

# FABRICATION OF AUTOMATED FLOOR CLEANER MODEL

Hema Latha K<sup>1</sup>, Venkat Harish S<sup>2</sup>, Vardhini C<sup>3</sup>, Tulasi Ram N<sup>4</sup>

<sup>1\*,2</sup>Dept. of Mechanical Engineering, St. Martin's Engineering College, Hyderabad, India

<sup>3</sup>Dept. of Mechanical Engineering, Sanjay Gandhi Polytechnic, Ballari, India

<sup>4</sup>Dept. of Mechanical Engineering, St. Martin's Engineering College, Hyderabad, India

## ABSTRACT

In this fast-paced world, one hardly finds time to do the basic necessities of life. Due to which households and industries of today are getting more efficient, convenient and smarter. The main objective of this project is to design and implement an automated floor cleaner which reduces human effort and saves human time. In previous years flooring of households and offices are cleaned using vacuum cleaners etc, which requires human effort. The robot shines in this area. It is fully automatic ie. Can detect obstacles and has a capability to maneuver by itself. Automatic floor cleaner is a compact robotics system which provides floor cleaning service in room and big offices reducing human labor. Basically as a robot it eliminates human error and provide cleaning activity with much more efficiency. If we clean the floor manually then there is a possibility that the operator will leave some portion of the floor. Also due to manual labor involved this is time consuming and irritating to clean the floor. Also in big offices floor area is very huge and the people involved there for cleaning purpose cannot clean it much more efficiently. This is where the robot comes as an advantage. Also the robot is small and compact in size.

**Keywords:** Arduino UNO, DC motor with driver, SERVO MOTOR, Rechargeable Battery, LED indication

## I. INTRODUCTION

Automatic floor cleaner is a compact robotics system which provides floor cleaning service in room and big offices reducing human labor. Basically, as a robot it eliminates human error and provide cleaning activity with much more efficiency. If we clean the floor manually then there is a possibility that the operator will leave some portion of the floor. Also due to manual labor involved this is time consuming and irritating to clean the floor. Also, in big offices floor area is very huge and the people involved there for cleaning purpose cannot clean it much more efficiently. This is where the robot comes as an advantage. Also, the robot is small and compact in size. So we can carry it and place it wherever we can on the house. Also, in industries the robot is very cost effective as compared to manual labor involved. The flexibility, time saving and efficiency make the robot a clean choice for cleaning the floor.

The project aims designing automatic floor cleaning robot using arduino . Arduino is an open source, computer hardware and software company, project, and user community that designs and manufactures microcontroller kits for building digital devices and interactive objects that can sense and control objects in the physical world.

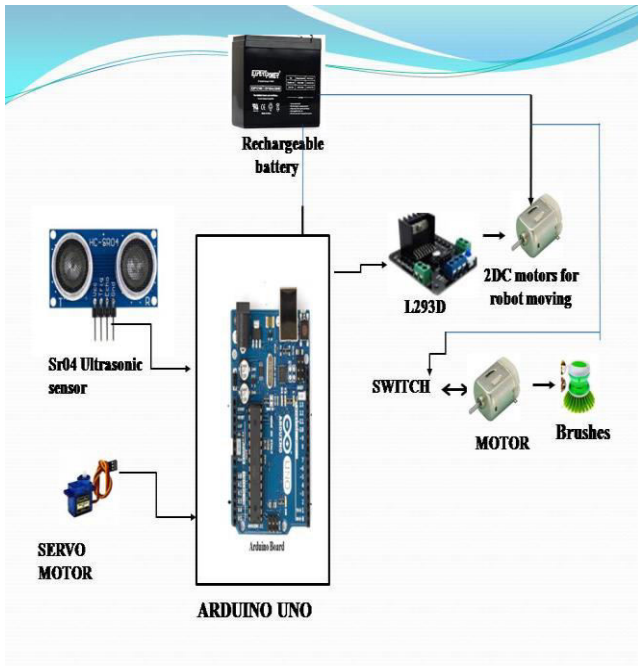
A dc motor uses electrical energy to produce mechanical

energy, very typically through the interaction of magnetic fields and current-carrying conductors. The reverse process, producing electrical energy from mechanical energy, is accomplished by an alternator, generator or dynamo.

