LAMPIRAN

Skrip program mesin pembuat minum otomatis

```
#include <Keypad.h>
#include <EEPROM.h>

#include <LiquidCrystal.h>

// initialize the library with the numbers of the interface pins
LiquidCrystal lcd(46, 47, 48, 49, 50, 51);

const byte ROWS = 4; // Four rows
const byte COLS = 4; // Three columns
// Define the Keymap
char keys[ROWS][COLS] = {
{'1','2','3','A'},
{'4','5','6','B'},
{'7','8','9','C'},
{'*','0','#','D'}};
// Connect keypad ROW0, ROW1, ROW2 and ROW3 to these Arduino pins.
byte rowPins[ROWS] = { 11, 10, 9, 8 };
// Connect keypad COL0, COL1 and COL2 to these Arduino pins.
byte colPins[COLS] = { 7, 6, 5, 4 };
// Create the Keypad
Keypad kpd = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );

#define sol_air   29
```

```
#define sol_gula  28
#define sol_kopi  27
#define sol_teh   26
#define sol_susu  25
#define mixer     24
#define peltier   23
#define heater    22
#define sol_kuras 30
#define sol_gelas 31
#define pom_mix   32
volatile int flow_air, flow_gula, flow_kopi, flow_teh, flow_susu, flow_gelas;
int gelas,gula,mix;
void setup(){
  pinMode(sol_air,OUTPUT);      digitalWrite(sol_air, HIGH);
  pinMode(sol_gula,OUTPUT);     digitalWrite(sol_gula, HIGH);
  pinMode(sol_kopi,OUTPUT);     digitalWrite(sol_kopi, HIGH);
  pinMode(sol_teh,OUTPUT);      digitalWrite(sol_teh, HIGH);
  pinMode(sol_susu,OUTPUT);     digitalWrite(sol_susu, HIGH);
  pinMode(mixer,OUTPUT);        digitalWrite(mixer, HIGH);
  pinMode(heater,OUTPUT);       digitalWrite(heater, HIGH);
  pinMode(peltier,OUTPUT);      digitalWrite(peltier, HIGH);
  pinMode(sol_kuras,OUTPUT);    digitalWrite(sol_kuras, HIGH);
  pinMode(sol_gelas,OUTPUT);    digitalWrite(sol_gelas, HIGH);
  pinMode(pom_mix,OUTPUT);      digitalWrite(pom_mix, HIGH);
  lcd.begin(20,4);
flow_air=flow_gula=flow_kopi=flow_teh=flow_susu=flow_gelas=0;
```

```
gula=0;
gelas=0;
mix=0;
detachInterrupt(0);
detachInterrupt(1);
detachInterrupt(2);
detachInterrupt(3);
detachInterrupt(4);
detachInterrupt(5);}
void loop(){
lcd.setCursor(0,1);
lcd.print("Ukuran=");

if      (gelas==150)  {lcd.setCursor(7,1); lcd.print("150 ml");}
else if (gelas==195)  {lcd.setCursor(7,1); lcd.print("195 ml");}
else if (gelas==280)  {lcd.setCursor(7,1); lcd.print("280 ml");}
else if (gelas>=280)  {gelas=280;}
else if (gelas<=150)  {gelas=150;}
lcd.setCursor(0,2);
lcd.print("Gula  =");
if      (gula==0)  {lcd.setCursor(7,2); lcd.print("0 ");}
else if (gula==5)  {lcd.setCursor(7,2); lcd.print("5 ");}
else if (gula==10) {lcd.setCursor(7,2); lcd.print("10");}
else if (gula==15) {lcd.setCursor(7,2); lcd.print("15");}
else if (gula==20) {lcd.setCursor(7,2); lcd.print("20");}
else if (gula>=20) {gula=20;}
```

```
else if (gula<=0)  {gula=0; }

EEPROM.read(0);
EEPROM.read(1);

if      (EEPROM.read(0)==1) {lcd.setCursor(5,0);lcd.print("Kopi");}
else if (EEPROM.read(0)==2) {lcd.setCursor(6,0);lcd.print("Teh");}
else if (EEPROM.read(0)==3) {lcd.setCursor(5,0);lcd.print("Susu");}
else if (EEPROM.read(0)==4) {lcd.setCursor(0,0);lcd.print("Kopi Susu");}
else if (EEPROM.read(0)==5) {lcd.setCursor(1,0);lcd.print("Teh Susu");}

if      (EEPROM.read(1)==1) {lcd.setCursor(10,0);lcd.print("Panas");}
else if (EEPROM.read(1)==2) {lcd.setCursor(10,0);lcd.print("Dingin");}
lcd.setCursor(14,3);
lcd.print("0=Help");
if  (digitalRead(33)==LOW)  {lcd.setCursor(0,3); lcd.print("*=Selesai     ");}
else                        {lcd.setCursor(0,3); lcd.print("Tdk Ada Gelas ");}
char key = kpd.getKey();
lcd.setCursor(19,0);
lcd.print(key);
if(key)  // Check for a valid key.{
switch (key){
case '1': EEPROM.put(0,1);  break;
case '2': EEPROM.put(0,2);  break;
case '3': EEPROM.put(0,3);  break;
case '4': EEPROM.put(0,4);  break;
```

```
case '5': EEPROM.put(0,5);  break;

case '6': setup();          break;

case '7': digitalWrite(pom_mix, LOW);digitalWrite(sol_gelas,
LOW);delay(5000);   break;

case '8': kuras();          break;

case '9': EEPROM.put(1,1);  break;

case '0': help();  break;

case '#': EEPROM.put(1,2);  break;

