોઝ પર કેવી રીતે ઇન્સ્ટોલ કરવું તેની સૂચનાઓ માટે:
http://www.arduino.cc/en/Guide/Windows

લિનક્સ પર કેવી રીતે ઇન્સ્ટોલ કરવું તેની સૂચનાઓ માટે:
http://www.arduino.cc/playground/Learning/Linux

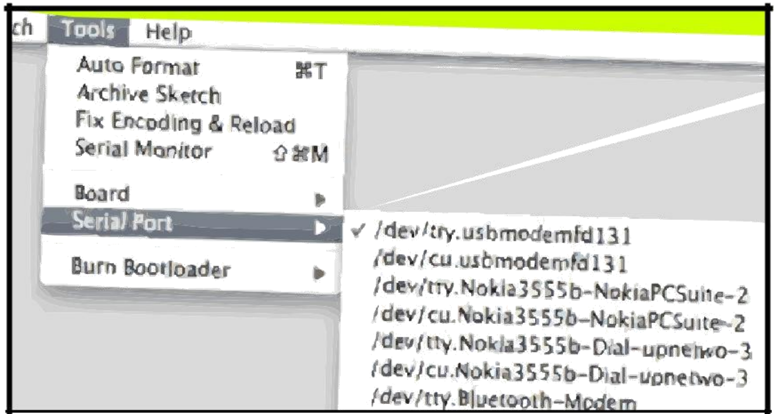
આ પ્લેટફોર્મ્સ પર સોફ્ટવેર ઇન્સ્ટોલ કરવા વિશે વિગતવાર સૂચનાઓ માટે ઉપરના લિંક પર જાઓ.



જ્યારે તમે સોફ્ટવેર ઇન્સ્ટોલ કરી લો, ત્યારે આર્ડુઇનો ને કનેક્ટ કરો. **ON** ચિહ્નિત થયેલ એલઇડી બોર્ડ પર પ્રકાશિત થવી જોઇએ.

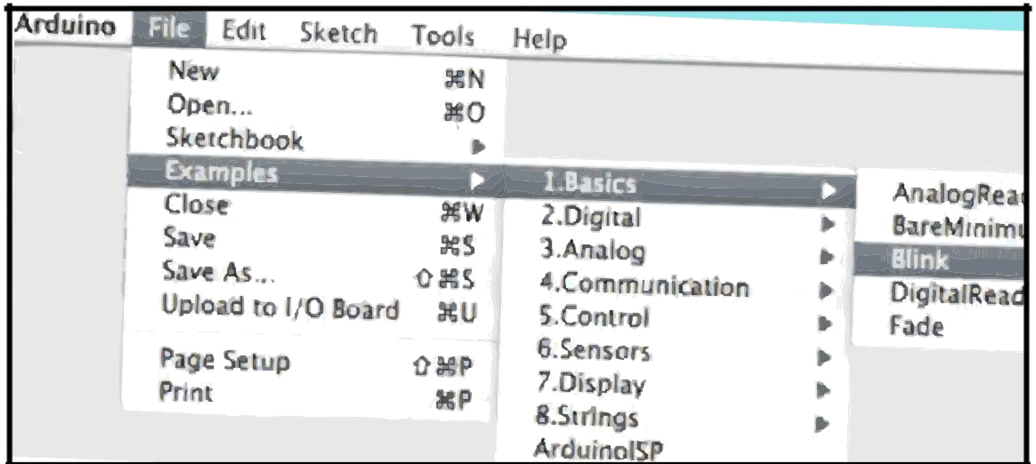


આર્ડુઇનો સોફ્ટવેર લોંચ કરો. ટૂલ્સ મેનૂમાં, તમે ઉપયોગ કરી રહ્યાં છો તે બોર્ડ પસંદ કરો (**ટૂલ્સ > બોર્ડ**). ઉદાહરણ તરીકે, **આર્ડુઇનો યુનો**.

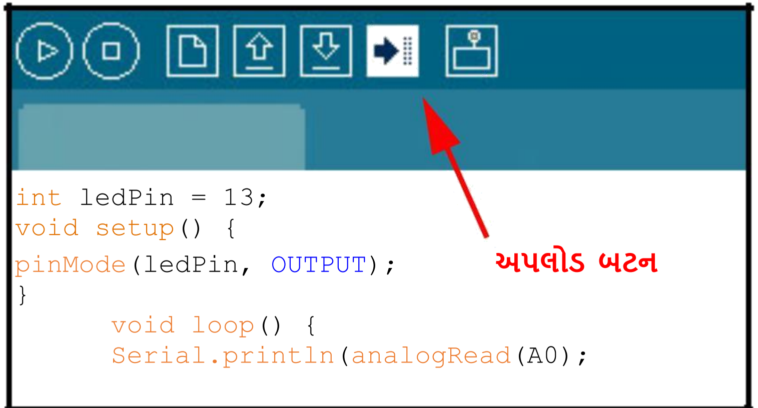


આગળ સીરીયલ પોર્ટ પસંદ કરો. (**ટૂલ્સ > સીરીયલ પોર્ટ**) મેક પર તે કંઇક **/dev/tty.usbmodem** જેવું હશે. વિન્ડોઝ મશીન પર, તે **COM3** અથવા તેના જેવું કંઇક હશે.