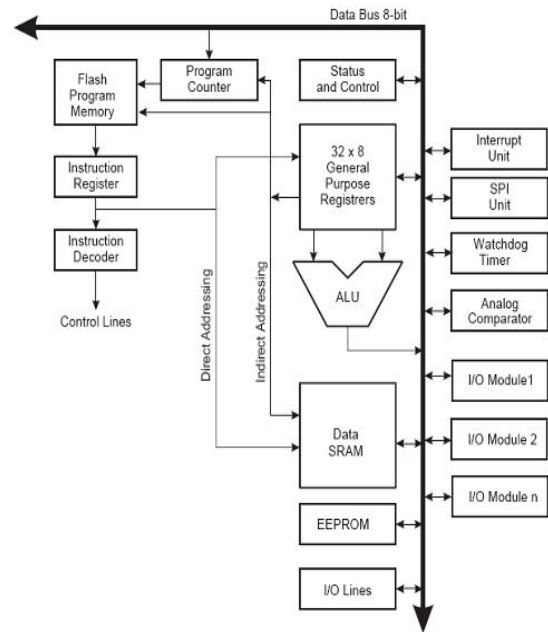
The controlling device of the robot system is Arduino Microcontroller. DC motors along with driver, sro4 ultrasonic sensor is interfaced to Microcontroller. And user can place a bottle on the robot with floor cleaning liquid. When the robot moves; automatically its pumps the liquid and cleans the floor and avoiding the obstacles in its path. To perform this intelligent task, Arduino Microcontroller is loaded with a program written in C language.

## II. THE MAIN BLOCKS OF THIS PROJECT:

1. Arduino UNO.
2. DC motor with driver.
3. SR04.
4. SERVO MOTOR.
5. Rechargeable Battery.
6. LED indication.



**Fig 1: Block diagram of construction of Automatic Floor Cleaner robot**



**Fig 3: Working Directions of Arduino**

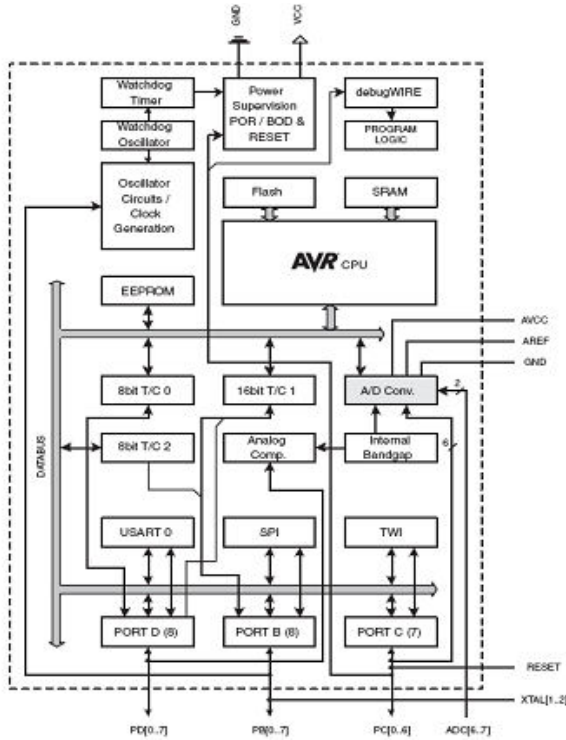
**ATMEGA328:**

**Features**

- High Performance, Low Power AVR® 8-Bit Microcontroller
- Advanced RISC Architecture
  - 131 Powerful Instructions – Most Single Clock Cycle Execution
  - 32 x 8 General Purpose Working Registers
  - Fully Static Operation
  - Up to 20 MIPS Throughput at 20 MHz
  - On-chip 2-cycle Multiplier
- **Peripheral Features**
  - Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode
  - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
  - Real Time Counter with Separate Oscillator
  - Six PWM Channels
  - 8-channel 10-bit ADC in TQFP and QFN/MLF package
  - Temperature Measurement
  - 6-channel 10-bit ADC in PDIP Package

**III. WORKING PROCESS**

The Automatic floor cleaner is designed a floor cleaning robot which is connected to Arduino. And user can place a bottle on the robot with floor cleaning liquid. When the robot moves; automatically its pumps the liquid and cleans the floor and avoiding the obstacles in its path. To perform this intelligent task, Arduino Microcontroller is loaded with a program written in C language.