case 'A': gelas+=15;    break;

case 'B': gelas-=15;    break;

case 'C': gula+=5;      break;

case 'D': gula-=5;      break;

case '*':

if      (EEPROM.read(0)==1 && gelas!=0  &&
digitalRead(33)==LOW) {lcd.clear();

lcd.setCursor(0,0);lcd.print("
Mohon Menunggu   ");

lcd.setCursor(0,1);lcd.print("
Kopi        ");

lcd.setCursor(0,2);lcd.print("
Sedang Dibuat    ");

buat_kopi();}

else if (EEPROM.read(0)==2 && gelas!=0  &&
digitalRead(33)==LOW) {lcd.clear();

lcd.setCursor(0,0);lcd.print("
Mohon Menunggu   ");

lcd.setCursor(0,1);lcd.print("
Teh         ");
```

```
                                    lcd.setCursor(0,2);lcd.print("
        Sedang Dibuat    ");

                                        buat_teh();}

    else if (EEPROM.read(0)==3 && gelas!=0  &&
digitalRead(33)==LOW) {lcd.clear();

                                    lcd.setCursor(0,0);lcd.print("
        Mohon Menunggu   ");

                                    lcd.setCursor(0,1);lcd.print("
            Susu        ");

                                    lcd.setCursor(0,2);lcd.print("
        Sedang Dibuat    ");

                                        buat_susu();}

    else if (EEPROM.read(0)==4 && gelas!=0  &&
digitalRead(33)==LOW) {lcd.clear();

                                    lcd.setCursor(0,0);lcd.print("
        Mohon Menunggu   ");

                                    lcd.setCursor(0,1);lcd.print("
          Kopi Susu      ");

                                    lcd.setCursor(0,2);lcd.print("
        Sedang Dibuat    ");

                                        buat_kopisusu();}

    else if (EEPROM.read(0)==5 && gelas!=0  &&
digitalRead(33)==LOW) {lcd.clear();

                                    lcd.setCursor(0,0);lcd.print("
        Mohon Menunggu   ");

                                    lcd.setCursor(0,1);lcd.print("
            Teh Susu        ");

                                    lcd.setCursor(0,2);lcd.print("
        Sedang Dibuat    ");

                                        buat_tehsusu();}
      else{}break;    } }else  {
```

```
digitalWrite(sol_air, HIGH);
digitalWrite(sol_gula, HIGH);
digitalWrite(sol_kopi, HIGH);
digitalWrite(sol_teh, HIGH);
digitalWrite(sol_susu, HIGH);
digitalWrite(mixer, HIGH);
digitalWrite(heater, HIGH);
digitalWrite(peltier, HIGH);
digitalWrite(sol_kuras, HIGH);
digitalWrite(sol_gelas, HIGH);
digitalWrite(pom_mix, HIGH);
}delay(100);
lcd.clear(); }
void count_air()    {flow_air++;}
void count_gula()   {flow_gula++;}
void count_kopi()   {flow_kopi++;}
void count_teh()    {flow_teh++;}
void count_susu()   {flow_susu++;}
void count_gelas()  {flow_gelas++;}
void kuras(){
do  {attachInterrupt(0, count_air, FALLING);  digitalWrite(sol_air, LOW);}
while (flow_air<=40);      digitalWrite(sol_air, HIGH);   detachInterrupt(0);
delay(500);

for(mix=0;mix<=50;mix++){
digitalWrite(mixer, LOW);
delay(50);}
```

```
digitalWrite(mixer, HIGH);
mix=0;
for(mix=0;mix<=500;mix++){
digitalWrite(pom_mix, LOW);
digitalWrite(sol_kuras, LOW);
delay(50);  }
digitalWrite(pom_mix, HIGH);
digitalWrite(sol_kuras, HIGH);
mix=0;
flow_air=0;
delay(1000);
}void buat_kopi()
{  kuras();

do  {attachInterrupt(0, count_air, FALLING);   digitalWrite(sol_air, LOW);}
while (flow_air<=60);      digitalWrite(sol_air, HIGH);  detachInterrupt(0);
delay(500);

do  {attachInterrupt(2, count_kopi, FALLING);  digitalWrite(sol_kopi, LOW);}
while (flow_kopi<=30);     digitalWrite(sol_kopi, HIGH); detachInterrupt(2);
delay(500);

do  {attachInterrupt(1, count_gula, FALLING);  digitalWrite(sol_gula, LOW);}
while (flow_gula<=gula);   digitalWrite(sol_gula, HIGH); detachInterrupt(1);
delay(500);

aduk();

}void buat_teh()

{    kuras();
```

```
do  {attachInterrupt(0, count_air, FALLING);   digitalWrite(sol_air, LOW);}
 while (flow_air<=60);      digitalWrite(sol_air, HIGH);  detachInterrupt(0);
                                delay(500);

do  {attachInterrupt(3, count_teh, FALLING);   digitalWrite(sol_teh, LOW);}