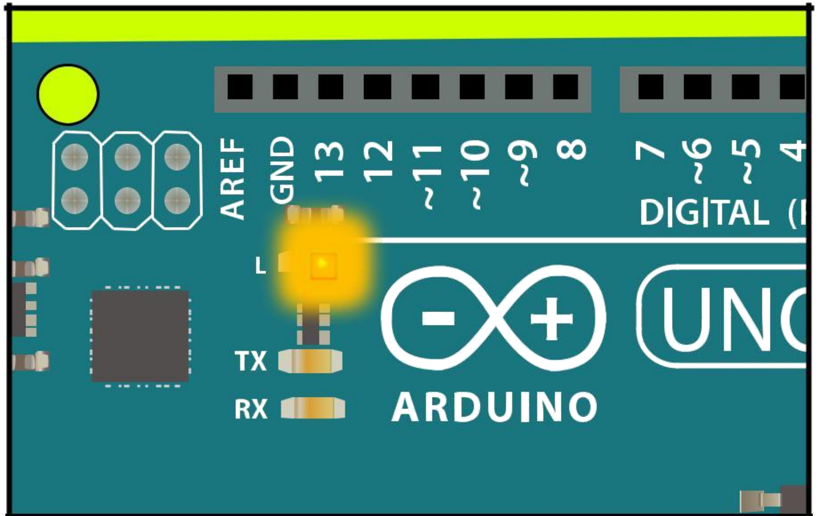
જ્યારે તમે આર્ડુઇનો સોફ્ટવેર ડાઉનલોડ કર્યું, ત્યારે તમે **IDE** ડાઉનલોડ કર્યું. તે પ્રોગ્રામરોને સોફ્ટવેર વિકસાવવામાં મદદ કરવા માટે કમ્પાઇલર અને અન્ય સુવિધાઓ સાથે ટેક્સ્ટ એડિટરને જોડે છે.



આર્ડુઇનો IDE તમને **સ્કેચ** અથવા **પ્રોગ્રામ** લખવા અને તેને આર્ડુઇનો બોર્ડ પર અપલોડ કરવાની પરવાનગી આપે છે. ફાઇલ મેનૂમાં **બ્લિંક** ઉદાહરણ ખોલો. **ફાઇલ > ઉદાહરણો > 1.બેઝિક્સ > બ્લિંક.**



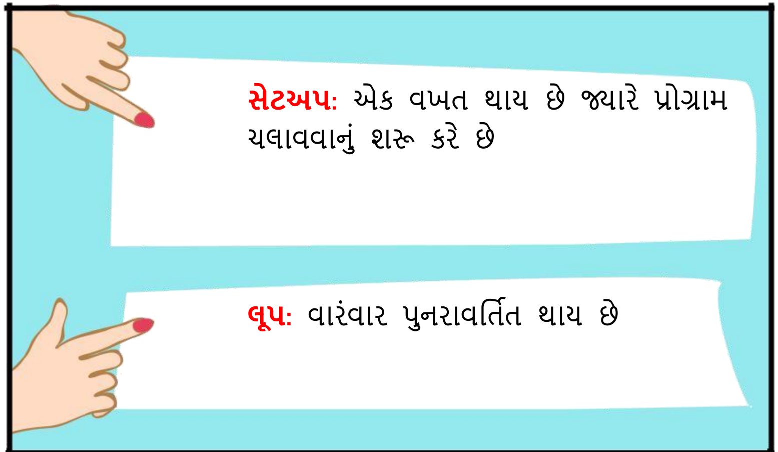
આર્ડુઇનો બોર્ડ પર સ્કેચ અપલોડ કરવા માટે, વિન્ડોની ટોચ પર બટનોની પટ્ટી પર **અપલોડ બટનને** ક્લિક કરો. કેટલાક સંદેશાઓ વિન્ડોની નીચે દેખાશે, છેલ્લે **અપલોડ કરવાનું પૂર્ણ** થયું.



આર્ડુઇનો પર પિન ૧૩ પર એલ.ઈ.ડી ઝબકવાનું શરૂ કરે છે.

```
void setup() {
 // initialize the digital pin as an output.
 // Pin 13 has LED connected on most Arduino
boards:
 pinMode(13, OUTPUT);
}
void loop() {
 digitalWrite(13, HIGH); // set the LED on
 delay(1000); // wait for a second
 digitalWrite(13, LOW); // set the LED off
 delay(1000); // wait for a second
}
```

સ્કેચ, કોઈપણ ભાષામાં લખેલા પ્રોગ્રામની જેમ, કમ્પ્યુટર માટેની સૂચનાઓનો સમૂહ છે. જો આપણે બ્લિંક સ્કેચને નજીકથી જોઈએ, તો આપણે જોઇએ છીએ કે ત્યાં ૨ મુખ્ય ભાગો, **સેટઅપ** અને **લૂપ** છે.



