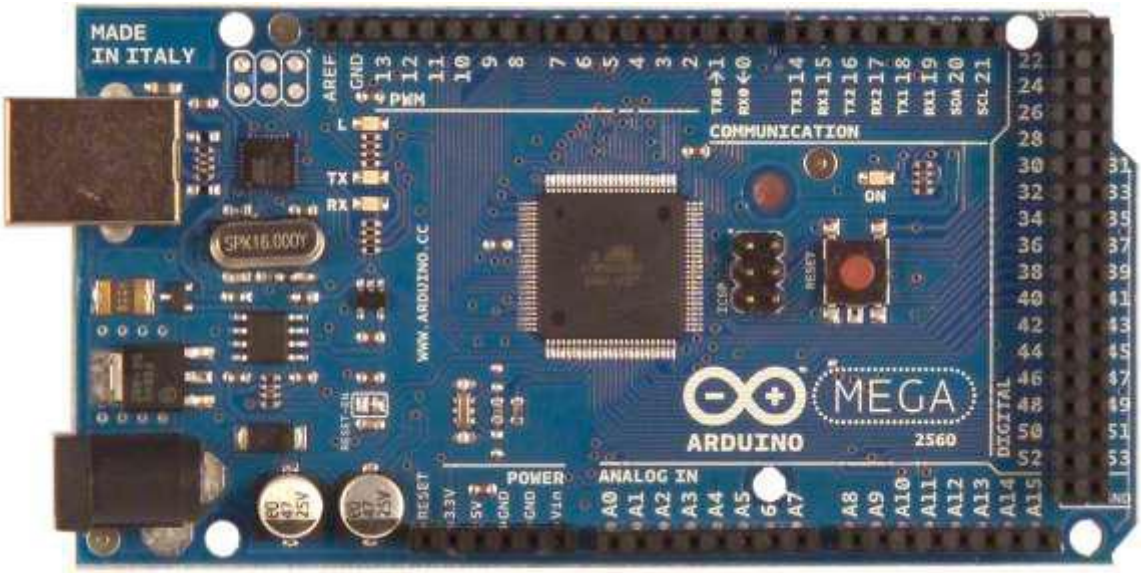
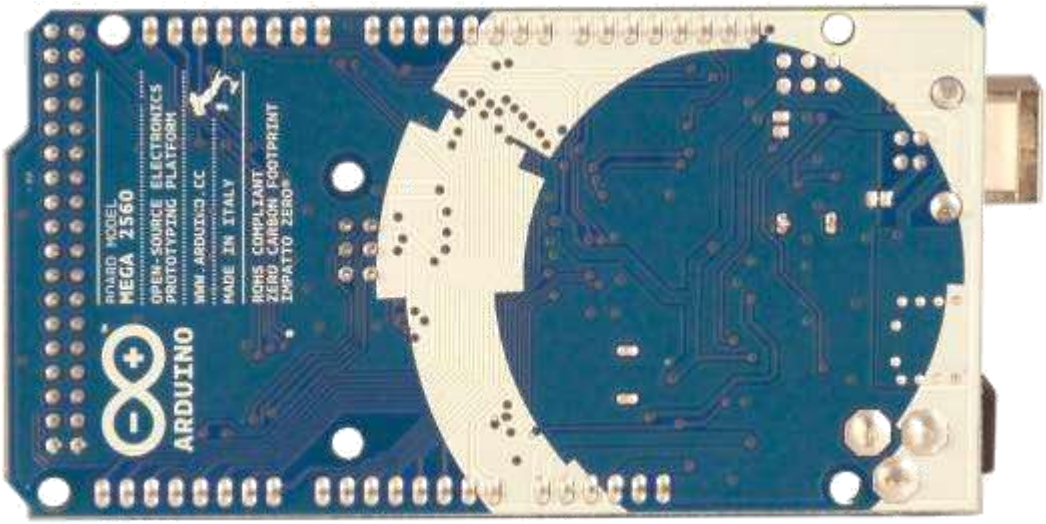
LAMPIRAN A

ARDUINO

Arduino Mega 2560 Datasheet





Overview

The Arduino Mega 2560 is a microcontroller board based on the ATmega2560 (datasheet). It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Mega is compatible with most shields designed for the Arduino Duemilanove or Diecimila.

Schematic & Reference Design

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

Power

The Arduino Mega can be powered via the USB connection or with an external power supply. The power source is selected automatically. External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers ofthe POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The Mega2560 differs from all preceding boards in that it does not use the FTDI USB-to- serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, accessit through this pin.
- **5V.** The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

Memory

The ATmega2560 has 256 KB of flash memory for storing code (of which 8 KB is used for the bootloader), 8 KB of SRAM and 4 KB of EEPROM (which can be read and written with theEEPROM library).

Input and Output

Each of the 54 digital pins on the Mega can be used as an input or output, using pinMode() , digitalWrite(), and digitalRead() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX); Serial 1: 19 (RX) and 18 (TX); Serial 2: 17 (RX) and 16 (TX); Serial 3: 15 (RX) and 14 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. Pins 0 and 1 are also connected to the corresponding pins of theATmega8U2 USB-to-TTL Serial chip.
- **External Interrupts: 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), and 21 (interrupt 2).** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a changein value. See the attachInterrupt() function for details.
- **PWM: 0 to 13.** Provide 8-bit PWM output with the analogWrite() function.
- **SPI: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS).** These pins support SPI communication using the SPI library. The SPI pins are also broken out on the ICSP header, which is physically compatible with the Uno, Duemilanove and Diecimila.
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH

value, the LED is on, when the pin is LOW, it's off.

- **I2C: 20 (SDA) and 21 (SCL).** Support I2C (TWI) communication using the Wire library (documentation on the Wiring website). Note that these pins are not in thesame location as the I2C pins on the Duemilanove or Diecimila.

The Mega2560 has 16 analog inputs, each of which provide 10 bits of resolution (i.e. 1024different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and analogReference() function.
There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with analogReference().
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

Communication

The Arduino Mega2560 has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega2560 provides four hardware UARTs for TTL (5V) serial communication. An ATmega8U2 on the board channels one of these
over USB and provides a virtual com port to software on the computer (Windows machines will need a .inf file, but OSX and Linux machines will recognize the board as a COM port automatically. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the ATmega8U2 chip and USB connection to the computer (butnot for serial communication on pins 0 and 1).
A SoftwareSerial library allows for serial communication on any of the Mega2560's digitalpins.
The ATmega2560 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the documentation on the Wiring website for details. For SPI communication, use the SPI library.

Programming

The Arduino Mega can be programmed with the Arduino software (download). For details, see the reference and tutorials.
The ATmega2560 on the Arduino Mega comes preburned with a bootloader that allowsyou to upload new code to it without the use of an external hardware programmer. It

communicates using the original STK500 protocol (reference, C header files). You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see these instructions for details.

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 Arduinosoftware 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. Whileit 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 afteropening the connection and before sending this data.

