THE UNIVERSITY OF JORDAN

School of Engineering Engineering

Department of Mechatronics

Bachelor of Science in Mechatronics Engineering

Senior Design Graduation Project Report

Design of Underground water leakage detection

Report by

Qusai albitar (0167487)

Rami abdelhameed (0162188)

Abdullah Malhas (0160714)

Mohammad arabyat (0162607)

Supervisor

Dr. Osama Al-Habahbeh

Date : 21/01/2021

DECLARATION STATEMENT

We, the undersigned students, confirm that the work submitted in this project report is entirely our own and has not been copied from any other source. Any material that has been used from other sources has been properly cited and acknowledged in the report.

We are fully aware that any copying or improper citation of references/sources used in this report will be considered plagiarism, which is a clear violation of the Code of Ethics of University of Jordan.

In addition, we have read and understood the legal consequences of committing any violation of the University's Code of Ethics.

	Student Name	Student ID	Signature	Date
1	Qusai albitar	0167487		18/01/2021
2	Rami abdelhameed	0162188		18/01/2021
3	Abdullah malhas	0160714		18/01/2021
4	Mohammad arabyat	0162607		18/01/2021

ABSTRACT

Smart systems are being used widely in our lives, and the use of robotic actuators is being implanted in almost every new technology these days, smart systems offer several benefits such as helping in a lot of day-to-day jobs, tasks, and routines, that can save both effort and time, and in some cases money, which will lead into higher efficiency and easier life. The subject of CCTV surveillance is topical and broadly used in lots of exceptional applications. The essential part of the CCTV device is a dependable photograph assessment through way of means of a human observer, whose effectiveness is motivated via way of means of many variables. Optimization of the effectiveness is a multidimensional hassle associated with each technical and human characteristic. In many implemented structures, the general overall performance is tormented by an actual overall performance of technical device (photograph compression, channel transmission, etc.). On the other hand those technical structures have exceptional optimization standards than a standard video device.

`TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

GLOSSARY

ABBREVIATION	DESCRIPTION
AC	Alternating Current
DC	Direct Current
APN	Access point name
CCTV	Close circuit television

Chapter 1 INTRODUCTION

1.1 Background

After the design of sewer, storm water and water reticulations as built drawings are drawn up. These drawings could be used as a reverence document to anyone who might be interested in the future to know exactly where these services are installed. Moreover, as-built drawings include maintenance contractors, quantity surveyors, project managers, clients and any other body involved with payments of the services or involved in further working thereon.

Figure 1-1

The main area of concern in developing countries is that as-built drawings are usually done wrong, due to various reasons such as no drawing until they are covered up, lack of the existing as-built plans and the loss of plans or the absence of original parties who were involved in drawing, which makes attaining them and the plans difficult.

This has the following effects:

- It is more difficult to find out the exact location of the pipes that needs maintenance, installation of other services and upgrading to the system. This means more labor and time consumption.
- An increase in the difficulty of checking and measuring the works.
- High Risk of damage to the existing services when installing further services or during maintenance on an existing service.

Another big problem in sewer, storm water and water reticulations is that when a leakage occurs in the reticulation it might be difficult to find its exact positions. This is mainly due to a number of facts:

· When a pipe leaks, the water surface will follow the route of the least resistance and might surface at a quite distance away from the actual problem.

· When blockages occur, the problem normally seen at the entry

Figure 1-2

positions of the fluid or an alternative exit point, not at the blockage itself.

1.2 Problem Definition

We found a lot of problems that start leaks through pipes in this method and the causes of the problems that we are going to find a solution for and figure it out:

- Foundation shifts - little shifts in your home's foundation because it settles over time will produce huge changes in your water lines, inflicting them to disconnect or rupture[2].

- High water pressure - ends up in damaging wear and tear on the pipes. Water zipping through the pipes at a high speed will increase the danger of pipes bursting. The explosive changes to the direction of water flow will be an excessive amount of for the pipes to handle, eventually it results in leaks [2].

- Tree roots - little cracks in pipes will unleash water vapour into the encompassing soil, wherever tree roots can note and burrow their thanks to your pipes, inflicting massive issues [2].

- Corrosion - whereas it's not a giant concern in a number of Cincinnati's newer neighbourhoods, several older homes within the space use copper or galvanized steel piping, which may be vulnerable to rust.

- Temperature changes - Extreme changes in temperature (mainly cold weather) can cause pipes to crack and leak.

- Damaged pipes joints - The weakest point in a line is often where the pipes connect. Over time, pipe joints may deteriorate, causing leaks. Unfortunately, most pipe joints aren't easily visible. If you have noisy pipes that make a ticking or banging sound, especially using the hot water, chances are that the pipe joints are under significant pressure.

- Broken seals - Not all plumbing consists of metal pipes. once appliances are installed, the contractor puts rubber sealer around associate connections or areas just like the dishwasher door to stay everything watertight. As your appliances age, the protection may degrade and even break over time. If you notice condensation on your appliances or puddles close to them, you will have a broken seal [2].

- Clogged lines – particularly in Jordan's tanks whereas a clog might not seem to be an imperative issue, generally it will result in overflowing or maybe exploding of pipes. This causes pressure to make up behind the blockage, doubtless resulting in structural damage. Moreover, if a clog consists of corrosive substances like home chemicals, it'd speed up the harm to the pipes.

- Wrong pipes distribution- As it's declared before, one in every of the foremost vulnerable spots of a pipe network is its joints. Joints include separate recurved items attaching to straight pipes on either aspect that is to blame for redirecting water throughout your home.

- Fixture cracks - If you notice leaking, a regulator or the pipes connected directly thereto could have advanced cracks. This typically happens because of physical impact reminiscent of somebody tripping and catching themselves on the faucet. It also can occur out of sight, because of perennial things being forced into the cupboard. If you notice any harm to a fixture or its close pipes, it's best to contact a artisan to visualize out the problem as presently as possible.

Figure 1-3

1.3 Literature Review

There are different methods to detect leakage in pipes.

A.1 Acoustic method

Sensitive part of acoustic device is placed on steel sensing base (later on – base). the bottom is hanging at snap diaphragm fabricated from sound-proof rubber and guarded from external noise with safety cuff. Overall style of the sensor reduces the distortion of external noise and prevents mechanical harm of body once operating with sensor, the cuff is placed absolutely against the surface, and also the base of sensor ought to bit the bottom surface.

Figure 1-4

A.2 Flow rate sensor using Arduino and GSM

Figure 1-5

The water flow sensor consists of a plastic valve body, a water rotor and a Hall sensor. When water flows through the rotor, the rotor rotates. Its speed changes with the change of flow rate, and the Hall-effect sensor generates a corresponding pulse signal. This is suitable for recording the flow in a water dispenser or coffee machine. The Arduino platform has begun to be widely welcomed by people. This is for good reason. They have just started to get involved in the electronics field. Unlike most motherboards. The previous Arduino programmable circuit does not require separate hardware (called a programmer) to upload new codes to the card. You can just use the USB cable. In addition, the Arduino IDE uses a simplified version of C++ to make programming easier. Finally, Arduino provides a standard form factor, which can decompose the functions of the microcontroller into a cheaper package**.**

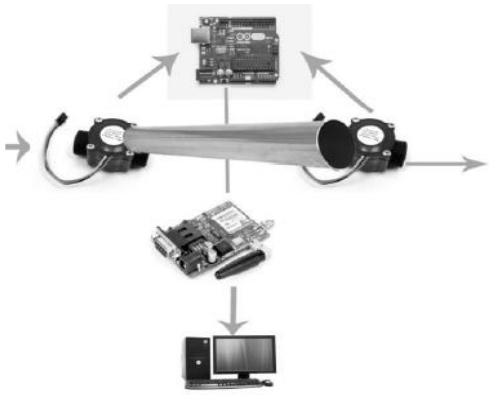


Figure 1-6

Figure 1-7

A.3 CCTV inspection

emote-controlled robots, detection of pipe leakage can be carried out using a Closed-Circuit Television (CCTV). The CCTV system comprised of a remote-controlled pan, and a camera is mounted on a robot traveling between two manholes inside the pipeline on trolled by certified operators [3], However, such passive systems have several drawbacks as such practices are time-consuming, labour intensity and have low reliability in detecting leaks as accuracy depends on the user's experience.

Figure 1-8

A.4 Thermograph

Water leaks and wet ingress will cause serious harm to a house or building, and it's troublesome to seek out till it is too late. Use a thermal image to spot hidden moisture and water issues, wherever appropriate, while not the necessity for harmful testing. There are several sources of moisture in buildings, as well as year-round moisture, condensate, pipe leaks, rain and snow, and even the breath of individuals and animals. Cause serious problems. The thermal image cannot "see" the moisture on the wall, however it can discover refined variations within the temperature and patterns of the visible water. [4]

Moisture Patterns

Moisture in buildings usually spreads in a recognizable pattern depending on where the water is:

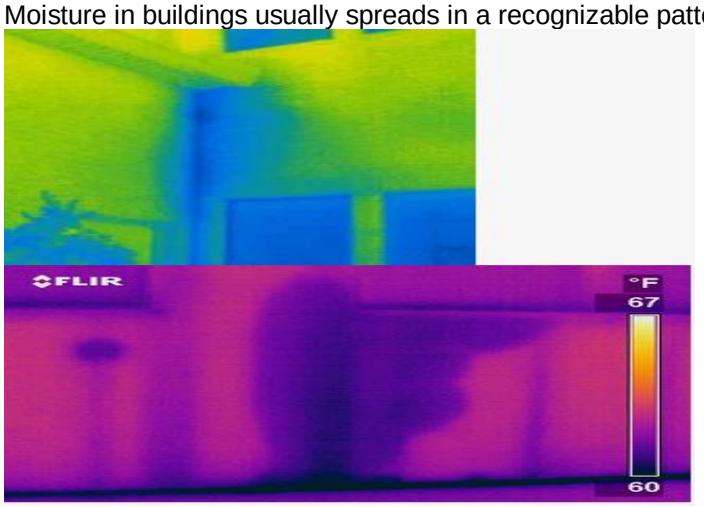


Figure 1-9

A.5Ground Penetrating Radar Operating Principle

The main in operation principle of a ground penetrating radio detection and ranging is to capture a picture of the underground pipe wherever the magnetic force signatures of discharge regions would manifest themselves within the captured images. leaky sources will then be determined throughout an on the spot interpretation of the pictures [5]. However, this detection methodology is influenced by the sort of soil encompassing the pipes. The results showed, that reflections at a lower place the leakage regions are relatively weaker than surrounding soil medium for many uniform soil and should not be potential to look at void development for most non-homogeneous soil. Furthermore, the relevance of this methodology for deeply buried pipes is limited, as either the wetness of the thereforeil or the non-homogeneity among the soil might not offer ample signal power levels on top of the noise floor [6]. In addition, it's tough to interpret the results [7] For tracing substances injection, it is an efficient and verified method and may sight even the tiniest leaks with low false alarm. Another pipeline with in-built sensors is sometimes inserted with the pipeline that needs watching so once discharge happens; the sensors can alert the engineers forthwith there could also be a requirement to filter or cleanse the water before victimisation that complicates the pipeline distribution network. Moreover, it should risk environmental contamination within the presence of leakage. Furthermore, for big unaggressive applications wherever tracer gas is used, the high volume of gas needed makes this methodology impractical [7]. Finally, installation of in-built sensors at the side of existing underground distribution system desires abundant physical labour to excavate the bottom that's quite impractical

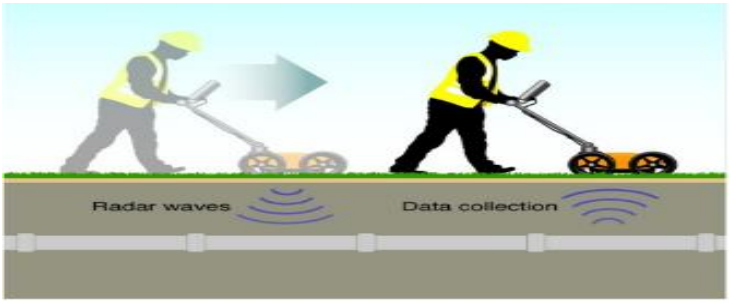


Figure 10

1.4 Aims and Objectives

The aim of this treatise is to investigate different technologies available to discover by a lot of methods to be mention later on leaks and/or the positions of underground water, storm water and sewer reticulation networks in addition the problem is that to detect leak of water in pipes, to save water bills and save your house from moisture cause that we think about devise to solve this problem.

1.5 Project Plan

First semester

Table 1-1

Task No.	Task Description	Semester Weeks																Students Involved			
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	1	2	3	4
1	Literature review	x	x	X														x	x	x	X
2	Model building			x	X	X	x											x	x	x	X
3	Validate the model							x	X									X	x	x	X
4	Making improvements to the model								x	X								x	x	x	X
5	Writing chapter one										x	x	X					x	x	x	X
6	Writing chapter two													x	x	x		x	x	x	x

Task No.	Task Description	Semester Weeks																Students Involved			
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	1	2	3	4
1	Design the module	X	X															X	X	X	X
2	Improve the system			X	X													X	X	X	X
3	Run the project					X	X											X	X	X	X
4	Writing chapter three						X	X	X									X	X	X	X
5	Writing chapter four									X	X	X						X	X	X	X

Second semester

Table 2

6	**Prepare the**											X	X	X			X	X	X	X

1.6 Report Organization

- In chapter one we talked about the problems that cause underground water leakage
- In chapter two we are going to talk about how to detect the leaks in pipes by some methods
- In the third chapter, designing and the stages of building the project will be presented with calculations and measures by new technology.
- In the fourth chapter, presenting the tests and results.

Chapter 2 DESIGN OPTIONS

2.1 Main components and Design

2.1.1 Controller

Arduino

Arduino platform is an open source for creating electronic projects. it consists of a physical programmable circuit board (usually called a microcontroller) and software or IDE (Integrated Development Environment) running on a computer, which is used to write and download computer code onto the physical board. The platform has begun to be more welcomed by people. This is for a good reason. They started to get involved in the electronics field. Unlike most previous programmable cards, Arduino does not require separate hardware (called a programmer). How to upload the new code to the whiteboard; by using the USB cable. In addition, the Arduino IDE uses a simplified version of C++, which makes learning programming easier. Finally, Arduino provides a standard form factor, which divides the function of the microcontroller into another package.

2.1.2 Software

Webots

Webots is a professional mobile robot simulation software package that provides a fast-prototyping environment, that allows users to create a 3D virtual world with physical characteristics (such as mass, joints, friction coefficient, etc.). Users are able to add simple passive objects or active objects, while the robot have different motion modes (wheeled robot, foot robot or flying robot). In addition, they can also be equipped with a series of detection and actuation devices, such as distance sensors, driving wheels, cameras, motors, sensors, transmitters, receivers, etc. Finally, the user can program each robot individually to demonstrate the desired behavior. Webots includes many robot models and sample control programs to help users get started. Webots also have many interfaces for real mobile robots. Once the simulated robot shows the expected behavior, you can transplant the control program to real robots such as E-Puck, Darwin OP, Nao, etc. through the connected system. [10]

Fritzing

Fritzing is an open-source hardware initiative that provides electronics accessibility as a creative material for anyone. We offer a software tool, a community website and services in the spirit of Processing and Arduino, fostering a creative ecosystem that allows users to document their prototypes, sharing them with others, teaching electronics, and layout and manufacture professional pcbs.[11]

2.1.3 Motor selection

Comparison between most used servo motors

Table 3

Made by	Model	Modulation	Weight gram	Dimensions (L×W×H) in mm	Torque Kg-cm	Speed	Gear Material	Typical Price
TowerPro	MG995	Digital signal	55	40.7*19.7*42.9	9.0	4.8V 0.20	Metal	8.00 Dollar
	SG90	Analog signal	9	23.0*12.2*29.0	1.6	4.8V 0.12	Plastic	2.0 Dollar

•SG90

Micro Servo Motor SG90 is a small and light server motor with high output power [12].
Servo rotation is almost 180° *and* 90°in each direction and works just like the basic versions the difference that its smaller. We can use any servo software, hardware, or library to control these servos.
It makes things run without utilizing a motor controller with feedback and gear box, especially when it fits in small spaces.

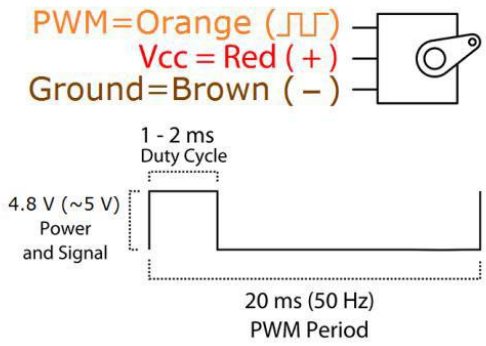


Figure 12

Design Constraints and Standards

-Design constraints

this section will discuss the main constrains that faced this project. constrains which are the limitations that engineers may counter during projects design, where constrains could affect the creativity of the project. Moreover, we mentioned the standards that will be adhered to in order to make the project more efficient to use.

• Cost: is a major factor in determining the efficiency, effectiveness, and quality of the system. Therefore, the approximate cost of the system must be calculated to work on improving the file Functional design with meticulous work under the condition required for use.

