Sudan University Of Sciences and Technology

College Of Engineering

Electrical Engineering Department

Development and Improvement of Electric Bike

مشروع تطوير وتحسين الدراجة الكهربائية

A Project Submitted In Partial Fulfillment for the Requirements of the Degree of B.Sc.(Honor) In Electrical Engineering

Prepared By:

1. علي خالد علي حسين
2. عصام الدين ياسر عباس الحسين
3. عبد اللطيف محمد عبداللطيف نقد الله
4. عمر عبدالعظيم عثمان سليمان

Supervised By:

Ust.Hanaa jaafar alameen

October 2018

الآيـــــــة

قال تعالي :

(قالُوا سُبْحَانَكَ لا عِلْمَ لَنَا إلا مَا عَلَّمْتَنَا إِنَّكَ أَنْتَ الْعَلِيمُ الْحَكِيمُ)

صدق الله العظيم

سورة البقرة الآية (32)

DEDICATION

I dedicate this work to the most precious human beings without whom this project wouldn’t be possible, my mother and my father. And I will love to thank them for their continuous believe and support. I also would like to dedicate this piece of work to the rest of my family members for always being there and there unconditional love and faith. My friends who have always support me and have a real faith on me. Thank you to everyone who helped me through this journey and believes in this project till it reaches to this point , Thank you from the heart .

ACKNOWLEDGEMENT

We wish to express our profound gratitude to our Supervisor Ust.Hanaa for her valuable guidance, continues encouragement, worthwhile suggestions and constructive ideas through out this project.Her support, pragmatic analysis and understanding made this study a success and knowledgeable experience for us. And We would also to thank Dr.OmerAbdelraziq for his help and his kindness .

ABSTRACT

To Enable the bike to walk longer by adding a new set of batteries that switch between them and the first group automatically to make one in the case of charging and the other in the case of discharging at the same moment.

The average distance traveled by the bike before the batteries run is between 35-45 km. After adding the new recycle of batteries, we will exceed this limit of distance, especially when providing a source of charge, which converts the mechanical movement to an electric power for charging the empty set of batteries while the bike is running.

After reaching max limit of charging the new range of batteries lies can control the process of switching in the input and output batteries.Reach to a smooth transition between the two sets of batteries through (micro controller).

المستخلص

تمكين الدراجة من السير لمسافه اطول عن طريق اضافة مجموعه جديده من البطاريات يتم التبديل بينها وبين المجموعه الاولي اتوماتيكا لجعل إحداهما في حالة شحن والاخري في حالة تفريغ في نفس اللحظه.

يتراوح متوسط المسافه التي تقطعها الدراجه قبل نفاذ بطارياتها بين (35-45)كم , اما بعد اضافة المجوعه الجديدة من البطاريات سنتجاوز هذا الحاجز من المسافة وخصوصا عند توفير مصدر لشحنها يقوم بتحويل الحركة الميكانيكيه الي طاقه كهربائية للإستفادة منها لشحن المجموعة الفارغه من البطاريات اثناء سير الدراجةبعد الوصول الي امكانية شحن المجوعه الجديده من البطاريات يكمن التحكم في عملية التبديل في مداخل ومخارج البطاريات للوصل الي انتقال سلس بين مجموعتي البطاريات عن طريق دائرة تحكم (micro controller).

TABLE OF CONTENTS

LITERATUERE OVERVIEW

CHAPTER THREE

MAIN CIRCUIT DESIGN AND PROGRAM

CHAPTER FOUR

EXPLANATION AND RESULTS

CHAPTER FIVE

CONCLUSION AND RECOMMENDATION

LIST OF FIGURES

LIST OF SYMBOLS

λ_S	The flux linkage
B	Flux density
F	Frequency
Θ	rotor position
Ωr	Rotor speed
N_S	Number of turns
B	Viscous friction
J	Inertia
T	Torque
R	Resistance
L	Reactance
E	Electro motive force
Ean	phase voltage

LIST OF ABBREVIATIONS

DC	Direct Current
BLDC	Brushless Direct Current
AC	Alternating Current
N	North
S	South
V	Voltage
A	Ampere
ADC	Analog To Digital Converter
E-Bike	Electric Bike
SMPM	Surface-Mounted Permanent Magnet
PM	Permanent Magnet
PWM	Pulse Width Modulation
FOC	Field-Oriented Control

CHAPTER ONE

INTRODUCTION

The electric Bike is made up of a bike with an integrated electric motor used to propel it with a large variety of different shapes available all over the world Its function is similar to the function of a motorcycle and relies on its technology on electric energy in the management of the engine in charge of the movement used by dry batteries (Led acid) to store energy . Electric bicycle is a vehicle empowered by electric motor in order to move. It is also known as e-bike. For the power source of the electric motor, certain country used different power, because it is depends on the Law of the country. The invention of the electric bike is as a proof that the engineering field keep advancing, the invention of the electric bicycle make it replacing the old bicycle in the market.

Although the electric bicycles are using electric motor, it still called as bicycle rather than motorcycle. This is because it identity as bicycle is still fixed which is most of it part are belong to bicycle. So, it not included in transportation law which require the certification and operation as on good motor vehicle. It is not need to have license to ride the electric bicycle.

The electric bicycle is not a fully motorized vehicle, it’s just semi motorized bicycle, which is still have pedal, gearing, brake, and frame design and so on. This electric bike use Led acid which is a common power supply used on the electric bicycle. This kind of the battery is rechargeable and a lighter and denser capacity batteries which is make it the designing of an electric bicycle more handful and easier. The electric bicycle is not like a motorcycle in many concept, either it design or it power supply. Besides that, the electric motor is also used lower power compared to motorcycle which is the bicycle still need the rider to pedal the bicycle.There is some type of electric bicycle that commonly used by all the people in term of weight and frame material type.

Weight of the bicycle also plays an important role in the speed of the bicycle. The weight of the bicycle depends on the purpose of the bicycle been used, it is either for competition so there is few of common weight that been used for the bicycle. On the older bicycle, the weight of the bicycle is about 35 or 40 pounds, this kind of weight was back older day before the technology of the bicycle still not growing. Now, the weight of the bicycle was improved, the weights of the bicycle have been reduced about to 15 and 25 pounds. The improvement of the weight of the bicycle is for the purpose of bicycle handling and speed of the bicycle.

1.1 objectives:

The main objectives of this project are :

i. Develop the current electric bikes to travel longer distance possible distance

taking into account operational weight .

ii. Easily handle this bike with the necessary safety.

iii. Fireworks in terms of construction, operational costs and stability in work.

1.2 Methodology

- Practically By adding newsett of battery.

- Write ATmega16 microcontroller program to control the entrances and exits of the two sets of batteries by relays.

1.3 layout:

This project consist of five chapters: Chapter on gives an introduction, motivation and objectives. Chapter two discusses the background of the electric bicycle industry in the world, explains the type used in this project , the parts, and the electrical theory used in it . Chapter three the main circuitand shows how it worksand presents the power circuit connection.Chapter four presents the results , the parts used, and how to control. Finally, Chapter five provides the conclusions and recommendations.

CHAPTER TWO

LITERATUERE OVERVIEW

2.1 Introduction:

In this chapter we will introduce electric bike construction and theory of operation.

2.2 Electric bicycle background:

It is surprising to know that the experiment done to make bicycle to function on electricity, was done quite a long time ago. The record said that the first electric bikes were already available during 1890s1. Various patents during that time prove that. On 19 September 1895, a patent application for an "electrical bicycle" was filed byField-Oriented ControlOgden Bolton Jr. of Canton Ohio (Patent number: 552271)2. The bicycle ran on 10 volt battery power, in which the motor could draw power up to 100 amperes. The hub motor was used placing in the back wheel. During that time gears was still a mysterious concept for the bicycles. So, it was made without it.

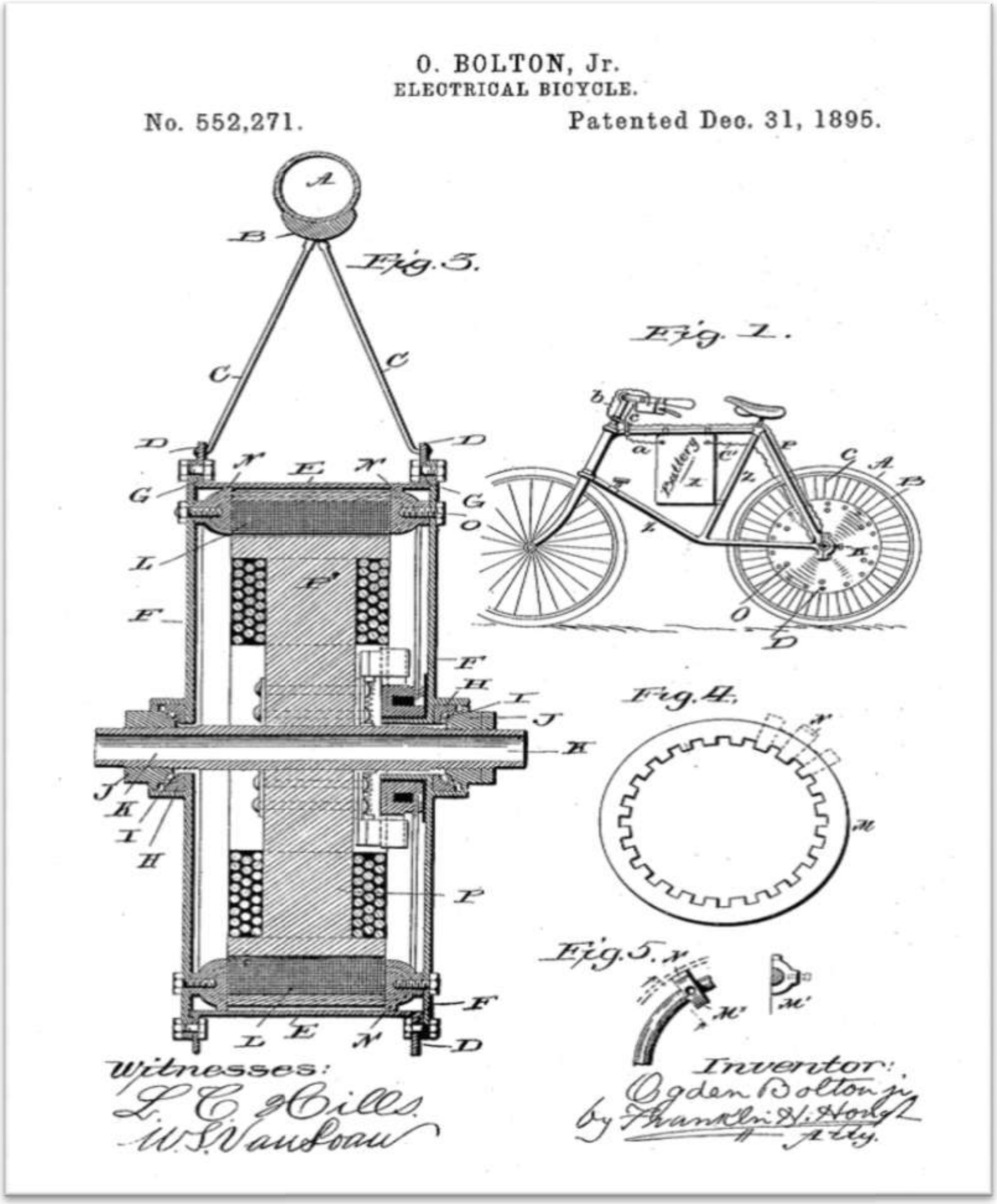


Figure2.1.:A patent by the name of Ogden Bolton Jr. of Canton Ohio on e-bike

On 8 November of the same year, another patent application for an "electric bicycle" wasfiled by Hosea W. Libbey of Boston (Patent number: 596272).

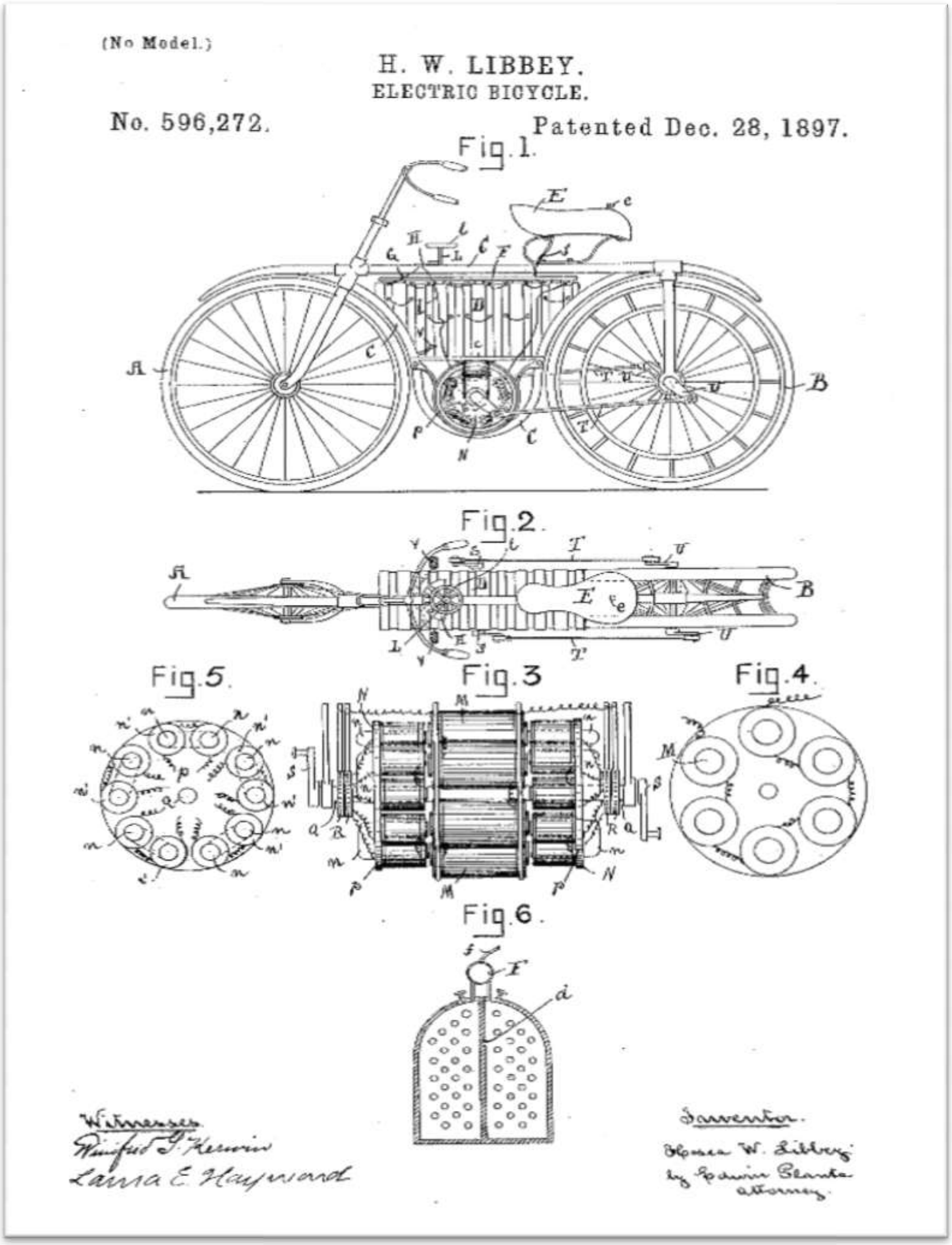


Figure 2.2: A patent under the name of H. W. Libbey

During that time in 1897, Hosea W. Libbey from Boston state invented an electric bike thatran on double electric motor. The motor having integrated within hub of the rear wheel. This model has been reused in various latest designs of e-bikes at present times.

It was by year 1898, a belt connecting outer edge of the wheel to the motor patented in the name of Mathew J. Steffens. In the same year John Schlep tried a back wheel friction“roller-wheel” style drive electric bicycle. It was in 1969 a modification of the same version was done consisting of 4 motors connected in series with the support of clock-wheel gears.

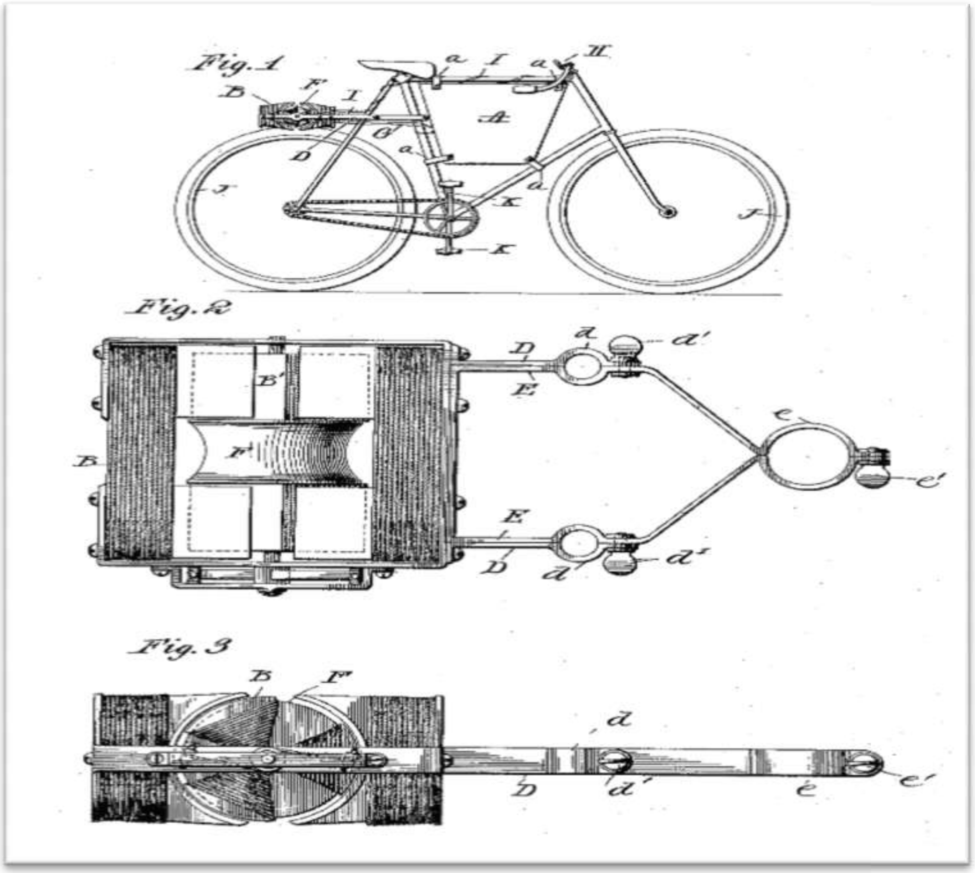


Figure 2.3: John Schnepf’s Friction Drive e-bike

Torque sensors and Power controls were developed recently in the 90s. The well-known commercial e-bike named ‘Zike’ was the modern e-bike which was launched in 1992, during that time hardly any commercial e-bikes were present in the market. Japan experimented in this and one patent relating to this under the name of Takada Yutky in 1997.