 while (flow_teh<=40);      digitalWrite(sol_teh, HIGH);  detachInterrupt(3);
                                delay(500);

do  {attachInterrupt(1, count_gula, FALLING);  digitalWrite(sol_gula, LOW);}
 while (flow_gula<=gula);   digitalWrite(sol_gula, HIGH); detachInterrupt(1);
                                delay(500);

                                aduk();}

                            void buat_susu()

                            {  kuras();

do  {attachInterrupt(0, count_air, FALLING);   digitalWrite(sol_air, LOW);}
 while (flow_air<=60);      digitalWrite(sol_air, HIGH);  detachInterrupt(0);
                                delay(500);

do  {attachInterrupt(4, count_susu, FALLING);  digitalWrite(sol_susu, LOW);}
 while (flow_susu<=40);     digitalWrite(sol_susu, HIGH); detachInterrupt(4);
                                delay(500);

do  {attachInterrupt(1, count_gula, FALLING);  digitalWrite(sol_gula, LOW);}
 while (flow_gula<=gula);   digitalWrite(sol_gula, HIGH); detachInterrupt(1);
                                delay(500);

                                aduk();}

                            void buat_kopisusu(){

                                kuras();

do  {attachInterrupt(0, count_air, FALLING);   digitalWrite(sol_air, LOW);}
 while (flow_air<=50);      digitalWrite(sol_air, HIGH);  detachInterrupt(0);
                                delay(500);

do  {attachInterrupt(2, count_kopi, FALLING);  digitalWrite(sol_kopi, LOW);}
 while (flow_kopi<=30);     digitalWrite(sol_kopi, HIGH); detachInterrupt(2);
                                delay(500);

do  {attachInterrupt(4, count_susu, FALLING);  digitalWrite(sol_susu, LOW);}
 while (flow_susu<=20);     digitalWrite(sol_susu, HIGH); detachInterrupt(4);
                                delay(500);
```

```
do  {attachInterrupt(1, count_gula, FALLING);  digitalWrite(sol_gula, LOW);}
while (flow_gula<=gula);   digitalWrite(sol_gula, HIGH); detachInterrupt(1);
delay(500);

aduk();}

void buat_tehsusu(){

kuras();

do  {attachInterrupt(0, count_air, FALLING);   digitalWrite(sol_air, LOW);}
while (flow_air<=50);      digitalWrite(sol_air, HIGH);  detachInterrupt(0);
delay(500);

do  {attachInterrupt(3, count_teh, FALLING);   digitalWrite(sol_teh, LOW);}
while (flow_teh<=30);      digitalWrite(sol_teh, HIGH);  detachInterrupt(3);
delay(500);

do  {attachInterrupt(4, count_susu, FALLING);  digitalWrite(sol_susu, LOW);}
while (flow_susu<=20);     digitalWrite(sol_susu, HIGH); detachInterrupt(4);
delay(500);

do  {attachInterrupt(1, count_gula, FALLING);  digitalWrite(sol_gula, LOW);}
while (flow_gula<=gula);   digitalWrite(sol_gula, HIGH); detachInterrupt(1);
delay(500);

aduk();}

void aduk(){

for(mix=0;mix<=100;mix++){

digitalWrite(mixer, LOW);

digitalWrite(pom_mix, HIGH);

digitalWrite(sol_gelas, HIGH);

delay(30);}

mix=0;

digitalWrite(mixer, HIGH);

for(mix=0;mix<=1000;mix++)

{ if     (EEPROM.read(1)==1)   {digitalWrite(heater, LOW);
digitalWrite(mixer, HIGH);}
```

```
else if (EEPROM.read(1)==2)   {digitalWrite(peltier, LOW);
digitalWrite(mixer, HIGH);}

lcd.setCursor(0,3);lcd.print("  ..............  "); delay(60);

lcd.setCursor(0,3);lcd.print("                    "); delay(60); }

digitalWrite(mixer,   HIGH);

digitalWrite(heater,  HIGH);

digitalWrite(peltier, HIGH);

delay(1000);

do  {

attachInterrupt(5, count_gelas, FALLING);

digitalWrite(pom_mix, LOW);

digitalWrite(sol_gelas, LOW);     }

while(flow_gelas<=gelas); digitalWrite(sol_gelas,
HIGH);digitalWrite(pom_mix, HIGH); detachInterrupt(5);    delay(500);

setup();}

void help(){

lcd.clear();

lcd.setCursor(0,0);

lcd.print("1.Kopi   4.Kopi Susu");

lcd.setCursor(0,1);

lcd.print("2.Teh    5.Teh Susu ");

lcd.setCursor(0,2);

lcd.print("3.Susu");

lcd.setCursor(0,3);

lcd.print("A.+Volume  B.-Volume");

delay(4000);
```

```
lcd.clear();
lcd.setCursor(0,1);
lcd.print("C.+Gula      D.-Gula");
lcd.setCursor(0,2);
lcd.print("9.Panas     #.Dingin");
delay(4000);
setup();}
```