• Dimensions: It is one of the most important challenges that needs attention, because if we couldn't pinpoint the exact dimension of each part (pipes), that would be it the main drawback is the small components and details of this project where it should be placed. This requires us to carefully select and select ingredients the most suitable ones which will result in a convenient and coordinated design.

CHAPTER 3 DESIGN STAGES

3.1 CCTV

CCTV (Closed Circuit TV, or Remote Visual) is the use of video cameras to see inside infrastructure, such as pipelines, turbines, vessels and tanks that are otherwise inaccessible for direct visual inspection. This technology provides the ability to inspect key areas of concern without disassembly, or the need for personnel to enter confined spaces. Cameras are mounted on push-rod cables, or crawlers that can travel into and through pipelines.

The inner surface images of the pipe are recorded in real time for the period of the inspection and the videos are then analyzed by the field engineer immediately. CCTV inspection services are ideal for monitoring pipe constructed with various materials, including metal, concrete, plastic and fiberglass. This particular CCTV system works best in 6” (150mm) to 60” (1524mm) pipe diameters.

Figure 13

Figure 14

CCTV Applications:

- Pipelines
- Gravity & storm water sewers
- Foreign object locating
- Water Intake Systems
- Storage tanks & vessels
- Trunk / Interceptor Systems
- Man holes & catch basins
- Laser Pipe Profiling & Analysis

3.2 Components

Figure 15(L298 Dual H-Bridge Motor Driver)
L298 Dual H-Bridge Motor Driver

Descriptions:

The Motor protect relies on the L298, that could be a twin full-bridge driver designed to drive inductive hundreds admire relays, solenoids, DC and stepping motors. It allows you to drive 2 DC motors,dominant the speed and direction of every one independently**.**

Table 4

How much voltage operating	4V to 35V
Motor controller	can drives 2 dc drive or just one stepper motor
current	2A or 4A max
Chip	ST L298N
power supply	5V
Power	25 W
Weight	35g
Size	55mm x 60mm x 30mm
Storage temperature	-25℃ to +135℃

L298N driver specification

About power:

The Motor Shield must be powered only by an external power supply.

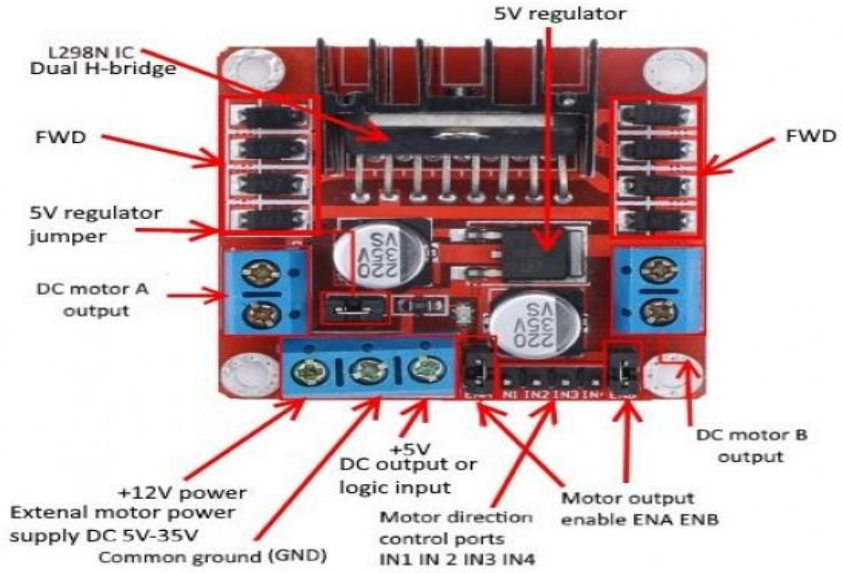


Figure 16

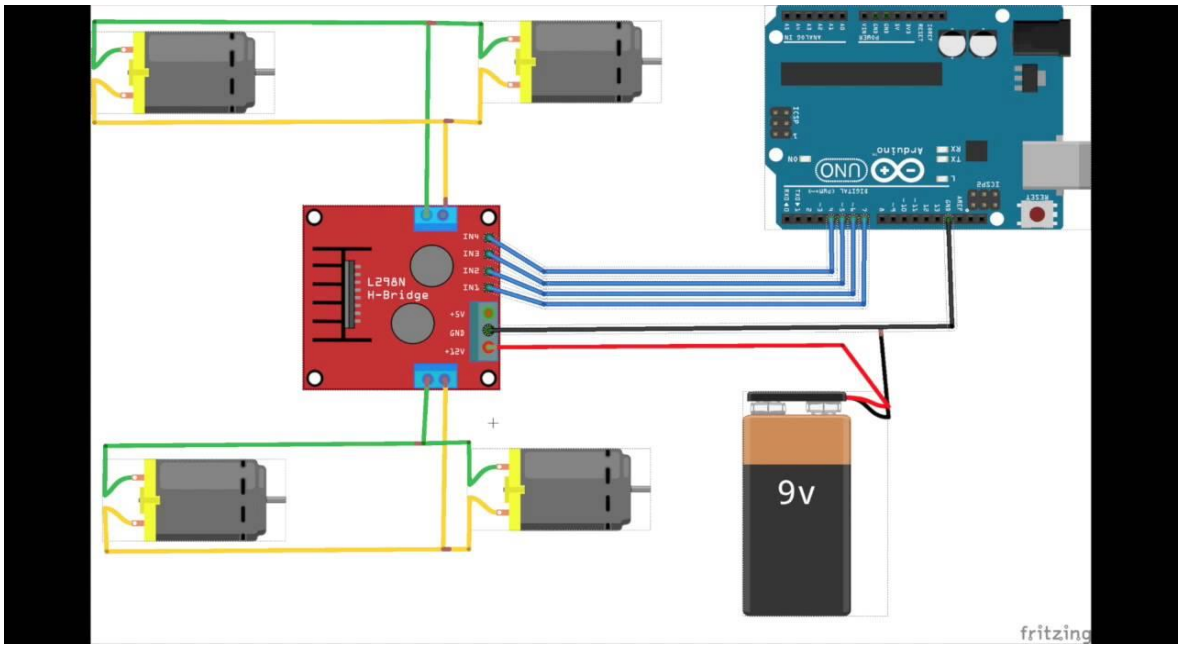


Figure 17
Connect L298N to control 4 motors

How to control the dc motors
Controllable parameters:

- Speed by varying the input voltage using PWM.
- Direction by reversing the polarity.

The 298n using the H-bridge.

- When switches 1 and 4 are closed and 2 and 3 are open, current flows through the motor in clockwise direction.
- When switches 2 and 3 are closed and 1 and 4 are open, polarity is reversed, and current flows from the supply to 3 to the motor in counterclockwise direction.

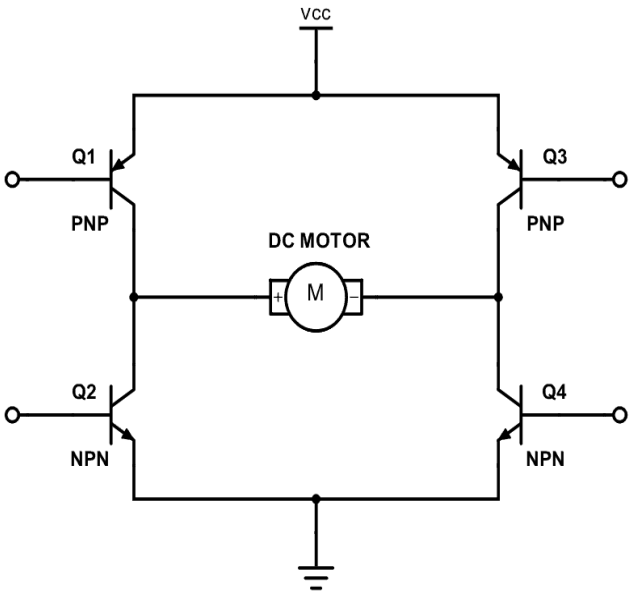


Figure 18

Motor speed using the PWM:

- Dc motor's speed proportional to supply the voltage
- If the voltage drops too far, the motor will not get enough power to run
- Usually at 50% at the rated voltage the motor run at varying speed
- Pulse width modulation approach is used to control the speed. This means that you pulse the motor ON and OFF at varying rates, to simulate a voltage.

When the signal is high, we call this on time described by the duty cycle

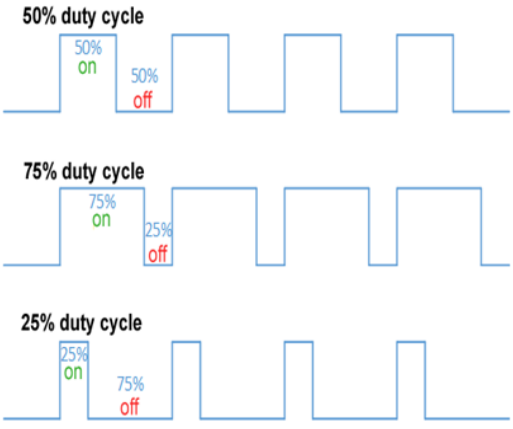


Figure 19

Figure 20
DC Gearbox Motor

Let's begin at the start with some definitions. What specifically could be a DC motor? an immediate current (DC) motor consists of a collection of magnets, rotor, Associate in Nursing switch. after you apply current to the rotor coil, it'll be converted into an magnet and repel the magnets. The commutator causes this within the rotor coil to change polarity because it rotates. This polarity switch causes the rotor coil to repel the magnets and generate continuous torque. Speed during a DC motor is proportional to the voltage applied to the rotor. The power made by the motor is proportional to the voltage increased by the current. once operating with DC motors, it's essential to recollect the connection between power, voltage, and current.

$$\boldsymbol{power = voltage * current}$$

And another relation is:

$$\boldsymbol{power = torque * angular\ velocity}$$

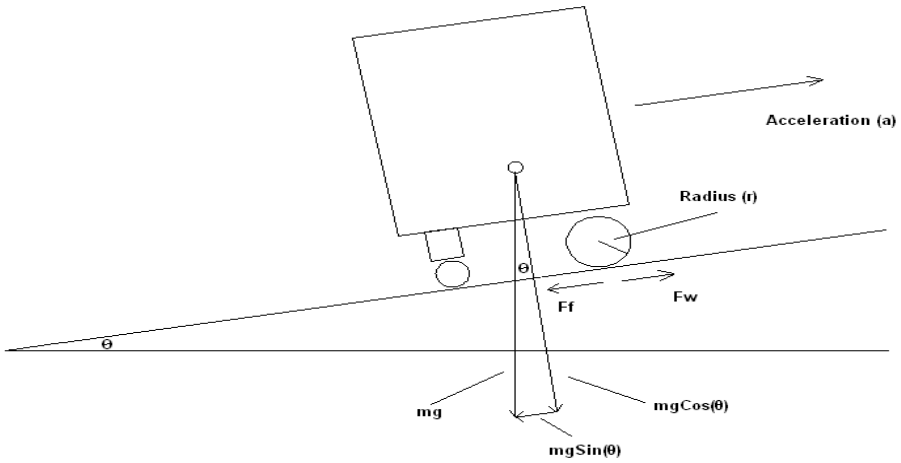


Figure 21

You may even look to 24 dc or 90 dc volt

When sizing a DC motor, we'll use the rated voltage of the motor. this can be the utmost voltage the motor is designed to handle. There are many differing types of DC motors to select from. In most cases, i exploit brushed DC gearhead motors. Gearhead motors have

a case put in as part of the motor. This gearhead could be a gearbox connected to the output shaft of the motor. a couple of alternative styles of DC motors include brushless and stepper motors.

$$\boldsymbol{F = ma}$$

$$\boldsymbol{Weight = mg}$$

Force pulling the robot down.

$$\boldsymbol{Fg = mgsin\theta}$$

Force pulling the robot incline.

$$\boldsymbol{Fn = mgcos\theta}$$

$$\boldsymbol{torque = force * distance}$$

$$\boldsymbol{T = M(a + gsin\theta) * r}$$

During the initial choice of the motor, we have a tendency to assumed that there was enough friction between the wheel and the surface thus there's no slip. this is often truly a foul assumption.Most wheels will spin once the golem is started at full speed on a slicker surface. In some cases, this could be a real problem. One resolution to the present state of affairs is to pick different wheels. Or, a far better solution is to build the motor speed instead of beginning at full speed

Table 5

Rated Voltage	3~6V
Continuous No-Load Current	150mA +/- 10%
Min. Operating Speed (3V)	90RPM
Min. Operating Speed (6V)	200RPM
Torque	0.15Nm ~0.60Nm
Stall Torque (6V)	0.8kg.cm
Body Dimensions	70 x 22 x 18mm
Weight	30.6g

DC Gearbox Motor specification

These steps we used to select a proper dc motor and gearbox

- **Step one** calculate the required force
- **Step two** calculate the torque and RPM
- **Step three** calculate the torque of motors
- **Step four** search for the motor could achieve that required

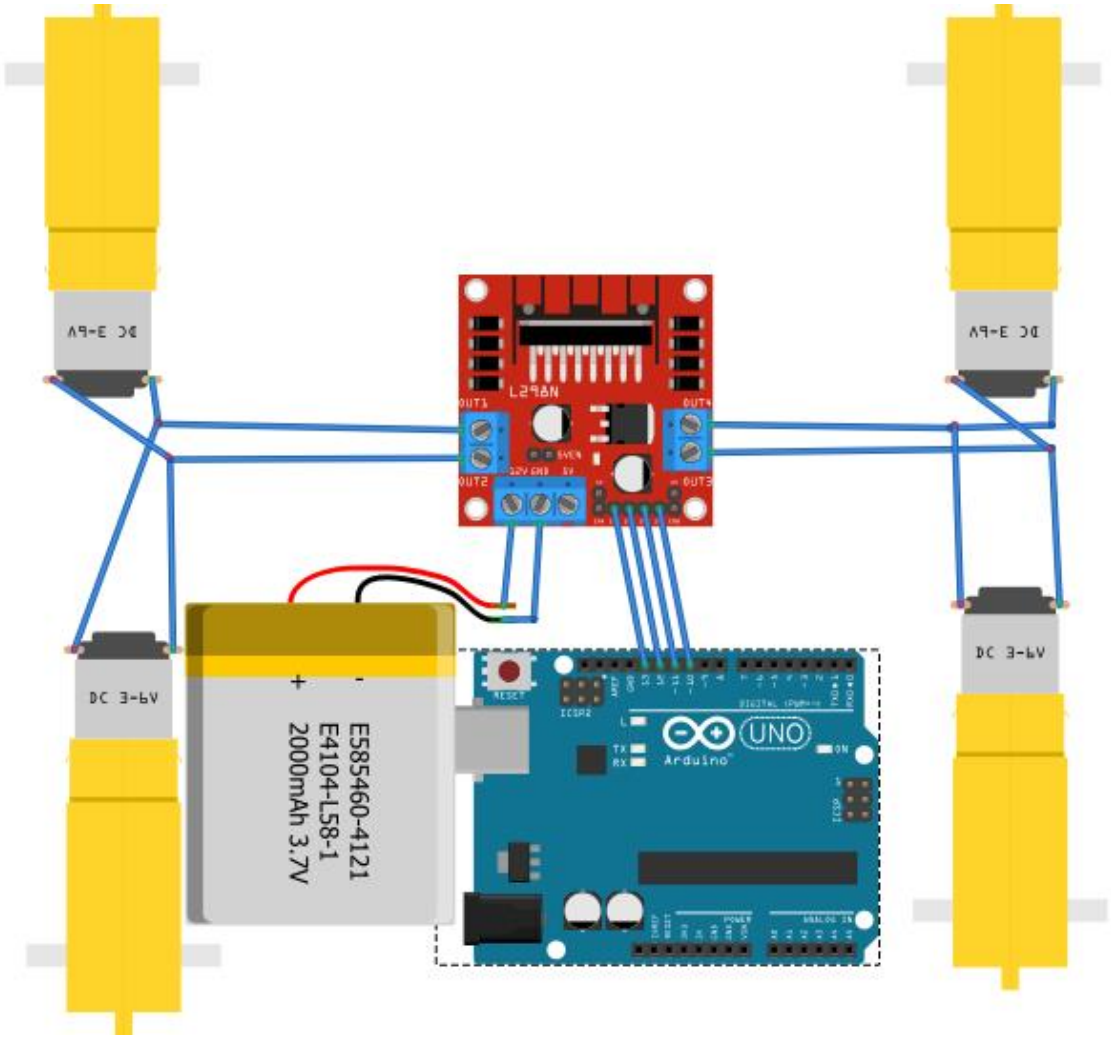


Figure 22
DC gearbox with L298N and Arduino connection

Figure 23
Obstacle tyres wheels

We here use the rubber wheels because we don’t need a slipping.

Our assumption is frictionless no slipping.

Figure 24
Arduino MEGA

Arduino is an open source gadget stage based on user-friendly hardware and computer programs. The Arduino worksheet can learn input (lights on sensors, fingers on buttons or Twitter messages) and turn it into performance art. -Activate the motor, turn on the drive, distribute something on the network, and then tell the card what to do by sending the lighting kit to the microcontroller on the card. For this, we use the Arduino programming dialect (wiring-based) and the processing-based Arduino program (IDE).