The Mega2560 contains a trace that can be cut to disable the auto-reset. The pads on eitherside 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 Mega2560 has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal

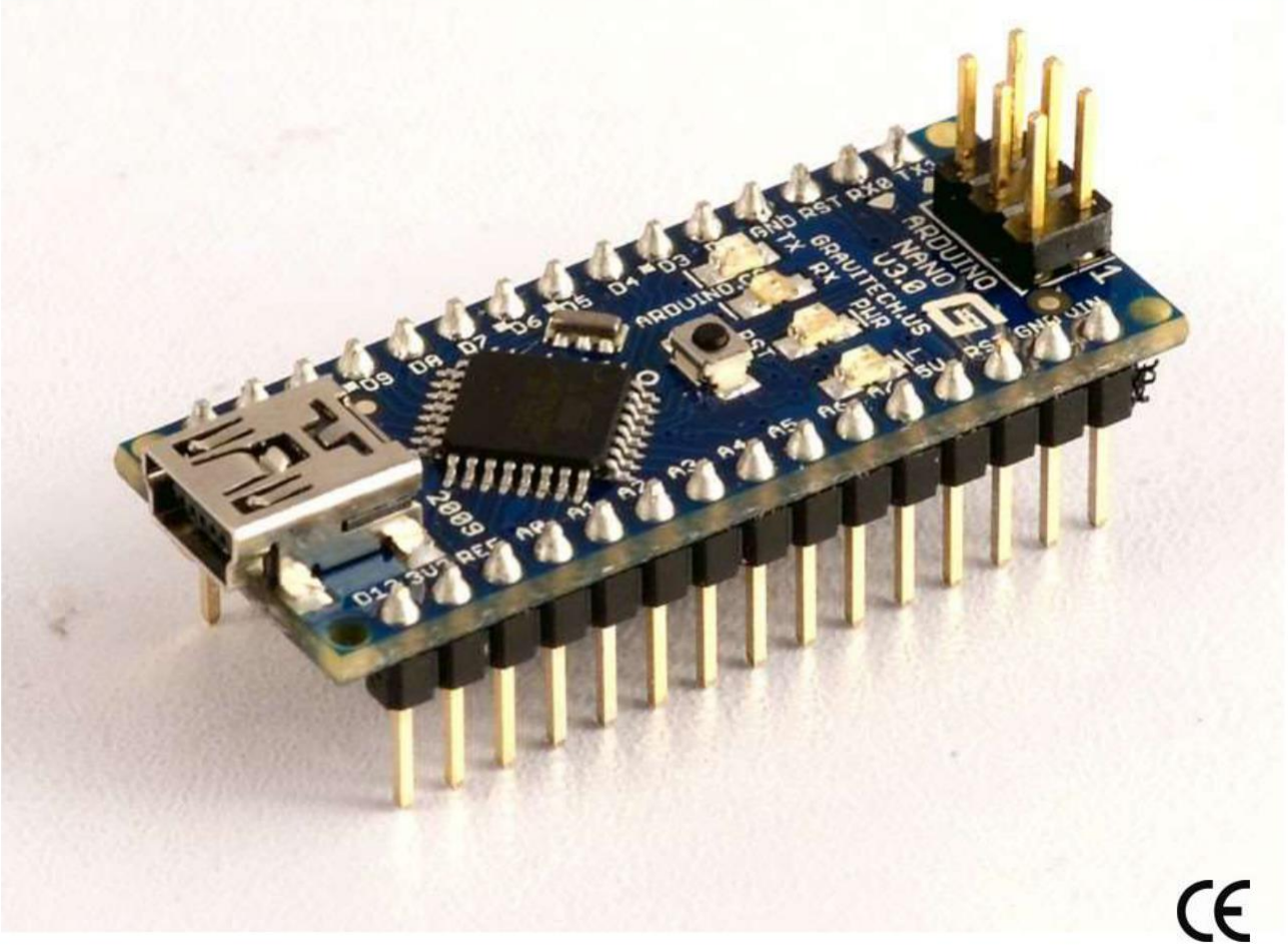
protection, the fuse provides an extra layer of protection. If more than 500 mA is applied tothe USB port, the fuse will automatically break the connection until the short or overload is removed.

Physical Characteristics and ShieldCompatibility

The maximum length and width of the Mega2560 PCB are 4 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Three screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.

The Mega2560 is designed to be compatible with most shields designed for the Uno, Diecimila or Duemilanove. Digital pins 0 to 13 (and the adjacent AREF and GND pins), analog inputs 0 to 5, the power header, and ICSP header are all in equivalent locations. Further the main UART (serial port) is located on the same pins (0 and 1), as are external interrupts 0 and 1 (pins 2 and 3 respectively). SPI is available through the ICSP header on both the Mega2560 and Duemilanove / Diecimila. *Please note that I^2C is not located on the same pins on the Mega (20 and 21) as the Duemilanove / Diecimila (analog inputs 4 and 5)*

Arduino Nano



Product Overview

The Arduino Nano is a small, complete, and breadboard-friendly board based on the ATmega328 (Arduino Nano 3.0) or ATmega168 (Arduino Nano 2.x). It has more or less the same functionality of the Arduino Duemilanove, but in a different package. It lacks only a DC power jack, and works with a Mini-B USB cable instead of a standard one. The Nano was designed and is being produced by Gravitech.

Index

Technical Specification

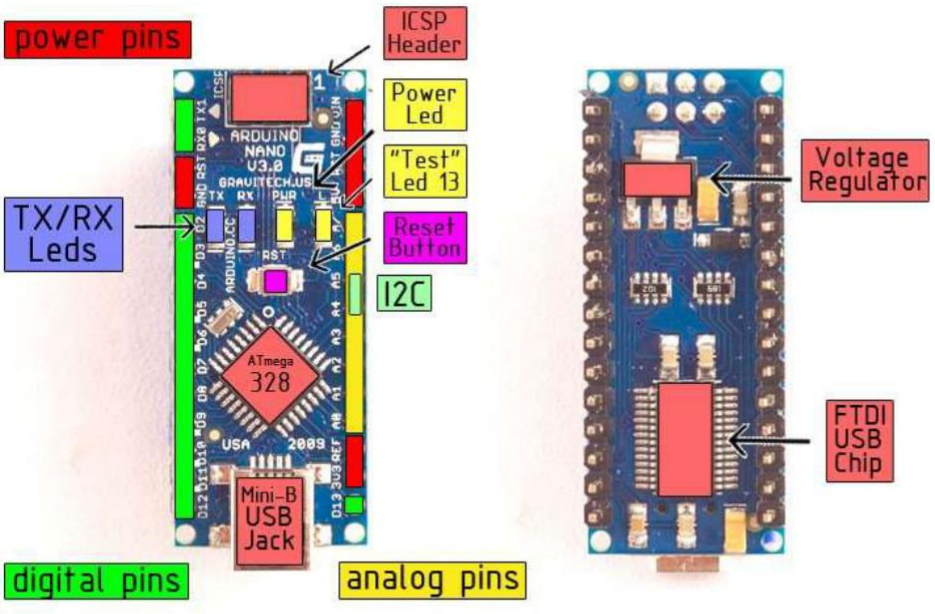
Arduino Nano 3.0 (ATmega328): schematic, Eagle files.

Arduino Nano 2.3 (ATmega168): manual (pdf), Eagle files. *Note:* since the free version of Eagle does not handle more than 2 layers, and this version of the Nano is 4 layers, it is published here unrouted, so users can open and use it in the free version of Eagle.

Summary

Microcontroller	Atmel ATmega168 or ATmega328
Operating Voltage (logic level)	5 V
Input Voltage (recommended)	7-12 V
Input Voltage (limits)	6-20 V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	8
DC Current per I/O Pin	40 mA
Flash Memory	16 KB (ATmega168) or 32 KB (ATmega328) of which 2 KB used by bootloader
SRAM	1 KB (ATmega168) or 2 KB (ATmega328)
EEPROM	512 bytes (ATmega168) or 1 KB (ATmega328)
Clock Speed	16 MHz
Dimensions	0.73" x 1.70"

the board



Power

The Arduino Nano can be powered via the Mini-B USB connection, 6-20V unregulated external power supply (pin 30), or 5V regulated external power supply (pin 27). The power source is automatically selected to the highest voltage source.