It was from 1993 onwards, when well-known Japanese companies involved in producing commercial e-bikes in huge numbers, it drew other companies' attentions into this, by estimating the huge market potentials, as a result the growth towards e-bikes increased by 35%, leading to the downfall in the production of regular bicycles.

In 1989, Michael Kutter, the founder of 'Dolphin E-bikes' done the first initiation in commercially producing e-bikes in the market. After his attempt, a well-known motorbike

2.2.1 Future of e-bikes:

The advantage of e-bikes has become more prominent in the recent times. The big companies' involvement helped to make it even better. They have tried to include many sophisticated technologies in the design of this e-bike. Brushless motors replaced the brushed ones to make it durable, efficient, and noise-free ride. Lithium battery inclusion has made e-bikes much lighter with better performance. Throttle replaced with Torque sensors has made the ride smoother. That is why today e-bike is growing popularity because of having the characteristics like lightweight, good-looking and able to make a long ride up to 55 miles on a single charge. E-bikes are now the rapidly growing name in the bicycle industry. Now with the demand for clean and safer world, there is only one possibility remains, success and just only success.

E-bike has gained its popularity in Europe slowly, and Germany is moving ahead in this, doing complete overhaul by replacing ordinary bikes. That is why e-bikes are taking market shares away from the conventional bicycles. It is also the case in India and China. In China they are slowly replacing other two wheel transportation vehicles based on fossil fuel. Firstly, it is due to high rise in petroleum price, secondly, the subsidies given by the government in promoting these e-bikes, helping it to make it affordable. This is also helping at the same time breaking the dependency upon oil and foreign market, and at the same time stabilizing globaltemperature thereby preserving the wellbeing of the planet

2.2.2 Why to choose e-bike (Analysis with respect to cost):

It is an interesting question to know as how much does it cost to charge an e-bike, that is what we are going to analyze in this to see if it is good enough in this regard as well.

As we know it takes around six hours on a normal house socket to refill the battery completely, which goes almost the same for both normal and high powered batteries.

Little calculation to test as how much cost it can there be to charge an e-bike.

1. **First calculating the kilowatt hours of our e-bike battery:**

Approximately taking the voltage of the battery as 60 volts and 12 ampere-hours (written also in the battery pack), we get the amount of kilowatt-hours that is required to charge this battery, which is around 60*12/1000= 0.72kilowatt-hours . This is the number which electric company will charge for each hour e-bike is attached with the power line.

2. **Second how much time e-bike battery takes to charge it completely:**

Considering the approximate time of 6 hours for completely charging, we get 0.72 kilowatt-hours∗ 6 hours = 4.32 kilowatt-hours , which is charged by the electric company for completely charging the e-bike.

3. **Third is to include electric company's billing rate for per kilowatt hours:**

Using the present rate of 1.5 SDG per KWh, we get (4.32 kilowatt hours * 1.5 SDG per KWh = 6.460 SDG for fully charging an e-bike battery).

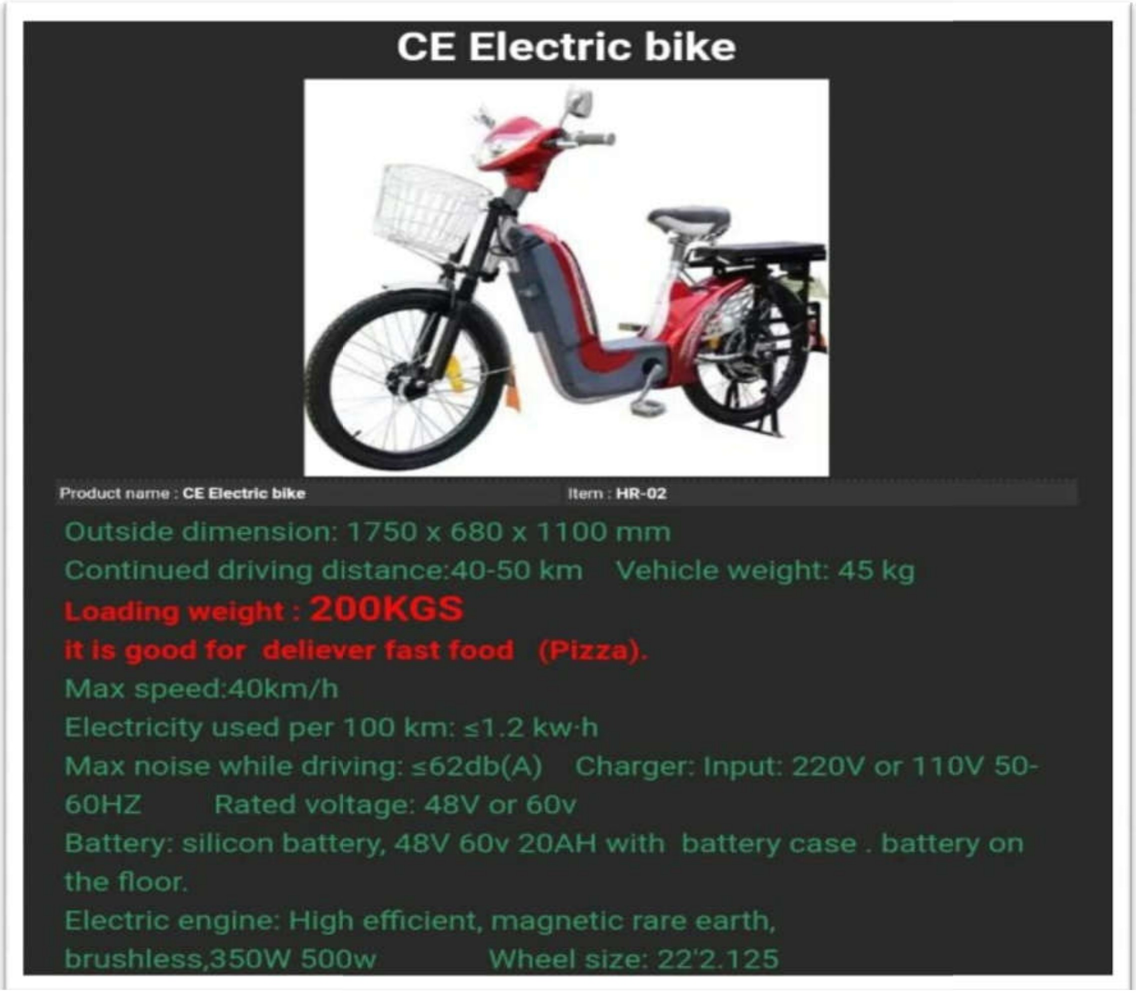
4. **Fourth to check the cost per mile:**

When battery is fully charged, it can give poweruntil 50 miles. Taking this idea into account, we check that it costs 6.460 / 50 = 1.292 SDG per mile.

Using this evidence it is clear that it does not cost the very minimum and virtually nothing ifwe compare with any cars.That is why it is also a way to save money, but also Save Earth as an extra contribution .

2.3 E-bike's components:

Pedelec differs from an ordinary bike by the inclusion of an electric motor, a battery, and electronic control system to monitor the motion of the cranks. There can also bepanel to view the status of battery, bike speed, and also the total distance covered by the bike. The security components like brakes, lighting, and structure body can be little stronger in e-bikes than in ordinary bikes, to withstand little heavier motion than ordinary bikes show the (Figure 2.4).



CE Electric bike

Product name : **CE Electric bike** Item : **HR-02**

Outside dimension: 1750 x 680 x 1100 mm
Continued driving distance:40-50 km Vehicle weight: 45 kg
Loading weight : 200KGS
it is good for deliever fast food (Pizza).
Max speed:40km/h
Electricity used per 100 km: ≤1.2 kw·h
Max noise while driving: ≤62db(A) Charger: Input: 220V or 110V 50-60HZ Rated voltage: 48V or 60v
Battery: silicon battery, 48V 60v 20AH with battery case . battery on the floor.
Electric engine: High efficient, magnetic rare earth, brushless,350W 500w Wheel size: 22'2.125

Figure2.4: CE electric bike

2.3.1 Battery:

Battery is the main component in e-bikes. There are many types which have been tested until this time but the popular ones Led-acid Batteries show the (Figure2.5). The battery capacity varies in bikes. In general the storage energy mostly up to about 400 Watt hour. Battery quality is measured by how many cycles they can be charged, and how much percentage it works still after a fixed duration length, which is measured by comparing with the original capacity at the manufacturing time. The charging time essentially depends upon the types of batteries that is used and accordingly it can be 2 to 9 hours. The most trustable battery at this moment is Led-acid battery, it is light that what makes the efficiency of the e-bikes to rise. Besides it is durable and importantly can deliver longer power into the system.

Figure2.5: led acid- battery

How can we charge the battery, and improve its efficiency level?

In some system it may be possible to charge the battery in three different ways, one is by using the 220V AC socket, another one is by using the solar panel attached to the bicycle carriage, (which is then used to convert solar power into electricity and that generated electricity is stored into the battery continuously, and that accumulated power helps to increase the efficiency of the battery, as it improves the durations). Finally the experimental option of recharging the battery can also be done by using dynamo. It is placed in such a waythat it starts revolving as soon as tires move, and all these rotations produce electricity to enhance battery power. It can have different variations, but it must be said that it is not applicable so widely because the power it generates is not so enough than the hindrance it creates, that is why it is still in the experimental stage. In our bicycle, we do not use any external charging unit attached to the bicycle. It is just that battery which is charged when it seems empty, by using normal electric socket that is available in houses. We simply take out the battery and recharge when the Power is too low to drive the bicycle easily.

2.3.1.1 Controller role in battery functions:

The technology that is used applying the controller topower the motor in e-bike is already very sophisticated. Motor consumes power when there is a need otherwise it remains completely disconnected with the main power supply, the battery unit. This system of functioning saves lots of battery power when it is not required. The microcontroller reads the need of the Power by the rider's behavior. When the rider is not moving the pedal, it means that the speed is already good enough, then there is no need for pedaling the bicycle or increasing the speed. During that time the process of supplying power to the motor remains completely idle. But as soon as the rider starts pedaling, it is almost quite instantly the controller comes into action and in few movements activates the power supply unit (the battery) to supply power to the motor hub. This mechanism makes the battery's function most sophisticated, as it works quite automatically to increase the efficiency of the battery. Besides by making the use of the motor at the right time, especially in difficult terrain, riding becomes rather easy and comfortable.

2.3.1.2 State of battery:

It is considerably the main hurdle in the process of bringing long durablebattery powered portable goods is because we still lacking the high quality batteries that canbe recharged quick, store maximum power and supply abundantly for long durations. Until this time we have batteries, which is not just big and heavy it is also not efficient to work for long durations. The capacity of the battery again goes down very rapidly over the time of use, Therefore quite incompetent, in this respect. Still we have developed to that level that we canmake use of batteries which is efficient enough to fulfill our task for limited time at least.

2.3.2 Motor:

Motor is made up of skillful wrapping of coilsshow the (Figure2.6) on a stator, a rotor for the rotation, and magnets to influence the rotations. The magnets used their work electromagnetically. That means electricity influences this iron to behave like a magnet, having both attraction and repulsion characteristics of a magnet into this, thereby helping it to generate the motion accompanying this. The principle in this is to switch the direction of the forces to keep the motor to move continuously, once it is started until the time it is stopped. There are two types of motors commonly used in e-bikes, one is brushed motor and another is brushless.

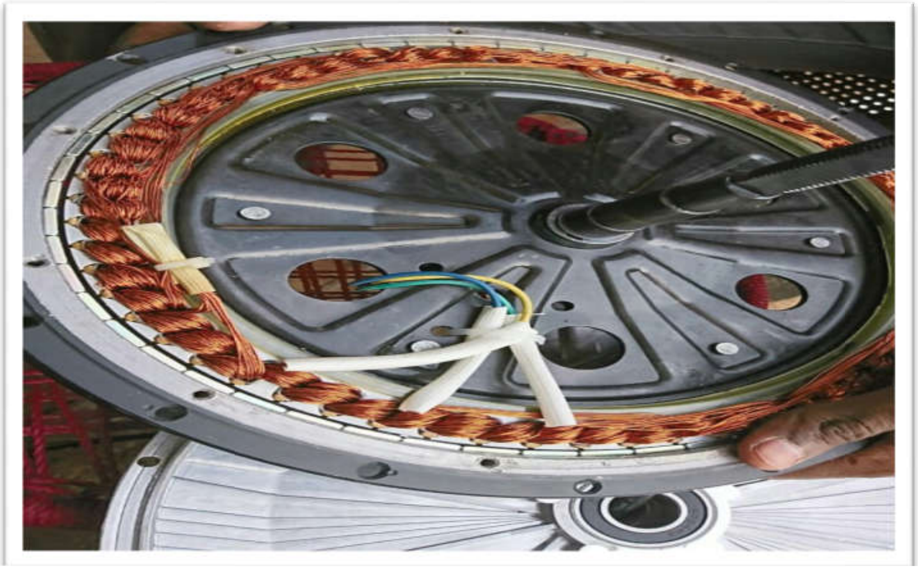


Figure2.6: BLDC motor

2.3.3 Controllers:

There are mainly two types of controllers which are designed to be effective on two types of motor, one is brushed, and another is brushless. According to the motor in use the controller function also varies. Brushless motors are popular nowadays because of high efficiency and durability, and it is also supported by the reduced cost factors, where as brushed motors because of less complex controller mechanism, is still in use fairlyshow the (Figure2.7) .

Figure2.7: the controller

2.3.3.1 Controllers used in brushless motors:

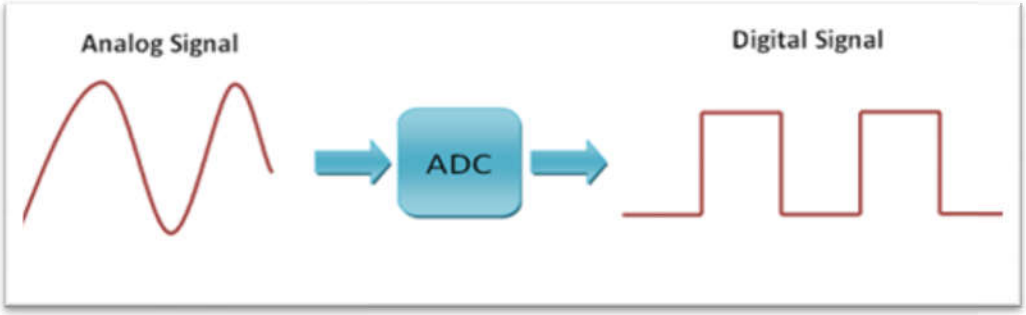
There are various sensors used to check and control the speed movements. To do this quite efficiently, Hall sensor is used The reason is also that e-bike requires strong initial torque to complement the low powered motor, this mechanism to control with safe the speed, the sensor has special functions to monitor the speed accurately. Various electronic controllers provide real time data input to the controller to react according to the situation. Usually the measuring values by the sensor are the ongoing force, and the present speed of the vehicle. The controllers work with closed-loop speed control mechanism for precise speed control, by adjusting the speed and also over-voltage surge, over-current input, or other levels of protections. Controller uses PWM (pulse width modulation) to adjust the power input to motor. In some e-bikes regenerative braking system brings additional role of power generation and management from the controller. In short, it has to maintain safety.

2.4 Microcontroller (Atmega 16)

Atmega16 is a 40-pin low power microcontroller which is developed using CMOS technology. CMOS is an advanced technology which is mainly used for developing integrated circuits. Itcomes with low power consumption and high noise immunity. Atmega16 is an 8-bit controller based on AVR advanced RISC (Reduced Instruction Set Computing) architecture. AVR is family of microcontrollers developed by Atmel in 1996.It is a single chip computer that comes with CPU, ROM, RAM, EEPROM, Timers, Counters, ADC and four 8-bit ports called PORTA, PORTB, PORTC, PORTD where each port consists of 8 I/O pins.Atmega16 has built-in registers that are used to make a connection between CPU and external peripherals devices. CPU has no direct connection with external devices. It can take input by reading registers and give output by writing registers. Atmega16 comes with two 8-bit timers and one 16-bit timer. All these timers can be used as counters when they are optimized to count the external signal. Most of the necessary peripherals required to run automatic functions are incorporated in this device like ADC (analog to digital converter), Analog comparator, USART, SPI, which make it economical as compared to a microprocessor that requires external peripheral to perform various functions. Atmega16 comes with 1KB of static RAM which is a volatile memory i.e stores information for short period of time and highly depends on the constant power supply. Whereas 16KB of flash memory, also known as ROM, is also incorporated in the device which is non-volatile in nature and can store information for long period oftime and doesn't lose any information when the power supply is disconnected. Atmega16 works on a maximum frequency of 16MHz where instructions are executed in one machine cycle.