For many years, Arduino has been the brain of thousands of projects, from everyday objects to complex scientific instruments. This open source platform gathers a community of developers from all over the world, including students, hobbyists, artists, programmers, and professionals. Your contribution adds an incredible amount of available knowledge, which may be of great help to beginners and others. Arduino was born in Ivrea Interactive Design College.

It is a simple rapid prototyping tool suitable for students who have no background in electronics or programming. Once entering the wider community, the Arduino development board began to change to adapt to new needs and challenges, making its products unique from simple 8-bit boards to IoT products. Applications, wearable devices, 3D printing and embedded environments. All Arduino development boards are completely open source, so users can create them independently and eventually adapt to their specific needs. Due to the participation of users from all over the world, the software is also open source and the number is growing.[13]

Table 6

Microcontroller	ATmega1280
Operating Voltage	5V
Input Voltage	7-12V
Input Voltage	6-20V
Digital I/O Pins	54 (of which 15 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	128 KB of which 4 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

Arduino MEGA specification

Figure 25

2 DOF pan and tilt kit

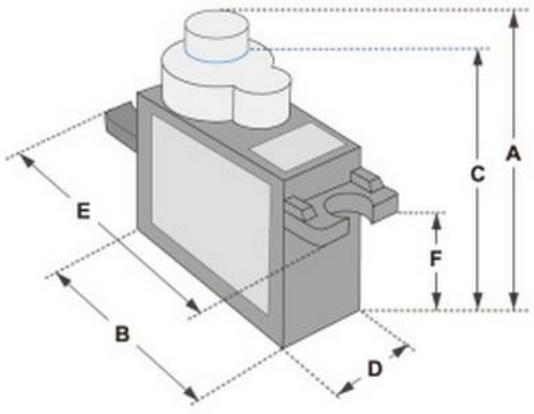
Specifications:

•Two axis platform, anti vibration

•Suitable for 9g-12g servos

•Net weight 16g

SERVO MOTOR SG90



Dimensions & Specifications
A (mm) : 32
B (mm) : 23
C (mm) : 28.5
D (mm) : 12
E (mm) : 32
F (mm) : 19.5
Speed (sec) : 0.1
Torque (kg-cm) : 2.5
Weight (g) : 14.7
Voltage : 4.8 - 6

Figure 26

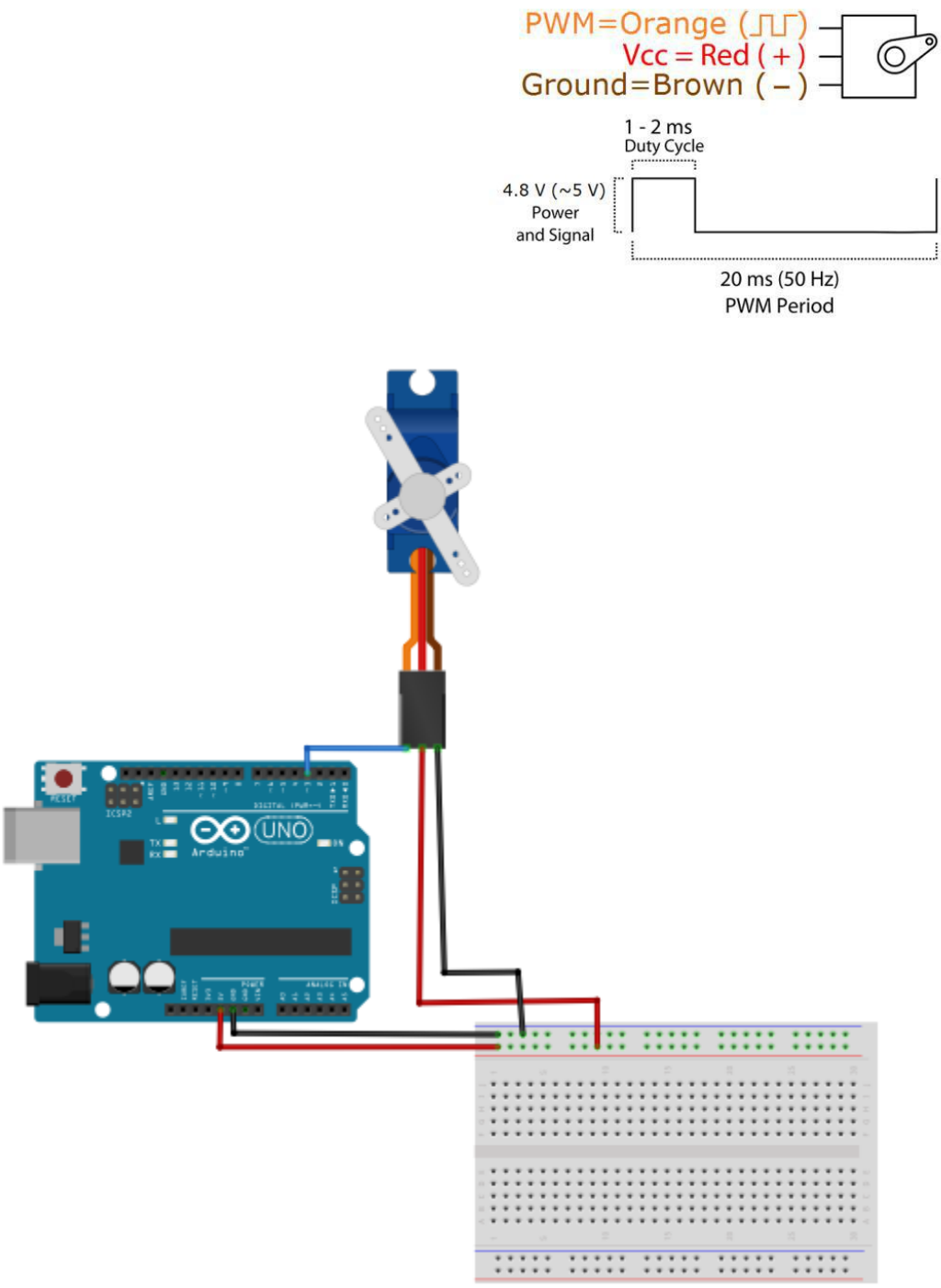


Figure 27

Servo motor with arduino connection

Figure 28

14.8V Li-ion 18650 Lithium ion Rechargeable Battery

Rechargeable atomic number 3 particle (Li-Ion) cells have a negative conductor (anode) made up of lithium compounds. atomic number 3 could be a extremely reactive material and my be abundant lighter than the hydrogen-absorbing metal alloy of the NiMH negative electrode. This ends up in higher hydrometric energy densities for the Li-Ion cell. Cells will keep in storage for twelve months while not requiring maintenance. The expected cycle lifetime of a Li-Ion in an application is concerning 1000+ cycles.

Table 7

Rated Capacity	2200mAh
Nominal Voltage	14.8V
Max Charge Voltage	16.8V
Discharge Cut Off Voltage	11V
Charging Current	0.2C
Max. Continuous Discharging Current	1C
Cycle Life	500 times
Dimensions (L*W*H)	71*38*38mm ±5mm / Customized
Weight	175g
Storage Temperature	-10 ~ 45℃

lithium ion battery Details

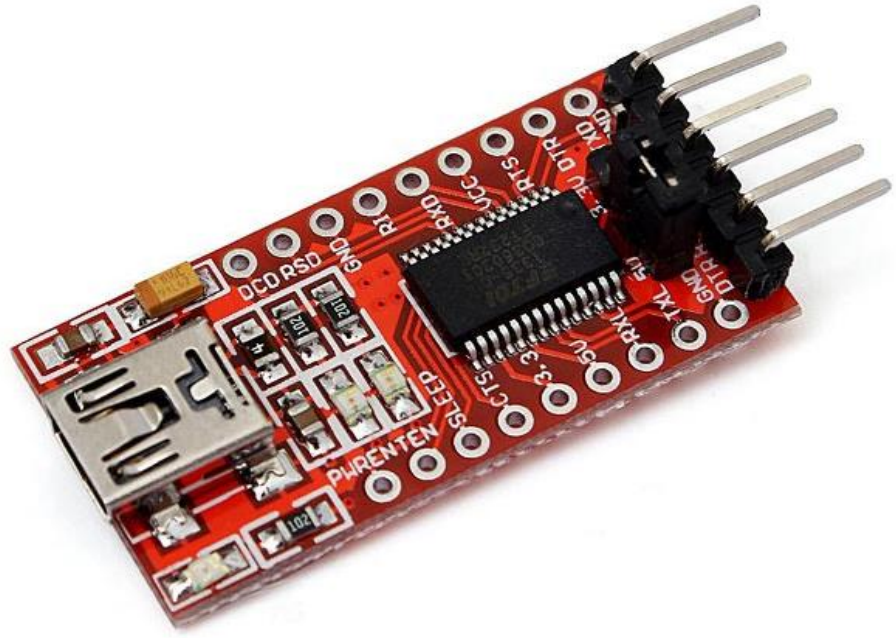
ESP32CAM With OV2640 Camera Module

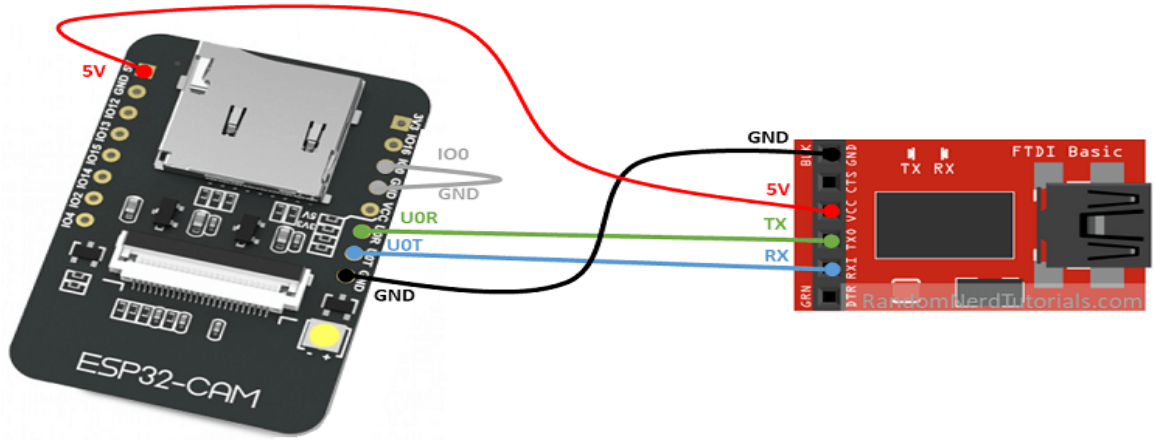
Figure 29

Features:

Features: Ultra-small 802.11b/g/n Wi-Fi BT/BLE SoC module Low-power dual-core 32-bit hardware for application processors Up to 240MHz, up to 600 DMIPS inherent 520 KB SRAM, external 4M PSRAM Supports interfaces resembling UART/SPI/I2C/PWM/ADC/DAC Support OV2640 and OV7670 cameras with built-in flash Support for pictures wireless local area network transfer Support TF card

FT232RL USB to TTL Serial Module





Description

The board is for FTDI's popular USB to UART IC. Now with internal oscillator and EEPROM, the FT232RL is an impressive IC!

Features

• Implements full v2.0 USB protocol

• Internal EEPROM for device ID and merchandise Description strings

• growth FT232RL IO for RS-232, bit-bang, or special perform modes.

• POWER, RXD, TXD, diode standing indicators.

• Power by USB, will be select 5V or 3.3V level (set by jumper)

• Royalty-Free Driver support for Windows, Linux, and macintosh OSX

Jumper Wires

Figure 30

Micro USB for TTL Serial Module

Figure 31

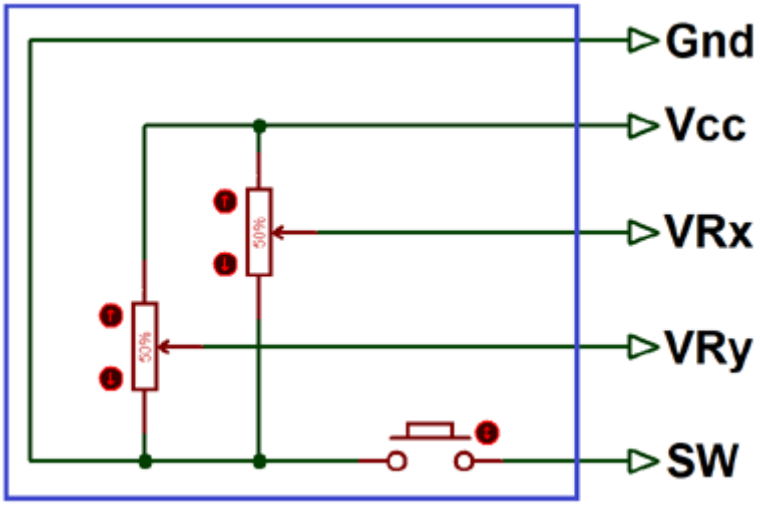
Figure 32
Joystick Module

When we have a tendency to listen the word “Joystick” we predict of Game controllers. If we point out Electronics, there are several helpful application of Joystick. These kinds of module are largely utilized in Arduino primarily based DIY comes and golem Control. As we know, the module provides analoge output so it are often used for feeding the analoge input supported direction or movement. It may also be connected to a movable camera to regulate its movement.[14]

Pin Configuration

Table 8

1	Gnd	Ground terminal of Module
2	+5v	Positive supply terminal of Module
3	VRx	Voltage changes in X axis
4	VRy	Voltage changes in Y axis
5	SW	Switch



Internal Structure

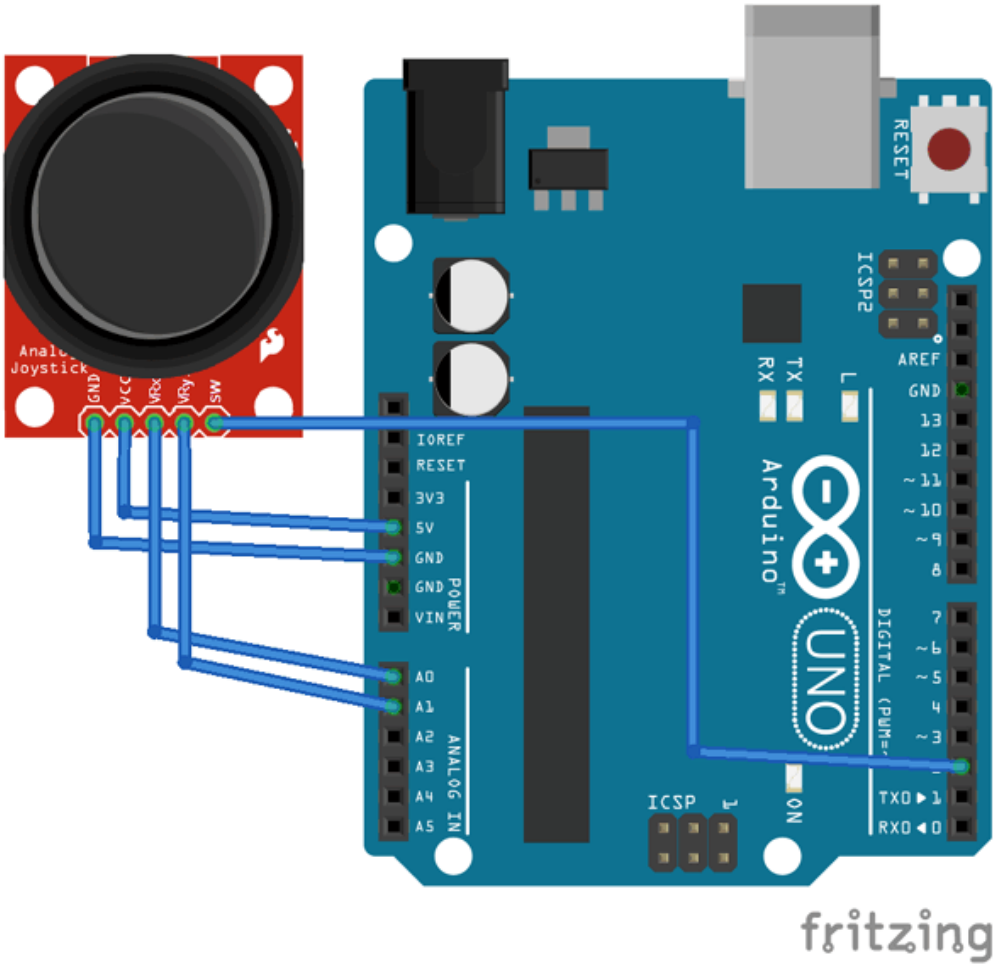


Figure 33
Joystick with Arduino connection

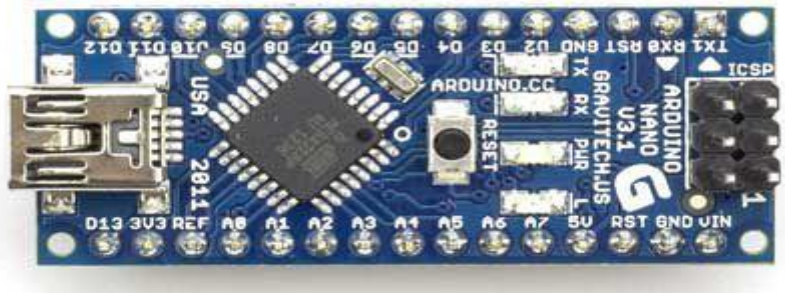


Figure 34
Arduino NANO

The Arduino Nano is a small, complete, and breadboard-friendly board based on the ATmega328 (Arduino Nano 3.x). It has more or less the same functionality of the Arduino Demilune, 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.[15]