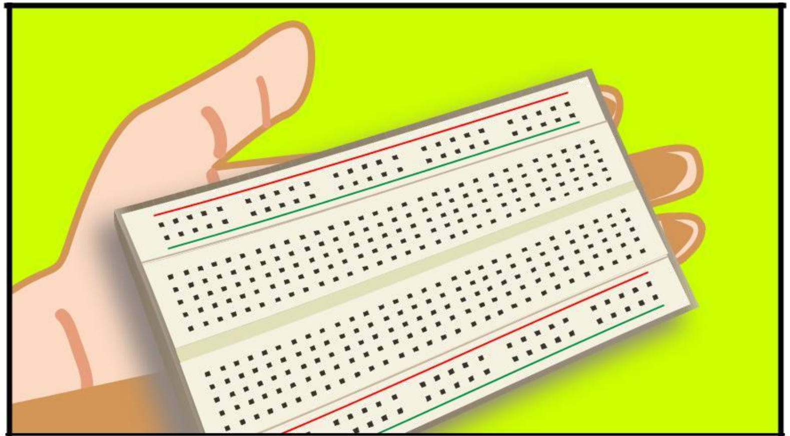
આ બંને કોડના બ્લોક્સ છે જેને **ફંક્શન** કહેવાય છે જે દરેક સ્કેચમાં હશે. તેઓ સર્પાકાર કૌંસ { } દ્વારા અવરોધિત છે.

http://arduino.cc/en/Reference/HomePage

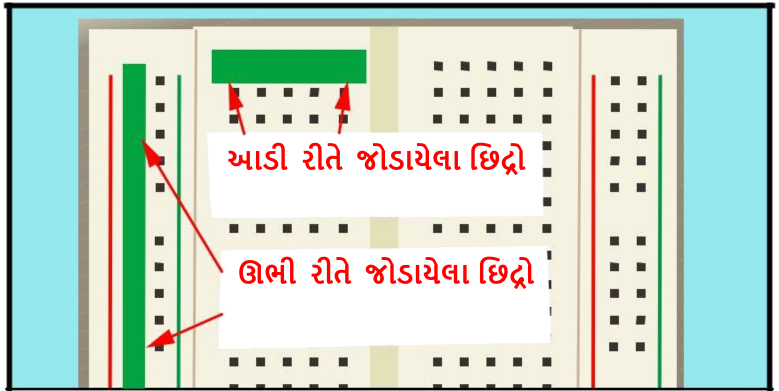
આર્ડુઇનો સંદર્ભ માર્ગદર્શિકા અને ભાષા શીખવા માટે અન્ય ઘણા સંસાધનો માટે આર્ડુઇનો વેબસાઇટ તપાસો.

```
void setup() {        //declares block of code
  pinMode(13, OUTPUT);  //sets pin 13 to output
}        //end block of code

void loop() {         //declares block of code
  digitalWrite(13, HIGH);     //sets pin 13 high
  delay(1000);        //pause 1 second
  digitalWrite(13, LOW);      //sets pin 13 low
  delay(1000);        //pause 1 second
}        //end block of code
```

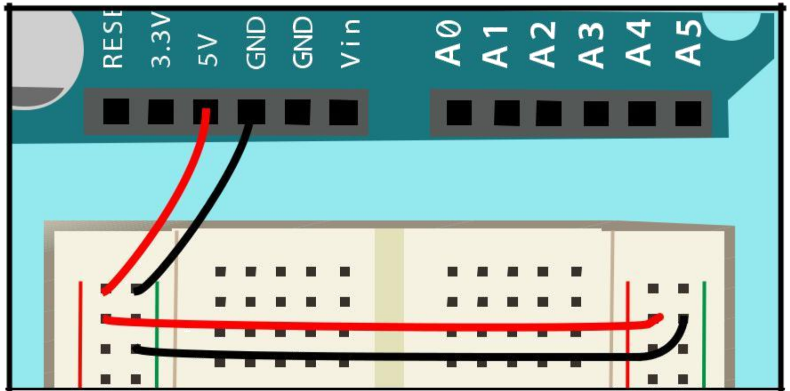
હમણાં માટે, ચાલો આ સરળ સ્ક્રિપ્ટ લાઇન બાય લાઇન જોઈએ અને જોઈએ કે દરેક લાઇન શું કરે છે.



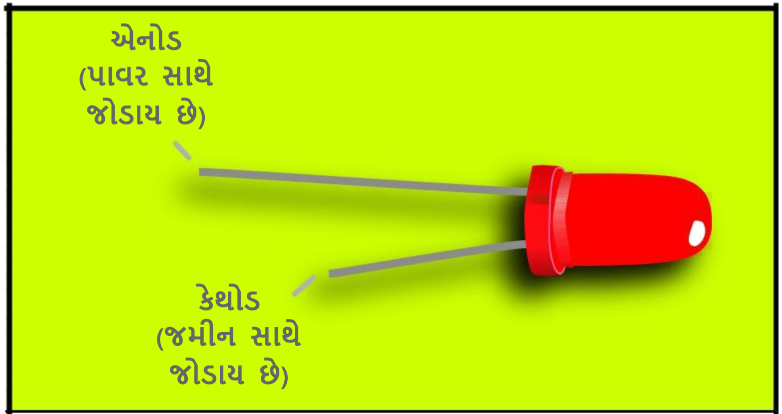
આર્ડુઇનો બોર્ડ પર ન હોય તેવા પદાર્થોને આપણે કેવી રીતે નિયંત્રિત કરી શકીએ? અમે આર્ડુઇનો ને **સોલ્ડરલેસ બ્રેડબોર્ડ** સાથે જોડીશું. આ અમને ઝડપથી સર્કિટ સેટ કરવા અને પરીક્ષણ કરવાની મંજૂરી આપશે.