2.4.1Sub system in Atmega 16(ADC):

The inputs available from the environment to the microcontroller are mostly analog in nature, i.e., they vary continuously with time. In order to understand the inputs by the digital processor, a device called Analog to Digital Converter (ADC) is used. As the name suggests this peripheral gathers the analog information supplied from the environment and converts it to the controller understandable digital format, microcontroller then processes the information and provides the desired result at the output end.



ATmega16 has an inbuilt 10 bit, 8-channel ADC system. Some of the basic features of Armega16 ADC are:

- 8 Channels.
- 10-bit Resolution.
- Input voltage range of 0 to Vcc.
- Selectable 2.56V of internal Reference voltage source.
- AREF pin for External Reference voltage.
- ADC Conversion Complete Interrupt.

ADC channels in Atmega16 are multiplexed with PORTA and use the common pins (pin33 to pin40) with PORTA. ADC system of Atmega16 microcontroller consists of following pins:

I **ADC0-ADC7:** 8 Channels from Pin 40 to Pin 33 of Atmega16 ADC peripheral.

II **AREF:** Pin32 of Atmega16 microcontroller, the voltage on AREF pin acts as the reference voltage for ADC conversion, reference voltage is always less than or equal to the supply voltage, i.e., Vcc

III **AVCC:** Pin30, this pin is the supply voltage pin for using PORTA and the ADC; AVCC pin must be connected to Vcc (microcontroller supply voltage) to use PORTA and ADC.

Note: External reference voltage source can be used at AREF pin. However, Atmega16 also has internal reference voltage options of 2.56V and Vref = Vcc.

The figure below shows the pin configuration for ADC system of Atmega16 microcontroller.

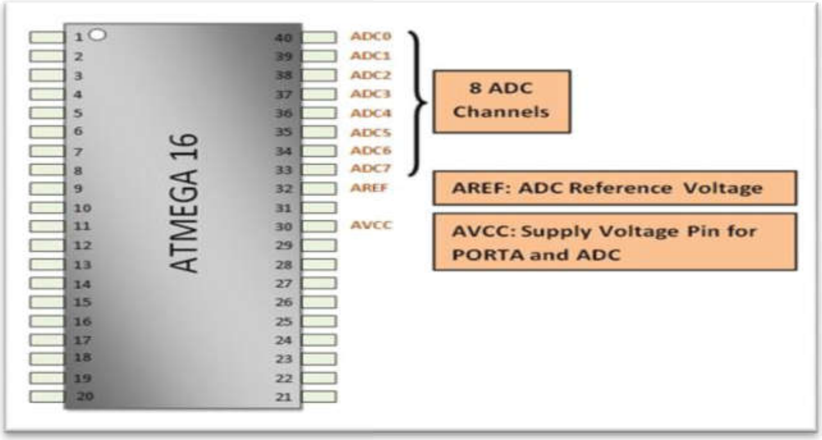
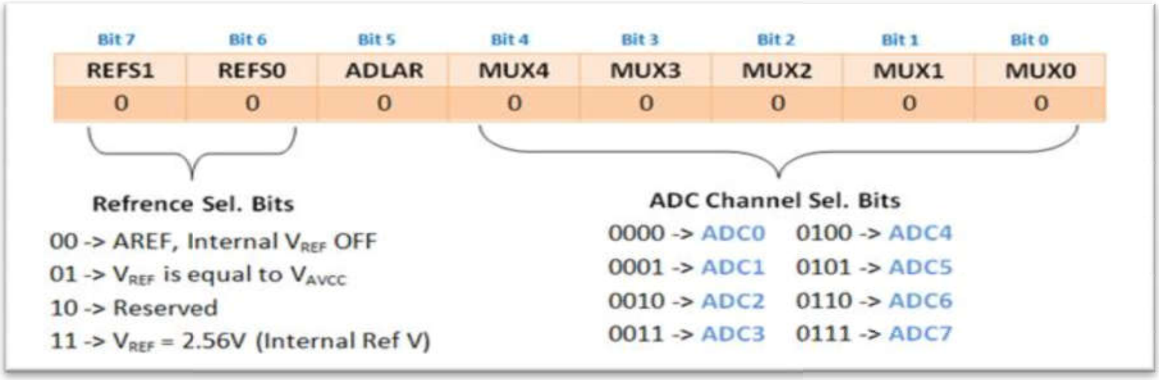


Figure2.8 : Pin numbers of ADC in ATmega16 AVR microcontroller

2.5.1.1 ADC Registers:

To use the ADC peripheral of Atmega16, certain registers need to be configured.

i. **ADMUX** (ADC Multiplexer And Selection Register)



Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
REFS1	REFS0	ADLAR	MUX4	MUX3	MUX2	MUX1	MUX0
0	0	0	0	0	0	0	0

Refrence Sel. Bits

00 -> AREF, Internal V_{REF} OFF
01 -> V_{REF} is equal to V_{AVCC}
10 -> Reserved
11 -> V_{REF} = 2.56V (Internal Ref V)

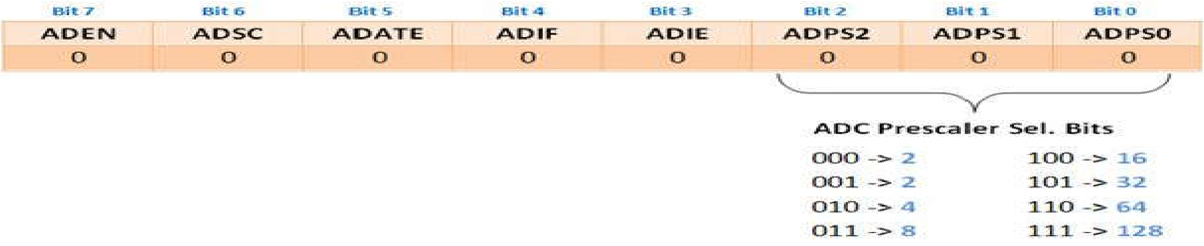
ADC Channel Sel. Bits

0000 -> ADC0
0001 -> ADC1
0010 -> ADC2
0011 -> ADC3
0100 -> ADC4
0101 -> ADC5
0110 -> ADC6
0111 -> ADC7

Figure 2.9: Bit Values of ADC Multiplexer And Selection Register to configure ADC Peripheral in AVR

REFS[0:1] bits determine the source of reference voltage whether it is internal or the external voltage source connected to AREF pin. MUX[4:0] bits are used to select between the channels which will provide data to ADC for conversion. ADLAR bit when set to 1 gives the left adjusted result in data registers ADCH and ADCL.

ii. **ADCSRA** (ADC Control and Status Register)



Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0
0	0	0	0	0	0	0	0

ADC Prescaler Sel. Bits

000 -> 2
001 -> 2
010 -> 4
011 -> 8
100 -> 16
101 -> 32
110 -> 64
111 -> 128

Figure 2.10: Bit Configuration of ADC Control and Status Register in AVR micro-controller

ADEN: ADC Enable bit, this bit must be set to 1 for turning ADC on.

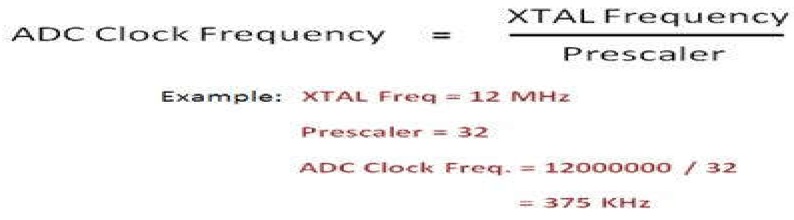
ADSC: ADC Start Conversion bit, this bit is set to 1 to start ADC conversion, as soon as conversion is completed this bit is set back to 0 by the hardware.

ADATE: ADC Auto Trigger Enable, this bit is set to 1 to enable auto triggering of ADC conversion.

ADIF: ADC Interrupt Flag, this bit is set to 1 when ADC conversion gets complete.

ADIE: ADC Interrupt Enable, this bit is set to 1 if we want to activate the ADC conversion complete interrupt.

ADPS[0:2]: ADC Prescaler bits, these bits are used to set the ADC clock frequency, the configuration of these bits determine the division factor by which the microcontroller clock frequency is divided to get the ADC clock frequency. The figure above shows the prescaler bit values for respective division factor.



$$\text{ADC Clock Frequency} = \frac{\text{XTAL Frequency}}{\text{Prescaler}}$$

Example: XTAL Freq = 12 MHz
Prescaler = 32
ADC Clock Freq. = 12000000 / 32
= 375 KHz

Fig. 2.11 *Equation of ADC Clock Frequency*

The ADC clock frequency must lie somewhere between 50 KHz to 200 KHz.

iii. **ADCH & ADCL** (ADC Data Registers)

When the ADC conversion is complete the data is stored in these two registers. The data configuration depends on the ADLAR bit value of ADMUX register. If ADLAR=0, data is right adjusted and if ADLAR=1, data is left adjusted. Always read ADCL first and then ADCH. In cases where the 8-bit precision is enough set the ADLAR bit to 1 to left adjust the data and read only the ADCH data register.

When ADLAR = 0

ADCH	-	-	-	-	-	-	ADC9	ADC8
ADCL	ADC7	ADC6	ADC5	ADC4	ADC3	ADC2	ADC1	ADC0

Figure 2.12 Bit Configuration of ADC Data Register at ADLAR=0 in ATmega16 AVR microcontroller

When ADLAR = 1,

ADCH	ADC9	ADC8	ADC7	ADC6	ADC5	ADC4	ADC3	ADC2
ADCL	ADC1	ADC0	-	-	-	-	-	-

Figure2.13 Bit Configuration of ADC Data Register at ADLAR=1 in ATmega16 AVR microcontroller

2.5BLDC Generator:

Brushless DC motors are usually motors that have permanent magnet rotors. It would be extremely unusual to find any other type of motor described as a brushless DC motor. All such motors can be used as generators, but some designs are easier to use as generators than others. A major example of a difficult motor is a BLDC fan motor found in a computer. Those have electronic circuitry built into them that must be removed or disconnected in order to use the motor as a generator. You might find some other design described as a BLDC motor that would be difficult to use as a generator, but most of them only require the shaft to be turned to produce AC at the terminals and a rectifier added if you want DC.

2.5.1Induced EMF of the BLDC Generator :

In order to control electrical output power of a BLDC generator, it is essential to understand the characteristics of the induced EMF generated in the BLDC generator.

Fig(2.15) shows the flux linkage and induced EMF of a single turn coil by the position variation. The flux linkage of a single turn coil is derived as:

$$\lambda_S = (\pi r l)\, Bf(\theta/(\pi/2)) \qquad (-\pi/2 \leq \theta \leq \pi/2) \qquad (2.1)$$

Where, l, B_f, and θ are rotor length, flux density of the permanent magnet, and rotor position, respectively. According to Faraday's law, the induced EMF is the result of the flux crossing the airgap in a radial direction and cutting the coils of the stator at a rate proportional to the rotor speed:

$$es = \frac{d\lambda_S}{dt} = \frac{d\lambda_S}{d\theta} \cdot \frac{d\theta}{dt} = \frac{\pi r l B f \omega r}{\pi/2} = 2Bf\, lr\omega r \quad (2.2)$$

The magnitude of the total induced EMF with N-turn concentric windings is derived as:

$E = 2N_S B_f l r \omega_r$ (2.3) Where, N_S is the number of turns in a phase winding.

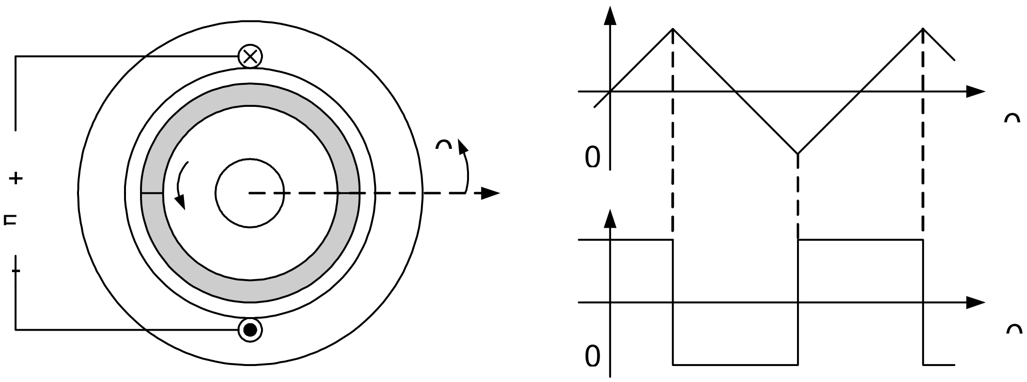


Figure2.15 Induced EMF waveform of a single turn coil.

The induced EMF waveforms of three-phase stator windings are shown in Fig(2.16) The sum of EMF that is induced in a N-turn coil generates a trapezoidal waveform with a 60° electrical angle slope region created by stator winding configuration. Each phase is shifted by 120° electrical angles. The distortion of the phase EMF due to many reasons, including manufacturing imperfections (machine geometry, winding, unevenness of the surface of the permanent magnet), leakage flux, and local saturation is neglected.

Therefore, the induced EMF waveform is ideally trapezoidal.

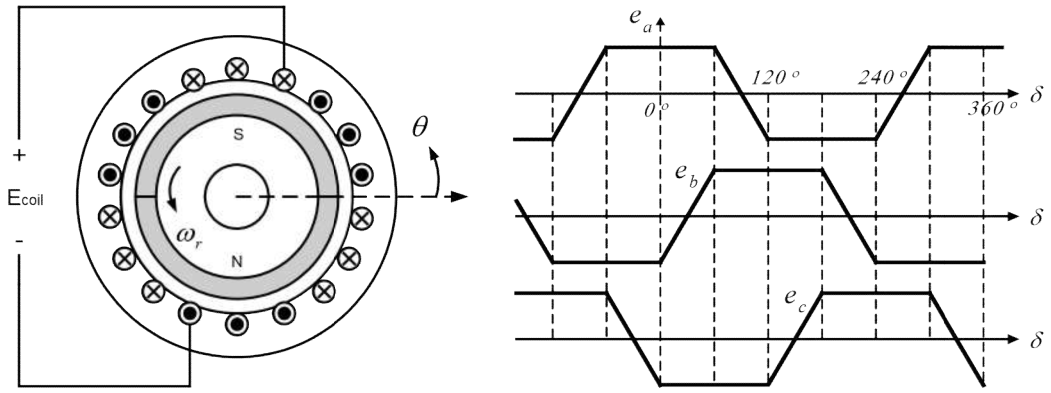


Figure2.16 Induced EMF waveforms of three-phase stator windings.

2.5.2Simplified Model of the BLDC Generator :

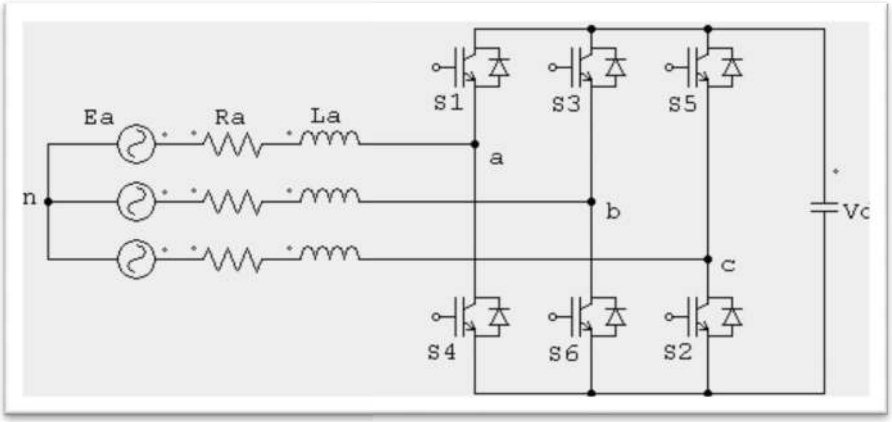


Figure2.17 Equivalent circuit of the BLDC generator.

Fig(2.17) shows the equivalent circuit of the BLDC generator. The analysis is based on the following assumptions for simplification:

- The generator is operated within the rated condition, so the generator is not saturated.
- Stator resistances and inductances of all the windings are equal.
- All three phases have an identical induced EMF shape.

- Power semiconductor devices in the converter are ideal.
- Iron losses are negligible.

Phase voltage equations of the BLDC generator can be expressed as:

$$ean= Raia+ \frac{d}{dt}(Laa(\theta,ia)\cdot ia+ Lab(\theta,ib)\cdot ib+ Lac\ (\theta,ic)\cdot ic)+ van\ e_{bn}=$$

$$R_b i_b + \frac{d}{dt}(L_{ba}(\theta,i_a)\cdot i_a+ L_{bb}(\theta,i_b)\cdot i_b+ L_{bc}(\theta,i_c)\cdot i_c)+ v_{bn} \qquad (2.4)$$

$$ecn= Rcic+ \frac{d}{dt}(Lca(\theta,ia)\cdot ia+ Lcb(\theta,ib)\cdot ib+ Lcc(\theta,ic)\cdot ic)+ vcn$$

Based on the assumptions:

$$R_a = R_b= R_c= R$$

$$Laa= Lbb= Lcc= Ls$$

$$L_{ba}= L_{ab} = L_{ca}= L_{ac} = L_{bc}= L_{cb}= L_m \qquad (2.5)$$

$$L_s - L_m = L$$

$$i_a+i_b+i_c=0$$

Equation (2.4) can be represented as:

$$ean= Ria+ (Ls - Lm) \frac{d}{dt}+ van = Ria+ L \frac{d}{dt}+ van$$

$$e_{bn}= Ri_b + (L_s - L_m) \frac{dib}{dt}+ v_{bn}= Ri_b + L\frac{dib}{dt}+ v_{bn} \qquad (2.6)$$

$$ecn= Ric+ (Ls- Lm) \frac{di}{dtc}+ vcn= Ric+ L \frac{di}{dtc}+ vcn$$

$$e_{xn} = Ri_x + L\frac{dix}{dt} + v_{xn} \tag{2.7}$$

Where, e_{xn}, $v_{xn,}$ i_x, R, L, L_s and L_m represent each phase EMF, each phase-toneutral voltage, each phase current, phase resistance, inductance, self-inductance, and mutual inductance, respectively. EMF calculation can be accomplished by sensing each phase current (i_x) and voltage (v_{xn}). And motion equation can be represented as:

$$Tmover = Tgenerator + B\omega r + J\,\frac{d}{dt}\omega r \Rightarrow \frac{d}{dt}\omega r = J\underline{1}\,(Tmover - Tgenerator - B\omega r)$$
(2.8)

Where, B and J represent viscous friction and inertia.