Arduino Nano Specifications

Table 9

Voltage can operating	5V
Input Voltage for Vin pin	7-12V
Analog Input Pins	6 pins (A0 – A5)
Digital I/O Pins	14 (Out of which 6 provide PWM output)
DC Current on I/O Pins	40 mA
DC Current on 3.3V Pin	50 mA
Flash Memory	32 KB (2 KB is used for Bootloader)
SRAM	2 KBite
EEPROM	1 KBite
Frequency (Clock Speed)	16 MHz

Figure 35
HC-05 Bluetooth module

The HC-05 may be a terribly cool module which may add 2-way (full-duplex) wireless practicality to your projects. you'll be able to use this module to speak between two microcontrollers like Arduino or communicate with any device with Bluetooth functionality sort of a Phone or Laptop. There are several golem applications that are already on the market that makes this method loads easier. The module communicates with the assistance of USART at 9600 information measure therefore it's simple to interface with any microcontroller that supports USART. we will additionally tack the default values of the module by victimisation the command mode. thus if you looking for a Wireless module that might transfer knowledge from your pc or mobile to microcontroller or the other way around then this module can be the correct alternative for you. but don't expect this module to transfer multimedia system like photos or songs; you would possibly have to be compelled to look at the CSR8645 module for that.[16]

Pin Configuration

Table 10

PIN NO.	Pin Name	Pin Description
1.	**KEY/En**	. In HC-05, the default baud speed in command mode is 38400bps and 9600 in data mode.
2.	**VCC**	3.3 to 5v
3.	**GND**	The ground pin of the module
4.	**TXD**	Connect this pin with the RXD pin of the Microcontroller. This pin transmits Serial data (wireless signals received by the Bluetooth module are converted by module and transmitted out serially on this pin)
5.	**RXD**	Connect to Rc nano microcontroller
6.	**STATE**	It is used to check if the module is connected or not. It acts as a status indicator.

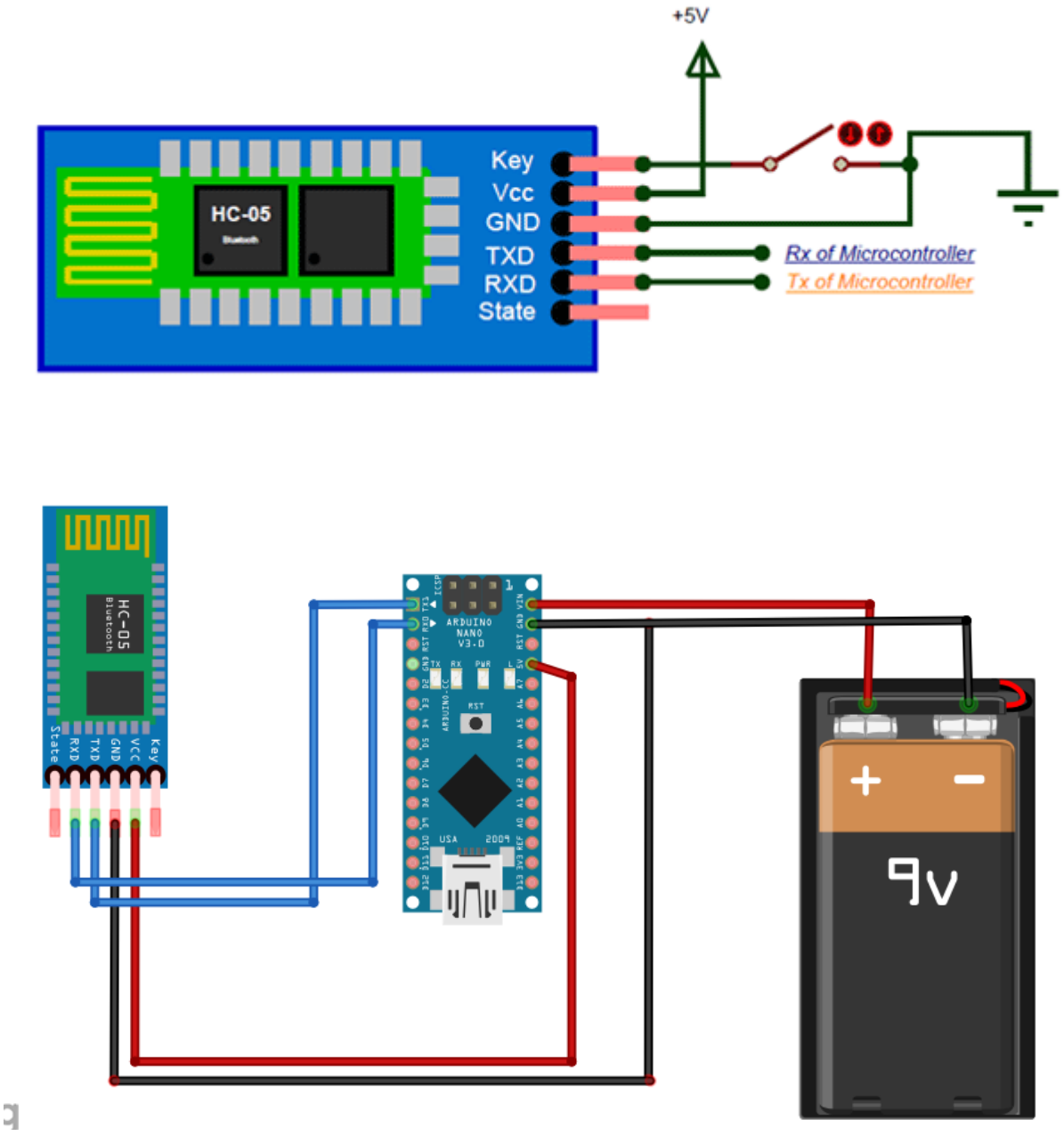
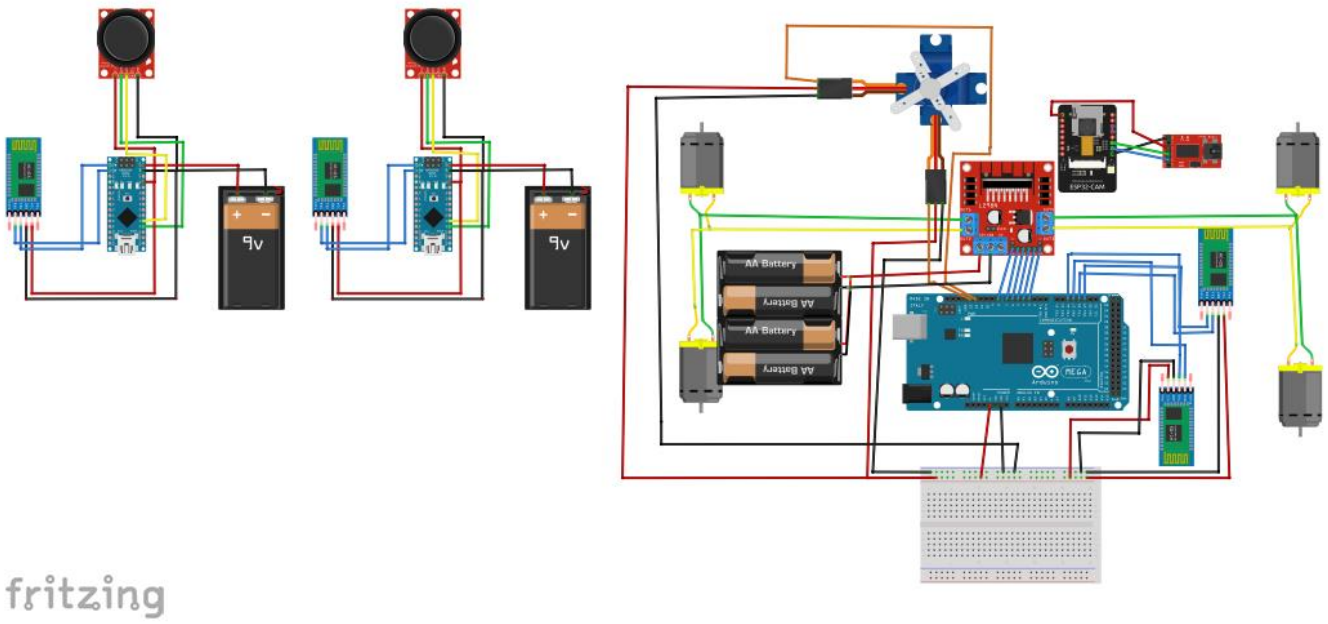


Figure 36
Hc-05 with Arduino connection

CHAPTER 4 DESIGN TESTING AND RESULTS

4.1 simulation



Project Components Connection

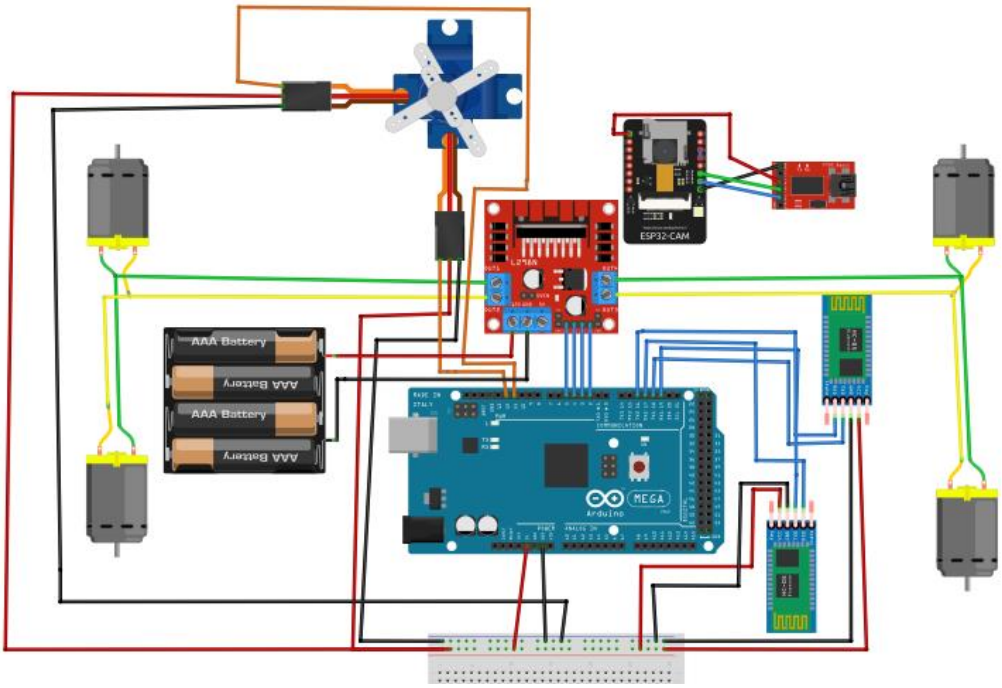


Figure 37
Robot Part Connection

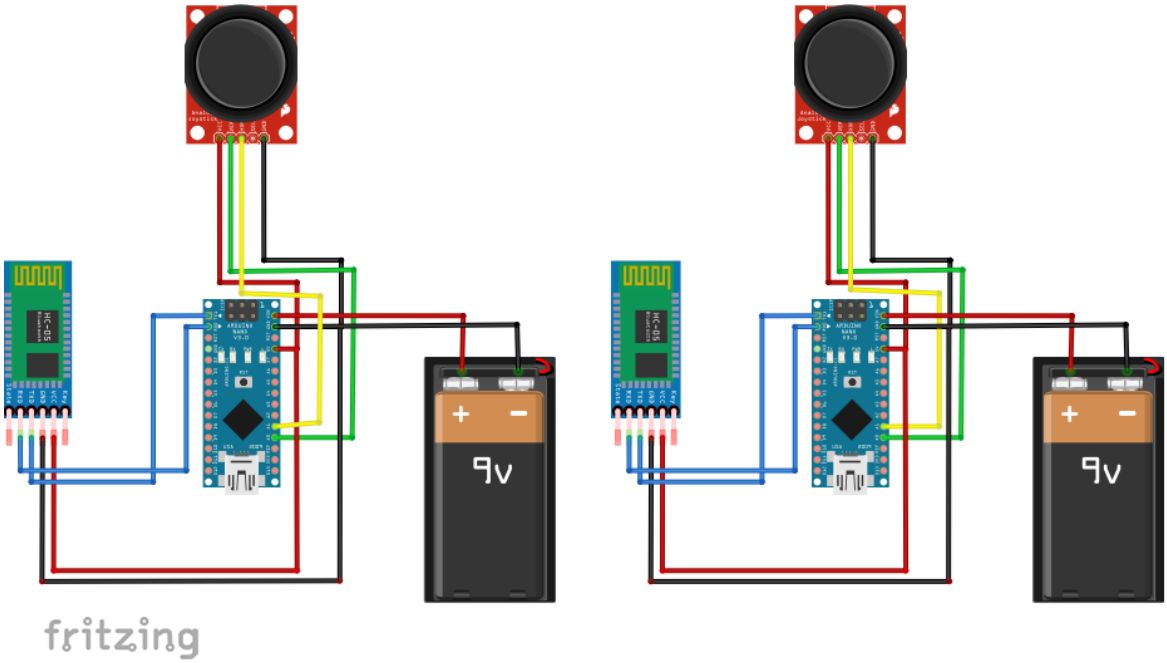


Figure 38
Controller Part Connection

Webots:

So, we tend to set to simulate our obstacle in webots machine it's very helpful let's imagine a short background of this simulator, you'll add heaps varieties of motors; liner and rotational, arena ,environment, hinge joints and wheels all kind of devices. it's simple to be told by victimisation c++ code generally The controller is that the keyboard to alter obstacle from to a different purpose and there's a distance sensor before of the golem to discoverion boundaries of pipes. we tend to add the camera with rotation motor to detect the discharge inside the pipe and liner motor to travel upward and downward and you'll use a keyboard to left and right

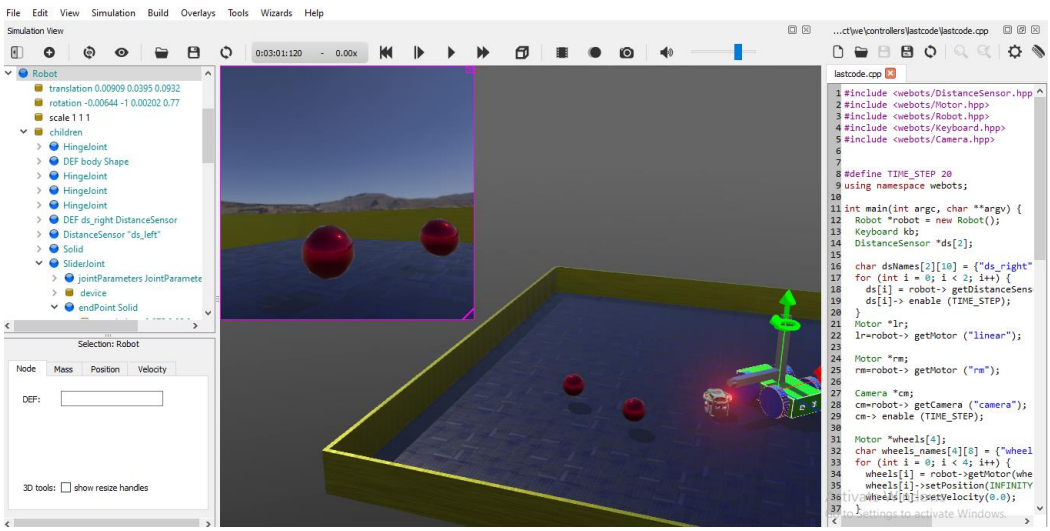


Figure 34

Our C++ webots code:

```
#include <webots/DistanceSensor.hpp>
#include <webots/Motor.hpp>
#include <webots/Robot.hpp>
#include <webots/Keyboard.hpp>
#include <webots/Camera.hpp>


#define TIME_STEP 25
using namespace webots;

int main(int argc, char **argv) {
Robot *robot = new Robot();
Keyboard kb;
DistanceSensor *ds[2];
char dsNames[2][10] = {"ds_right", "ds_left"};
for (int i = 0; i < 2; i++) {
ds[i] = robot-> getDistanceSensor(dsNames[i]);
ds[i]-> enable (TIME_STEP);
}
Motor *lr;
lr=robot-> getMotor ("linear");
Motor *rm;
rm=robot-> getMotor ("rm");
Camera *cm;
cm=robot-> getCamera ("camera");
cm-> enable (TIME_STEP);
Motor *wheels[4];
char wheels_names[4][8] = {"wheel1", "wheel2", "wheel3", "wheel4"};
for (int i = 0; i < 4; i++) {
wheels[i] = robot->getMotor(wheels_names[i]);
wheels[i]->setPosition(INFINITY);
wheels[i]->setVelocity(0.0);
}
kb.enable(TIME_STEP);
double leftSpeed = 0.0;
```

```
double rightSpeed = 0.0;
double linear=0.0;
double rotate=0.0;
while (robot->step(TIME_STEP) != -1) {
int key=kb.getKey();
if (key==315){
leftSpeed = 1.0;
rightSpeed = 1.0;
} else if (key==317){
leftSpeed = -1.0;
rightSpeed = -1.0;
}else if (key==316){
leftSpeed = 1.0;
rightSpeed = -1.0;
}else if (key==314){
leftSpeed = -1.0;
rightSpeed = 1.0;
}else {
leftSpeed = 0.0;
rightSpeed = 0.0;
}
wheels[0]->setVelocity(leftSpeed);
wheels[1]->setVelocity(rightSpeed);
wheels[2]->setVelocity(leftSpeed);
wheels[3]->setVelocity(rightSpeed);
if (key==87 && linear<0.19){
linear += 0.005;
} else if (key==83 && linear>0){
linear += -0.005;
}else {
linear+=0;
}
lr->setPosition(linear);
std::cout<<key<<std::endl;
if (key==65 && rotate<1.57){
rotate += 0.05;
} else if (key==68 && rotate>-1.57){
rotate += -0.05;
}else {
rotate+=0;
}
rm-> setPosition (rotate);
}
delete robot;
return 0; // EXIT_SUCCESS
}
```

Webots description you can follow the link below I hope to check it out to see how we simulate our obstacle description in YouTube :

webots simulator

4.2 System Limitations and Compliance with Design Constraints

There are some limitations for this system:

- The system can reach certain distances due to the weakness of the Wi-Fi network or the Bluetooth underground.
- There should be no high-water pumping in the pipes. The water pumping should be disconnected for a period until the location of the leak is determined.
- Visual Data only, with cable tracked encoder providing approximate meter age.
- Not available in Amman's house connection cause the connection provided the small diameter pipelines we can use accelerometer .