**Fig 2: Pin Circuit of Arduino**

#### 4. Procedural steps for compilation, simulation and dumping:

##### Compilation and simulation steps:

##### Step 1: Parts

- 1 x Arduino on a Breadboard
- 1 x Arduino UNO
- Connecting Wires
- Arduino IDE installed on your PC

##### Step 2: The Approach

We use the Arduino UNO to bootload the ATmega328 that is sitting on the Arduino-on-a-Breadboard. This is fairly straightforward having an ATmega328P-PU, but needs an extra step for an ATmega328-PU.

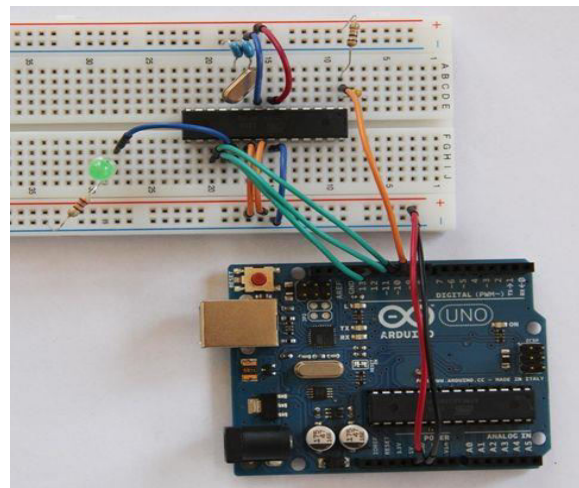


Fig 4: Arduino Breadboard

##### Step 3: Program your Arduino UNO as an ISP

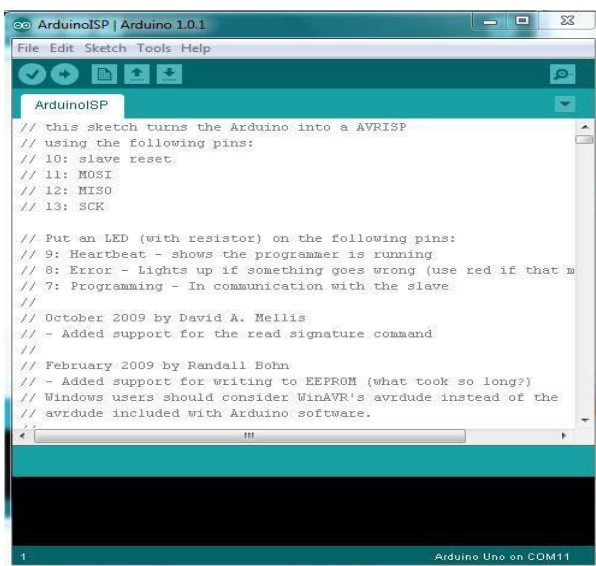


Fig 5: Arduino UNO ISP

We need to program the Arduino UNO to act as an ISP (In-System Programmer), so that it can burn the bootloader onto the Breadboard chip.

1. Open the Arduino IDE
2. Open the ArduinoISP sketch (under File, Examples)
3. If you're using version 1.0 of the IDE:

Search for void heartbeat and change the line that reads: delay(40); to delay(20);

Connect your UNO to the PC, making sure it's not connected to the Arduino on a Breadboard.

Ensure your UNO is selected under the Boards menu option, and upload the sketch.