2.5.2Conventional Rectification Methods For The BLDC Generator:

The characteristics of the BLDC generator were described. From the induced EMF waveform, it is evident that the BLDC generator is a nonsinusoidal AC (Alternating Current) power supply system. PM generators such as BLDC generators are used for small rated power supply systems because of PM, and the electrical output power is charged in the battery as a DC voltage source. Therefore, using an AC-to-DC converter between the BLDC generator and charging equipment is essential. In this chapter, conventional rectification methods for AC-to-DC conversion are described.

2.5.2.1 Conventional Rectification of the Power Supply System :

Full-bridge diode rectifier

In most power electronics applications, a simple full-bridge diode rectifier is used for the AC-to-DC conversion. Its advantages are:

- Simple construction: A full-bridge diode rectifier has six diodes in one package.

  No additional hardware is required.

- No control: The diode is a passive element in the power electronics. There is no control to conduct circuits.
- Low cost

However, it has also disadvantages:

- The phase voltage and current is not in phase.
- Current waveform cannot be controlled as ideal waveform, and diode rectifiers draw highly distorted current from the AC side.
- Power per ampere is low because of the uncontrolled current.

Fig(2.19) shows the equivalent circuit of the BLDC generator with a diode rectifier. As it can be seen, there is no switch to control the phase current. And rectified electrical power is charged to the battery as a DC voltage source. Fig(2.20) shows each phase induced EMF and current waveform of the diode rectifier.

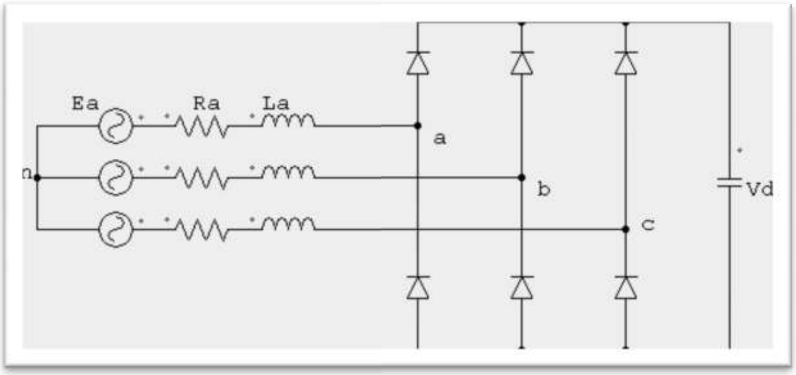


Figure2.18 Equivalent circuit of the BLDC generator with a diode rectifier.

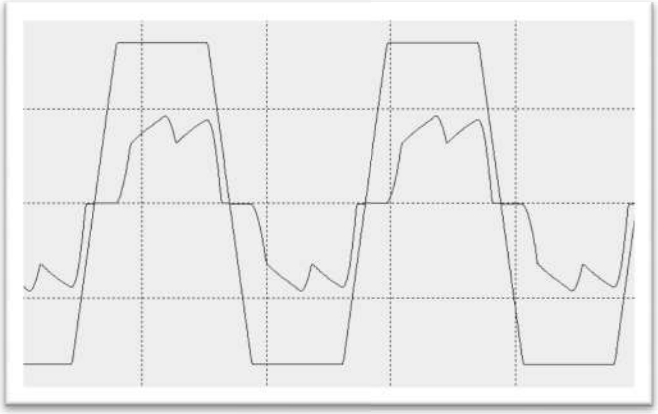


Figure 2.19 Phase EMF and current waveform of the full-bridge diode rectifier.

2.6Brushless dc motor

2.6.1 Introduction:

Electrical equipment often has at least one motor used to rotate or displace an object from its initial position. There are a variety of motor types available in the market, including induction motors, servomotors, DC motors (brushed and brushless), etc. Depending upon the applicationrequirements a particular motor can be selected. However a current trend is that most new designs are moving towards Brushless DC motors, popularly known as BLDC motors.This article will concentrate on the following aspects of BLDC motor design:

- Construction of the BLDC motor
- Operation of the BLDC motor
- Torque and Efficiency requirements
- Comparison with Induction and Brushed DC motors
- Selection criteria for a BLDC motor
- Motor control – Speed, Position and Torque, to be covered in Part II of this article.

2.6.2 Construction :

BLDC motors have many similarities to AC induction motors and brushed DC motors in terms of construction and working principles respectively. Like all other motors, BLDC motors also have a rotor and a stator.

1-Stator:

Similar to an Induction AC motor, the BLDC motor stator is made out of laminated steel stacked up to carry the windings. Windings in a stator can be arranged in two patterns; i.e. a star pattern (Y) or delta pattern (Δ). The major difference between the two patterns is that the Y pattern gives high torque at low RPM and the Δ pattern gives low torque at low RPM. This is because in the Δ configuration, half of the voltage is applied across the winding that is not driven, thus increasing losses and, in turn, efficiency and torque.

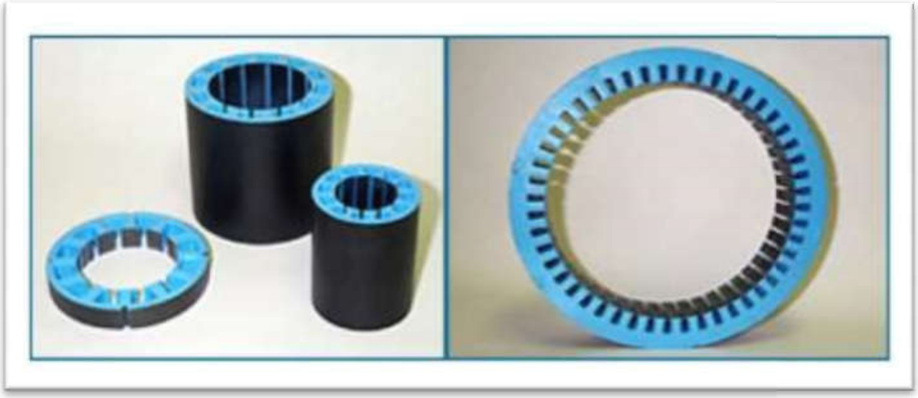


Figure2.20: Steel laminations stator

Steel laminations in the stator can be slotted or slotless as shown in Fig (2.20)aslotless core has lower inductance, thus it can run at very high speeds. Because of the absence of teeth in the lamination stack, requirements for the cogging torque also go down, thus making them an ideal fit for low speeds too (when permanent magnets on rotor and tooth on the stator align with each other then, because of the interaction between the two, an undesirable cogging torque develops and causes ripples in speed). The main disadvantage of a slotless core is higher cost because it requires more winding to compensate for the larger air gap.

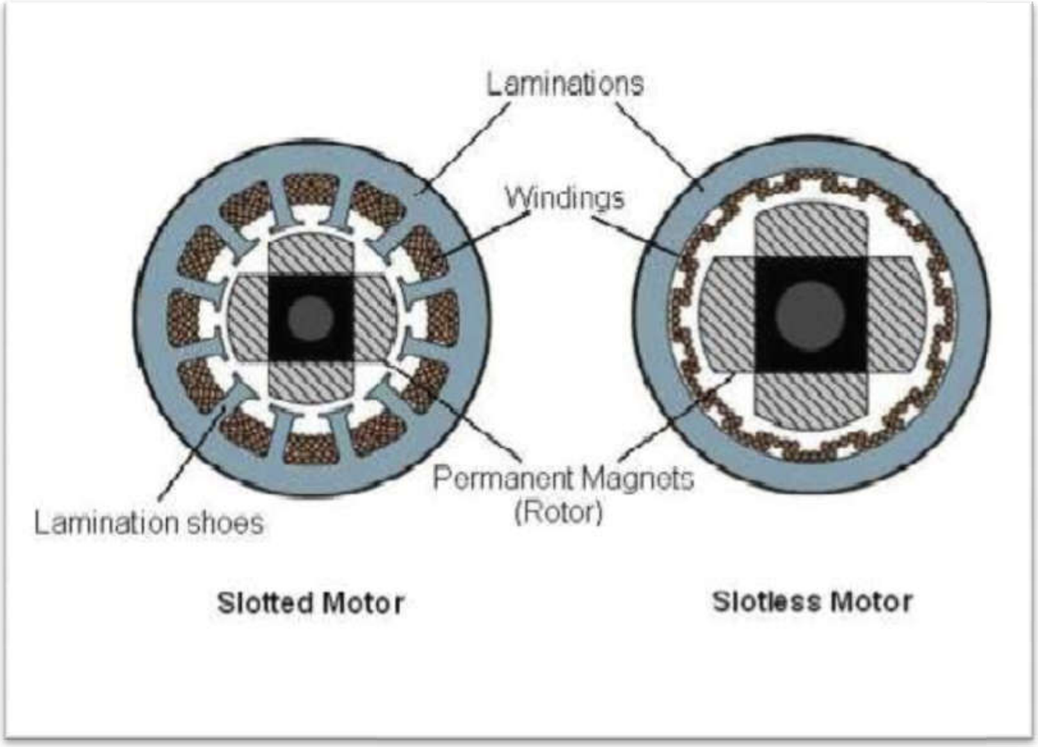


Figure2. 21: slotted and slotless stator

Proper selection of the laminated steel and windings for the construction of stator are crucial to motor performance. An improper selection may lead to multiple problems during production, resulting in market delays and increased design costs.

2-Rotor :

The rotor of a typical BLDC motor is made out of permanent magnets. Depending upon the application requirements, the number of poles in the rotor may vary. Increasing the number of poles does give better torque but at the cost of reducing the maximum possible speed.

Figure2.22: permanent magnets rotor

Another rotor parameter that impacts the maximum torque is the material used for the construction of permanent magnet; the higher the flux density of the material, the higher the torque.

2.6.3 Working Principles and Operation :

The underlying principles for the working of a BLDC motor are the same as for a brushed DC motor; i.e., internal shaft position feedback. In case of a brushed DC motor, feedback is implemented using a mechanical commutator and brushes. With a in BLDC motor, it is achieved using multiple feedback sensors. The most commonly used sensors are hall sensors and optical encoders.

Note:Hall sensors work on the hall-effect principle that when a current-carrying conductor is exposed to the magnetic field, charge carriers experience a force based on the voltage developed across the two sides of the conductor.

If the direction of the magnetic field is reversed, the voltage developed will reverse as well. For Hall effect sensors used in BLDC motors, whenever rotor magnetic poles (N or S) pass near the hall sensor, they generate a HIGH or LOW level signal, which can be used to determine the position of the shaft.

In a commutation system – one that is based on the position of the motor identified using feedback sensors – two of the three electrical windings are energized at a time as shown in figure (2.23).

In figure (2.23) (A), the GREEN winding labeled “001” is energized as the NORTH pole and the BLUE winding labeled as “010” is energized as the SOUTH pole. Because of this excitation, the SOUTH pole of the rotor aligns with the GREEN winding and the NORTH pole aligns with the RED winding labeled “100”. In order to move the rotor, the “RED” and “BLUE” windings are energized in the direction shown in figure (2.23) (B). This causes the RED winding to become the NORTH pole and the BLUE winding to become the SOUTH pole. This shifting of the magnetic field in the stator produces torque because of the development of repulsion (Red winding – NORTH-NORTH alignment) and attraction forces (BLUE winding – NORTH-SOUTH alignment), which moves the rotor in the clockwise direction.

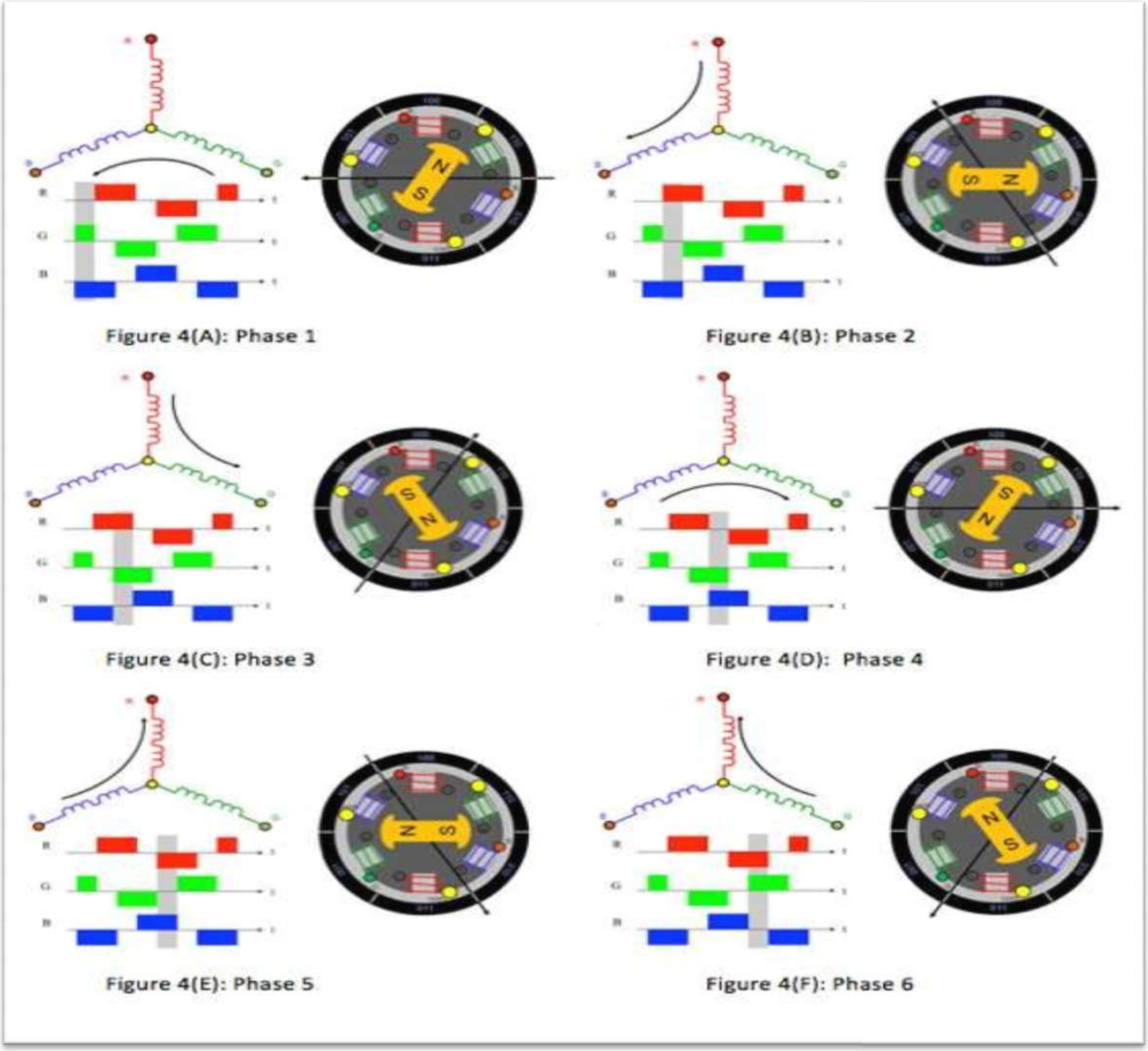


Figure 4(A): Phase 1
Figure 4(B): Phase 2
Figure 4(C): Phase 3
Figure 4(D): Phase 4
Figure 4(E): Phase 5
Figure 4(F): Phase 6

Figure 2.23: operation of BLDC motor

CHAPTER THREE

Main Circuit Design And Program

3.1 System Components:

Selection of appropriate material for a mechanical part is an essential element of all engineering projects.The main mechanical parts of the system are the base support .

3.1.1 BLDC motor

show the (Figure3.1)BLDC motor we used for this project and it specification are rated voltage =60v, output power=500w, torque=50-200n.m, speed=450-500 R.P.M, continuous current=10A-83A, efficiency=93%, rated speed=40km/h.

Figure3.1: BLDC motor

3.1.2 THE CAPACITOR

show capacitor (1000µf,100v)

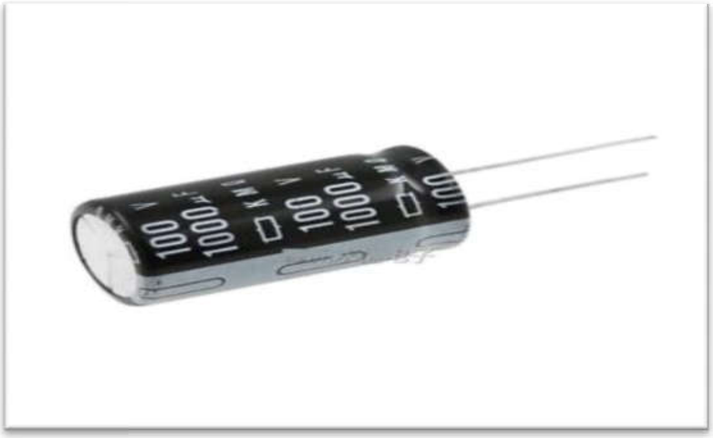


Figure 3.2: capacitor

3.1.3 The Rectifier:

show three phase full wave – bridge rectifier.