CHAPTER 5 PROJECT IMPACT

5.1 Financially

Table 11

Arduino MEGA	8 JD
x2 Ardunio NANO	18 JD
x4 Gearbox Motor	8 JD
x2 SG90 Servo motor	4 JD
ESP 32 with cam	15 JD
x2 Joystick	3 JD
x4 HC-05 Bluetooth Module	32 JD
L298 Dual H-Bridge Motor Driver	3 JD
4 Robot wheels	4 JD
FT232RL USB to TTL Serial Module	6 JD
9 Volt battery and 4x rechargeable battery	22 JD
Wires	1 JD
2 DOF pan and tilt kit	2 JD
Total price	126 JD

Project Cost

5.2 Environmentally

Our project has multiple parts that are recyclable.

We can say our project eco-friendly.

CHAPTER 6 CONCLUSION AND FUTURE WORK

6.1 Conclusion

All of components have chosen with high bear of pressure and waterproof

We have successfully built our project smart robot car using ESP32 CAM , a prototype of our robot that needs multiple adjustments and trials for it to work properly ,so it could detect any leakage of pipes, but unfortunately we couldn't apply in small size diameter .

The esp32 cam and arduino work perfectly taking a video , pictures and save them in TF card .

Since our robot moves through wheels, it will needs to stop the flow of water in the pipes .

Finally, we cannot leave the robot and control it with a distance of more than 10 meters, the controller must follow it from the surface, and it is worth noting that the control distance can be developed by adding antennas.

6.2 Future Work

Our project can be widely upgraded, adding state of art technology to our current design and making it more beneficial to the user.

First, an application for iOS and Android devices can be built so the users can control the obstacle from anywhere using internet connection, and it could be a micro ship to be allowed and get used in home pipelines connections.

It will be upgraded to use it with high flow of water in pipes means no more constrains.

REFERENCES

[2]. Liu and Y. Kleiner, ‘‘State of the art review of inspection technologies for condition assessment of water pipes, ’’Measurement”, vol. 46, no. 1,pp. 1–15, Jan. 2013

[3] http://flir.com

[4] S Demirci, E. Yigit, I. H. Eskidemir, and C. Ozdemir, ‘‘Ground pene-trating radar imaging of water leaks from buried pipes based on back-projection method,’’NDT&E Int., vol. 47, pp. 35–42, Apr. 2012.

[5] S. Demirci, E. Yigit, I. H. Eskidemir, and C. Ozdemir, ‘‘Ground pene-trating radar imaging of water leaks from buried pipes based on back-projection method,’’NDT&E Int., vol. 47, pp. 35–42, Apr. 2012.

[6] S. Demirci, E. Yigit, I. H. Eskidemir, and C. Ozdemir, ‘‘Ground pene-trating radar imaging of water leaks from buried pipes based on back-projection method,’’NDT&E Int., vol. 47, pp. 35–42, Apr. 2012.

[7] https://www.technoac.com

[8] . Liu and Y. Kleiner, ‘‘State of the art review of inspection technologies for condition assessment of water pipes, 'Measurement, vol. 46, no. 1,pp. 1–15, Jan. 2013.

[9] https://cyberbotics.com/doc/guide/introduction-to-webots

[10] https://fritzing.org/

[11] Towerpro, “SERVO MOTOR SG90” datasheet .

[12] https://www.arduino.cc/en/guide/introduction

[13] https://components101.com/modules/joystick-module

[14] https://www.arduino.cc/en/pmwiki.php?n=Main/ArduinoBoardNano

[15] https://components101.com/wireless/hc-05-bluetooth-module

[16] https://components101.com/wireless/hc-05-bluetooth-module

APPENDICES

APPENDIX A: PROJECT CODE

HC-05 Master code: (This code we must uplode on Arduino Nano #1)

```
int xAxis, yAxis;

void setup() {
  Serial.begin(38400);
}

void loop() {
  xAxis = analogRead(A0);
  yAxis = analogRead(A1);


  Serial.write(xAxis/4);
  Serial.write(yAxis/4);
  delay(50);
}
```

****

HC-05 Slave code: (This code we must uplode on Arduino MEGA)

```
#define enA 9
#define in1 4
#define in2 5
#define enB 10
#define in3 6
#define in4 7

int xAxis, yAxis;
int  x = 0;
int  y = 0;

int motorSpeedA = 0;
int motorSpeedB = 0;

void setup() {
  pinMode(enA, OUTPUT);
  pinMode(enB, OUTPUT);
  pinMode(in1, OUTPUT);
  pinMode(in2, OUTPUT);
  pinMode(in3, OUTPUT);
  pinMode(in4, OUTPUT);

  Serial.begin(38400);
```

```
}

void loop() {

  xAxis = 510;
  yAxis = 510;

  // Read the incoming data from the Smartphone Android App
  while (Serial.available() >= 2) {
    x = Serial.read();
    delay(10);
    y = Serial.read();
  }
  delay(10);


  if (x > 60 & x < 220) {
    xAxis = map(x, 220, 60, 1023, 0);
  }
  if (y > 60 & y < 220) {
    yAxis = map(y, 220, 60, 0, 1023);
  }


  if (yAxis < 470) {
    // Set Motor A backward
    digitalWrite(in1, HIGH);
    digitalWrite(in2, LOW);
    // Set Motor B backward
    digitalWrite(in3, HIGH);
    digitalWrite(in4, LOW);

    motorSpeedA = map(yAxis, 470, 0, 0, 255);
    motorSpeedB = map(yAxis, 470, 0, 0, 255);
  }
  else if (yAxis > 550) {
    // Set Motor A forward
    digitalWrite(in1, LOW);
    digitalWrite(in2, HIGH);
    // Set Motor B forward
    digitalWrite(in3, LOW);
    digitalWrite(in4, HIGH);
    motorSpeedA = map(yAxis, 550, 1023, 0, 255);
    motorSpeedB = map(yAxis, 550, 1023, 0, 255);
  }

  else {
    motorSpeedA = 0;
    motorSpeedB = 0;
  }

  // X-axis used for left and right control
  if (xAxis < 470) {
    // Convert the declining X-axis readings from 470 to 0 into increasing 0 to 255 value
    int xMapped = map(xAxis, 470, 0, 0, 255);
    // Move to left - decrease left motor speed, increase right motor speed
    motorSpeedA = motorSpeedA - xMapped;
    motorSpeedB = motorSpeedB + xMapped;
    // Confine the range from 0 to 255
```

```
    if (motorSpeedA < 0) {
      motorSpeedA = 0;
    }
    if (motorSpeedB > 255) {
      motorSpeedB = 255;
    }
  }
  if (xAxis > 550) {

    int xMapped = map(xAxis, 550, 1023, 0, 255);
    // Move right
    motorSpeedA = motorSpeedA + xMapped;
    motorSpeedB = motorSpeedB - xMapped;
// motor a in right side should be low
    // Confine the range from 0 to 255
    if (motorSpeedA > 255) {
      motorSpeedA = 255;
    }
    if (motorSpeedB < 0) {
      motorSpeedB = 0;
    }
  }

  if (motorSpeedA < 70) {
    motorSpeedA = 0;
  }
  if (motorSpeedB < 70) {
    motorSpeedB = 0;
  }
  analogWrite(enA, motorSpeedA); // Send PWM signal to motor A
  analogWrite(enB, motorSpeedB); // Send PWM signal to motor B
}
```

* * * * * *

Servomotor code:

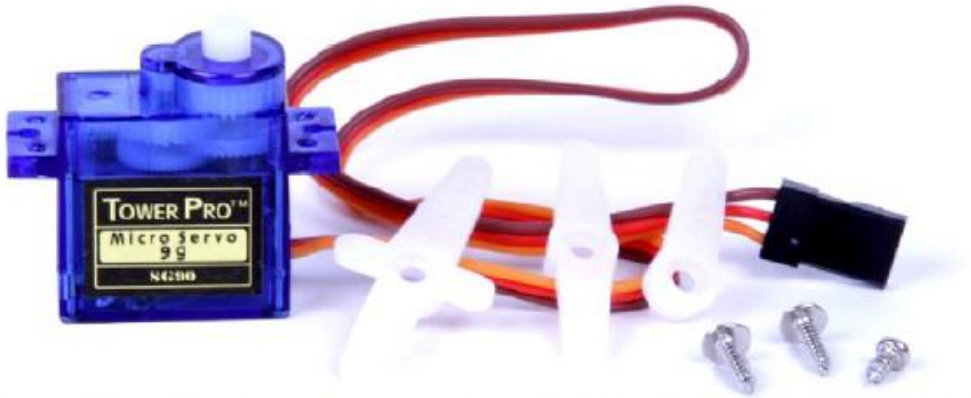
```
#include <SoftwareSerial.h> // TX RX software library for bluetooth


#include <Servo.h> // servo library
Servo myservo1, myservo2; // servo name


int bluetoothTx = 10; // bluetooth tx to 10 pin
int bluetoothRx = 11; // bluetooth rx to 11 pin


SoftwareSerial bluetooth(bluetoothTx, bluetoothRx);


void setup()
{
  myservo1.attach(3); // attach servo signal wire pin 3 and 5
```

```
  myservo2.attach(5);

  //Setup usb serial connection to computer
  Serial.begin(9600);

  //Setup Bluetooth serial connection to master bluetooth
  bluetooth.begin(9600);
}

#include <SoftwareSerial.h> // TX RX software library for bluetooth

#include <Servo.h> // servo library
Servo myservo1, myservo2; // servo name

int bluetoothTx = 10; // bluetooth tx to 10 pin
int bluetoothRx = 11; // bluetooth rx to 11 pin

SoftwareSerial bluetooth(bluetoothTx, bluetoothRx);

void setup()
{
  myservo1.attach(3); // attach servo signal wire to pin 3 and 5
  myservo2.attach(5);

  //Setup usb serial connection to computer
  Serial.begin(9600);

  //Setup Bluetooth serial connection to android
  bluetooth.begin(9600);
}
void loop()
{
  //Read from bluetooth and write to usb serial
  if(bluetooth.available()>= 2 )
  {
    unsigned int servopos = bluetooth.read();
```

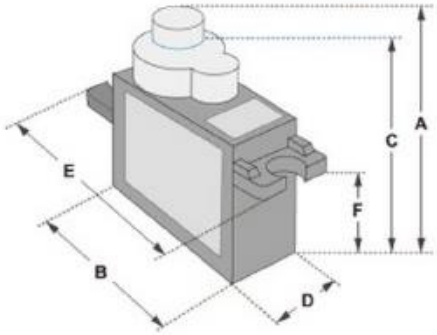
```
  unsigned int servopos1 = bluetooth.read();
  unsigned int realservo = (servopos1 *256) + servopos;
  Serial.println(realservo);

  if (realservo >= 1000 && realservo <1180) {
    int servo1 = realservo;
    servo1 = map(servo1, 1000, 1180, 0, 180);
    myservo1.write(servo1);
    Serial.println("Servo 1 ON");
    delay(10);
  }
  if (realservo >= 2000 && realservo <2180) {
    int servo2 = realservo;
    servo2 = map(servo2, 2000, 2180, 0, 180);
    myservo2.write(servo2);
    Serial.println("Servo 2 ON");
    delay(10);
```

APPENDIX B: SERVO MOTOR DATASHEET

SERVO MOTOR SG90 DATA SHEET

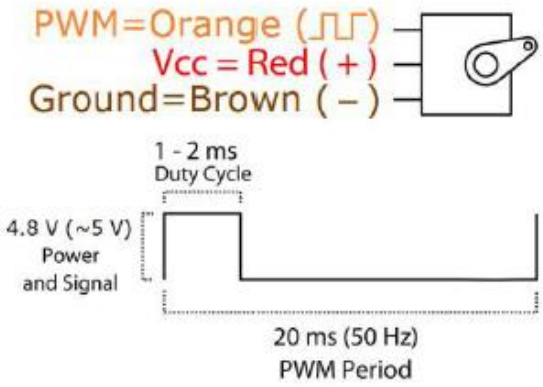


Tiny and lightweight with high output power. Servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but smaller. You can use any servo code, hardware or library to control these servos. Good for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. It comes with a 3 horns (arms) and hardware.



Dimensions & Specifications
A (mm) : 32
B (mm) : 23
C (mm) : 28.5
D (mm) : 12
E (mm) : 32
F (mm) : 19.5
Speed (sec) : 0.1
Torque (kg-cm) : 2.5
Weight (g) : 14.7
Voltage : 4.8 - 6

Position "0" (1.5 ms pulse) is middle, "90" (~2ms pulse) is middle, is all the way to the right, "-90" (~1ms pulse) is all the way to the left.



APPENDIX C: Gearbox Motor DATASHEET

DC Gearbox Motor - "TT Motor" - 200RPM - 3 to 6VDC

PRODUCT ID: 3777

Perhaps you've been assembling a new robot friend, adding a computer for a brain and other fun personality touches. Now the time has come to let it leave the nest and fly on its own wings- err, *wheels!*

These durable (but affordable!) plastic gearbox motors (also known as 'TT' motors) are an easy, low-cost way to get your projects moving. This is a TT DC Gearbox Motor with a gear ratio of 1:48, and it comes with 2 x 200mm wires with breadboard-friendly 0.1" male connectors. Perfect for plugging into a breadboard or terminal blocks.

You can power these motors with 3VDC up to 6VDC, they'll of course go a little faster at the higher voltages. We grabbed one motor and found these stats when running it from a bench-top supply

- At 3VDC we measured 150mA @ 120 RPM no-load, and 1.1 Amps when stalled
- At 4.5VDC we measured 155mA @ 185 RPM no-load, and 1.2 Amps when stalled
- At 6VDC we measured 160mA @ 250 RPM no-load, and 1.5 Amps when stalled

Note that these are very basic motors, and have no built-in encoders, speed control or positional feedback. Voltage goes in, rotation goes out! There will be variation from motor to motor, so a separate feedback system is required if you need precision movement.

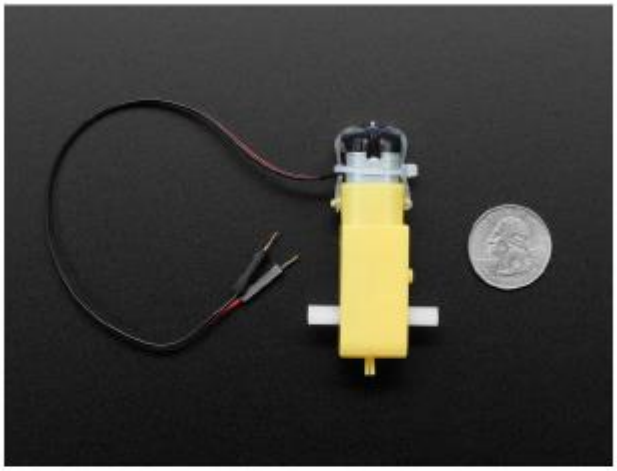
Comes 1 x per order, with just the motor + wires. You *cannot* drive these directly from a microcontroller, a high-current motor driver is required! We recommend our DRV8833 motor driver for these motors, as it works well down to 3V and can be set up with current limiting since the stall current on these can get high. The TB6612 can also be used, it's on our shields and wings, but you'll need to supply at least 4.5V - which is what you'll likely want to run these motors at anyhow!