Skrip program pengujian Flow meter

```
volatile int flow_air;

#include <LiquidCrystal.h>

// initialize the library with the numbers of the interface pins
LiquidCrystal lcd(46, 47, 48, 49, 50, 51);

void setup() {
// put your setup code here, to run once:

detachInterrupt(0);}
void loop() {
// put your main code here, to run repeatedly:

attachInterrupt(0,count_air, FALLING);

lcd.setCursor(0,0);
lcd.print(flow_air);}
void count_air()    {flow_air++;}
```

Foto – foto mesin pembuat minuman otomatis

Foto tampak depan mesin pembuat minuman otomatis

Foto tampak LCD dan Key Pad mesin pembuat minuman otomatis

Foto tampak atas mesin pembuat minuman otomatis

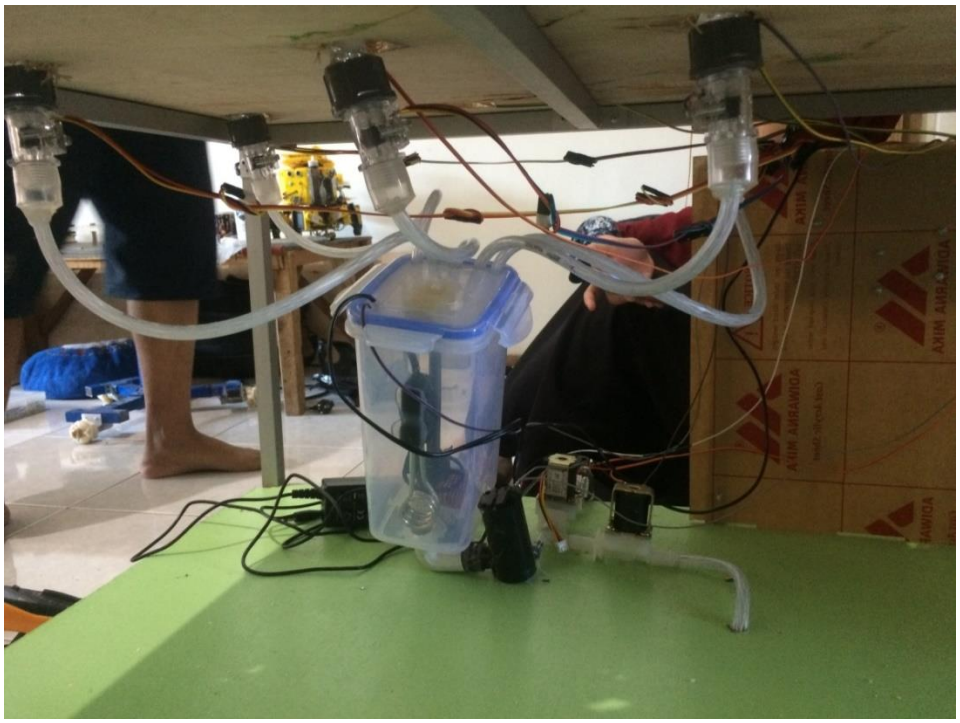
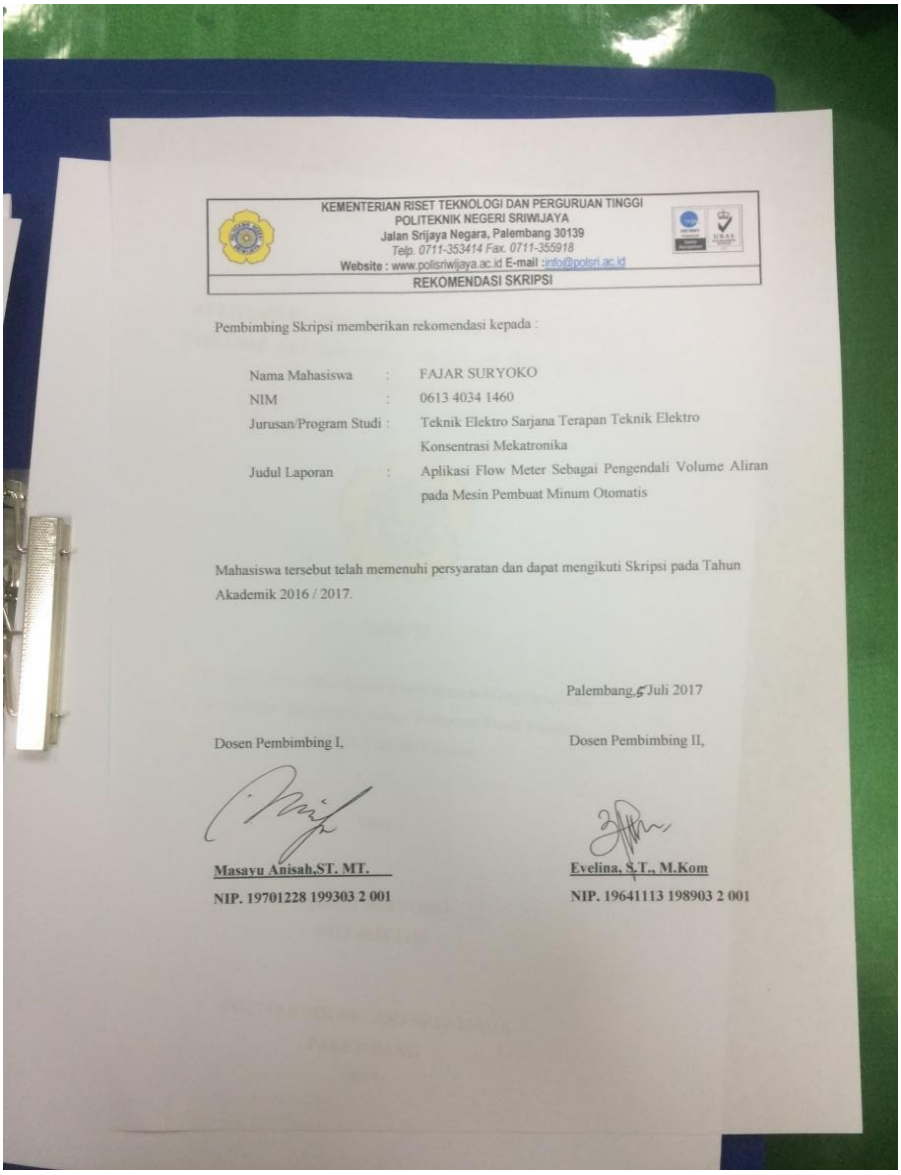


Foto tampak dalam mesin pembuat minuman otomatis

Rekomendasi Ujian Skripsi



KEMENTERIAN RISET TEKNOLOGI DAN PERGURUAN TINGGI
POLITEKNIK NEGERI SRIWIJAYA
Jalan Srijaya Negara, Palembang 30139
Telp. 0711-353414 Fax. 0711-355918
Website : www.polisriwijaya.ac.id E-mail :info@polsri.ac.id