Figure 3.3: Rectifier

3.1.4 Voltage Sensor:

Voltage Detection Sensor Module 25V module is based on the principle of resistive voltage divider design, it can make the red terminal connector input voltage to 5 times smaller microcontroller analog input voltages up to 5 v. The voltage detection module input voltage not greater than 5Vx5=25V (if using 3.3V systems, input voltage not greater than 3.3Vx5=16.5V).

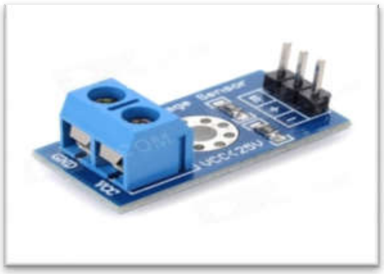


Figure 3.4: Voltage Sensor

3.1.5 Relay module:

This 1 channel 5V 10A relay control board module with optocoupler modules is compliant with international safety standards, control and load areas isolation trenches it has a single relay a genuine. The power supply and relay instructions, lit, a disconnect are off. The input signal common Terminal and start conducting. It can be use as a single chip module for appliance control and work with both DC or AC signals where you can control the 220V AC load.

Figure 3.5: Relay module

3.2Circuit description:

Connect the circuit as shown in the circuit diagram(Figure3.6), connected in between AVcc (pin 30) ,Aref (pin 32) and Reset (pin 9). AVcc (pin 30) is connected to external supply +5V , and connected in between (pin 31)and (pin11) connected to ground , it has also been connected port A (ADC) in a (pin 40) to battery of the first group and connected (pin 39) to battery of the second group ,it has also been connected port B in a (pin 1) to relay module 1 and connected (pin 2) to relay module 2.

3.3 Software Type

The voltage sensor sense the charge of the batteries groups and send analog signal to the microcontroller in port A (ADC) when the percentage of the charge of one group that is discharging arrive to 10% the microcontroller send a signal to the relays the change their contacts as show in figure (3.6) .

CodevisionAVR used to programming the microcontroller (AT mega 16) and proteus professional used in simulation.

3.4 System Implementation

Figure (3.7) show circuit components and how it connected. The generator will generate (ac) current. This current goes to the rectifier which will change in shape of wave and make the current DC current. The DC current go through capacitor which make the wave of the current smoothie. Let us assume it connected to relay 1 that who will choice which group of battery will charge.

Relay 2 will take the current from other group to supply the motor When the percentage of charge for the group of battery that connected to the motor arrive to 10% The both relays will change their contacts in the same time and in this case relay 1 will be connected to the group 2 of batteries and relay 2 will be connected to group 1 two of batteries.

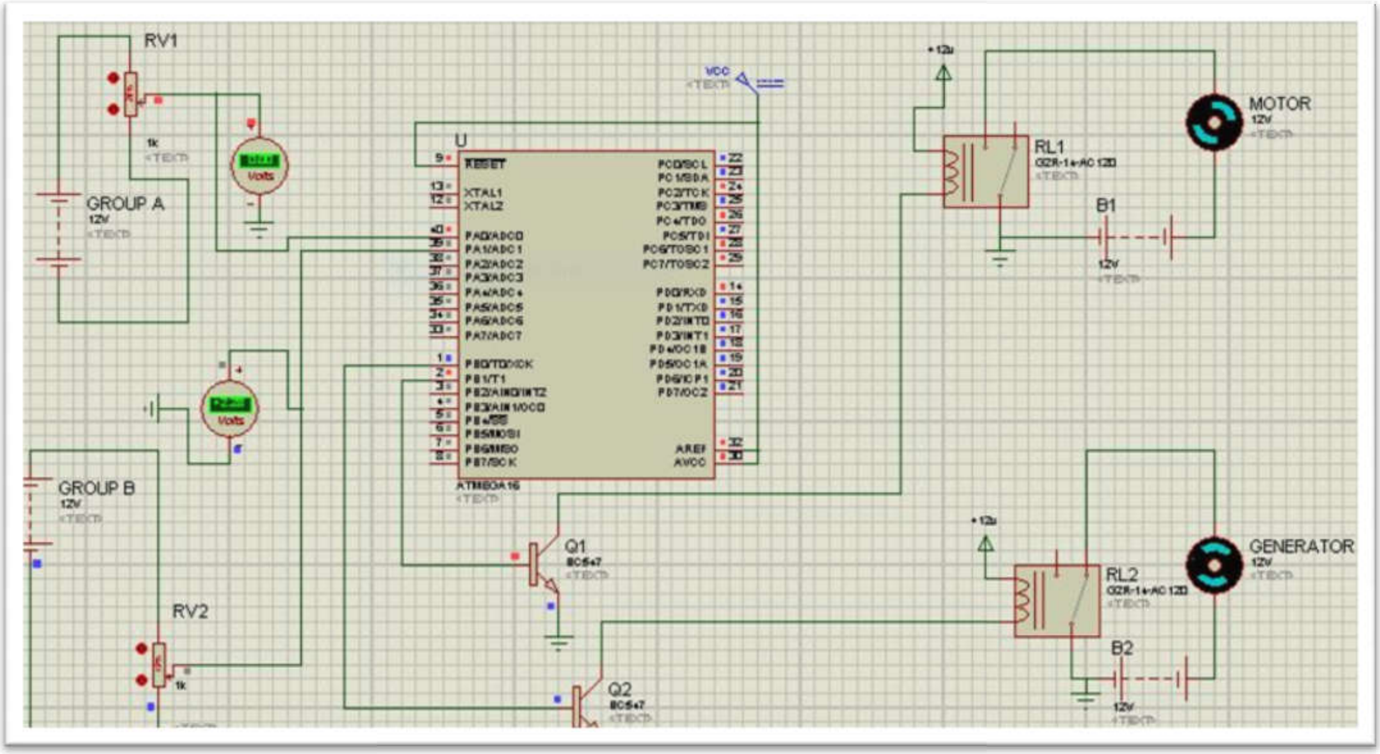


Figure 3.6: proteus simulation

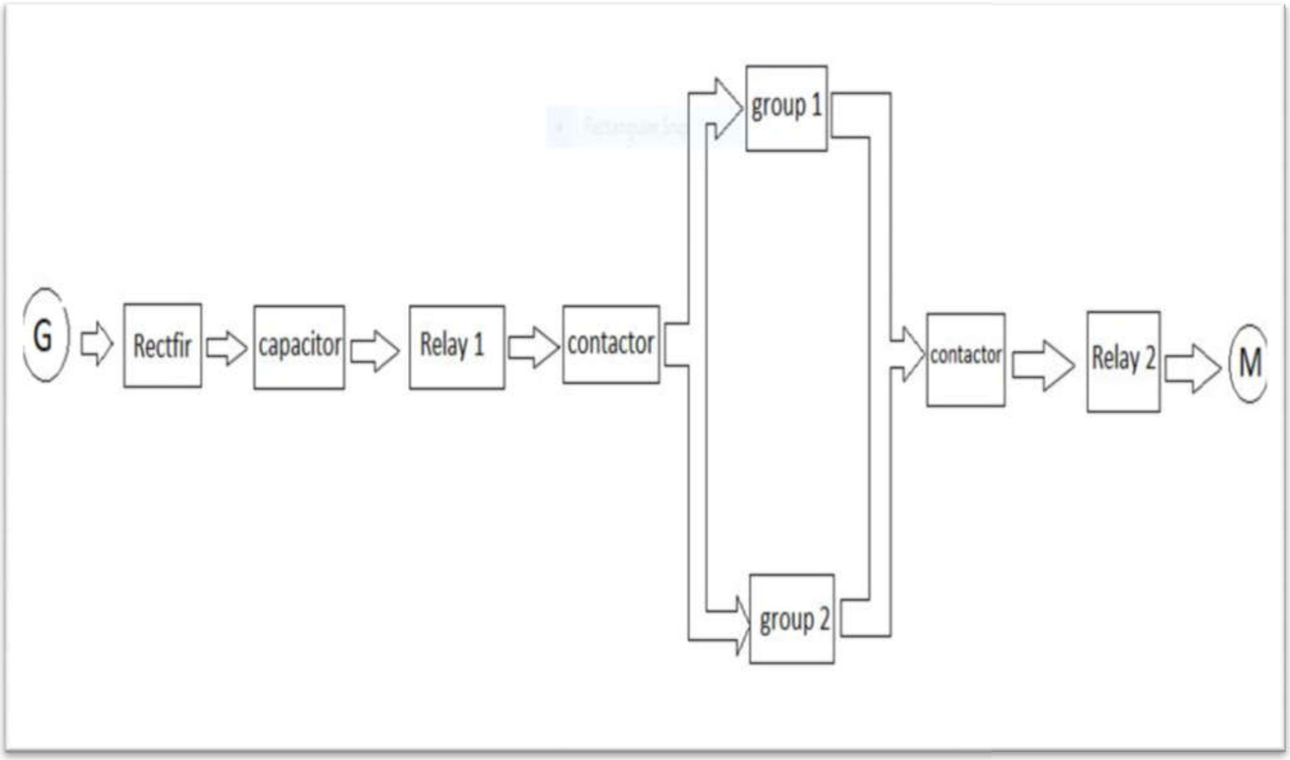


Figure 3.7: circuit connection

CHAPTER FOUR

Explanation And Results

4.1 Introduction:

In this chapter we will explain and discussed the related results as a result of the improvement of charging the other set of batteries and the results of control circuit used to control the switch between them and the primary group .

4.2 Results of battery charging:

Distance (km)	Group 1 voltage (v)	Group 2 voltage (v)
0	66	54.1
1	65.2	55.6
2	64.7	56.3
3	63.9	57.2
4	62.4	57.9
5	61.2	58.5
6	59.5	59.3
7	58.3	60
8	57.5	60.9
9	56.2	61.4
10	55.4	61.7

Table 4.1: The relation between charging group and discharging group to distance:

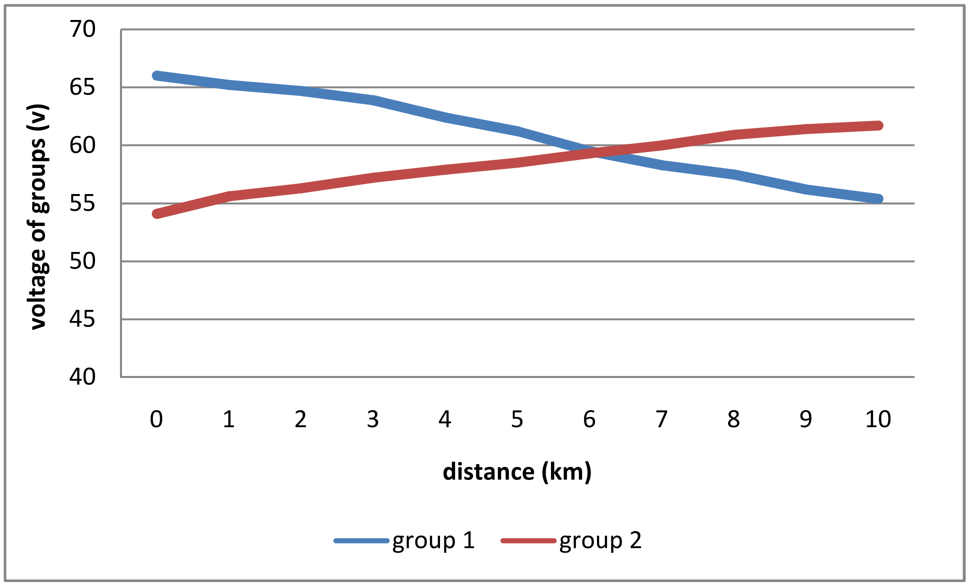


Figure4.1: The relation between charging group and discharging group to distance.

When the electric scooter is moving group 1 of battary is discharging and group 2 is charging. when it is charging percentage of group 1 arrive to (11%) group 2 charge level arrive to (64%) This means we will earn at each charging and discharging cycle what equals (64%) of the charged battery in the discharge mode .

4.3 System Experimental Results:

The one battery charge have been measured and it value at 11% approximately equal (12.4v). The voltage sensor divides this value over 5 then change it to hexadecimal this value equal (1F).

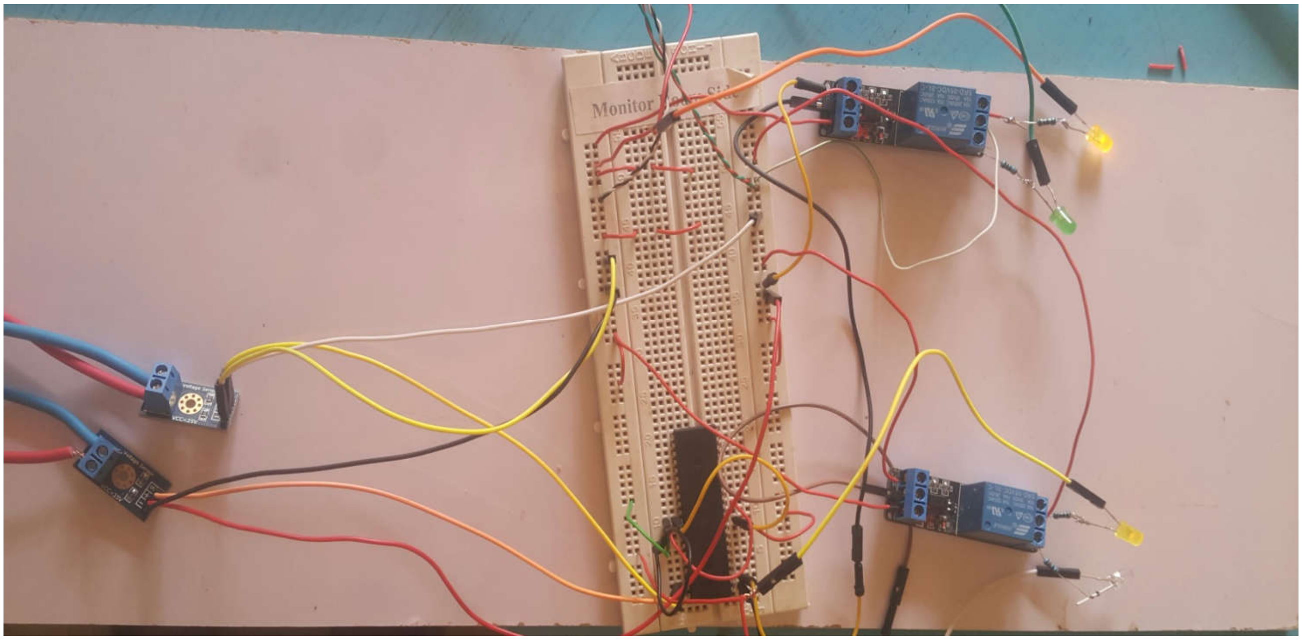


Figure 4.2: control system circuit

The microcontroller performs a process of switching between the entrances and exits of the batteries , which is charging current coming from the source (electricity + generator) when charging and choosing which of the two sets is unloaded in the engine .

CHAPTER FIVE

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

E-bikes are particularly terrible for long distance riding, since with current battery technology they tend to only 15-40 km on single charge and take hours to recharge, bikes often needs to move more than 15-40 km, to fix this problem we use self-generation , extra battery, microcontroller, voltage sensor to read battery charge value, and relays to specifies the battery to be charged from generator and the battery that feeds the motor and we use microcontroller to programming the process, some techniques, relays and controller have been applied to fulfill the requirements to fix the problem of bikes , this way is one of many technique that is used to achieve this goal.

The control Architecture that is used is a microcontroller as method to program the avoidance obstacle bike. Now the bike can travel very long distance without stopping and wasting time to charging battery by switching between the two battery groups .

5.2 Recommendation

1.Use a Contactor to control it through the relay for smooth batteries switching.

2. Use larger capacity batteries.

3. Using high voltage and high current components

4. Reduce overall weight.

5.Can add solar panels as a second source.

REFERENCES

[1] https://en.m.wikipedia.org/wiki/electric_bicycle/

[2] http://ww.electricbike.com/e-bike-patents-from-the-1800s/

[3]http://www.bicyclehistory.net/motorcycle-history/electric-bicycle/

[4] Paul Rosen, Poter Cox , David Horton " cycling and society ".

[5] Tony Foale , " motorcycle and handling and chassis design: The Art and science "

[6]William C. Morchin , Henry Oman" Electric Bicycles : A Guide to Design and Use "

[7] HYOUNG_WOO LEE " Advance Control for power Density maximization of The Brush Less Dc generator "

[8] M.Mubeen " Brushless DC Motor primer" otion tech trends,2008.

[9] P. Yedamale, "Brushless DC (BLDC) motor fundamental," microship technology lnc, vol. 20,pp.3-15, 2003.

[10] N.Rethinam and P.Abhishek," technology for controlling a brushless DC (BLDC) electric motor," ed: Google patents.