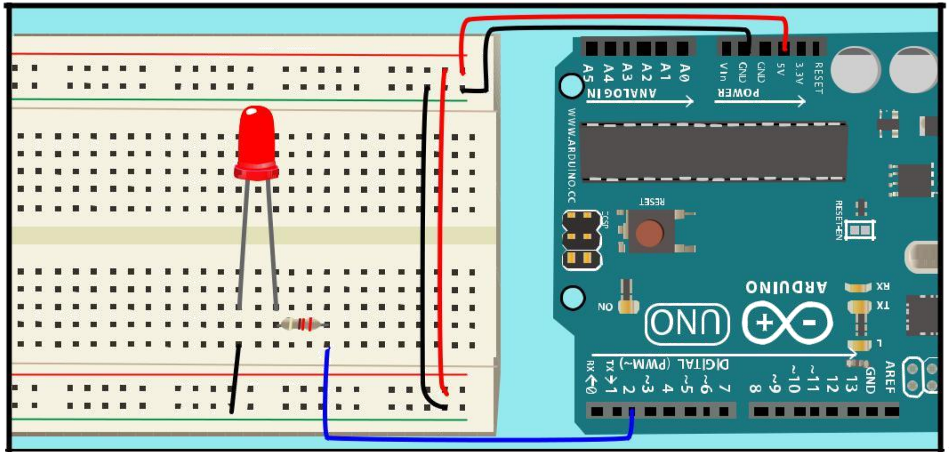
આ બ્રેડબોર્ડમાં ડાબી અને જમણી બાજુએ નીચે ચાલતા છિદ્રોની ૨ પંક્તિઓ છે, અને મધ્ય ઇન્ડેન્ટેશનની બંને બાજુએ છિદ્રોની ૫ પંક્તિઓ છે. બાજુની પંક્તિઓ **ઊભી** રીતે જોડાયેલ છે, મધ્યમાં ૫ છિદ્રોની દરેક પંક્તિ **આડી** રીતે જોડાયેલ છે.



અમે આર્ડુઇનો બોર્ડમાંથી **પાવર** અને **ગ્રાઉન્ડને** ૨૨ ગેજ વાયર વડે ડાબી અને જમણી બાજુએ ઊભી રીતે જોડાયેલ સ્ટ્રીપ્સ સાથે જોડીશું. અન્ય ઘટકો મધ્યમાં છિદ્રો અને પાવર અને જમીન પર જરૂરિયાત મુજબ જોડી શકાય છે.



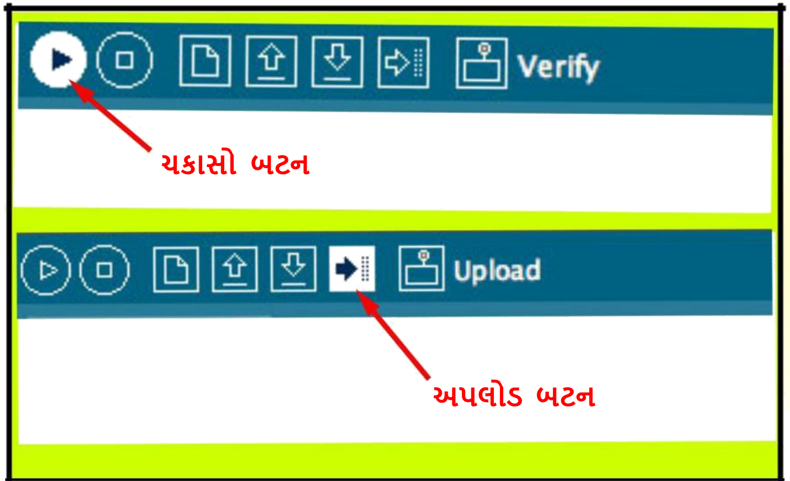
જ્યારે વર્તમાન **એલઇડી (લાઇટ ઇમિટિંગ ડાયોડ)** દ્વારા યોગ્ય દિશામાં વહે છે, ત્યારે તે પ્રકાશિત થાય છે. અમે બ્રેડબોર્ડ પર એલઇડી જોડીશું, પછી અરડિનો સાથે જેથી અમે તેને કોડ સાથે નિયંત્રિત કરી શકીએ.