The FTDI FT232RL chip on the Nano is only powered if the board is being powered over USB. As a result, when running on external (non-USB) power, the 3.3V output (which is supplied by the FTDI chip) is not available and the RX and TX LEDs will flicker if digital pins 0 or 1 are high.

Memory

The ATmega168 has 16 KB of flash memory for storing code (of which 2 KB is used for the bootloader); the ATmega328 has 32 KB, (also with 2 KB used for the bootloader). The ATmega168 has 1 KB of SRAM and 512 bytes of EEPROM (which can be read and written with the EEPROM library); the ATmega328 has 2 KB of SRAM and 1 KB of EEPROM.

Input and Output

Each of the 14 digital pins on the Nano can be used as an input or output, using pinMode(), digitalWrite(), and digitalRead() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the FTDI USB-to-TTL Serial chip.
- **External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attachInterrupt() function for details.
- **PWM: 3, 5, 6, 9, 10, and 11.** Provide 8-bit PWM output with the analogWrite() function.
- **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Nano has 8 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the analogReference() function. Additionally, some pins have specialized functionality:

- **I^2C: 4 (SDA) and 5 (SCL).** Support I^2C (TWI) communication using the Wire library (documentation on the Wiring website).

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with analogReference().
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

See also the mapping between Arduino pins and ATmega168 ports.

Communication

The Arduino Nano has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega168 and ATmega328 provide UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An FTDI FT232RL on the board channels this serial communication over USB and the FTDI drivers (included with the Arduino software) provide a virtual com port to software on the computer. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the FTDI chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A SoftwareSerial library allows for serial communication on any of the Nano's digital pins.

The ATmega168 and ATmega328 also support I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the documentation for details. To use the SPI communication, please see the ATmega168 or ATmega328 datasheet.

Programming

The Arduino Nano can be programmed with the Arduino software (download). Select "Arduino Diecimila, Duemilanove, or Nano w/ ATmega168" or "Arduino Duemilanove or Nano w/ ATmega328" from the **Tools > Board** menu (according to the microcontroller on your board). For details, see the reference and tutorials.

The ATmega168 or ATmega328 on the Arduino Nano comes preburned with a bootloader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol (reference, C header files).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see these instructions for details.

Automatic (Software) Reset

Rather then requiring a physical press of the reset button before an upload, the Arduino Nano 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 FT232RL is connected to the reset line of the ATmega168 or ATmega328 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 Nano 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 Nano. 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.

How to use Arduino

Arduino can sense the environment by receiving input from a variety of sensors and can affect its surroundings by controlling lights, motors, and other actuators. The microcontroller on the board is programmed using the Arduino programming language (based on Wiring) and the Arduino development environment (based on Processing). Arduino projects can be stand-alone or they can communicate with software on running on a computer (e.g. Flash, Processing, MaxMSP).

Arduino is a cross-platoform program. You'll have to follow different instructions for your personal OS. Check on the Arduino site for the latest instructions. *http://arduino.cc/en/Guide/HomePage*

Linux Install

Windows Install

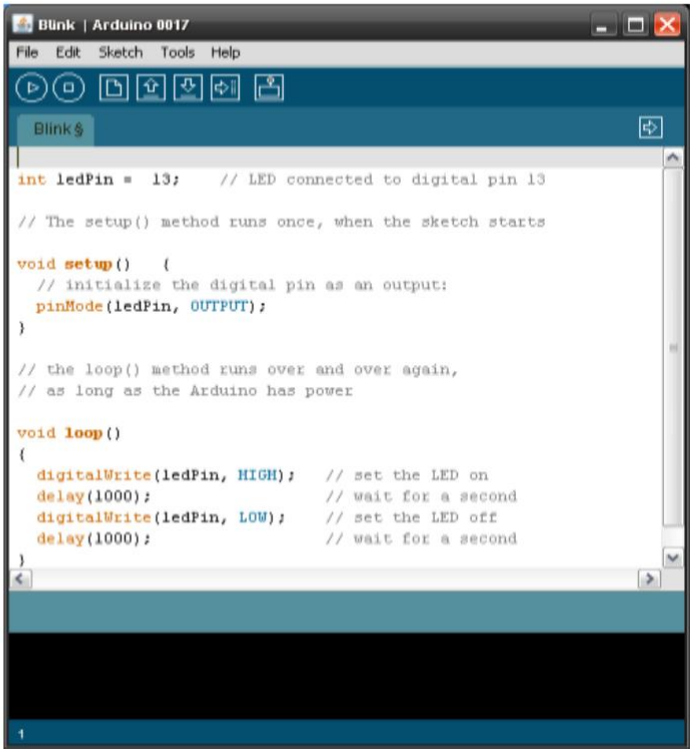
Mac Install

Once you have downloaded/unzipped the arduino IDE, you'll need to install the FTDI Drivers to let your PC talk to the board. First **Plug the Arduino to your PC via USB cable.**

Blink led

Now you're actually ready to "burn" your first program on the arduino board. To select "blink led", the physical translation of the well known programming "hello world", select

File>Sketchbook>
Arduino-0017>Examples>
Digital>Blink



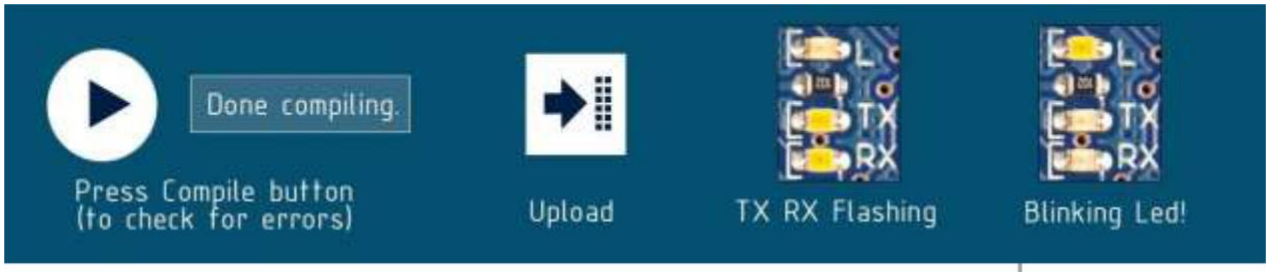
Once you have your skecth you'll see something very close to the screenshot on the right.