[11]www.engineersgarage.com/Atmega16/ADC/

[12] https://en.wikipedia.org/wiki/Brushless DC Motor/

APPENDIX A

```
*******************************************************
   This program was created by the CodeWizardAVR V3.32a
   Automatic Program Generator
© Copyright 1998-2017 PavelHaiduc, HP InfoTech s.r.l.
   http://www.hpinfotech.com
Project :
Version :
   Date    : 8/28/2018
Author  :
Company :
   Comments:
   Chip type              : ATmega16
   Program type           : Application
   AVR Core Clock frequency: 1.000000 MHz
   Memory model           : Small
   External RAM size      : 0
   Data Stack size        : 256
   *********************************************************/
```

```
   #include <mega16.h>

   #include <delay.h>

   // Declare your global variables here

   #define FIRST_ADC_INPUT 0

   #define LAST_ADC_INPUT 1

unsigned char adc_data[LAST_ADC_INPUT-FIRST_ADC_INPUT+1];

   // Voltage Reference: AREF pin

   #define ADC_VREF_TYPE ((0<<REFS1) | (0<<REFS0) | (1<<ADLAR))

   // ADC interrupt service routine

   // with auto input scanning

interrupt [ADC_INT] void adc_isr(void)

   {

static unsigned char input_index=0;

   // Read the 8 most significant bits

   // of the AD conversion result

adc_data[input_index]=ADCH;

   // Select next ADC input

if (++input_index> (LAST_ADC_INPUT-FIRST_ADC_INPUT))

input_index=0;
```

```
   ADMUX=(FIRST_ADC_INPUT | ADC_VREF_TYPE)+input_index;
   // Delay needed for the stabilization of the ADC input voltage
delay_us(10);
   // Start the AD conversion
   ADCSRA|=(1<<ADSC);
   }
void main(void)
   {
   // Declare your local variables here
   // Input/Output Ports initialization
   // Port A initialization
   // Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In Bit0=In
   DDRA=(0<<DDA7) | (0<<DDA6) | (0<<DDA5) | (0<<DDA4) | (0<<DDA3) | (0<<DDA2) |
(0<<DDA1) | (0<<DDA0);
   // State: Bit7=T Bit6=T Bit5=T Bit4=T Bit3=T Bit2=T Bit1=P Bit0=P
   PORTA=(0<<PORTA7) | (0<<PORTA6) | (0<<PORTA5) | (0<<PORTA4) | (0<<PORTA3) |
(0<<PORTA2) | (1<<PORTA1) | (1<<PORTA0);
   // Port B initialization
   // Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In Bit0=In
   DDRB=(0<<DDB7) | (0<<DDB6) | (0<<DDB5) | (0<<DDB4) | (0<<DDB3) | (0<<DDB2) |
(0<<DDB1) | (0<<DDB0);
   // State: Bit7=T Bit6=T Bit5=T Bit4=T Bit3=T Bit2=T Bit1=T Bit0=T
```

```
PORTB=(0<<PORTB7) | (0<<PORTB6) | (0<<PORTB5) | (0<<PORTB4) | (0<<PORTB3) | (0<<PORTB2) | (0<<PORTB1) | (0<<PORTB0);

// Port C initialization

// Function: Bit7=Out Bit6=Out Bit5=Out Bit4=Out Bit3=Out Bit2=In Bit1=Out Bit0=Out

DDRC=(1<<DDC7) | (1<<DDC6) | (1<<DDC5) | (1<<DDC4) | (1<<DDC3) | (0<<DDC2) | (1<<DDC1) | (1<<DDC0);

// State: Bit7=0 Bit6=0 Bit5=0 Bit4=0 Bit3=0 Bit2=T Bit1=0 Bit0=0

PORTC=(0<<PORTC7) | (0<<PORTC6) | (0<<PORTC5) | (0<<PORTC4) | (0<<PORTC3) | (0<<PORTC2) | (0<<PORTC1) | (0<<PORTC0);

// Port D initialization

// Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=Out Bit0=Out

DDRD=(0<<DDD7) | (0<<DDD6) | (0<<DDD5) | (0<<DDD4) | (0<<DDD3) | (0<<DDD2) | (1<<DDD1) | (1<<DDD0);

// State: Bit7=T Bit6=T Bit5=T Bit4=T Bit3=T Bit2=T Bit1=0 Bit0=0

PORTD=(0<<PORTD7) | (0<<PORTD6) | (0<<PORTD5) | (0<<PORTD4) | (0<<PORTD3) | (0<<PORTD2) | (0<<PORTD1) | (0<<PORTD0);

// Timer/Counter 0 initialization

// Clock source: System Clock

// Clock value: Timer 0 Stopped

// Mode: Normal top=0xFF

// OC0 output: Disconnected

TCCR0=(0<<WGM00) | (0<<COM01) | (0<<COM00) | (0<<WGM01) | (0<<CS02) | (0<<CS01) | (0<<CS00);
```

```
TCNT0=0x00;

OCR0=0x00;

// Timer/Counter 1 initialization

// Clock source: System Clock

// Clock value: Timer1 Stopped

// Mode: Normal top=0xFFFF

// OC1A output: Disconnected

// OC1B output: Disconnected

// Noise Canceler: Off

// Input Capture on Falling Edge

// Timer1 Overflow Interrupt: Off

// Input Capture Interrupt: Off

// Compare A Match Interrupt: Off

// Compare B Match Interrupt: Off

TCCR1A=(0<<COM1A1) | (0<<COM1A0) | (0<<COM1B1) | (0<<COM1B0) | (0<<WGM11) |
(0<<WGM10);

TCCR1B=(0<<ICNC1) | (0<<ICES1) | (0<<WGM13) | (0<<WGM12) | (0<<CS12) | (0<<CS11) |
(0<<CS10);

TCNT1H=0x00;

TCNT1L=0x00;

ICR1H=0x00;
```

```
ICR1L=0x00;

OCR1AH=0x00;

OCR1AL=0x00;

OCR1BH=0x00;

OCR1BL=0x00;


// Timer/Counter 2 initialization

// Clock source: System Clock

// Clock value: Timer2 Stopped

// Mode: Normal top=0xFF

// OC2 output: Disconnected

ASSR=0<<AS2;

TCCR2=(0<<PWM2) | (0<<COM21) | (0<<COM20) | (0<<CTC2) | (0<<CS22) | (0<<CS21) | (0<<CS20);

TCNT2=0x00;

OCR2=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization

TIMSK=(0<<OCIE2) | (0<<TOIE2) | (0<<TICIE1) | (0<<OCIE1A) | (0<<OCIE1B) | (0<<TOIE1) | (0<<OCIE0) | (0<<TOIE0);

// External Interrupt(s) initialization

// INT0: Off
```

```
// INT1: Off
// INT2: Off
MCUCR=(0<<ISC11) | (0<<ISC10) | (0<<ISC01) | (0<<ISC00);
MCUCSR=(0<<ISC2);
// USART initialization
// USART disabled
UCSRB=(0<<RXCIE) | (0<<TXCIE) | (0<<UDRIE) | (0<<RXEN) | (0<<TXEN) | (0<<UCSZ2) | (0<<RXB8) | (0<<TXB8);
// Analog Comparator initialization
// Analog Comparator: Off
// The Analog Comparator's positive input is
// connected to the AIN0 pin
// The Analog Comparator's negative input is
// connected to the AIN1 pin
ACSR=(1<<ACD) | (0<<ACBG) | (0<<ACO) | (0<<ACI) | (0<<ACIE) | (0<<ACIC) | (0<<ACIS1) | (0<<ACIS0);
// ADC initialization
// ADC Clock frequency: 500.000 kHz
// ADC Voltage Reference: AREF pin
// ADC Auto Trigger Source: ADC Stopped
// Only the 8 most significant bits of
```

```
   // the AD conversion result are used

   ADMUX=FIRST_ADC_INPUT | ADC_VREF_TYPE;

   ADCSRA=(1<<ADEN) | (1<<ADSC) | (0<<ADATE) | (0<<ADIF) | (1<<ADIE) | (0<<ADPS2) |
(0<<ADPS1) | (1<<ADPS0);

   SFIOR=(0<<ADTS2) | (0<<ADTS1) | (0<<ADTS0);

   // SPI initialization

   // SPI disabled

   SPCR=(0<<SPIE)  |  (0<<SPE)  |  (0<<DORD)  |  (0<<MSTR)  |  (0<<CPOL)  |  (0<<CPHA)  |
(0<<SPR1) | (0<<SPR0);

   // TWI initialization

   // TWI disabled

   TWCR=(0<<TWEA) | (0<<TWSTA) | (0<<TWSTO) | (0<<TWEN) | (0<<TWIE);

   // Globally enable interrupts

   #asm("sei")

while (1)

      {

      // Place your code here

      PORTC=0x00;

delay_ms(1000);

do

      {
```

```
          PORTB=0X01;

          ADMUX=0x00;

          ADCSRA=0XC3;

while(ADCSRA &(1<<ADSC));

          PORTC=ADCL;

          PORTD=ADCH;

          } while(PORTC > 0X1F);

if (PORTC <= 0X1f)  PORTB=0X02;

           PORTC=0x00;

delay_ms(1000);

           PORTC= 0xFF;

delay_ms(1000);

           PORTC=0x00;

delay_ms(1000);

do

          {

           ADMUX=0x01;

          ADCSRA=0XC3;

while(ADCSRA &(1<<ADSC));

          PORTC=ADCL;
```

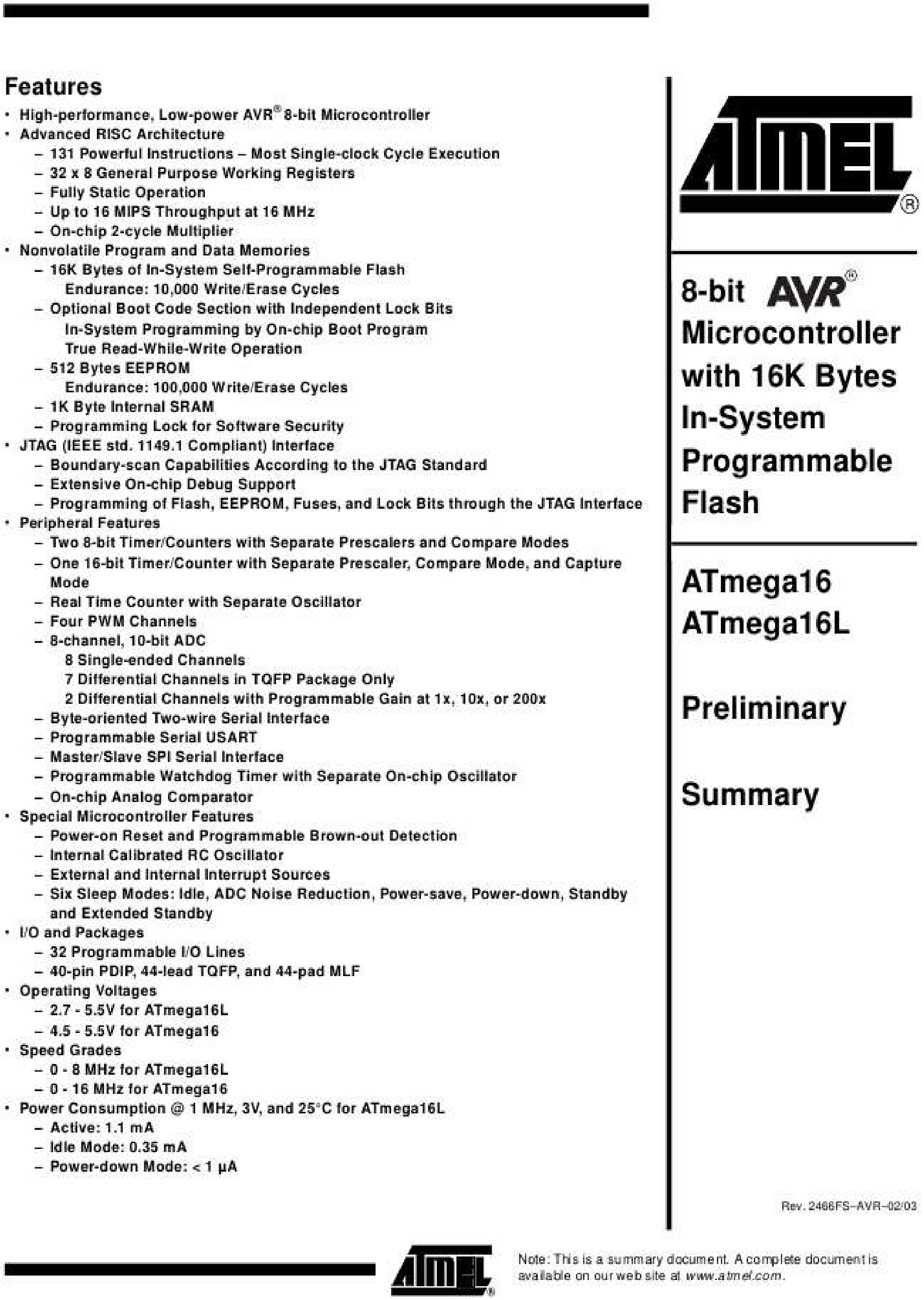
```
          PORTD=ADCH;

}while (PORTC > 0X1F);

if (PORTC<=0X1f)   PORTB=0X01;

        } }
```

APPENDIXB



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 16 MIPS Throughput at 16 MHz
  - On-chip 2-cycle Multiplier
- Nonvolatile Program and Data Memories
  - 16K Bytes of In-System Self-Programmable Flash
    Endurance: 10,000 Write/Erase Cycles
  - Optional Boot Code Section with Independent Lock Bits
    In-System Programming by On-chip Boot Program
    True Read-While-Write Operation
  - 512 Bytes EEPROM
    Endurance: 100,000 Write/Erase Cycles
  - 1K Byte Internal SRAM
  - Programming Lock for Software Security
- JTAG (IEEE std. 1149.1 Compliant) Interface
  - Boundary-scan Capabilities According to the JTAG Standard
  - Extensive On-chip Debug Support
  - Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface
- Peripheral Features
  - Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes
  - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
  - Real Time Counter with Separate Oscillator
  - Four PWM Channels
  - 8-channel, 10-bit ADC
    8 Single-ended Channels
    7 Differential Channels in TQFP Package Only
    2 Differential Channels with Programmable Gain at 1x, 10x, or 200x
  - Byte-oriented Two-wire Serial Interface
  - Programmable Serial USART
  - Master/Slave SPI Serial Interface
  - Programmable Watchdog Timer with Separate On-chip Oscillator
  - On-chip Analog Comparator
- Special Microcontroller Features
  - Power-on Reset and Programmable Brown-out Detection
  - Internal Calibrated RC Oscillator
  - External and Internal Interrupt Sources
  - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby
- I/O and Packages
  - 32 Programmable I/O Lines
  - 40-pin PDIP, 44-lead TQFP, and 44-pad MLF
- Operating Voltages
  - 2.7 - 5.5V for ATmega16L
  - 4.5 - 5.5V for ATmega16
- Speed Grades
  - 0 - 8 MHz for ATmega16L
  - 0 - 16 MHz for ATmega16
- Power Consumption @ 1 MHz, 3V, and 25°C for ATmega16L
  - Active: 1.1 mA
  - Idle Mode: 0.35 mA
  - Power-down Mode: < 1 µA

8-bit AVR® Microcontroller with 16K Bytes In-System Programmable Flash

ATmega16
ATmega16L

Preliminary

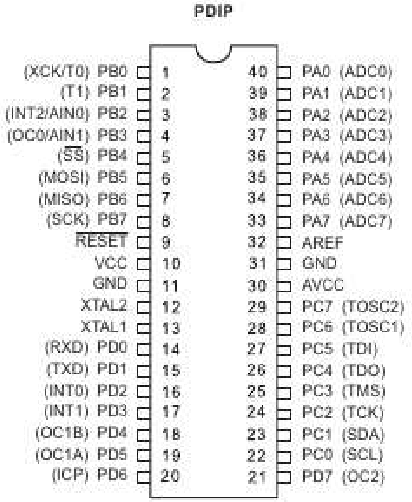
Summary

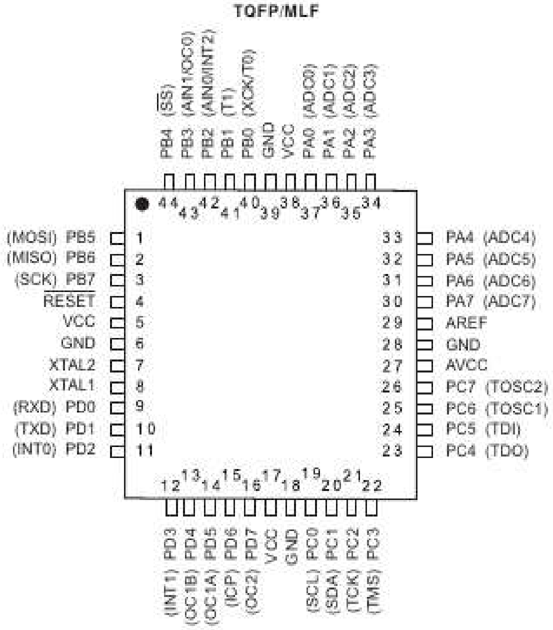
Rev. 2466FS–AVR–02/03

Note: This is a summary document. A complete document is available on our web site at *www.atmel.com*.

Pin Configurations

Figure 1. Pinouts ATmega16


PDIP


TQFP/MLF

Disclaimer

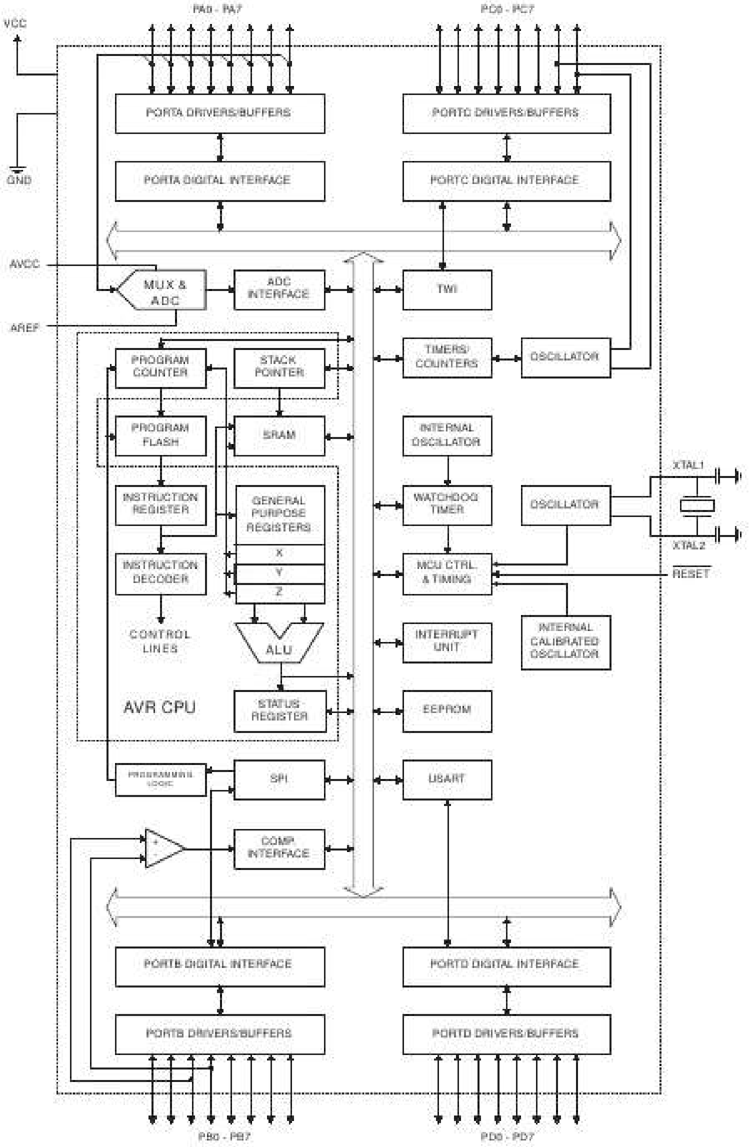
Typical values contained in this data sheet are based on simulations and characterization of other AVR microcontrollers manufactured on the same process technology. Min and Max values will be available after the device is characterized.