##### Step 4: Connect your ATmega328

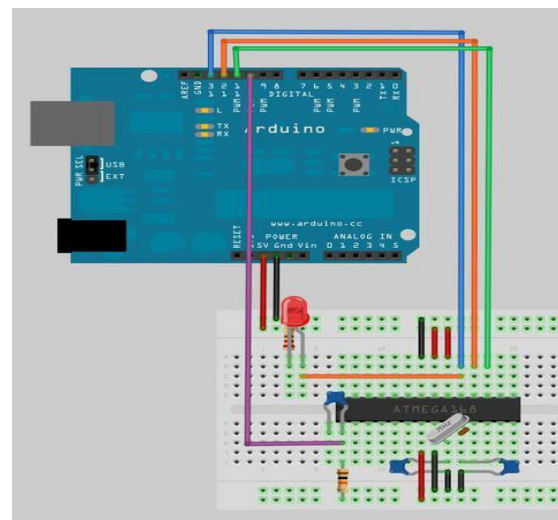


Fig 6: ATmega328

(PCINT14/RESET) PC6	1	28	PC5 (ADC5/SCL/PCINT13)
(PCINT16/RXD) PD0	2	27	PC4 (ADC4/SDA/PCINT12)
(PCINT17/TXD) PD1	3	26	PC3 (ADC3/PCINT11)
(PCINT18/INT0) PD2	4	25	PC2 (ADC2/PCINT10)
(PCINT19/OC2B/INT1) PD3	5	24	PC1 (ADC1/PCINT9)
(PCINT20/XCK/T0) PD4	6	23	PC0 (ADC0/PCINT8)
VCC	7	22	GND
GND	8	21	AREF
(PCINT6/XTAL1/TOSC1) PB6	9	20	AVCC
(PCINT7/XTAL2/TOSC2) PB7	10	19	PB5 (SCK/PCINT5)
(PCINT21/OC0B/T1) PD5	11	18	PB4 (MISO/PCINT4)
(PCINT22/OC0A/AIN0) PD6	12	17	PB3 (MOSI/OC2A/PCINT3)
(PCINT23/AIN1) PD7	13	16	PB2 (SS/OC1B/PCINT2)
(PCINT0/CLKO/ICP1) PB0	14	15	PB1 (OC1A/PCINT1)

Fig 7: ATmega328 Parts

Now connect your ATmega to your UNO as follows:

- UNO 5v ---> ATmega pin 7 (VCC)
- UNO GND ---> ATmega pin 8 (GND)
- UNO pin 10 ---> ATmega pin 1 (RESET)
- UNO pin 11 ---> ATmega pin 17 (MOSI)
- UNO pin 12 ---> ATmega pin 18 (MISO)
- UNO pin 13 ---> ATmega pin 19 (SCK)

### Step 5: Which ATmega328 are you using?



**Fig 8: ATmega328 Pin**

I learnt the hard way that there is more than one type of ATmega328. The two variants that are of interest to us are the ATmega328-PU and the ATmega328P-PU.

The **-PU** suffix means that the chips are in a PDIP package, the format we need for our breadboard.

The **328P** is a picoPower processor, designed for low power consumption, and is used on the Arduino boards. Given low power consumption this is first choice.

The **328** does not have picoPower technology, and is not used on the Arduino boards – and is not explicitly supported by the Arduino IDE.

What this means is that we can easily bootload the ATmega328P, but not the ATmega328. Unfortunately the websites that sell these chips don't always differentiate between them and forums are filled with people struggling to use the ATmega328-PU.

Luckily there is a workaround - take a look at my Crash Bang website.

### Step 6: ATmega328-PU workaround

Each microprocessor has a **signature** – a unique code that identifies its model. When you bootload a chip (or even upload a sketch) the Arduino IDE checks that the chip

selected matches the type it's connected to. Even though the ATmega328-PU in essence functions in the same way as the ATmega328P-PU, it has a different signature, and one that isn't recognised by the Arduino IDE.

In your Arduino folder, find the subfolder: `..hardware\tools\avr\etc`

1. Make a backup copy of the file: `avrdude.conf`
2. Open the file `avrdude.conf` in a text editor
3. Search for: `"0x1e 0x95 0x0F"` (this is the ATmega328P signature)
4. Replace it with: `"0x1e 0x95 0x14"` (this is the ATmega328 signature)
5. Save the file
6. Restart the Arduino IDE
7. Continue with the rest of the steps in the instructable, and once boot loading is complete restore the backup copy you made.