REKOMENDASI SKRIPSI

Pembimbing Skripsi memberikan rekomendasi kepada :

Nama Mahasiswa	:	FAJAR SURYOKO
NIM	:	0613 4034 1460
Jurusan/Program Studi	:	Teknik Elektro Sarjana Terapan Teknik Elektro Konsentrasi Mekatronika
Judul Laporan	:	Aplikasi Flow Meter Sebagai Pengendali Volume Aliran pada Mesin Pembuat Minum Otomatis

Mahasiswa tersebut telah memenuhi persyaratan dan dapat mengikuti Skripsi pada Tahun Akademik 2016 / 2017.

Palembang, 5 Juli 2017

Dosen Pembimbing I,

Masayu Anisah,ST. MT.
NIP. 19701228 199303 2 001

Dosen Pembimbing II,

Evelina, S.T., M.Kom
NIP. 19641113 198903 2 001

Data Sheet/ specification

Search document 1 of 1

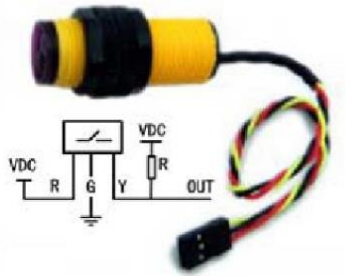
E18-D80NK-N

Adjustable Infrared Sensor Switch Manual

61mcu.Com 北京亿学通电子

Introduction

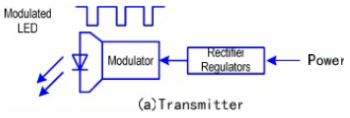
This is an infrared distance switch. It has an adjustable detection range, 3cm - 80cm. It is small, easy to use/assemble, inexpensive. Useful for robot, interactive media, industrial assembly line, etc.



Search document 1 of 1

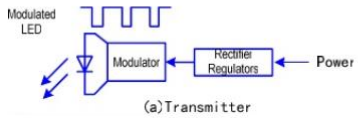
Specification

Model NO: E18-D80NK-N	Diameter: 18mm, Length: 45mm
Sensing range: 3-80cm adjustable	Appearance: Threaded cylindrical
Sensing object: Translucency, opaque	Material: Plastic
Supply voltage: DC5V	Guard mode: Reverse polarity protection
Load current: 100mA	Ambient temperature: -25-70℃
Output operation: Normally open(O)	Red: +5V; Yellow:Signal;Green:GND
Output: DC three-wire system(NPN)	

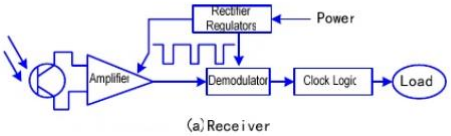


(a)Transmitter

Search document 1 of 1



(a)Transmitter



(a)Receiver

北京亿学通电子

Tel: 010-62669059
Website: www.61mcu.com
E-mail: fae_61mcu@163.com

G1/2" Water Flow Sensor Enclosure

Water flow sensor consists of a plastic valve body, a water rotor, and a hall-effect sensor. When water flows through the rotor, rotor rolls. Its speed changes with different rate of flow. The hall-effect sensor outputs the corresponding pulse signal. This one is suitable to detect flow in water dispenser or coffee machine.

We have a comprehensive line of water flow sensors in different diameters. Check them out to find the one that meets your need most.

Feature:

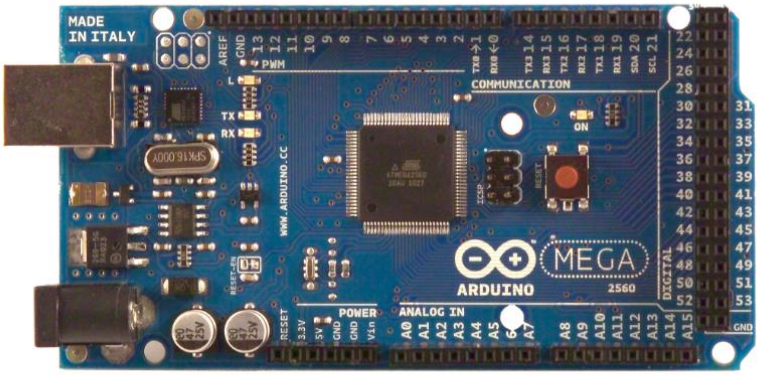
- Compact, Easy to Install
- High Sealing Performance
- High Quality Hall Effect Sensor
- RoHS Compliant
- Transparent