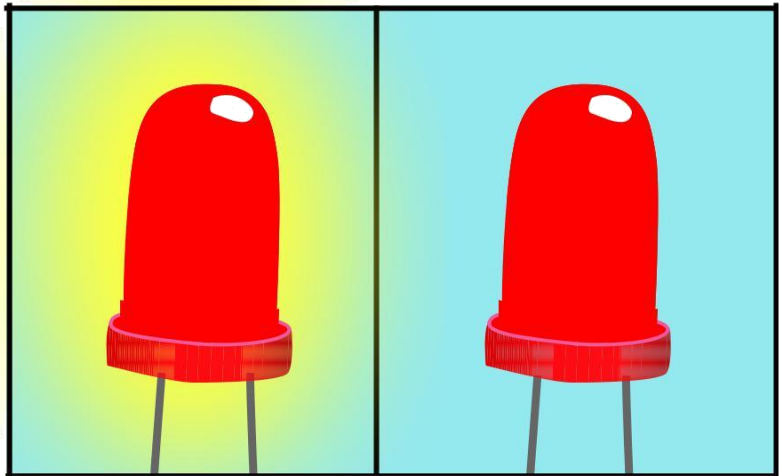
એ **૨૨૦ ઓહ્મ રેઝિસ્ટર** દ્વારા આર્ડુઇનો પર **પિન ૨** સાથે જોડાયેલ છે. **કેથોડ જમીન** સાથે જોડાયેલ છે. પિન ૨ થી ૧૩ ને ડિજિટલ ઇનપુટ્સ અથવા આઉટપુટ તરીકે ગોઠવી શકાય છે. સ્કેચ શરૂ કરવા માટે **નવું** બટન ક્લિક કરો.

```
void setup() {
      pinMode(2, OUTPUT);
}
void loop() {
      digitalWrite(2, HIGH);
      delay(500);
      digitalWrite(2, LOW);
      delay(500);
      }
```

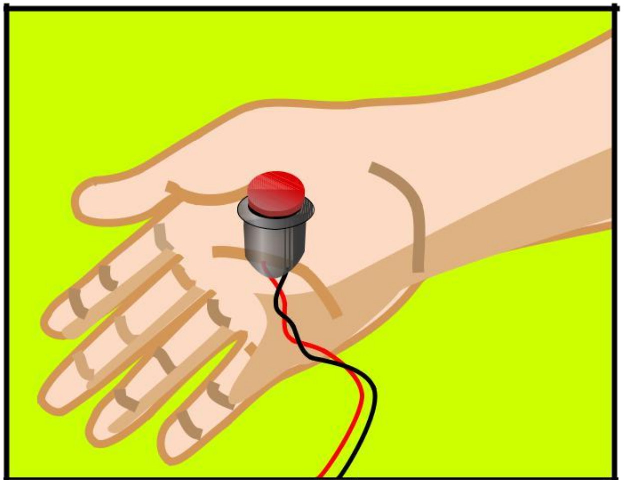
સેટઅપમાં, અમે પિન ૨ ને આઉટપુટ તરીકે સેટ કરીએ છીએ. **લૂપમાં**, પહેલા આપણે પિન ૨ ને ઊંચો સેટ કરીએ છીએ જે લીડને લાઇટ કરે છે. વિલંબ ૫૦૦ મિલીસેકન્ડ અથવા અડધી સેકન્ડ થોભાવે છે. જ્યારે પિન ૨ નીચો સેટ થાય છે, ત્યારે એલઇડી બંધ થઇ જાય છે, અમે બીજી અડધી સેકન્ડ થોભાવીએ છીએ.



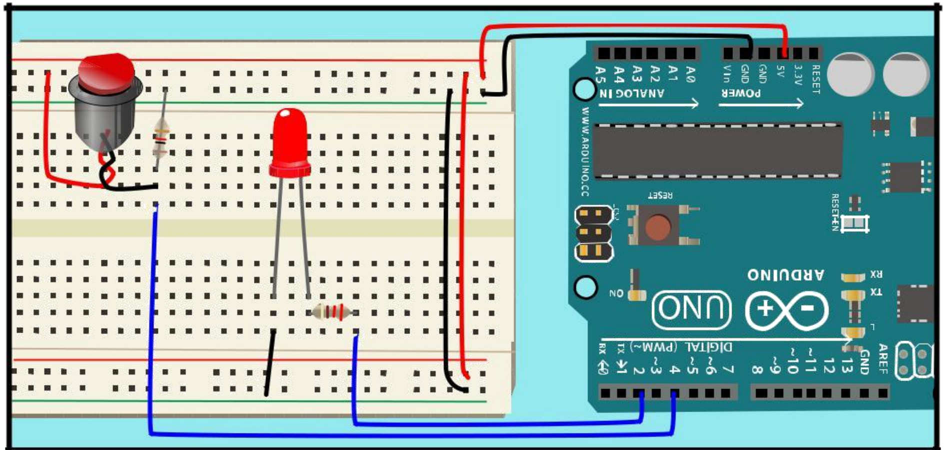
તમારો કોડ તપાસવા માટે મેનૂ પર ચકાસો પર ક્લિક કરો. જો ત્યાં કોઈ ભૂલો ન હોય, તો તમારા પ્રોગ્રામને આર્ડુઇનો પર મૂકવા માટે અપલોડ પર ક્લિક કરો.



એલ.ઈ.ડી અડધી સેકન્ડ માટે ઝબકી જાય છે, પછી અડધી સેકન્ડ માટે વારંવાર ઝબકી જાય છે.