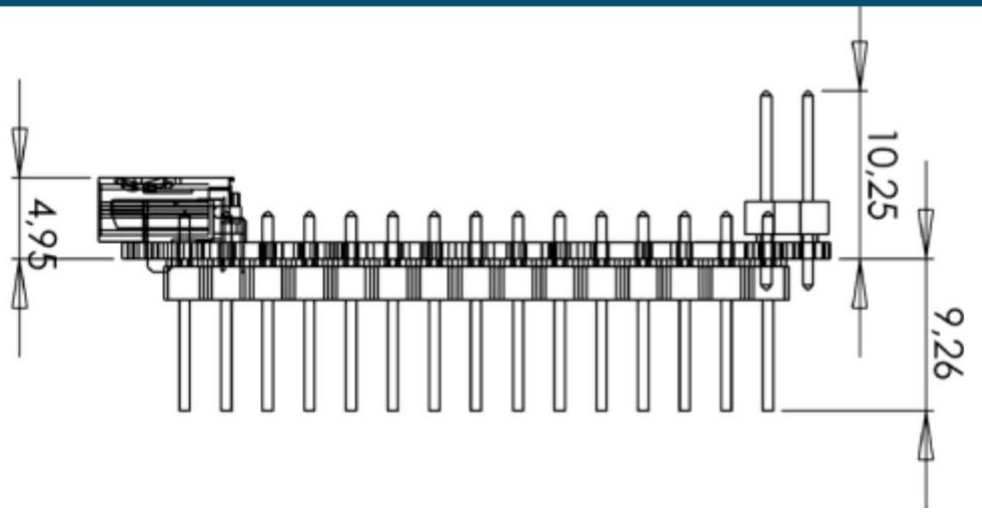
In **Tools>Board** select Arduino NANO and with the AtMEGA you're using (probably 328)

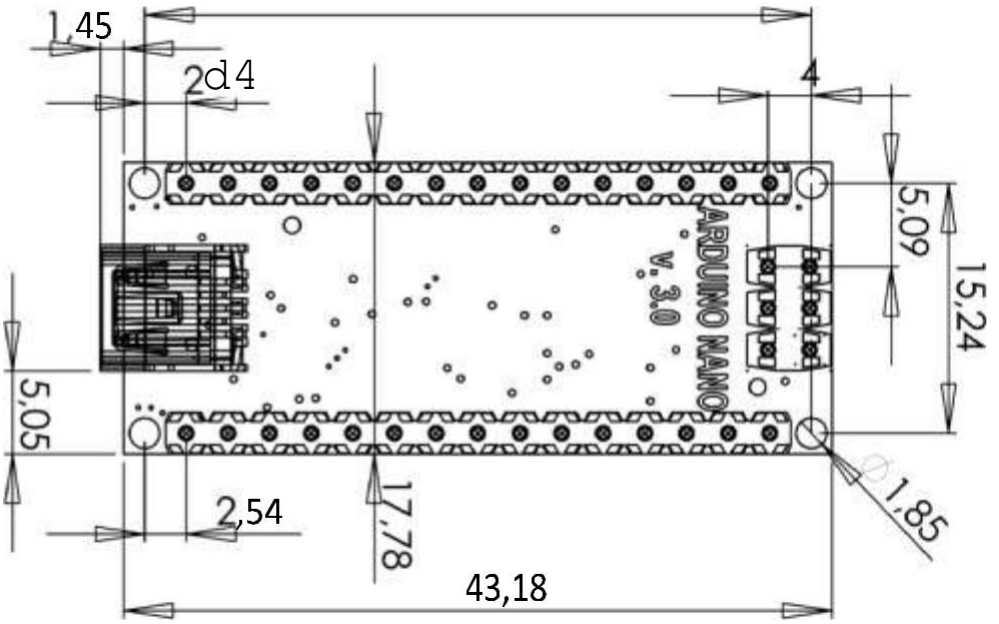
Now you have to go to

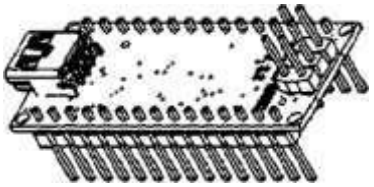
Tools>SerialPort

and select the right serial port, the one arduino is attached to.









Terms & Conditions

1. Warranties

1.1 The producer warrants that its products will conform to the Specifications. This warranty lasts for one (1) years from the date of the sale. The producer shall not be liable for any defects that are caused by neglect, misuse or mistreatment by the Customer, including improper installation or testing, or for any products that have been altered or modified in any way by a Customer. Moreover, The producer shall not be liable for any defects that result from Customer's design, specifications or instructions for such products. Testing and other quality control techniques are used to the extent the producer deems necessary.

1.2 If any products fail to conform to the warranty set forth above, the producer's sole liability shall be to replace such products. The producer's liability shall be limited to products that are determined by the producer not to conform to such warranty. If the producer elects to replace such products, the producer shall have a reasonable time to replacements. Replaced products shall be warranted for a new full warranty period.

1.3 EXCEPT AS SET FORTH ABOVE, PRODUCTS ARE PROVIDED "AS IS" AND "WITH ALL FAULTS." THE PRODUCER DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, REGARDING PRODUCTS, INCLUDING BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE

1.4 Customer agrees that prior to using any systems that include the producer products, Customer will test such systems and the functionality of the products as used in such systems. The producer may provide technical, applications or design advice, quality characterization, reliability data or other services. Customer acknowledges and agrees that providing these services shall not expand or otherwise alter the producer's warranties, as set forth above, and no additional obligations or liabilities shall arise from the producer providing such services.

1.5 The Arduino products are not authorized for use in safety-critical applications where a failure of the product would reasonably be expected to cause severe personal injury or death. Safety-Critical Applications include, without limitation, life support devices and systems, equipment or systems for the operation of nuclear facilities and weapons systems. Arduino products are neither designed nor intended for use in military or aerospace applications or environments and for automotive applications or environment. Customer acknowledges and agrees that any such use of Arduino products which is solely at the Customer's risk, and that Customer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

1.6 Customer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products and any use of Arduino products in Customer's applications, notwithstanding any applications- related information or support that may be provided by the producer.

2. Indemnification

The Customer acknowledges and agrees to defend, indemnify and hold harmless the producer from and against any and all third-party losses, damages, liabilities and expenses it incurs to the extent directly caused by: (i) an actual breach by a Customer of the representation and warranties made under this terms and conditions or (ii) the gross negligence or willful misconduct by the Customer.

3. Consequential Damages Waiver

In no event the producer shall be liable to the Customer or any third parties for any special, collateral, indirect, punitive, incidental, consequential or exemplary damages in connection with or arising out of the products provided hereunder, regardless of whether the producer has been advised of the possibility of such damages. This section will survive the termination of the warranty period.

4. Changes to specifications

The producer may make changes to specifications and product descriptions at any time, without notice. The Customer must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." The producer reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The product information on the Web Site or Materials is subject to change without notice. Do not finalize a design with this information.

Ultrasonic Ranging Module HC - SR04

Product features:

Ultrasonic ranging module HC - SR04 provides 2cm - 400cm non-contact measurement function, the ranging accuracy can reach to 3mm. The modules includes ultrasonic transmitters, receiver and control circuit. The basic principle of work:

(1) Using IO trigger for at least 10us high level signal,
(2) The Module automatically sends eight 40 kHz and detect whether there is a pulse signal back.
(3) IF the signal back, through high level , time of high output IO duration is the time from sending ultrasonic to returning.
Test distance = (high level time×velocity of sound (340M/S) / 2,

Wire connecting direct as following:

5V Supply
Trigger Pulse Input
Echo Pulse Output
0V Ground

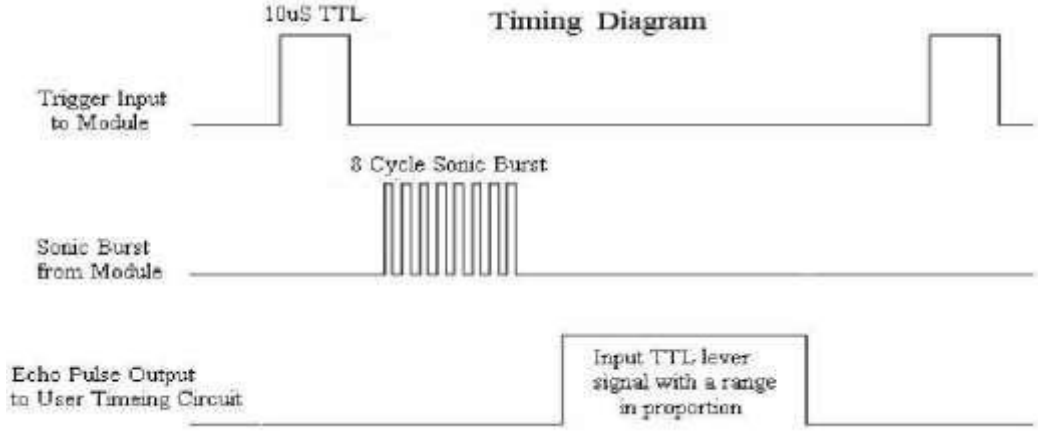
Electric Parameter

Working Voltage	**DC5V**
Working Current	**15mA**
Working Frequency	**40Hz**
Max Range	**4m**
Min Range	**2cm**
MeasuringAngle	**15 degree**
Trigger Input Signal	**10uS TTL pulse**
Echo Output Signal	**Input TTL lever signal and the range in proportion**
Dimension	**45*20*15mm**