Specification ：

- Mini. Wokring Voltage: DC 4.5V
- Max. Working Current: 15mA (DC 5V)
- Working Voltage: DC 5V~24V
- Flow Rate Range: 1~30L/min

- Flow Pulse: F(Hz)=(5.0*Q)±3% Q=L/Min
- Load Capacity: ≤10mA (DC 5V)
- Operating Temperature: ≤80℃
- Liquid Temperature: ≤120℃
- Operating Humidity: 35%～90%RH
- Water Pressure: ≤1.75MPa
- Storage Temperature: -25～+ 80℃
- Storage Humidity: 25%～95%RH

Arduino MEGA 2560



CE

Technical Specification

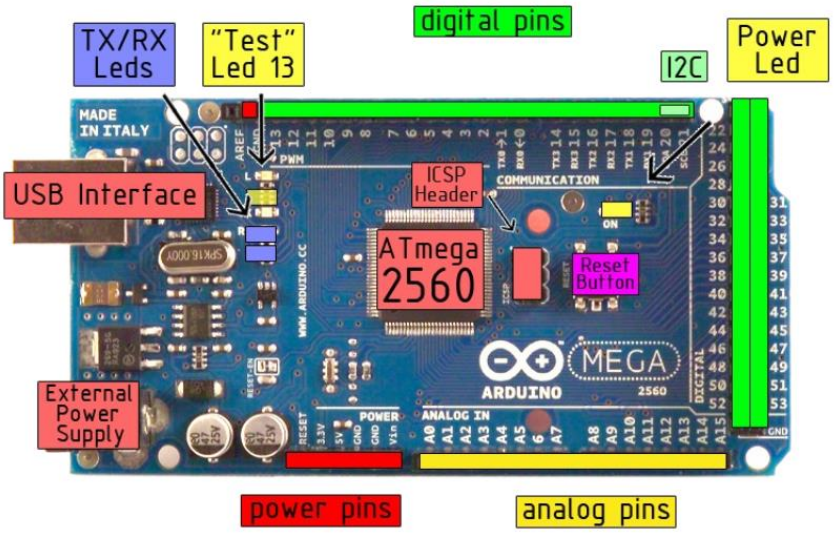
EAGLE files: arduino-mega2560-reference-design.zip Schematic: arduino-mega2560-schematic.pdf

Summary

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 14 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

the board

the board



Automatic (Software) Reset

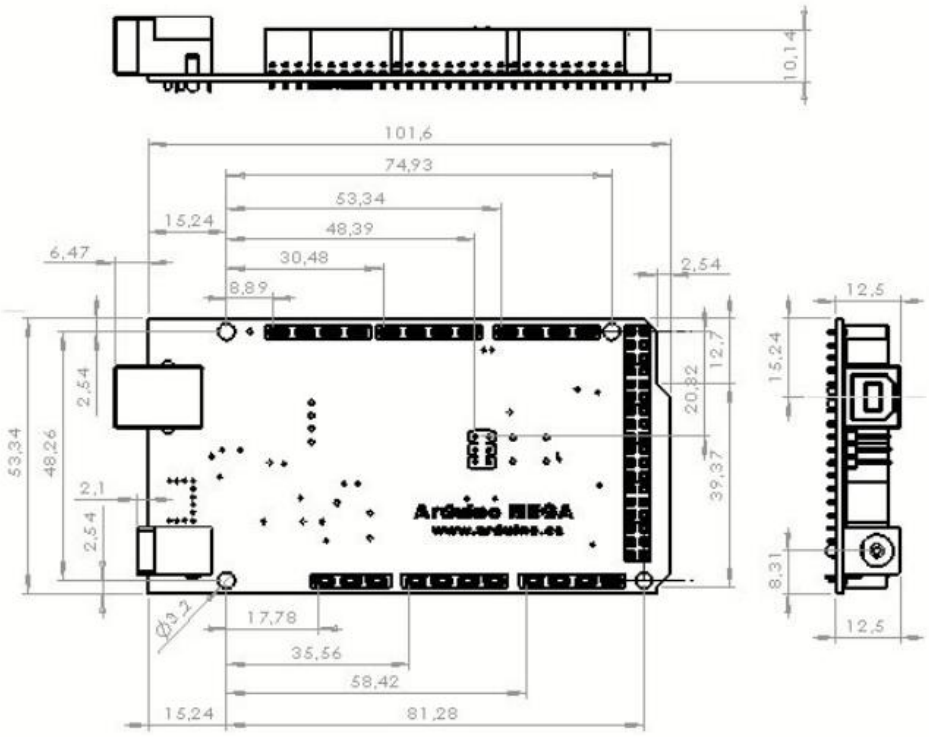
Rather then requiring a physical press of the reset button before an upload, the Arduino Mega2560 is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2 is connected to the reset line of the ATmega2560 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