Overview

The ATmega16 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega16 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

Block Diagram

Figure 2. Block Diagram



The AVR core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The ATmega16 provides the following features: 16K bytes of In-System Programmable Flash Program memory with Read-While-Write capabilities, 512 bytes EEPROM, 1K byte SRAM, 32 general purpose I/O lines, 32 general purpose working registers, a JTAG interface for Boundary-scan, On-chip Debugging support and programming, three flexible Timer/Counters with compare modes, Internal and External Interrupts, a serial programmable USART, a byte oriented Two-wire Serial Interface, an 8-channel, 10-bit ADC with optional differential input stage with programmable gain (TQFP package only), a programmable Watchdog Timer with Internal Oscillator, an SPI serial port, and six software selectable power saving modes. The Idle mode stops the CPU while allowing the USART, Two-wire interface, A/D Converter, SRAM, Timer/Counters, SPI port, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next External Interrupt or Hardware Reset. In Power-save mode, the Asynchronous Timer continues to run, allowing the user to maintain a timer base while the rest of the device is sleeping. The ADC Noise Reduction mode stops the CPU and all I/O modules except Asynchronous Timer and ADC, to minimize switching noise during ADC conversions. In Standby mode, the crystal/resonator Oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low-power consumption. In Extended Standby mode, both the main Oscillator and the Asynchronous Timer continue to run.

The device is manufactured using Atmel's high density nonvolatile memory technology. The On-chip ISP Flash allows the program memory to be reprogrammed in-system through an SPI serial interface, by a conventional nonvolatile memory programmer, or by an On-chip Boot program running on the AVR core. The boot program can use any interface to download the application program in the Application Flash memory. Software in the Boot Flash section will continue to run while the Application Flash section is updated, providing true Read-While-Write operation. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash on a monolithic chip, the Atmel ATmega16 is a powerful microcontroller that provides a highly-flexible and cost-effective solution to many embedded control applications.

The ATmega16 AVR is supported with a full suite of program and system development tools including: C compilers, macro assemblers, program debugger/simulators, In-Circuit Emulators, and evaluation kits.

Pin Descriptions

VCC

Digital supply voltage.

GND

Ground.

Port A (PA7..PA0)

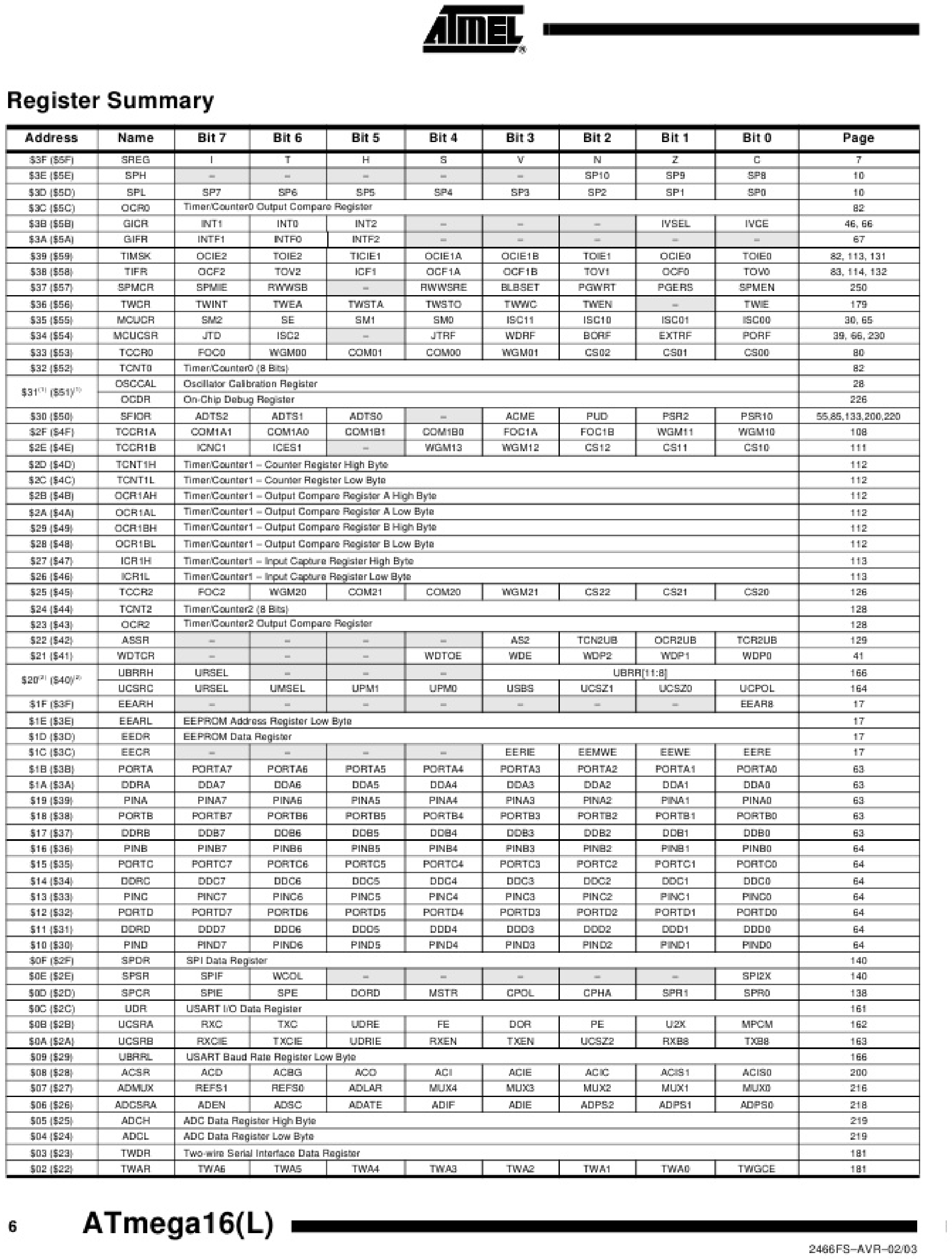
Port A serves as the analog inputs to the A/D Converter.

Port A also serves as an 8-bit bi-directional I/O port, if the A/D Converter is not used. Port pins can provide internal pull-up resistors (selected for each bit). The Port A output buffers have symmetrical drive characteristics with both high sink and source capability. When pins PA0 to PA7 are used as inputs and are externally pulled low, they will source current if the internal pull-up resistors are activated. The Port A pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port B (PB7..PB0)	Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running. Port B also serves the functions of various special features of the ATmega16 as listed on page 56.
Port C (PC7..PC0)	Port C is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port C output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running. If the JTAG interface is enabled, the pull-up resistors on pins PC5(TDI), PC3(TMS) and PC2(TCK) will be activated even if a reset occurs. Port C also serves the functions of the JTAG interface and other special features of the ATmega16 as listed on page 59.
Port D (PD7..PD0)	Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running. Port D also serves the functions of various special features of the ATmega16 as listed on page 61.
$\overline{\text{RESET}}$	Reset Input. A low level on this pin for longer than the minimum pulse length will generate a reset, even if the clock is not running. The minimum pulse length is given in Table 15 on page 36. Shorter pulses are not guaranteed to generate a reset.
XTAL1	Input to the inverting Oscillator amplifier and input to the internal clock operating circuit.
XTAL2	Output from the inverting Oscillator amplifier.
AVCC	AVCC is the supply voltage pin for Port A and the A/D Converter. It should be externally connected to V_{CC}, even if the ADC is not used. If the ADC is used, it should be connected to V_{CC} through a low-pass filter.
AREF	AREF is the analog reference pin for the A/D Converter.

About Code Examples

This documentation contains simple code examples that briefly show how to use various parts of the device. These code examples assume that the part specific header file is included before compilation. Be aware that not all C Compiler vendors include bit definitions in the header files and interrupt handling in C is compiler dependent. Please confirm with the C Compiler documentation for more details.



Register Summary

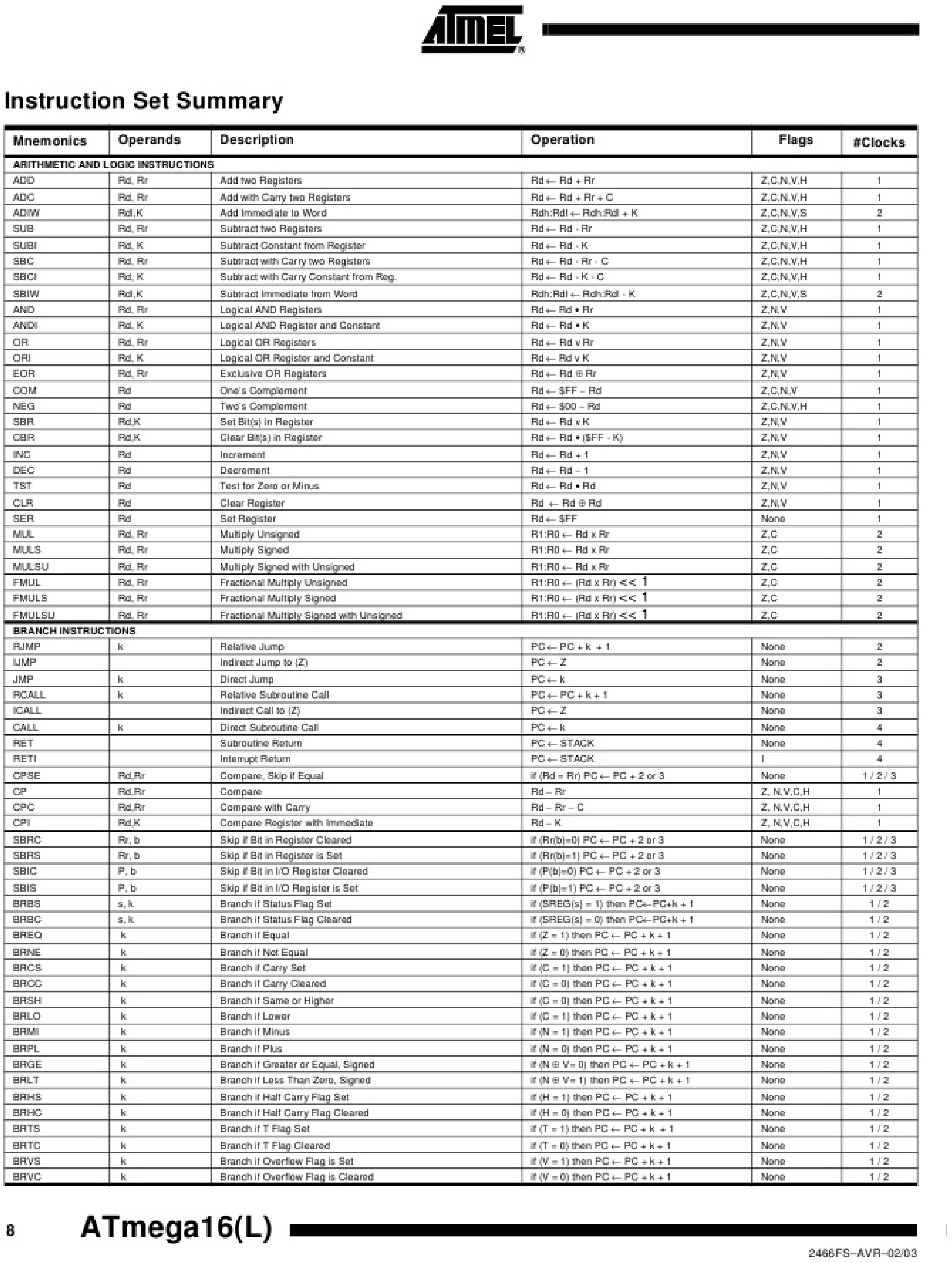
Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
$3F ($5F)	SREG	I	T	H	S	V	N	Z	C	7
$3E ($5E)	SPH	–	–	–	–	–	SP10	SP9	SP8	10
$3D ($5D)	SPL	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	10
$3C ($5C)	OCR0	Timer/Counter0 Output Compare Register								82
$3B ($5B)	GICR	INT1	INT0	INT2	–	–	–	IVSEL	IVCE	46, 66
$3A ($5A)	GIFR	INTF1	INTF0	INTF2	–	–	–	–	–	67
$39 ($59)	TIMSK	OCIE2	TOIE2	TICIE1	OCIE1A	OCIE1B	TOIE1	OCIE0	TOIE0	82, 113, 131
$38 ($58)	TIFR	OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	OCF0	TOV0	83, 114, 132
$37 ($57)	SPMCR	SPMIE	RWWSB	–	RWWSRE	BLBSET	PGWRT	PGERS	SPMEN	250
$36 ($56)	TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	–	TWIE	179
$35 ($55)	MCUCR	SM2	SE	SM1	SM0	ISC11	ISC10	ISC01	ISC00	30, 65
$34 ($54)	MCUCSR	JTD	ISC2	–	JTRF	WDRF	BORF	EXTRF	PORF	39, 66, 230
$33 ($53)	TCCR0	FOC0	WGM00	COM01	COM00	WGM01	CS02	CS01	CS00	80
$32 ($52)	TCNT0	Timer/Counter0 (8 Bits)								82
$31(1) ($51)(1)	OSCCAL	Oscillator Calibration Register								28
	OCDR	On-Chip Debug Register								226
$30 ($50)	SFIOR	ADTS2	ADTS1	ADTS0	–	ACME	PUD	PSR2	PSR10	55,85,133,200,220
$2F ($4F)	TCCR1A	COM1A1	COM1A0	COM1B1	COM1B0	FOC1A	FOC1B	WGM11	WGM10	108
$2E ($4E)	TCCR1B	ICNC1	ICES1	–	WGM13	WGM12	CS12	CS11	CS10	111
$2D ($4D)	TCNT1H	Timer/Counter1 – Counter Register High Byte								112
$2C ($4C)	TCNT1L	Timer/Counter1 – Counter Register Low Byte								112
$2B ($4B)	OCR1AH	Timer/Counter1 – Output Compare Register A High Byte								112
$2A ($4A)	OCR1AL	Timer/Counter1 – Output Compare Register A Low Byte								112
$29 ($49)	OCR1BH	Timer/Counter1 – Output Compare Register B High Byte								112
$28 ($48)	OCR1BL	Timer/Counter1 – Output Compare Register B Low Byte								112
$27 ($47)	ICR1H	Timer/Counter1 – Input Capture Register High Byte								113
$26 ($46)	ICR1L	Timer/Counter1 – Input Capture Register Low Byte								113
$25 ($45)	TCCR2	FOC2	WGM20	COM21	COM20	WGM21	CS22	CS21	CS20	126
$24 ($44)	TCNT2	Timer/Counter2 (8 Bits)								128
$23 ($43)	OCR2	Timer/Counter2 Output Compare Register								128
$22 ($42)	ASSR	–	–	–	–	AS2	TCN2UB	OCR2UB	TCR2UB	129
$21 ($41)	WDTCR	–	–	–	WDTOE	WDE	WDP2	WDP1	WDP0	41
$20(2) ($40)(2)	UBRRH	URSEL	–	–	–	UBRR[11:8]				166
	UCSRC	URSEL	UMSEL	UPM1	UPM0	USBS	UCSZ1	UCSZ0	UCPOL	164
$1F ($3F)	EEARH	–	–	–	–	–	–	–	EEAR8	17
$1E ($3E)	EEARL	EEPROM Address Register Low Byte								17
$1D ($3D)	EEDR	EEPROM Data Register								17
$1C ($3C)	EECR	–	–	–	–	EERIE	EEMWE	EEWE	EERE	17
$1B ($3B)	PORTA	PORTA7	PORTA6	PORTA5	PORTA4	PORTA3	PORTA2	PORTA1	PORTA0	63
$1A ($3A)	DDRA	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0	63
$19 ($39)	PINA	PINA7	PINA6	PINA5	PINA4	PINA3	PINA2	PINA1	PINA0	63
$18 ($38)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	63
$17 ($37)	DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	63
$16 ($36)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	64
$15 ($35)	PORTC	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	64
$14 ($34)	DDRC	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0	64
$13 ($33)	PINC	PINC7	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0	64
$12 ($32)	PORTD	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	64
$11 ($31)	DDRD	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	64
$10 ($30)	PIND	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	64
$0F ($2F)	SPDR	SPI Data Register								140
$0E ($2E)	SPSR	SPIF	WCOL	–	–	–	–	–	SPI2X	140
$0D ($2D)	SPCR	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	138
$0C ($2C)	UDR	USART I/O Data Register								161
$0B ($2B)	UCSRA	RXC	TXC	UDRE	FE	DOR	PE	U2X	MPCM	162
$0A ($2A)	UCSRB	RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8	163
$09 ($29)	UBRRL	USART Baud Rate Register Low Byte								166
$08 ($28)	ACSR	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	200
$07 ($27)	ADMUX	REFS1	REFS0	ADLAR	MUX4	MUX3	MUX2	MUX1	MUX0	216
$06 ($26)	ADCSRA	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0	218
$05 ($25)	ADCH	ADC Data Register High Byte								219
$04 ($24)	ADCL	ADC Data Register Low Byte								219
$03 ($23)	TWDR	Two-wire Serial Interface Data Register								181
$02 ($22)	TWAR	TWA6	TWA5	TWA4	TWA3	TWA2	TWA1	TWA0	TWGCE	181

6 **ATmega16(L)**

2466FS–AVR–02/03

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
$01 ($21)	TWSR	TWS7	TWS6	TWS5	TWS4	TWS3	–	TWPS1	TWPS0	180
$00 ($20)	TWBR	Two-wire Serial Interface Bit Rate Register								179

Notes:
1. When the OCDEN Fuse is unprogrammed, the OSCCAL Register is always accessed on this address. Refer to the debugger specific documentation for details on how to use the OCDR Register.
2. Refer to the USART description for details on how to access UBRRH and UCSRC.
3. For compatibility with future devices, reserved bits should be written to zero if accessed. Reserved I/O memory addresses should never be written.
4. Some of the status flags are cleared by writing a logical one to them. Note that the CBI and SBI instructions will operate on all bits in the I/O register, writing a one back into any flag read as set, thus clearing the flag. The CBI and SBI instructions work with registers $00 to $1F only.