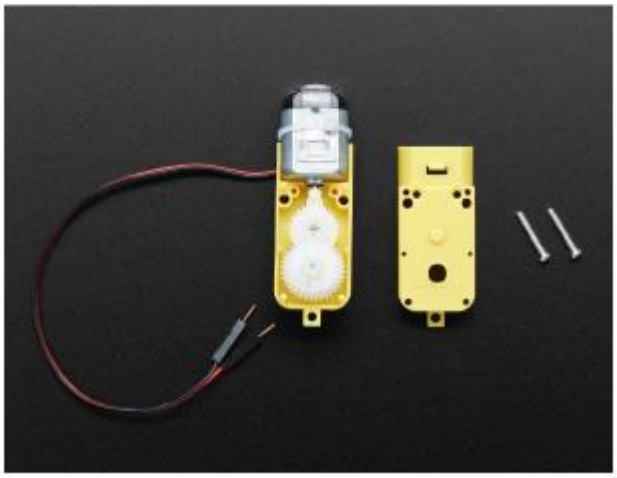
We have a range of wheels, add-ons and accessories for these motors so you can bling out your bot just the way you like.

TECHNICAL DETAILS

- Rated Voltage: 3~6V
- Continuous No-Load Current: 150mA +/- 10%
- Min. Operating Speed (3V): 90+/- 10% RPM
- Min. Operating Speed (6V): 200+/- 10% RPM
- Torque: 0.15Nm ~0.60Nm
- Stall Torque (6V): 0.8kg.cm
- Gear Ratio: 1:48
- Body Dimensions: 70 x 22 x 18mm
- Wires Length: 200mm & 28 AWG
- Weight: 30.6g

Product Weight: 30.6g / 1.1oz





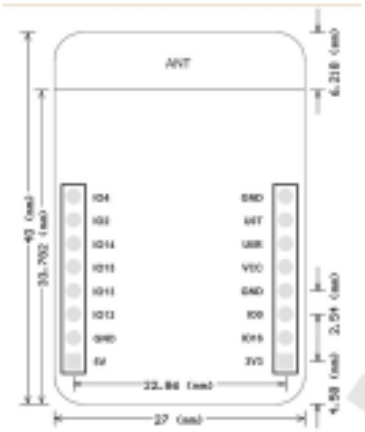
https://www.adafruit.com/product/3777 4-30-18

APPENDIX D: ESP32 CAM DATASHEET

安信可科技 Ai-Thinker

ESP32-CAM Wi-Fi+BT SoC Module V1.0

ESP32-CAM Module



Features

- The smallest 802.11b/g/n Wi-Fi BT SoC Module
- Low power 32-bit CPU,can also serve the application processor
- Up to 160MHz clock speed, Summary computing power up to 600 DMIPS
- Built-in 520 KB SRAM, external 4MPSRAM
- Supports UART/SPI/I2C/PWM/ADC/DAC
- Support OV2640 and OV7670 cameras,Built-in Flash lamp.
- Support image WiFI upload
- Support TF card
- Supports multiple sleep modes.
- Embedded Lwip and FreeRTOS
- Supports STA/AP/STA+AP operation mode
- Support Smart Config/AirKiss technology
- Support for serial port local and remote firmware upgrades (FOTA)

Overview

The ESP32-CAM has a very competitive small-size camera module that can operate independently as a minimum system with a footprint of only 27*40.5*4.5mm and a deep sleep current of up to 6mA.

ESP-32CAM can be widely used in various IoT applications. It is suitable for home smart devices, industrial wireless control, wireless monitoring, QR wireless identification, wireless positioning system signals and other IoT applications. It is an ideal solution for IoT applications.

ESP-32CAM adopts DIP package and can be directly inserted into the backplane to realize rapid production of products, providing customers with high-reliability connection mode, which is convenient for application in various IoT hardware terminals.

Page 1 of 4

Product Specifications

Module Model	ESP32-CAM
Package	DIP-16
Size	27*40.5*4.5（±0.2）mm
SPI Flash	Default 32Mbit
RAM	520KB SRAM +4M PSRAM
Bluetooth	Bluetooth 4.2 BR/EDR and BLE standards
Wi-Fi	802.11 b/g/n/
Support interface	UART、SPI、I2C、PWM
Support TF card	Maximum support 4G
IO port	9
UART Baudrate	Default 115200 bps
Image Output Format	JPEG(OV2640 support only),BMP,GRAYSCALE
Spectrum Range	2412 ~2484MHz
Antenna	Onboard PCB antenna, gain 2dBi
Transmit Power	802.11b: 17±2 dBm (@11Mbps) 802.11g: 14±2 dBm (@54Mbps) 802.11n: 13±2 dBm (@MCS7)
Receiving Sensitivity	CCK, 1 Mbps : -90dBm CCK, 11 Mbps: -85dBm 6 Mbps (1/2 BPSK): -88dBm 54 Mbps (3/4 64-QAM): -70dBm MCS7 (65 Mbps, 72.2 Mbps): -67dBm
Power Dissipation	Turn off the flash lamp:180mA@5V Turn on the flash lamp and turn on the brightness to the maximum:310mA@5V Deep-sleep: Minimum power consumption can be achieved 6mA@5V Modem-sleep: Minimum up to 20mA@5V Light-sleep: Minimum up to 6.7mA@5V
Security	WPA/WPA2/WPA2-Enterprise/WPS
Power Supply Range	5V
Operating Temperature	-20 ℃ ~ 85 ℃
Storage Environment	-40 ℃ ~ 90 ℃ , < 90%RH

Weight	10g

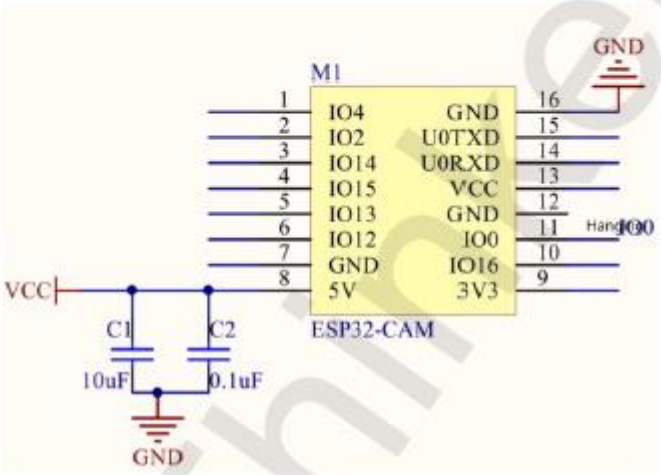
ESP32-CAM module picture output format rate

Format Size	QQVGA	QVGA	VGA	SVGA
JPEG	6	7	7	8
BMP	9	9	-	-
GRAYSCALE	9	8	-	-

Internal Pin Connect

CAM	ESP32	SD	ESP32
D0	PIN5	CLK	PIN14
D1	PIN18	CMD	PIN15
D2	PIN19	DATA0	PIN2
D3	PIN21	DATA1/Flash lamp	PIN4
D4	PIN86	DATA2	PIN12
D5	PIN89	DATA3	PIN13
D6	PIN84		
D7	PIN85		
XCLK	PIN0		
PCLK	PIN22		
VSYNC	PIN25		
HREF	PIN23		
SDA	PIN26		
SCL	PIN27		
POWER PIN	PIN32		

Minimum system diagram



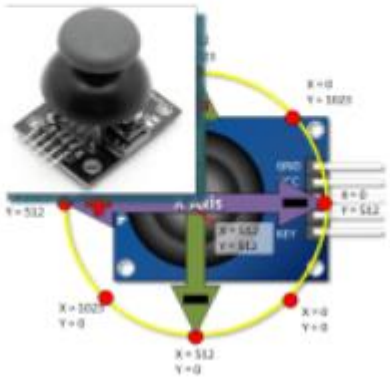
Contact US

Shenzhen Ai-Thinker Technology Co., Ltd
Address: 7/F, Fengze Building B, Huafeng Industrial Park 2th, Hangkong street,Xixiang Raod, Baoan, Shenzhen China
Website:www.ai-thinker.com Tel: 0755-29162996 E-mail:support@aithinker.com

APPENDIX F: JOYSTICK MODULE DATASHEET

Arduino PS2 Joystick Tutorial:

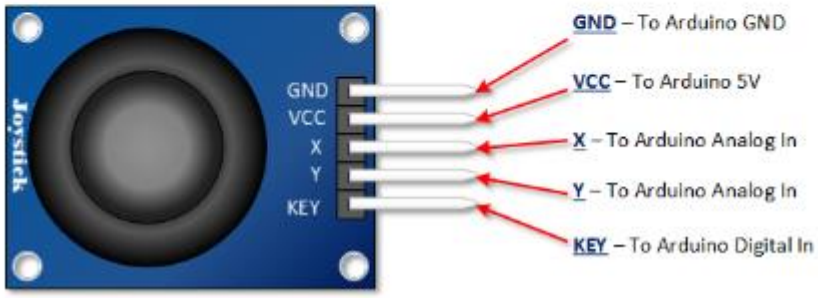
A Versatile Input Device:



The PS2 style joystick is a thumb operated device, that when put to creative use, offers a convenient way of getting operator input. Its fundamentally consists of two potentiometers and a push button switch. The two potentiometers indicate which direction the potentiometer is being pushed.The switch sends a low (or ground) when the joy stick knob is pressed.

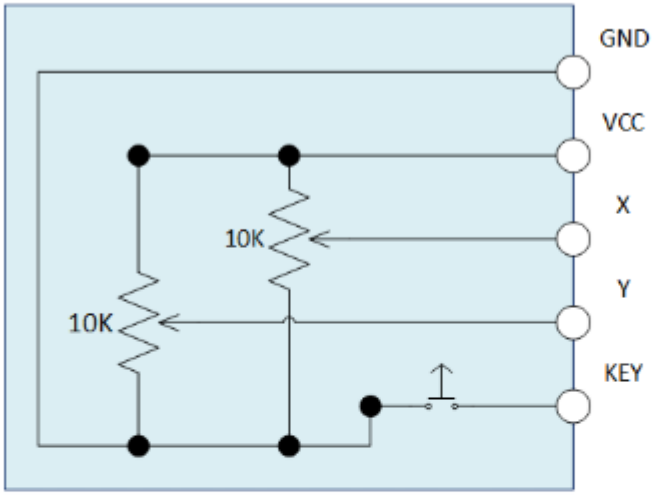
Arduino PS2 Joystick Pin Outs

This input device interfaces to your Arduino via five pins. Three of which are inputs to your Arduino, while the remaining two supply voltage and ground.



Arduino PS2 Joystick Schematic

As you can see in the schematic below, full deflection of a potentiometer in either direction will provide ground or the supply voltage as an output.

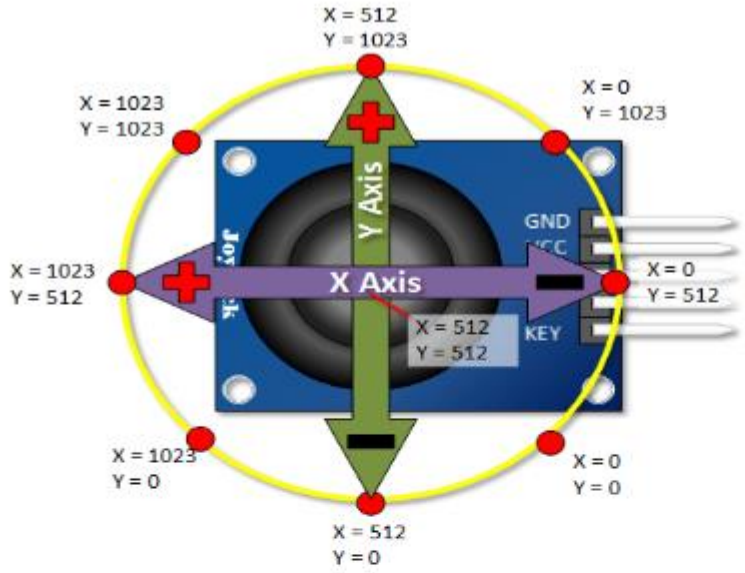


Arduino PS2 Joystick Output Orientation

In order to put this thumb control to use, you are going to want to understand which direction is X and which direction is Y. You will also need to decipher the direction it is being pushed in either the X or the Y direction.

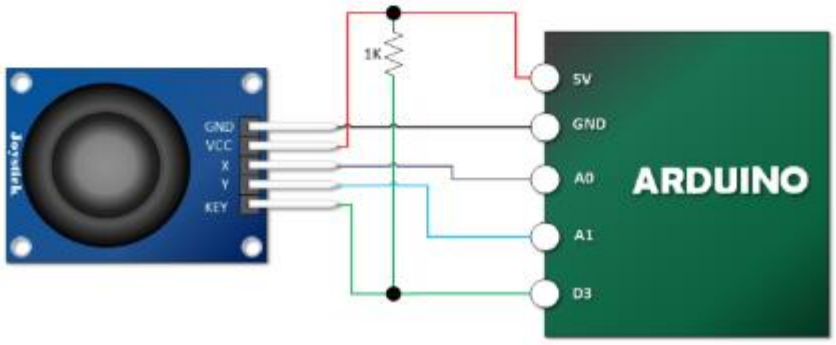
In this tutorial we are using analog inputs to measure the joystick position. The analog inputs provided indications that range between 0 and 1023.

The graphic below shows the X and Y directions and also gives an indication of how the outputs will respond when the joystick is pushed in various directions. Keep in mind, the graphic you see is based on my Deek-Robot model and may in fact differ a little with yours. It that's the case, experiment a little and draw your own sketch so that the orientations are clear.



Assemble the PS2 Joystick Project:

Note that you use pull up resistor between the key switch and the digital input. Once you move beyond experimentation, It is highly recommend some sort of software or hardware de-bounce for this switch as well.



APPENDIX G: L298 H-BRIDGE DATASHEET

L298 Dual H-Bridge Motor Driver

User's Guide

L298 Dual H-Bridge Motor Driver User's Guide

Overview

The Motor Shield is based on the L298, which is a dual full-bridge driver designed to drive inductive loads such as relays, solenoids, DC and stepping motors. It lets you drive two DC motors , controlling the speed and direction of each one independently.

Summary

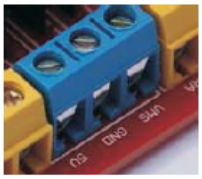
Operating Voltage 4V to 35V
Motor controller L298N, Drives 2 DC motors or 1 stepper motor
Max current 2A per channel or 4A max
Free running stop and brake function
Chip: ST L298N
Logic power supply:5v
Max power:25w
Weight: 35g
Size:55mm x 60mm x 30mm
Storage temperature:-25 to +135

L298 Dual H-Bridge Motor Driver User's Guide

CSA: Between this pin and ground is conncted the sense resistor to control the current of the load. Enable-------- Ignore current detection function
CSB: Between this pin and ground is conncted the sense resistor to control the current of the load. Enable-------- Ignore current detection function

Logic power indicator

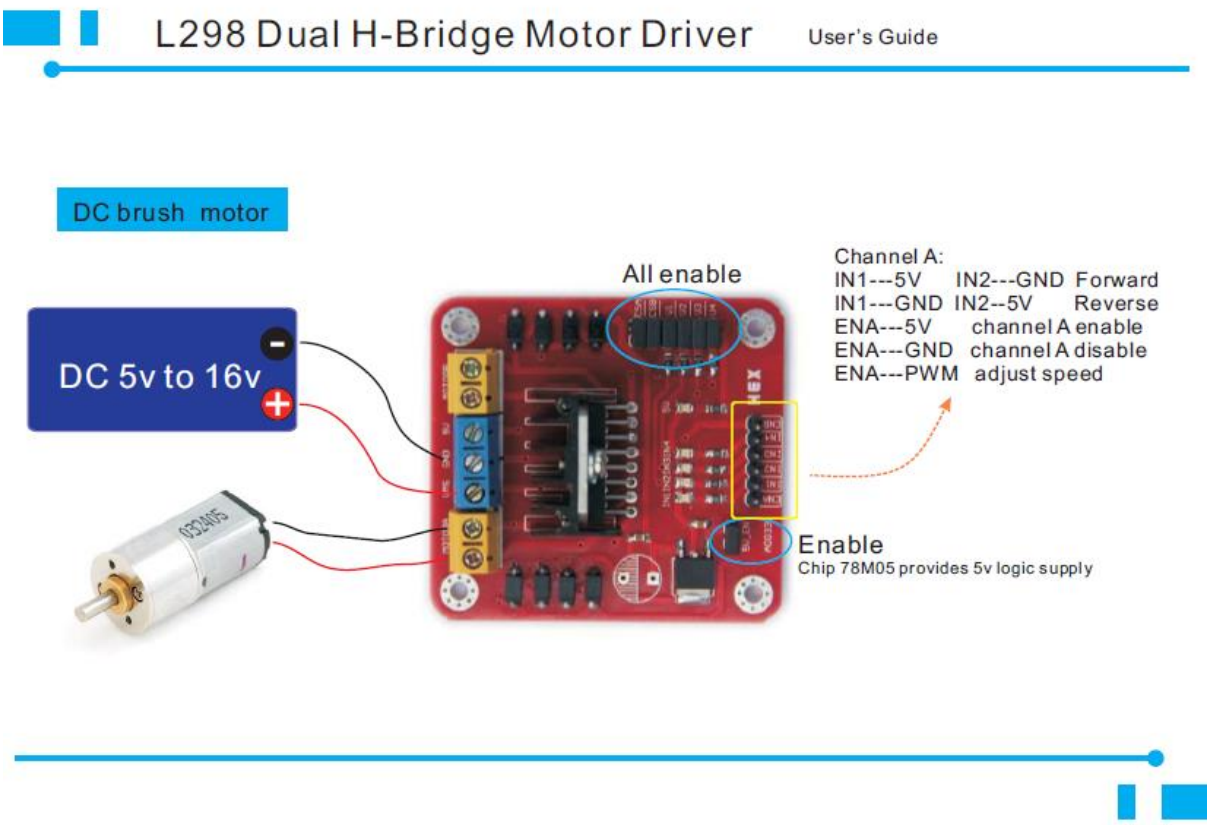
5V-EN: Enable----78M05 worked ,output DC 5V
Disable----78M05 do not work . Need input DC 5V
The module need DC 5V always, for logic supply.