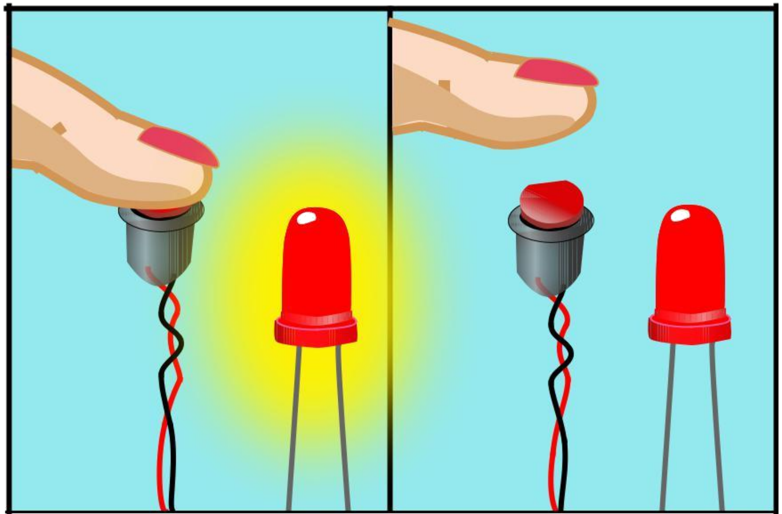
આગળ આપણે એક સ્વીચ, એક ડિજિટલ ઇનપુટ ઉમેરીશું, જેથી આપણે એલ.ઈ. ડી ને બંધ અને ચાલુ કરી શકીએ.



આર્ડુઇનો પર પિન ૪ પર ક્ષણિક સ્વીચના એક છેડાને કનેક્ટ કરોએક જ અંત સાથે જોડાયેલ જમીન સાથે જોડાયેલ ૧૦ કે રેઝિસ્ટર સાથે. બીજા છેડાને પાવર સાથે જોડો. અમે એ જ પિન સાથે જોડાયેલ એલઇડી છોડીશું.

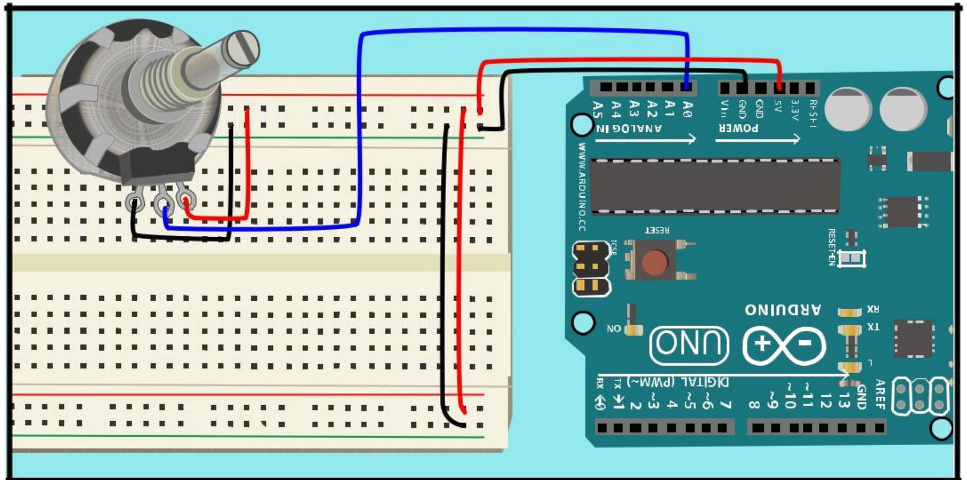
```
void setup() {
     pinMode(2, OUTPUT);
     pinMode(4, INPUT);
}

void loop() {
     if(digitalRead(4)){
      digitalWrite(2, HIGH);
     }else{
      digitalWrite(2, LOW);
     }
}
```

આગળ, અમે કોડ લખીશું. સેટઅપમાં, અમે પિન ૨ ને આઉટપુટ અને પિન ૪ ને ઇનપુટ જાહેર કરીએ છીએ. લૂપમાં, અમે જો સ્ટેટમેન્ટનો ઉપયોગ કરીએ છીએ, જો આપણે પિન ૪ ને ઉચ્ચ તરીકે વાંચીએ છીએ, તો અમે એલઇડી પિનને ઉચ્ચ પર સેટ કરીએ છીએ, અન્યથા અમે એલઇડી પિનને નીચા પર સેટ કરીએ છીએ, તેને બંધ કરીએ છીએ.



જ્યારે સ્વીચ દબાવી રાખવામાં આવે ત્યારે એલ.ઈ. ડીલાઇટ થાય છે.

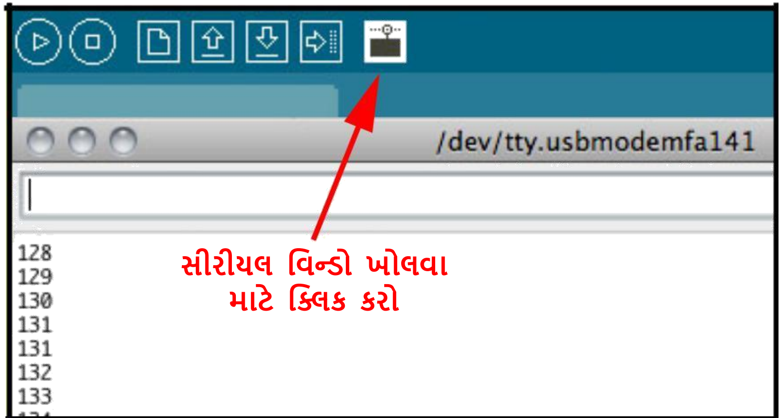
હવે આપણે એનાલોગ ઇનપુટ સેટ કરીશું. અમે પોટેન્ટિઓમીટરનો ઉપયોગ કરીશું.