### Step 7: Bootload the ATmega328

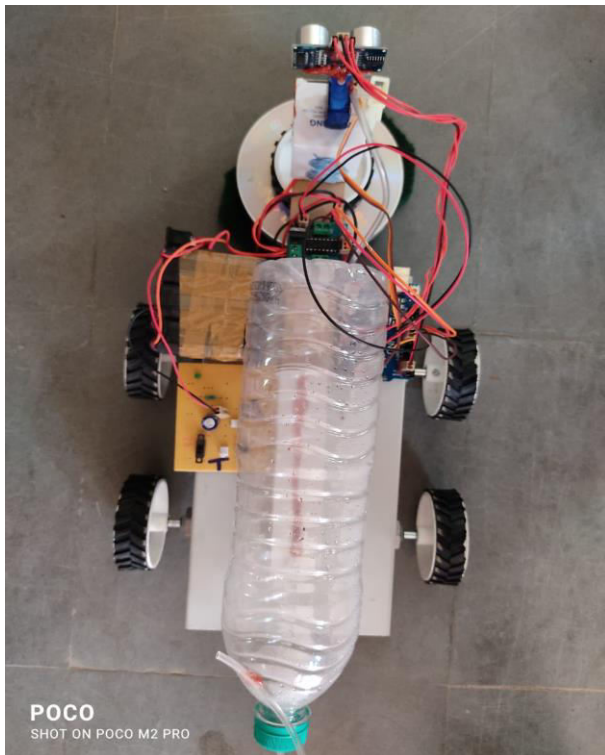
In the Arduino IDE, from the Tools menu:

- under the Board option choose Arduino UNO
- under the Serial Port option ensure the correct port is selected
- under the Programmer option choose Arduino as ISP

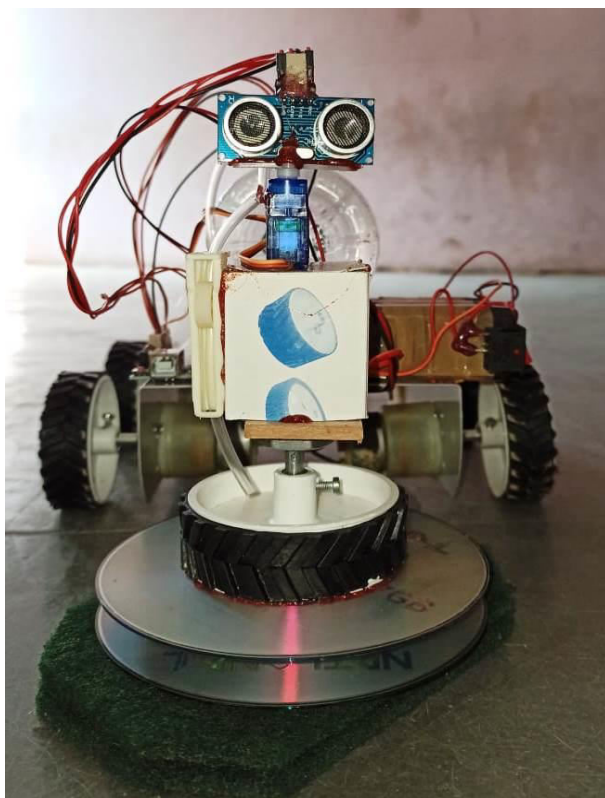
**To burn the Boot loader**, choose Burn Boot loader from the Tools menu. You should see a message "Burning boot loader to I/O Board (this may take a minute)". Once the boot loader has been burned, a message of confirming the success gets displayed. **"Congratulations: You're now ready to load sketches onto your Arduino on a breadboard!"**



#### IV. WORKING MODEL



**Fig 9: Top View of Working Model**



**Fig 10: Front View of Working Model**

#### V. Conclusion:

Integrating features of all the hardware components used have been developed in it. Presence of every module has been reasoned out and placed carefully, thus contributing to the best working of the unit. Secondly, using highly advanced IC's with the help of growing technology, the project has been successfully implemented. Thus the project has been successfully designed and tested.

#### References:

- [1] Jens-Steffen Gutmann , Kristen Culp , Mario E. Munich and Paolo Pirjanian. The Social Impact of a Systematic Floor Cleaner . In IEEE international workshop on advance robotics an its social impacts , Technische University munchen, Germany May 21-23,2012.
- [2]Evolution Robotics Inc. Introducing Mint-the evolution of floor care, www.mintcleaner.com,2011.
- [3]J-S. Gutmann, E.Eade, P.Fong and M.E. Munich. Vector field SLAM. IN Int. conf. on Robotics and automation (ICRA), 2010.
- [4]J-Y. SUNG, R.E.Grinter, and H.I.Chrstensen, and L.Go.Housewives domestic robot technology int. Journal of social robotics, 2(4):417-429,2010.
- [5]Youngkak Ma, seungwoo Kim, Dongik Oh and Youngwan Cho.A study on development of home mess-cleanup robot McBot.In IEEE/ASME international conference on advanced mechatronics July 2-5, 2008, Xian, China.