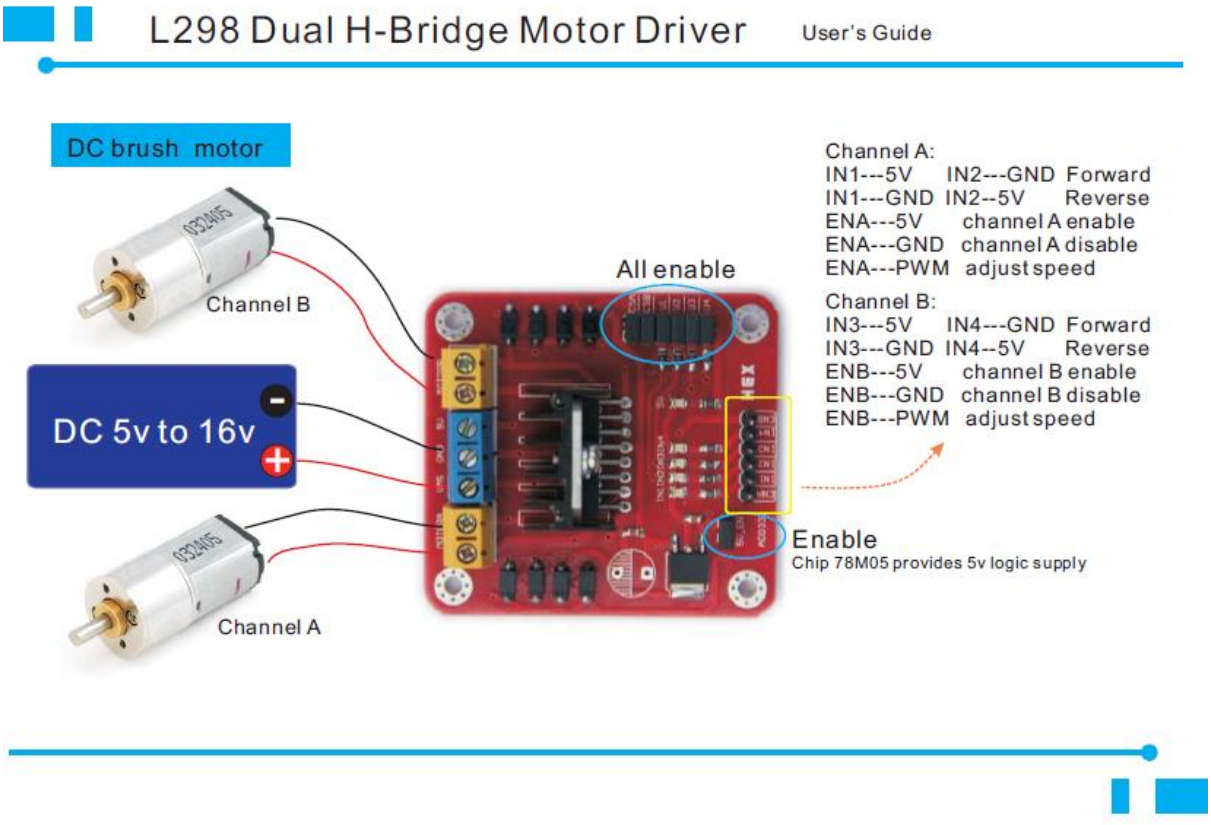


5V_EN:
Enable: [5V] can output DC 5V.
Disable:[5v] need input DC 5V.

The pull-up resistor enabled.
U1---Enable In1 pull-up resistor [10k].
U2---Enable In2 pull-up resistor [10k].
U3---Enable In3 pull-up resistor [10k].
U4---Enable In4 pull-up resistor [10k].

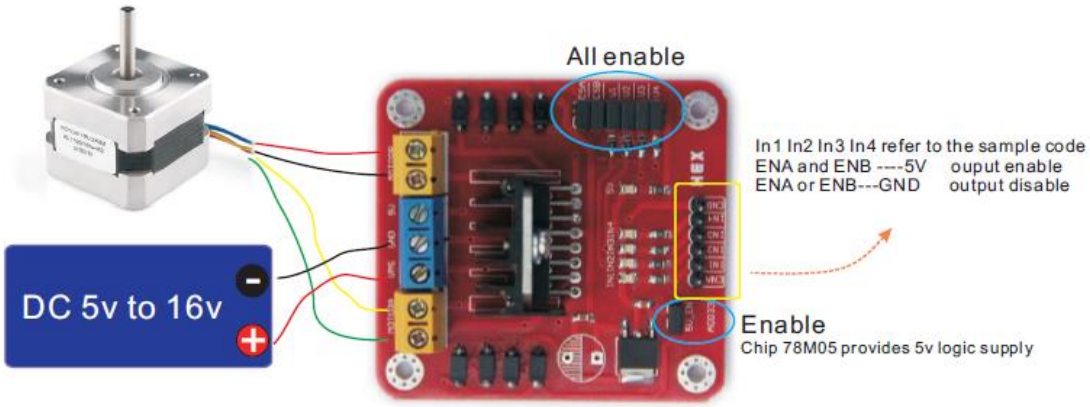
IN1 IN2 :TTL Compatible Inputs of the Bridge A
In3 In4 :TTL Compatible Inputs of the Bridge B.
ENA ENB:TTL Compatible Enable Input: the L state disables the bridge A(enable A) and/or the bridge B (enable B).


L298 Dual H-Bridge Motor Driver
User's Guide
DC brush motor
DC 5v to 16v
All enable
Channel A:
IN1---5V IN2---GND Forward
IN1---GND IN2--5V Reverse
ENA---5V channel A enable
ENA---GND channel A disable
ENA---PWM adjust speed
Enable
Chip 78M05 provides 5v logic supply


L298 Dual H-Bridge Motor Driver
User's Guide
DC brush motor
Channel B
DC 5v to 16v
Channel A
All enable
Channel A:
IN1---5V IN2---GND Forward
IN1---GND IN2--5V Reverse
ENA---5V channel A enable
ENA---GND channel A disable
ENA---PWM adjust speed
Channel B:
IN3---5V IN4---GND Forward
IN3---GND IN4--5V Reverse
ENB---5V channel B enable
ENB---GND channel B disable
ENB---PWM adjust speed
Enable
Chip 78M05 provides 5v logic supply

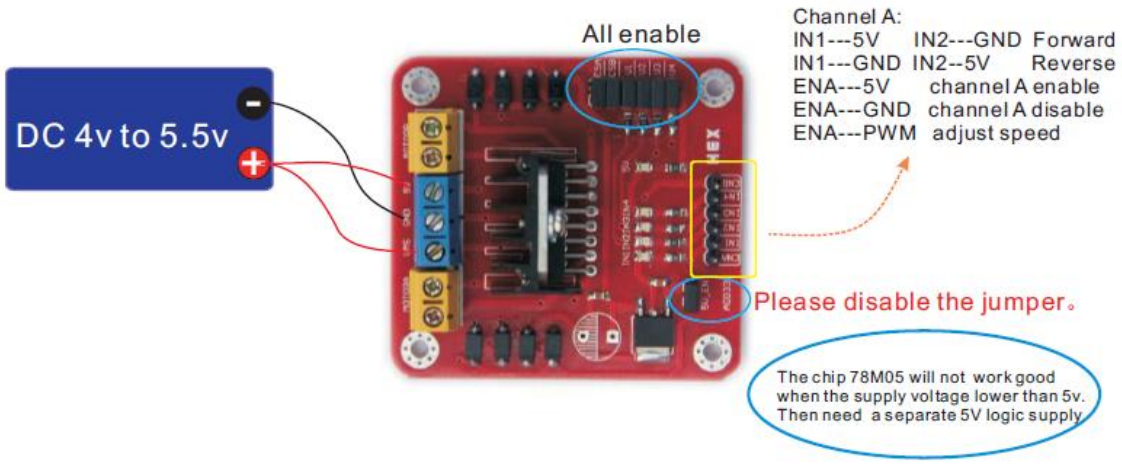
L298 Dual H-Bridge Motor Driver User's Guide

Stepper motor



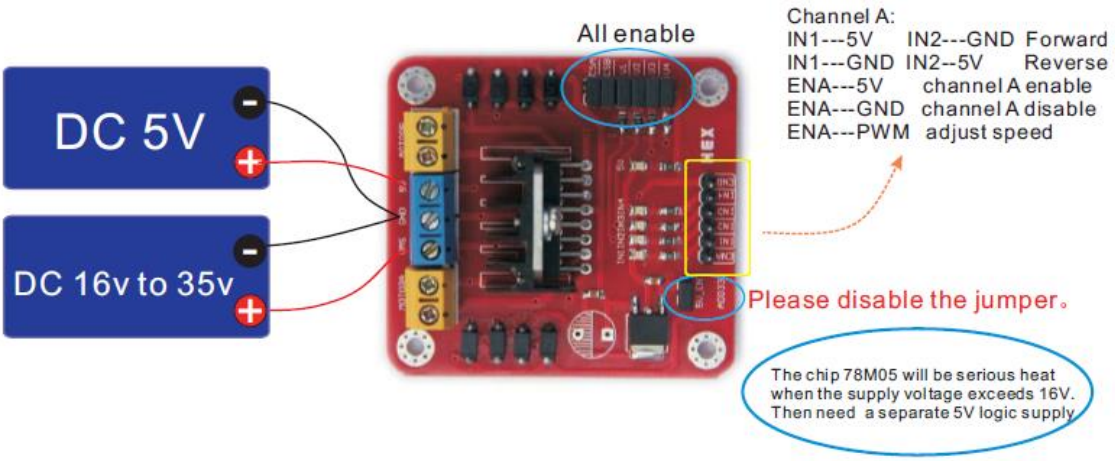
L298 Dual H-Bridge Motor Driver User's Guide

Motor powered with DC 4V to 5.5V



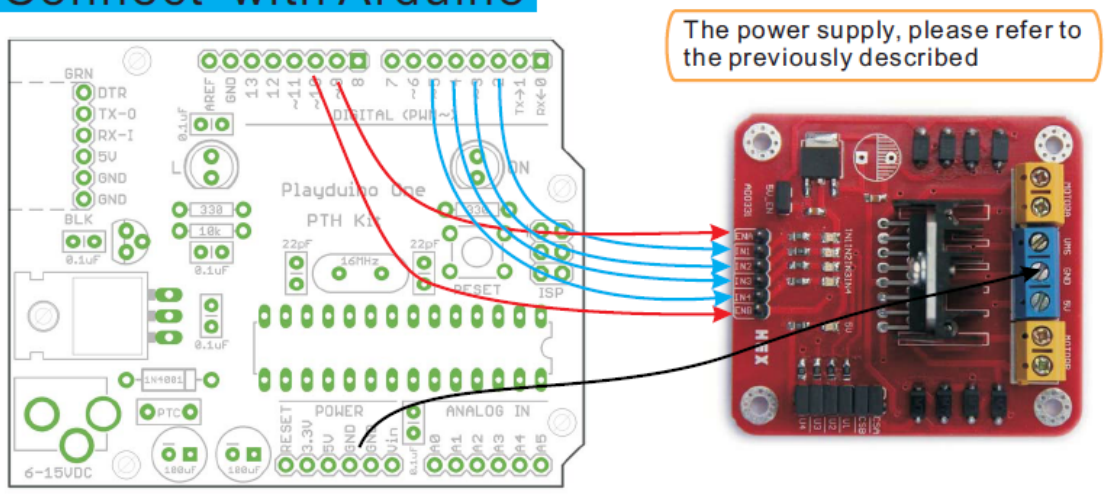
L298 Dual H-Bridge Motor Driver User's Guide

Motor powered with DC 16V to 35V



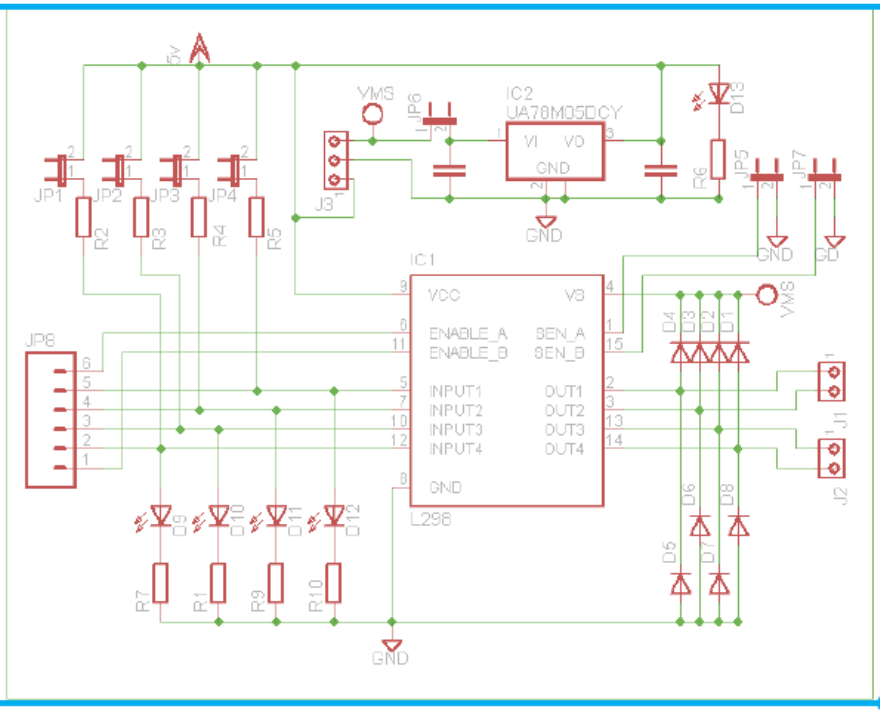
L298 Dual H-Bridge Motor Driver User's Guide

Connect with Arduino



L298 Dual H-Bridge Motor Driver User's Guide

Schematic:



APPENDIX H: HC-05 BLUETOOTH DATASHEET

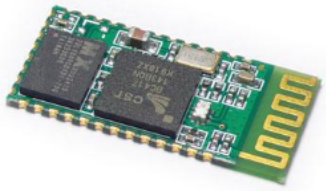
ITead Studio
Make innovation easier

Tech Support: info@iteadstudio.com

HC-05

-Bluetooth to Serial Port Module

Overview



HC-05 module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup.

Serial port Bluetooth module is fully qualified Bluetooth V2.0+EDR (Enhanced Data Rate) 3Mbps Modulation with complete 2.4GHz radio transceiver and baseband. It uses CSR Bluecore 04-External single chip Bluetooth system with CMOS technology and with AFH(Adaptive Frequency Hopping Feature). It has the footprint as small as 12.7mmx27mm. Hope it will simplify your overall design/development cycle.