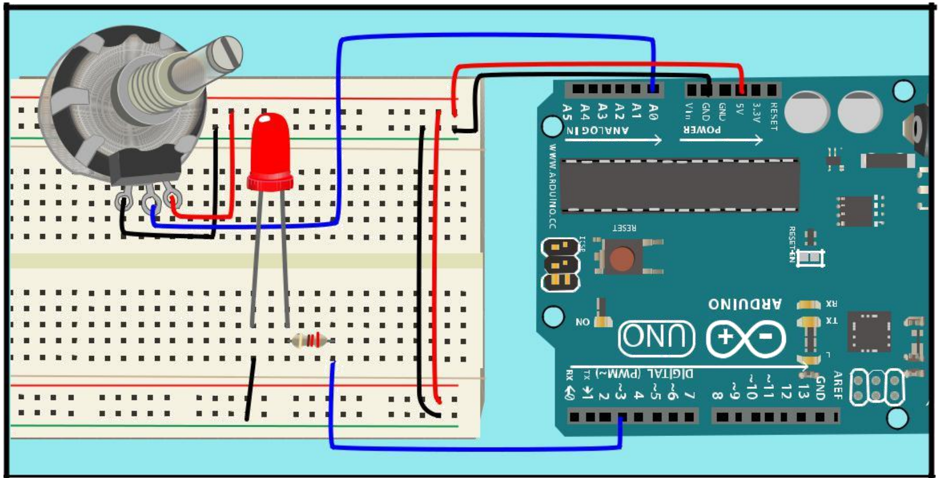
પોટેન્ટિઓમીટર પર મધ્ય પિનને એનાલોગ પિન A0 સાથે જોડો. પોટના એક છેડાને પાવર સાથે, બીજાને જમીન સાથે જોડો.

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

void loop() {
     Serial.println(analogRead(A0));
}
```

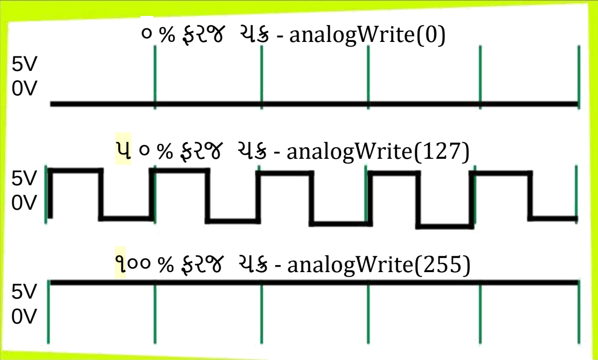
પ્રથમ આપણે **સીરીયલ મોનિટરનો** ઉપયોગ કરીને પોટ ફેરવીને આપણને મળેલી કિંમતોની શ્રેણી જોઈશું. અમારા કોડમાં, અમે સેટઅપમાં સીરીયલ ઑબ્જેક્ટને પ્રારંભ કરીએ છીએ, બૉડ રેટ ૯૬૦૦ સેટ કરીએ છીએ. લૂપમાં, અમે એનાલોગ પિન a0 માંથી મૂલ્ય વાંચીએ છીએ અને **printLn** ફંક્શનનો ઉપયોગ કરીને તેને સીરીયલ ઑબ્જેક્ટ પર પ્રિન્ટ કરીએ છીએ.



તમે આર્ડિનો પર સ્ક્રિપ્ટ અપલોડ કર્યા પછી, તમે પોટ ચાલુ કરો ત્યારે મૂલ્યો જોવા માટે **સીરીયલ મોનિટર** બટનને ક્લિક કરો. એક વિંડો ખુલશે, અને તમે ૦ થી ૧૦૨૩ સુધીના મૂલ્યો જોશો કારણ કે પોટ ચાલુ છે.



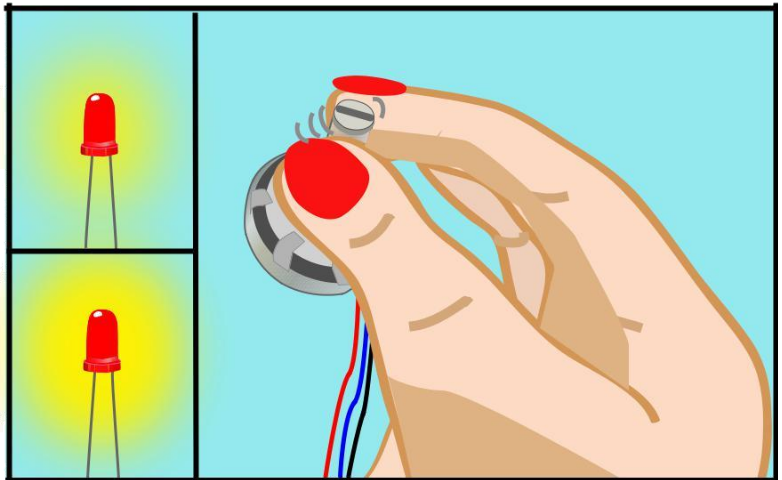
ચાલો એલઇડીને નિયંત્રિત કરવા માટે પોટમાંથી મળેલા બદલાતા મૂલ્યોનો ઉપયોગ કરીએ. પિન 3, કેથોડથી જમીન પર બોર્ડ પર રેઝિસ્ટર દ્વારા એનોડ જોડો.



અમે **પલ્સ વિડ્થ મોડ્યુલેશન** (પી.ડબલ્યુ.એમ) નો ઉપયોગ કરીશું. વોલ્ટેજની હેરફેર કરીને, તેને અલગ-અલગ દરે ચાલુ અને બંધ કરીને અથવા ફરજ ચક્ર દ્વારા એનાલોગ મૂલ્યનું અનુકરણ કરવાની આ પદ્ધતિ છે. તમે પીન 3, ૫, ૬, ૯, ૧૦ અને ૧૧ સાથે પી.ડબલ્યુ.એમ નો ઉપયોગ કરી શકો છો.