Vcc Trig Echo GND

Timing diagram

The Timing diagram is shown below. You only need to supply a short 10uS pulse to the trigger input to start the ranging, and then the module will send out an 8 cycle burst of ultrasound at 40 kHz and raise its echo. The Echo is a distance object that is pulse width and the range in proportion .You can calculate the range through the time interval between sending trigger signal and receiving echo signal. Formula: uS / 58 = centimeters or uS / 148 =inch; or: the range = high level time * velocity (340M/S) / 2; we suggest to use over 60ms measurement cycle, in order to prevent trigger signal to the echo signal.



Attention:

The module is not suggested to connect directly to electric, if connected electric, the GND terminal should be connected the module first, otherwise,
it will affect the normal work of the module.

When tested objects, the range of area is not less than 0.5 square meters and the plane requests as smooth as possible, otherwise ,it will affect the results of measuring.

www.Elecfreaks.com

SX1278 Lora Module

The SX1278 transceivers feature the LoRaTM long range modem that provides ultra-long range spread spectrum communication and high interference immunity whilst minimising current consumption. Using Semtech's patented LoRaTM modulation technique SX1278 can achieve a sensitivity of over -148dBm using a low cost crystal and bill of materials.

The high sensitivity combined with the integrated +20 dBm power amplifier yields industry leading link budget making it optimal for any application requiring range or robustness. LoRaTM also provides significant advantages in both blocking and selectivity over conventional modulation techniques, solving the traditional design compromise between range, interference immunity and energy consumption.

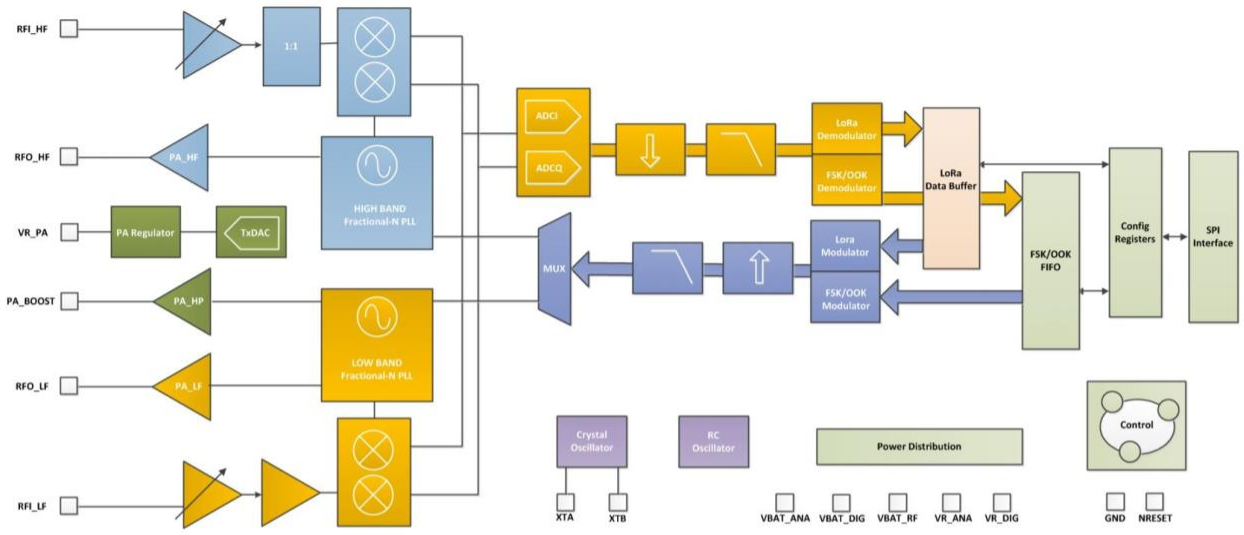
These devices also support high performance (G)FSK modes for systems including WMBus, IEEE802.15.4g. The SX1278 deliver exceptional phase noise, selectivity, receiver linearity and IIP3 for significantly lower current consumption than competing devices.

FEATURES:

- LoRa Modem
- Operating Voltage: 3.3V
- Operating Frequency: 433Mhz
- Half-Duplex SPI communication
- Modulation Technique FSK, GFSK, MSK, GMSK, LORA
- Packet size: 256 bytes
- Sensitivity: -148db

FUNCTIONAL DESCRIPTION:

, This section gives a high-level overview of the functionality of the SX1276/77/78/79 low-power, highly integrated transceiver. The following figure shows a simplified block diagram of the SX1276/77/78/79



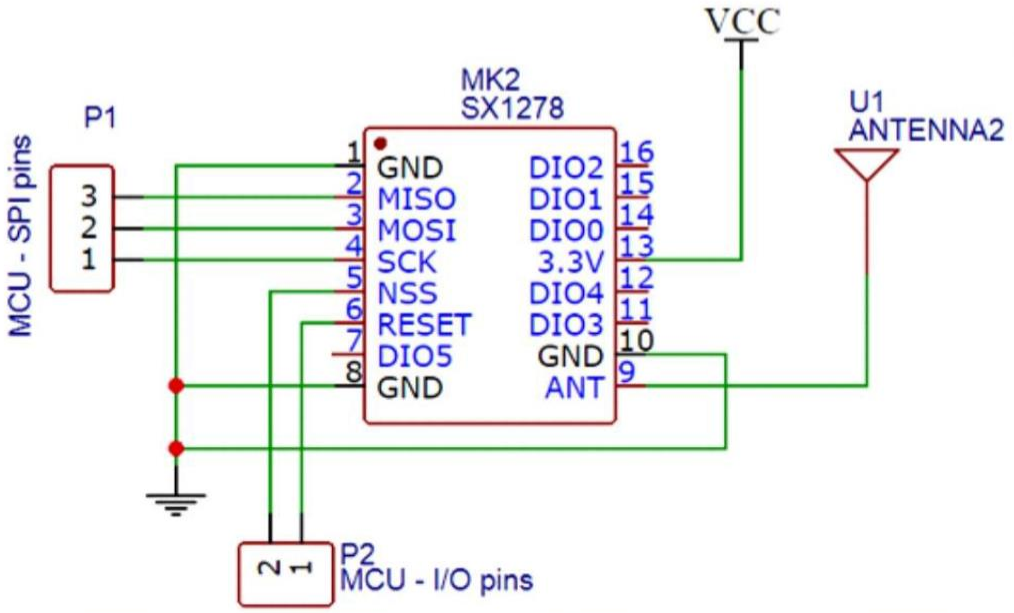
- SX1276/77/78/79 is a half-duplex, low-IF transceiver. Here the received RF signal is first amplified by the LNA. The LNA inputs are single ended to minimize the external BoM and for ease of design. Following the LNA inputs, the conversion to differential is made

to improve the second order linearity and harmonic rejection. The signal is then down-converted to in-phase and quadrature (I&Q) components at the intermediate frequency (IF) by the mixer stage. A pair of sigma delta ADCs then perform data conversion, with all subsequent signal processing and demodulation performed in the digital domain. The digital state machine also controls the automatic frequency correction (AFC), received signal strength indicator (RSSI) and automatic gain control (AGC). It also features the higher-level packet and protocol level functionality of the top level sequencer (TLS), only available with traditional FSK and OOK modulation schemes.