Specifications

Hardware features

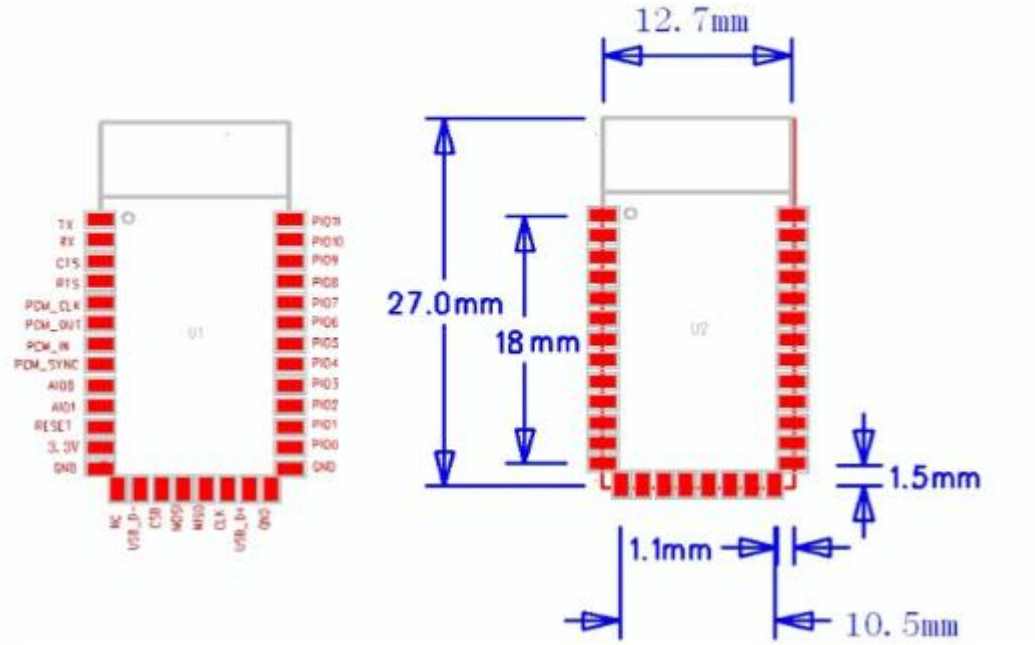
- Typical -80dBm sensitivity
- Up to +4dBm RF transmit power
- Low Power 1.8V Operation ,1.8 to 3.6V I/O
- PIO control
- UART interface with programmable baud rate
- With integrated antenna
- With edge connector

HC-05 Bluetooth module iteadstudio.com 06.18.2010

Software features

- Default Baud rate: 38400, Data bits:8, Stop bit:1,Parity:No parity, Data control: has. Supported baud rate: 9600,19200,38400,57600,115200,230400,460800.
- Given a rising pulse in PIO0, device will be disconnected.
- Status instruction port PIO1: low-disconnected, high-connected;
- PIO10 and PIO11 can be connected to red and blue led separately. When master and slave are paired, red and blue led blinks 1time/2s in interval, while disconnected only blue led blinks 2times/s.
- Auto-connect to the last device on power as default.
- Permit pairing device to connect as default.
- Auto-pairing PINCODE:"0000" as default
- Auto-reconnect in 30 min when disconnected as a result of beyond the range of connection.

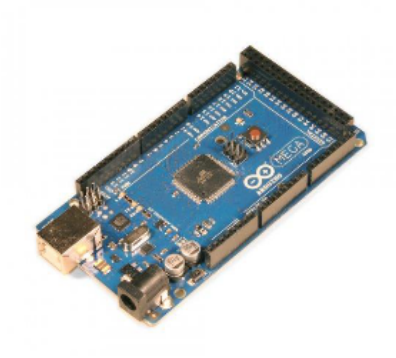
Hardware



PIN Name	PIN #	Pad type	Description	Note
GND	13 21 22	VSS	Ground pot	
3.3 VCC	12	3.3V	Integrated 3.3V (+) supply with On-chip linear regulator output within 3.15-3.3V	
AIO0	9	Bi-Directional	Programmable input/output line	
AIO1	10	Bi-Directional	Programmable input/output line	
PIO0	23	Bi-Directional RX EN	Programmable input/output line, control output for LNA(if fitted)	
PIO1	24	Bi-Directional TX EN	Programmable input/output line, control output for PA(if fitted)	
PIO2	25	Bi-Directional	Programmable input/output line	
PIO3	26	Bi-Directional	Programmable input/output line	
PIO4	27	Bi-Directional	Programmable input/output line	
PIO5	28	Bi-Directional	Programmable input/output line	
PIO6	29	Bi-Directional	Programmable input/output line	
PIO7	30	Bi-Directional	Programmable input/output line	
PIO8	31	Bi-Directional	Programmable input/output line	
PIO9	32	Bi-Directional	Programmable input/output line	
PIO10	33	Bi-Directional	Programmable input/output line	
PIO11	34	Bi-Directional	Programmable input/output line	

APPENDIX I: ARDUINO MEGA DATASHEET

Arduino Mega2560 Rev3



The Arduino Mega 2560 is a microcontroller board based on the ATmega2560. 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 Uno, Duemilanove or Diecimila.

The Mega 2560 is an update to the Arduino Mega, which it replaces.

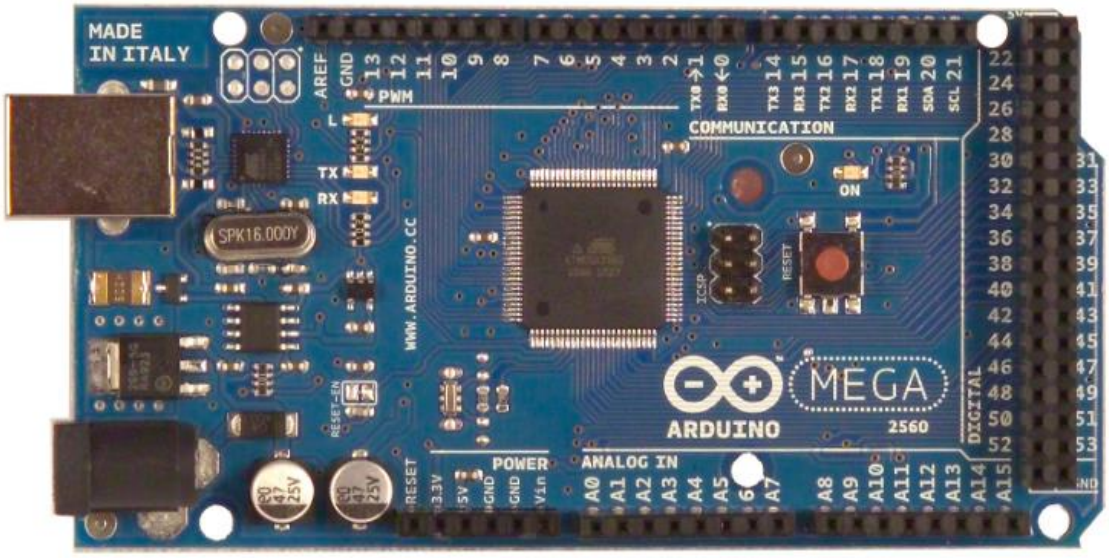
The Mega 2560 is an update to the Arduino Mega, which it replaces.

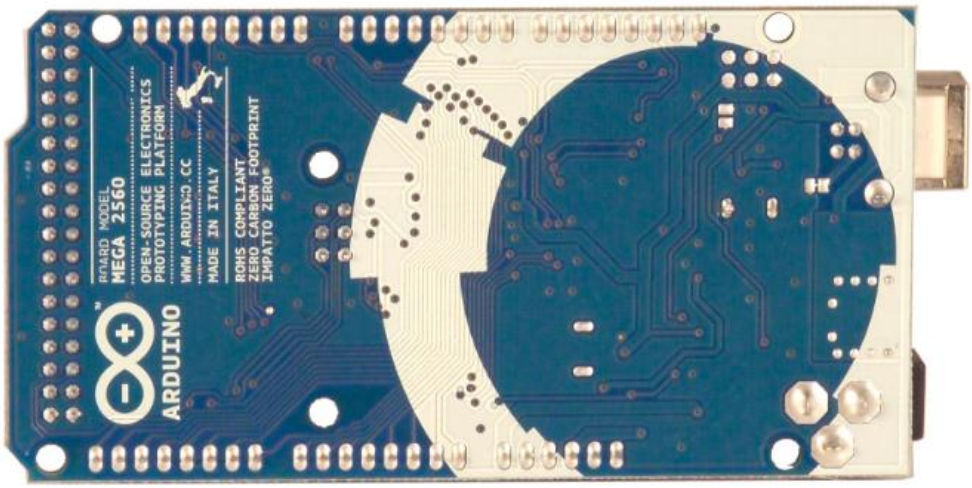
Additional features coming with the R3 version are:

- ATmega16U2 instead 8U2 as USB-to-Serial converter.
- 1.0 pinout: added SDA and SCL pins for TWI communication placed near to the AREF pin and two other new pins placed near to the RESET pin, the IOREF that allow the shields to adapt to the voltage provided from the board and the second one is a not connected pin, that is reserved for future purposes.
- stronger RESET circuit.

Technical Specifications

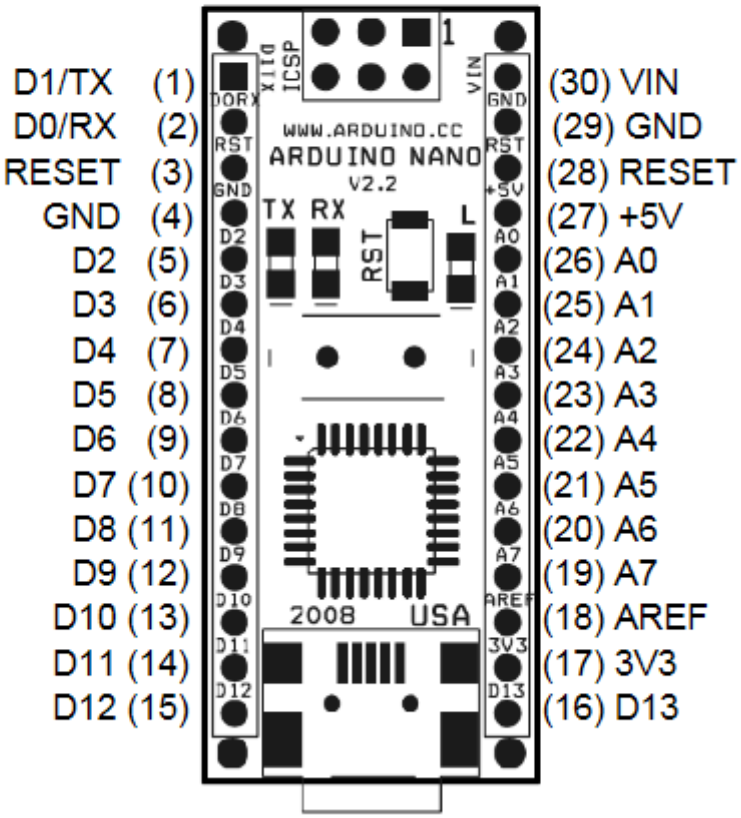
Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 14 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM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


MADE
IN ITALY
COMMUNICATION
MEGA
ARDUINO
2560
POWER
ANALOG IN
DIGITAL


ARDUINO
MEGA 2560
OPEN-SOURCE ELECTRONICS
PROTOTYPING PLATFORM
MADE IN ITALY
ROHS COMPLIANT
ZERO CARBON FOOTPRINT
IMPATTO ZERO®

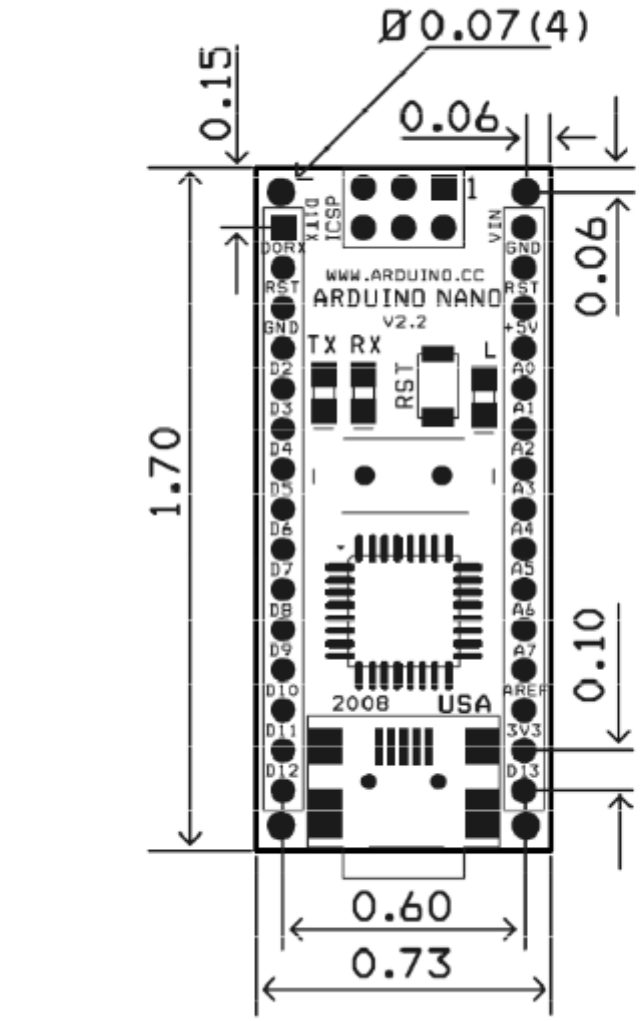
APPENDIX J: ARDUINO NANO DATASHEET

Arduino Nano Pin Layout



Pin No.	Name	Type	Description
1-2, 5-16	D0-D13	I/O	Digital input/output port 0 to 13
3, 28	RESET	Input	Reset (active low)
4, 29	GND	PWR	Supply ground
17	3V3	Output	+3.3V output (from FTDI)
18	AREF	Input	ADC reference
19-26	A7-A0	Input	Analog input channel 0 to 7
27	+5V	Output or Input	+5V output (from on-board regulator) or +5V (input from external power supply)
30	VIN	PWR	Supply voltage

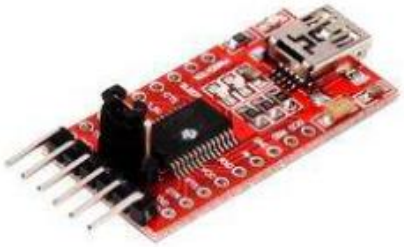
Arduino Nano Mechanical Drawing



ALL DIMENTIONS ARE IN INCHES

APPENDIX K: FT232RL USB TO TTL SERIAL MODULE DATASHEET

FT232RL USB TO TTL 5V 3.3V Convertor



The USB to TTL serial adapter is based on the high quality and very popular FTDI FT232RL chipset and is an excellent way to connect TTL serial devices to a PC through a USB port.

This USB to TTL serial adapter is ideal for many uses, including:

- Programming microprocessors such as ARM, AVR, etc
- Working with computing hardware such as routers and switches
- Serial communication with many devices such as GPS devices
- Serial terminals on devices like the Raspberry Pi

Unlike most USB to TTL serial adapters, this adapter supports both 5V AND 3.3V operation! Simply set the jumper as required to choose between 5V and 3.3V as labelled on the board.

The adapter comes with a right-angle connector fitted allowing you to use it straight away. If you need to access any of the other inputs or outputs of the FT232RL, all the useful signals are provided as through-hole solder pads - ideal for use with straight headers into a breadboard, for example.

The main connector has 6 pins:

- **DTR:** Data Terminal Ready - an output used for flow control
- **RX:** Serial data Receive pin
- **TX:** Serial data Transmit pin
- **VCC:** Positive voltage output - this is controlled by the jumper. If the jumper is set to 5V, this will provide a 5V output. If the jumper is set to 3.3V, this will provide a 3.3V output.
- **CTS:** Clear To Send - an input used for flow control
- **GND:** Ground or 0V

For most uses, you can simply connect the following pins:

- **RX** on this board to the **TX** pin on your device
- **TX** on this board to the **RX** pin on your device
- **GND** on this board to **GND** on your device

The **VCC** pin is ideal for powering small devices such as homemade circuits. This pin should not be connected when a device has a separate power supply as this may damage both devices.

Please note that in 5V mode the maximum current draw on this pin is approximately 500mA. In 3.3V mode the maximum current draw on **VCC** is approximately 50mA.

There are also several pins available as solder pads. These pins are labelled on the board. Connecting to these pins is not usually required and you should check the FTDI datasheet before doing so.

This adapter supports the following operating systems:

- Windows 2000 (32 bit)
- Windows XP (32 and 64 bit)
- Windows Vista (32 and 64 bit)
- Windows 7 (32 and 64 bit)
- Windows 8 (32 and 64 bit)
- Windows 8.1 (32 and 64 bit)
- Linux 2.6+
- Mac OS X 10.4, 10.5, 10.6, 10.7, 10.8 and 10.9

The FT232RL is a USB to serial UART interface IC with the following advanced features:

- Single chip USB to asynchronous serial data transfer interface.
- Entire USB protocol handled on the chip. No USB specific firmware programming required.
- Fully integrated 1024 bit EEPROM storing device descriptors and CBUS I/O configuration.
- Fully integrated USB termination resistors.
- Fully integrated clock generation with no external crystal required plus optional clock output selection enabling a glue-less interface to external MCU or FPGA.
- Data transfer rates from 300 baud to 3 Mbaud (RS422, RS485, RS232) at TTL levels.
- 128 byte receive buffer and 256 byte transmit buffer utilising buffer smoothing technology to allow for high data throughput.

Features of FT232RL USB TO TTL 5V 3.3V Convertor

- Material: PCB + Electronic Component
- Support 3.3V, 5V
- Main Colour: Red
- Chipset: FT232RL
- USB power has over current protection, using 500MA self-restore fuse
- RXD/TXD transceiver communication indicator
- Pin definition: DTR,RXD,TX,VCC,CTS,GND
- Pitch:2.54mm
- Module Size: About 36mm(length)*17.5mm(width)
- Interface : Mini USB