```
int sensorValue = 0;

void setup() {
      pinMode(3,OUTPUT);
}

void loop() {
      sensorValue = analogRead(A0);
      analogWrite(3, sensorValue/4);
}
```

પ્રથમ આપણે પોટની કિંમત સ્ટોર કરવા માટે એક ચલ બનાવીએ છીએ. સેટઅપમાં આપણે પિન 3 ને આઉટપુટ બનાવીએ છીએ. લૂપમાં, આપણે આપણા વેરીએબલમાં પિન a0 થી વાંચેલી વેલ્યુ સ્ટોર કરીએ છીએ. પછી અમે પિન 3, અમારી LED પિનનું મૂલ્ય લખીએ છીએ. આપણે વેરીએબલને ૪ વડે વિભાજિત કરવું પડશે, તેથી આપણી પાસે ૦ થી ૨૫૫ સુધીની કિંમતોની શ્રેણી, અથવા બાઈટ હશે.



જ્યારે તમે પોટ ફેરવો છો ત્યારે એલઇડીની બ્રાઇટનેસ સંપૂર્ણપણે બંધથી લઈને ખૂબ જ તેજસ્વી સુધી બદલાય છે.

લિંક્સ

સોફ્ટવેર

સોફ્ટવેર ડાઉનલોડ
http://www.arduino.cc/en/Main/Software

ભાષા સંદર્ભ
http://arduino.cc/en/Reference/HomePage

પુરવઠો

સ્પાર્કફન ઇલેક્ટ્રોનિક્સ
http://www.sparkfun.com/

અદાફ્રૂટ ઇન્ડસ્ટ્રીઝ
http://adafruit.com/

મેકર શેડ
http://www.makershed.com/

જેમકો ઈલેક્ટ્રોનિક્સ
http://www.jameco.com/

ટ્યુટોરિયલ્સ

આર્ડુઇનો સાઇટ ટ્યુટોરિયલ્સ
http://www.arduino.cc/en/Tutorial/HomePage

લેડી એડા
http://www.ladyada.net/learn/arduino/

સૂચનાઓ
http://www.instructables.com/tag/type-id/category-technology/channel-arduino/

પુસ્તકો

Getting Started with Arduino by Massimo Banzi
Making Things Talk: Using Sensors, Networks, and Arduino to See, Hear, and Feel Your World by Tom Igoe
Physical Computing: Sensing and Controlling the Physical World with Computers by Dan O'Sullivan & Tom Igoe
Arduino Cookbook by Michael Margolis

વધુ માટે **Jody Culkin** દ્વારા તમામ ટેક્સ્ટ અને રેખાંકનો, તપાસો jodyculkin.com

ગુજરાતી લિપિમાં રૂપાંતરણની સમીક્ષા કરવા બદલ UniConverge Technologies Pvt Ltd ની IoT ટીમનો વિશેષ આભાર. ખાસ આભાર Tom Igoe, Marianne Petit, Calvin Reid, NYU ખાતે ઇન્ટરેક્ટિવ ટેલિકોમ્યુનિકેશન્સ પ્રોગ્રામના ફેકલ્ટી અને સ્ટાફ ને, ખાસ કરીને Dan O'Sullivan, Danny Rozin અને Red Burns. આભાર Cindy Karasek, Chris Stein, Sarah Teitler, Kathy Goncharov અને Zannah Marsh.

આર્ડુઇનો ટીમનો ખૂબ ખૂબ આભાર અમને આ મજબૂત અને લવચીક ઓપન સોર્સ પ્લેટફોર્મ લાવવા બદલ.
અને જીવંત, સક્રિય અને સતત વિકસતા આર્ડુઇનો સમુદાય માટે આભાર.

Jody Culkin દ્વારા આર્ડુઇનો પરિચય ક્રિએટિવ કોમન્સ એટ્રિબ્યુશન-નોન કોમર્શિયલ-શેરએલાઈક ૩.૦ અનપોર્ટેડ લાયસન્સ હેઠળ લાઇસન્સ પ્રાપ્ત છે.