- The frequency synthesizers generate the local oscillator (LO) frequency for both receiver and transmitter, one covering the lower UHF bands (up to 525 MHz), and the other one covering the upper UHF bands (from 779 MHz). The PLLs are optimized for user-transparent low lock time and fast auto-calibrating operation. In transmission, frequency modulation is performed digitally within the PLL bandwidth. The PLL also features optional pre-filtering of the bit stream to improve spectral purity.
- SX1276/77/78/79 feature three distinct RF power amplifiers. Two of those, connected to RFO_LF and RFO_HF, can deliver up to +14 dBm, are unregulated for high power efficiency and can be connected directly to their respective RF receiver inputs via a pair of passive components to form a single antenna port high efficiency transceiver. The third PA, connected to the PA_BOOST pin and can deliver up to +20 dBm via a dedicated matching network. Unlike the high efficiency PAs, this high-stability PA covers all frequency bands that the frequency synthesizer addresses.SX1276/77/78/79 also include two timing references, an RC oscillator and a 32 MHz crystal oscillator.
- All major parameters of the RF front end and digital state machine are fully configurable via an SPI interface which gives access to SX1276/77/78/79's configuration registers.
  This includes a mode auto sequencer that oversees the transition and calibration of the SX1276/77/78/79 between intermediate modes of operation in the fastest time possible.The SX1276/77/78/79 are equipped with both standard FSK and long range spread spectrum (LoRaTM) modems. Depending upon the mode selected either conventional OOK or FSK modulation may be employed or the LoRaTM spread spectrum modem.

WORKING PRINCIPLE:

- The LoRa SX1278 works with SPI communication protocol so it can be used with any microcontroller that supports SPI. It is mandatory to use an Ariel (antenna) along with the module else it might damage the module permanently. The module should be powered only with 3.3V and the SPI line can be connected to uP/uC as shown in the image below.



- To communicate with the module some standard libraries are also available for Arduino like LoRa by Sandeep and other platforms. The module comes in a surface mount style package hence care should be taken while soldering. It is also a common practice to solder wires and pins with the module to use it as through hole module.

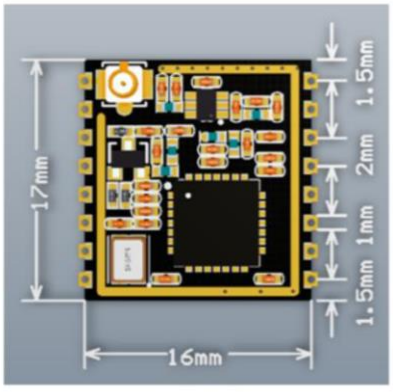
PIN FUNCTION:

Pin NO	Pin Name	Description
1,2,9,16	Ground	Ground pin of the Module
3	3.3V	Input Supply voltage
4	Reset	Resets the Module
5	DIO0	Digital I/O – 0
6	DIO1	Digital I/O – 1
7	DIO2	Digital I/O – 2
8	DIO3	Digital I/O – 3
10	DIO4	Digital I/O – 4
11	DIO5	Digital I/O – 5
12	SCK	SPI – Clock Input
13	MISO	SPI – Data Out
14	MOSI	SPI – Data In
15	NSS	SPI – Chip Select

APPLICATIONS:

- IoT Applications
- Home and Building Automation
- Long range communication
- Mesh or star topology networks
- Industrial monitoring and control

DIMENSION:



LAMPIRAN B

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN
RISET DAN TEKNOLOGI
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

LEMBAR BIMBINGAN LAPORAN TUGAS AKHIR

Pembimbing Tugas Akhir memberikan rekomendasi kepada,

Nama	:	Ria Febriyanti
NIM	:	061840341364
Jurusan/Program Studi	:	Teknik Elektro
Judul Tugas Akhir	:	Sistem Monitoring Sampah Pada Bak Robot Pengumpul Sampah Di Permukaan Air Berbasis Iot

Mahasiswa tersebut telah memenuhi persyaratan dan dapat mengikuti Ujian Tugas Akhir (TA) pada Tahun Akademik ..2022.....

Palembang, ...01 Agustus 2022

Pembimbing I,	Pembimbing II,
(Ir. Faisal Damsi, M.T.)	(r. Iskandar Lutfi, M.T.)
NIP. 196302181994031001	NIP. 196501291991031002

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN
RISET, DAN TEKNOLOGI
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

LEMBAR BIMBINGAN TUGAS AKHIR (TA)

Nama : Ria Febriyanti
NIM : 061840341364
Jurusan/Program Studi : Teknik Elektro DIV
Judul Laporan Akhir : Sistem Monitoring Sampah Pada Bak Robot Pengumpul Sampah di Permukaan Air
Pembimbing I : Ir. Faisal Damsi, M.T

No.	Tanggal	Uraian Bimbingan	Tanda Tangan Pembimbing
1.	15 Februari 2022	Pengajuan Judul Tugas Akhir	
2.	21 Februari 2022	Revisi Judul Tugas Akhir	
3.	16 Maret 2022	Rancang desain perahu	
4.	30 Maret 2022	Revisi desai perahu	
5.	19 April 2022	Merancang desain rangkaian elektronika	
6.	18 Mei 2022	~~Bab~~ Bimbingan Bab 3	
7.	20 Juni 2022	Revisi Bab 3	

No.	Tanggal	Uraian Bimbingan	Tanda Tangan Pembimbing
8.	8 Juli 2022	Tingkat kerjo alat sistem monitoring	
9.	27 Juli 2022	Bimbingan Bab 4 dan bab 5	
10.	20 Agustus 2022	Revisi Bab 4 dan 5	
11.	28 Juli 2022	Revisi Bab 4 dan 5	
12.	1 Agustus 2022	ACC Bab I, II, dan 5 lengkapi laporan	
13.			
14.			

Palembang, ...

Ketua Jurusan/KPS,

(Masayu Anisah, S.T., M.T.)
NIP **197012281993032001**

Catatan:
*) melingkari angka yang sesuai.
Ketua Jurusan/Ketua Program Studi harus memeriksa jumlah pelaksanaan bimbingan sesuai yang dipersyaratkan dalam Pedoman Laporan Akhir sebelum menandatangani lembar bimbingan ini.
Lembar pembimbingan LA ini harus dilampirkan dalam Laporan Akhir.

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN
RISET, DAN TEKNOLOGI
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

LEMBAR BIMBINGAN TUGAS AKHIR (TA)

Nama : Ria Febriyanti
NIM : 061840341364
Jurusan/Program Studi : Teknik Elektro DIV
Judul Laporan Akhir : Sistem Monitoring Sampah Pada Bak Robot Pengumpul Sampah di Permukaan Air
Pembimbing I : Ir. Iskandar Luthfi, M.T.