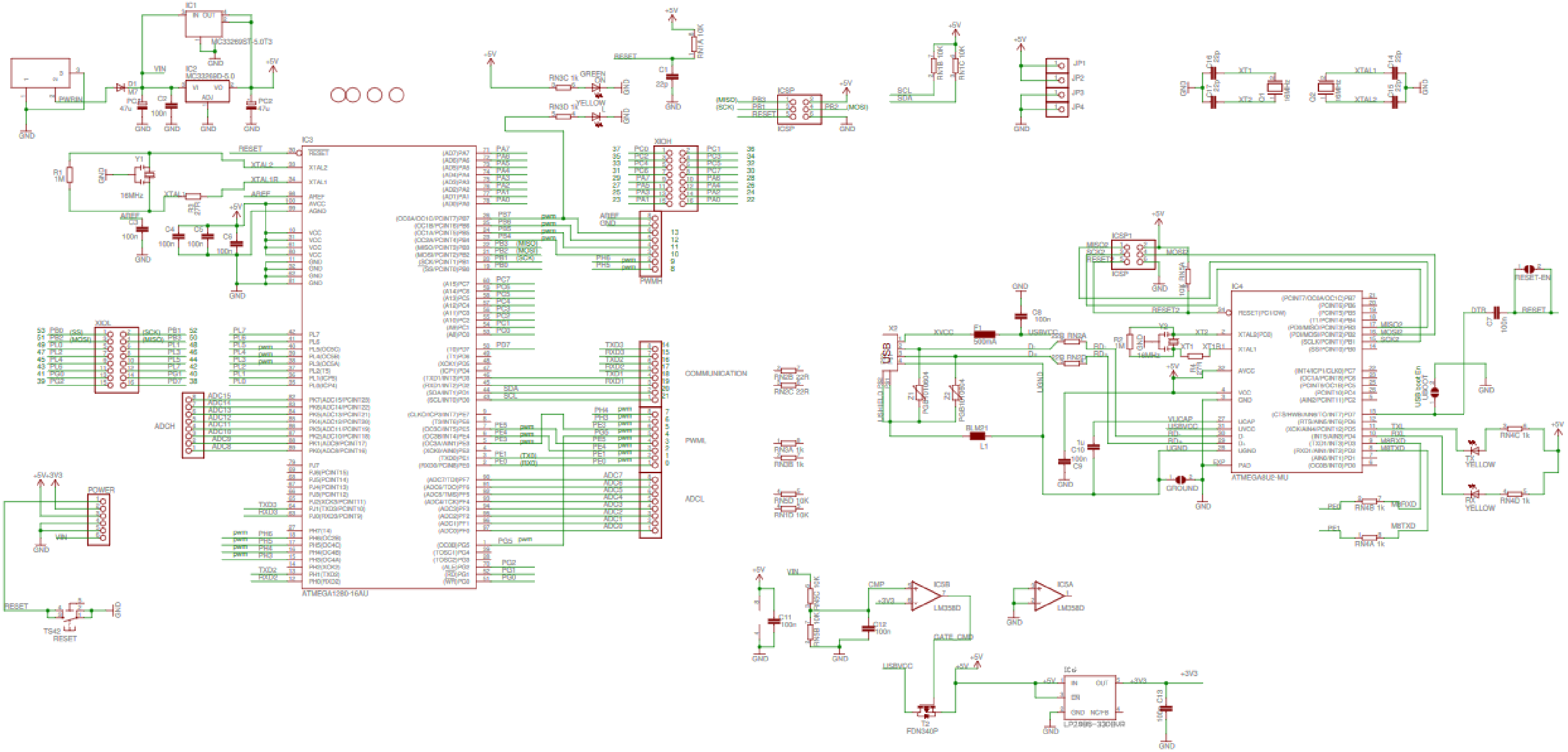
This setup has other implications. When the Mega2560 is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Mega2560. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

The Mega contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see this forum thread for details.

USB Overcurrent Protection

The Arduino Mega has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.





ATMEGA1280-16AU
ATMEGA8U2-MU
COMMUNICATION
PWML
ADCL
ADCH
POWER
LM358D
FDN340P