No.	Tanggal	Uraian Bimbingan	Tanda Tangan Pembimbing
1.	17 Februari 2022	Pengajuan judul Tugas Akhir	
2.	22 Februari 2022	Revisi judul tugas akhir	
3.	18 maret 2022	Rancangan desain perahu	
4.	28 maret 2022	Revisi rancangan perahu	
5.	19 April 2022	Bimbingan rancangan desain rangkaian elektronika	
6.	24 Mei 2022	Bimbingan Bab 3	
7.	22 Juni 2022	Revisi Bab 3	

No.	Tanggal	Uraian Bimbingan	Tanda Tangan Pembimbing
8.	27 Juni 2022	Tingkat kerja alat sistem monitoring	
9.	11 Juli 2022	Bimbingan Bab 4	
10.	18 Juli 2022	Revisi Bab 4	
11.	28 Juli 2022	Revisi Bab 4 dan Bab 5	
12.	1/8 2022	Rekomendasi Utk Mengikuti Uji Akhir	

Palembang, 01 Agustus 2022

Ketua Jurusan/KPS,

1/8 2022

(Ir. Iskandar Luthfi, M.T)
NIP **196501291991031002**

Catatan:
*) melingkari angka yang sesuai.
Ketua Jurusan/Ketua Program Studi harus memeriksa jumlah pelaksanaan bimbingan sesuai yang dipersyaratkan dalam Pedoman Laporan Akhir sebelum menandatangani lembar bimbingan ini.
Lembar pembimbingan LA ini harus dilampirkan dalam Laporan Akhir.

POLITEKNIK NEGERI SRIWIJAYA

HIMPUNAN MAHASISWA JURUSAN

TEKNIK ELEKTRO

Jalan Srijaya Negara Bukit Besar, Palembang 30139
Telepon 0711 353414 Faximili 0711-355918
Website :hmjelektro.polsri.ac.id, Email:hmjelektro@polsri.ac.id, CP: 082351238316

Letter of Acceptance

Kepada, 8 Juli 2022
Ria Febriyanti
Jurusan Teknik Elektro, Politeknik Negeri Sriwijaya
Palembang, Sumatera Selatan

Dengan ini kami menyampaikan bahwa, berdasarkan hasil seleksi yang dilakukan oleh *reviewer Electro National Conference* (ENACO) dengan tema ***"Empowering Technology for Artificial Intelligence Development in 5.0 Industrial Era to Achieve SDG's 2030"***, makalah dengan rincian:

Judul	: Penerapan *Internet of Things* sebagai Pemantau Bak Sampah pada Robot Pengumpul Sampah di Permukaan Air
Penyaji	: Ria Febriyanti
Email	: riaf38888@gmail.com

Dinyatakan **DITERIMA** untuk dipublikasikan di dalam buku prosiding *Electro National Conference* (ENACO) dengan e-ISSN 2797-0515 dan p-ISSN 2777-0958. Artikel tersebut akan tersedia secara online di https://enacoelektropolsri.com.

Demikian hal ini kami sampaikan, atas perhatian dan kerjasamanya kami ucapkan terima kasih

Ketua Panitia,

Destra Andika Pratama, S.T., M.T.
NIP 197712202008121001

KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
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

PELAKSANAAN REVISI TUGAS AKHIR

Mahasiswa berikut,

Nama	:	Ria Febriyanti
NIM	:	061840341364
Jurusan/Program Studi	:	Teknik Elektro/Sarjana Terapan Teknik Elektro
Judul Laporan Akhir	:	Sistem Monitoring Sampah Pada Bak Robot Pengumpul Sampah di Permukaan Air Berbasis Internet of Things (IoT)

Telah melaksanakan revisi terhadap Laporan Akhir yang diujikan pada hari Kamis tanggal 04 bulan Agustus tahun 2022 Pelaksanaan revisi terhadap Laporan Akhir tersebut telah disetujui oleh Dosen Penguji yang memberikan revisi:

No.	Komentar	Nama Dosen Penguji *)	Tanggal	Tanda Tangan
1.	Acc Laporan LA	Ir. Yordan Hasan,M.Kom	18/08-022	
2.	Revisi LA nee	Amperawan,ST.,M.T	7/2022 /9	
3.	Tdk Revisi	Yudi Wijanarko,ST.,M.T	10/8-22	
4.	Acc Laporan LA	Agum Try Wardhana,B.Eng,M.Tr.T	11/8 2022	

Palembang, Agustus 2022
Ketua Penguji **),

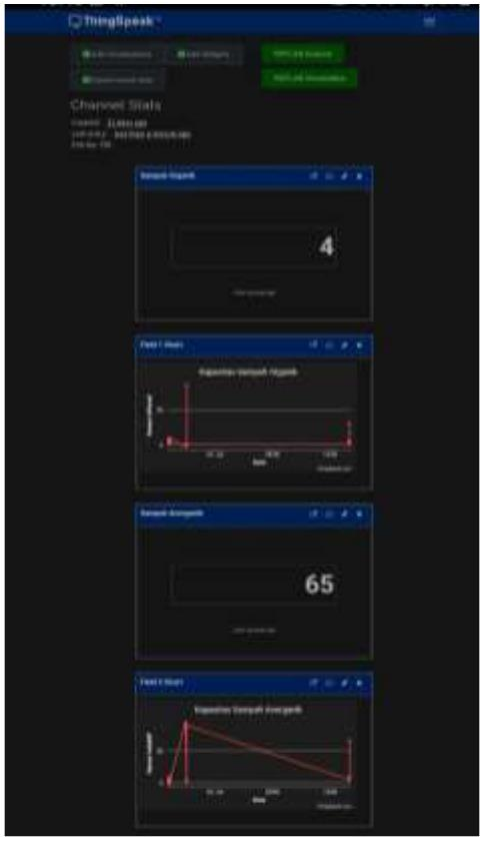
(Ir. Yordan Hasan,M.Kom.)
NIP. 195910101990031004

Catatan:
*) Dosen penguji yang memberikan revisi saat ujian laporan akhir.
**) Dosen penguji yang ditugaskan sebagai Ketua Penguji saat ujian LA.
Lembaran pelaksanaan revisi ini harus dilampirkan dalam Laporan Akhir.

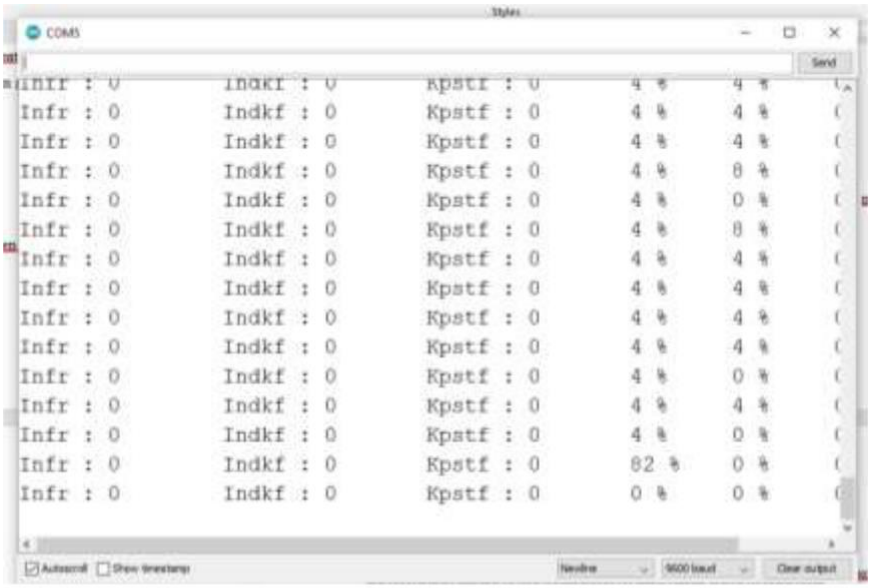
LAMPIRAN C

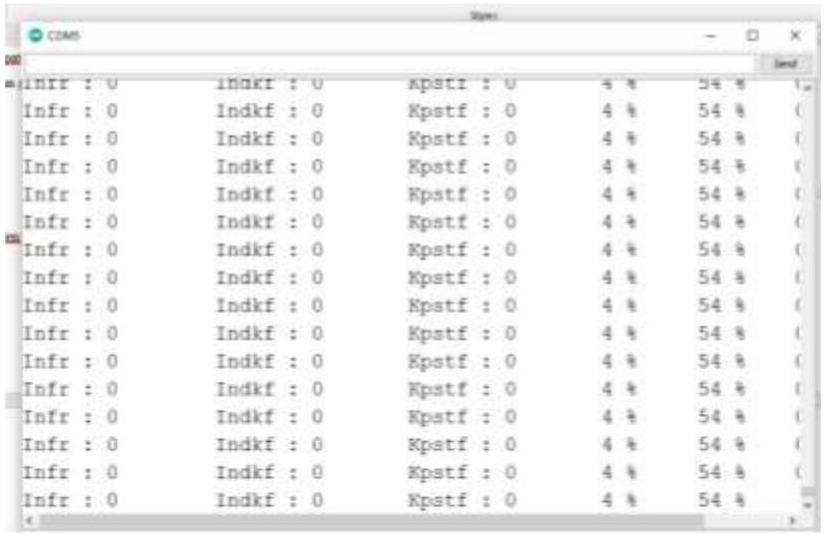
Robot Perahu Pengumpul Sampah di Permukaan Air

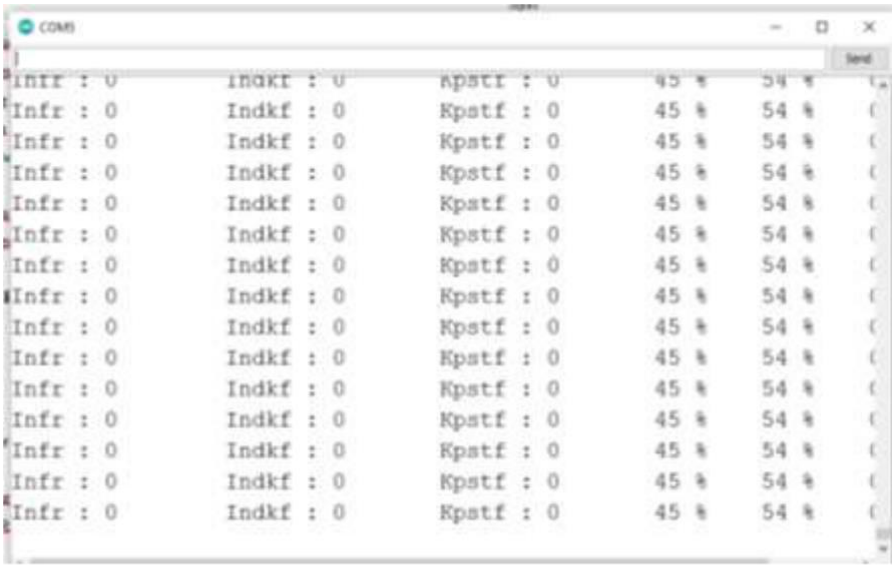
Lampiran pada thingspeak ketika mengukur jarak

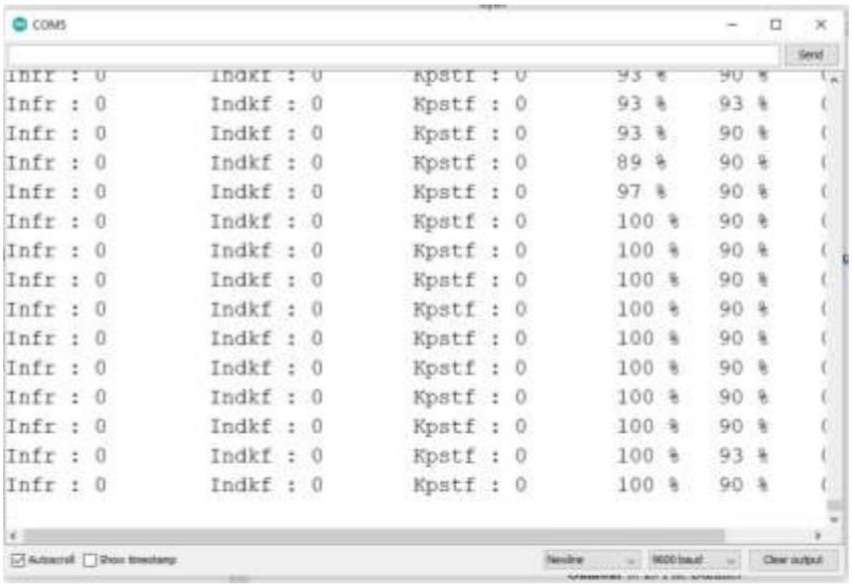


Lampiran serial monitor ketika menguji sensor ultrasonik pada bak sampah






COM5
Send
Infr : 0 Indkf : 0 Kpstf : 0 45 % 54 %
Infr : 0 Indkf : 0 Kpstf : 0 45 % 54 %
Infr : 0 Indkf : 0 Kpstf : 0 45 % 54 %
Infr : 0 Indkf : 0 Kpstf : 0 45 % 54 %
Infr : 0 Indkf : 0 Kpstf : 0 45 % 54 %
Infr : 0 Indkf : 0 Kpstf : 0 45 % 54 %
Infr : 0 Indkf : 0 Kpstf : 0 45 % 54 %
Infr : 0 Indkf : 0 Kpstf : 0 45 % 54 %
Infr : 0 Indkf : 0 Kpstf : 0 45 % 54 %
Infr : 0 Indkf : 0 Kpstf : 0 45 % 54 %
Infr : 0 Indkf : 0 Kpstf : 0 45 % 54 %
Infr : 0 Indkf : 0 Kpstf : 0 45 % 54 %
Infr : 0 Indkf : 0 Kpstf : 0 45 % 54 %
Infr : 0 Indkf : 0 Kpstf : 0 45 % 54 %


COM5
Send
Infr : 0 Indkf : 0 Kpstf : 0 93 % 93 %
Infr : 0 Indkf : 0 Kpstf : 0 93 % 90 %
Infr : 0 Indkf : 0 Kpstf : 0 89 % 90 %
Infr : 0 Indkf : 0 Kpstf : 0 97 % 90 %
Infr : 0 Indkf : 0 Kpstf : 0 100 % 90 %
Infr : 0 Indkf : 0 Kpstf : 0 100 % 90 %
Infr : 0 Indkf : 0 Kpstf : 0 100 % 90 %
Infr : 0 Indkf : 0 Kpstf : 0 100 % 90 %
Infr : 0 Indkf : 0 Kpstf : 0 100 % 90 %
Infr : 0 Indkf : 0 Kpstf : 0 100 % 90 %
Infr : 0 Indkf : 0 Kpstf : 0 100 % 90 %
Infr : 0 Indkf : 0 Kpstf : 0 100 % 90 %
Infr : 0 Indkf : 0 Kpstf : 0 100 % 93 %
Infr : 0 Indkf : 0 Kpstf : 0 100 % 90 %
Autoscroll
Show timestamp
Clear output