Instruction Set Summary

Mnemonics	Operands	Description	Operation	Flags	#Clocks
ARITHMETIC AND LOGIC INSTRUCTIONS					
ADD	Rd, Rr	Add two Registers	Rd ← Rd + Rr	Z,C,N,V,H	1
ADC	Rd, Rr	Add with Carry two Registers	Rd ← Rd + Rr + C	Z,C,N,V,H	1
ADIW	Rdl,K	Add Immediate to Word	Rdh:Rdl ← Rdh:Rdl + K	Z,C,N,V,S	2
SUB	Rd, Rr	Subtract two Registers	Rd ← Rd - Rr	Z,C,N,V,H	1
SUBI	Rd, K	Subtract Constant from Register	Rd ← Rd - K	Z,C,N,V,H	1
SBC	Rd, Rr	Subtract with Carry two Registers	Rd ← Rd - Rr - C	Z,C,N,V,H	1
SBCI	Rd, K	Subtract with Carry Constant from Reg.	Rd ← Rd - K - C	Z,C,N,V,H	1
SBIW	Rdl,K	Subtract Immediate from Word	Rdh:Rdl ← Rdh:Rdl - K	Z,C,N,V,S	2
AND	Rd, Rr	Logical AND Registers	Rd ← Rd • Rr	Z,N,V	1
ANDI	Rd, K	Logical AND Register and Constant	Rd ← Rd • K	Z,N,V	1
OR	Rd, Rr	Logical OR Registers	Rd ← Rd v Rr	Z,N,V	1
ORI	Rd, K	Logical OR Register and Constant	Rd ← Rd v K	Z,N,V	1
EOR	Rd, Rr	Exclusive OR Registers	Rd ← Rd ⊕ Rr	Z,N,V	1
COM	Rd	One's Complement	Rd ← $FF – Rd	Z,C,N,V	1
NEG	Rd	Two's Complement	Rd ← $00 – Rd	Z,C,N,V,H	1
SBR	Rd,K	Set Bit(s) in Register	Rd ← Rd v K	Z,N,V	1
CBR	Rd,K	Clear Bit(s) in Register	Rd ← Rd • ($FF - K)	Z,N,V	1
INC	Rd	Increment	Rd ← Rd + 1	Z,N,V	1
DEC	Rd	Decrement	Rd ← Rd – 1	Z,N,V	1
TST	Rd	Test for Zero or Minus	Rd ← Rd • Rd	Z,N,V	1
CLR	Rd	Clear Register	Rd ← Rd ⊕ Rd	Z,N,V	1
SER	Rd	Set Register	Rd ← $FF	None	1
MUL	Rd, Rr	Multiply Unsigned	R1:R0 ← Rd x Rr	Z,C	2
MULS	Rd, Rr	Multiply Signed	R1:R0 ← Rd x Rr	Z,C	2
MULSU	Rd, Rr	Multiply Signed with Unsigned	R1:R0 ← Rd x Rr	Z,C	2
FMUL	Rd, Rr	Fractional Multiply Unsigned	R1:R0 ← (Rd x Rr) << 1	Z,C	2
FMULS	Rd, Rr	Fractional Multiply Signed	R1:R0 ← (Rd x Rr) << 1	Z,C	2
FMULSU	Rd, Rr	Fractional Multiply Signed with Unsigned	R1:R0 ← (Rd x Rr) << 1	Z,C	2
BRANCH INSTRUCTIONS					
RJMP	k	Relative Jump	PC ← PC + k + 1	None	2
IJMP		Indirect Jump to (Z)	PC ← Z	None	2
JMP	k	Direct Jump	PC ← k	None	3
RCALL	k	Relative Subroutine Call	PC ← PC + k + 1	None	3
ICALL		Indirect Call to (Z)	PC ← Z	None	3
CALL	k	Direct Subroutine Call	PC ← k	None	4
RET		Subroutine Return	PC ← STACK	None	4
RETI		Interrupt Return	PC ← STACK	I	4
CPSE	Rd,Rr	Compare, Skip if Equal	if (Rd = Rr) PC ← PC + 2 or 3	None	1 / 2 / 3
CP	Rd,Rr	Compare	Rd – Rr	Z, N,V,C,H	1
CPC	Rd,Rr	Compare with Carry	Rd – Rr – C	Z, N,V,C,H	1
CPI	Rd,K	Compare Register with Immediate	Rd – K	Z, N,V,C,H	1
SBRC	Rr, b	Skip if Bit in Register Cleared	if (Rr(b)=0) PC ← PC + 2 or 3	None	1 / 2 / 3
SBRS	Rr, b	Skip if Bit in Register is Set	if (Rr(b)=1) PC ← PC + 2 or 3	None	1 / 2 / 3
SBIC	P, b	Skip if Bit in I/O Register Cleared	if (P(b)=0) PC ← PC + 2 or 3	None	1 / 2 / 3
SBIS	P, b	Skip if Bit in I/O Register is Set	if (P(b)=1) PC ← PC + 2 or 3	None	1 / 2 / 3
BRBS	s, k	Branch if Status Flag Set	if (SREG(s) = 1) then PC←PC+k + 1	None	1 / 2
BRBC	s, k	Branch if Status Flag Cleared	if (SREG(s) = 0) then PC←PC+k + 1	None	1 / 2
BREQ	k	Branch if Equal	if (Z = 1) then PC ← PC + k + 1	None	1 / 2
BRNE	k	Branch if Not Equal	if (Z = 0) then PC ← PC + k + 1	None	1 / 2
BRCS	k	Branch if Carry Set	if (C = 1) then PC ← PC + k + 1	None	1 / 2
BRCC	k	Branch if Carry Cleared	if (C = 0) then PC ← PC + k + 1	None	1 / 2
BRSH	k	Branch if Same or Higher	if (C = 0) then PC ← PC + k + 1	None	1 / 2
BRLO	k	Branch if Lower	if (C = 1) then PC ← PC + k + 1	None	1 / 2
BRMI	k	Branch if Minus	if (N = 1) then PC ← PC + k + 1	None	1 / 2
BRPL	k	Branch if Plus	if (N = 0) then PC ← PC + k + 1	None	1 / 2
BRGE	k	Branch if Greater or Equal, Signed	if (N ⊕ V= 0) then PC ← PC + k + 1	None	1 / 2
BRLT	k	Branch if Less Than Zero, Signed	if (N ⊕ V= 1) then PC ← PC + k + 1	None	1 / 2
BRHS	k	Branch if Half Carry Flag Set	if (H = 1) then PC ← PC + k + 1	None	1 / 2
BRHC	k	Branch if Half Carry Flag Cleared	if (H = 0) then PC ← PC + k + 1	None	1 / 2
BRTS	k	Branch if T Flag Set	if (T = 1) then PC ← PC + k + 1	None	1 / 2
BRTC	k	Branch if T Flag Cleared	if (T = 0) then PC ← PC + k + 1	None	1 / 2
BRVS	k	Branch if Overflow Flag is Set	if (V = 1) then PC ← PC + k + 1	None	1 / 2
BRVC	k	Branch if Overflow Flag is Cleared	if (V = 0) then PC ← PC + k + 1	None	1 / 2

Mnemonics	Operands	Description	Operation	Flags	#Clocks
BRIE	k	Branch if Interrupt Enabled	if (I = 1) then PC ← PC + k + 1	None	1 / 2
BRID	k	Branch if Interrupt Disabled	if (I = 0) then PC ← PC + k + 1	None	1 / 2
DATA TRANSFER INSTRUCTIONS					
MOV	Rd, Rr	Move Between Registers	Rd ← Rr	None	1
MOVW	Rd, Rr	Copy Register Word	Rd+1:Rd ← Rr+1:Rr	None	1
LDI	Rd, K	Load Immediate	Rd ← K	None	1
LD	Rd, X	Load Indirect	Rd ← (X)	None	2
LD	Rd, X+	Load Indirect and Post-Inc.	Rd ← (X), X ← X + 1	None	2
LD	Rd, - X	Load Indirect and Pre-Dec.	X ← X - 1, Rd ← (X)	None	2
LD	Rd, Y	Load Indirect	Rd ← (Y)	None	2
LD	Rd, Y+	Load Indirect and Post-Inc.	Rd ← (Y), Y ← Y + 1	None	2
LD	Rd, - Y	Load Indirect and Pre-Dec.	Y ← Y - 1, Rd ← (Y)	None	2
LDD	Rd,Y+q	Load Indirect with Displacement	Rd ← (Y + q)	None	2
LD	Rd, Z	Load Indirect	Rd ← (Z)	None	2
LD	Rd, Z+	Load Indirect and Post-Inc.	Rd ← (Z), Z ← Z+1	None	2
LD	Rd, -Z	Load Indirect and Pre-Dec.	Z ← Z - 1, Rd ← (Z)	None	2
LDD	Rd, Z+q	Load Indirect with Displacement	Rd ← (Z + q)	None	2
LDS	Rd, k	Load Direct from SRAM	Rd ← (k)	None	2
ST	X, Rr	Store Indirect	(X) ← Rr	None	2
ST	X+, Rr	Store Indirect and Post-Inc.	(X) ← Rr, X ← X + 1	None	2
ST	- X, Rr	Store Indirect and Pre-Dec.	X ← X - 1, (X) ← Rr	None	2
ST	Y, Rr	Store Indirect	(Y) ← Rr	None	2
ST	Y+, Rr	Store Indirect and Post-Inc.	(Y) ← Rr, Y ← Y + 1	None	2
ST	- Y, Rr	Store Indirect and Pre-Dec.	Y ← Y - 1, (Y) ← Rr	None	2
STD	Y+q,Rr	Store Indirect with Displacement	(Y + q) ← Rr	None	2
ST	Z, Rr	Store Indirect	(Z) ← Rr	None	2
ST	Z+, Rr	Store Indirect and Post-Inc.	(Z) ← Rr, Z ← Z + 1	None	2
ST	-Z, Rr	Store Indirect and Pre-Dec.	Z ← Z - 1, (Z) ← Rr	None	2
STD	Z+q,Rr	Store Indirect with Displacement	(Z + q) ← Rr	None	2
STS	k, Rr	Store Direct to SRAM	(k) ← Rr	None	2
LPM		Load Program Memory	R0 ← (Z)	None	3
LPM	Rd, Z	Load Program Memory	Rd ← (Z)	None	3
LPM	Rd, Z+	Load Program Memory and Post-Inc	Rd ← (Z), Z ← Z+1	None	3
SPM		Store Program Memory	(Z) ← R1:R0	None	-
IN	Rd, P	In Port	Rd ← P	None	1
OUT	P, Rr	Out Port	P ← Rr	None	1
PUSH	Rr	Push Register on Stack	STACK ← Rr	None	2
POP	Rd	Pop Register from Stack	Rd ← STACK	None	2
BIT AND BIT-TEST INSTRUCTIONS					
SBI	P,b	Set Bit in I/O Register	I/O(P,b) ← 1	None	2
CBI	P,b	Clear Bit in I/O Register	I/O(P,b) ← 0	None	2
LSL	Rd	Logical Shift Left	Rd(n+1) ← Rd(n), Rd(0) ← 0	Z,C,N,V	1
LSR	Rd	Logical Shift Right	Rd(n) ← Rd(n+1), Rd(7) ← 0	Z,C,N,V	1
ROL	Rd	Rotate Left Through Carry	Rd(0)←C,Rd(n+1)← Rd(n),C←Rd(7)	Z,C,N,V	1
ROR	Rd	Rotate Right Through Carry	Rd(7)←C,Rd(n)← Rd(n+1),C←Rd(0)	Z,C,N,V	1
ASR	Rd	Arithmetic Shift Right	Rd(n) ← Rd(n+1), n=0..6	Z,C,N,V	1
SWAP	Rd	Swap Nibbles	Rd(3..0)←Rd(7..4),Rd(7..4)←Rd(3..0)	None	1
BSET	s	Flag Set	SREG(s) ← 1	SREG(s)	1
BCLR	s	Flag Clear	SREG(s) ← 0	SREG(s)	1
BST	Rr, b	Bit Store from Register to T	T ← Rr(b)	T	1
BLD	Rd, b	Bit load from T to Register	Rd(b) ← T	None	1
SEC		Set Carry	C ← 1	C	1
CLC		Clear Carry	C ← 0	C	1
SEN		Set Negative Flag	N ← 1	N	1
CLN		Clear Negative Flag	N ← 0	N	1
SEZ		Set Zero Flag	Z ← 1	Z	1
CLZ		Clear Zero Flag	Z ← 0	Z	1
SEI		Global Interrupt Enable	I ← 1	I	1
CLI		Global Interrupt Disable	I ← 0	I	1
SES		Set Signed Test Flag	S ← 1	S	1
CLS		Clear Signed Test Flag	S ← 0	S	1
SEV		Set Twos Complement Overflow.	V ← 1	V	1
CLV		Clear Twos Complement Overflow	V ← 0	V	1
SET		Set T in SREG	T ← 1	T	1
CLT		Clear T in SREG	T ← 0	T	1
SEH		Set Half Carry Flag in SREG	H ← 1	H	1

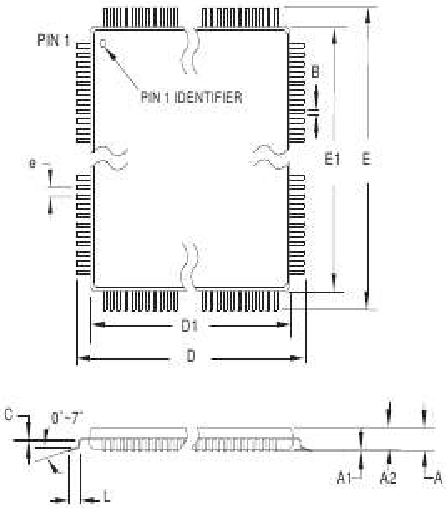
Ordering Information

Speed (MHz)	Power Supply	Ordering Code	Package	Operation Range
8	2.7 - 5.5V	ATmega16L-8AC ATmega16L-8PC ATmega16L-8MC	44A 40P6 44M1	Commercial (0°C to 70°C)
		ATmega16L-8AI ATmega16L-8PI ATmega16L-8MI	44A 40P6 44M1	Industrial (-40°C to 85°C)
16	4.5 - 5.5V	ATmega16-16AC ATmega16-16PC ATmega16-16MI	44A 40P6 44M1	Commercial (0°C to 70°C)
		ATmega16-16AI ATmega16-16PI ATmega16-16MC	44A 40P6 44M1	Industrial (-40°C to 85°C)

Package Type	
44A	44-lead, Thin (1.0 mm) Plastic Gull Wing Quad Flat Package (TQFP)
40P6	40-pin, 0.600” Wide, Plastic Dual Inline Package (PDIP)
44M1	44-pad, 7 x 7 x 1.0 mm body, lead pitch 0.50 mm, Micro Lead Frame Package (MLF)

Packaging Information

44A



COMMON DIMENSIONS
(Unit of Measure = mm)

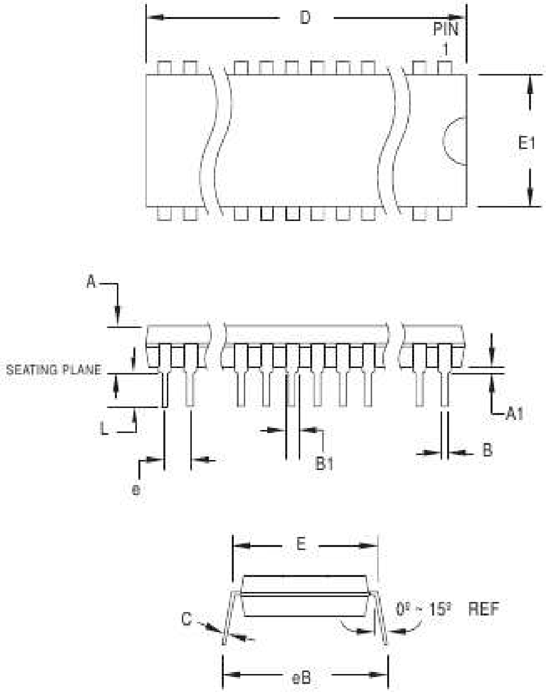
SYMBOL	MIN	NOM	MAX	NOTE
A	–	–	1.20	
A1	0.05	–	0.15	
A2	0.95	1.00	1.05	
D	11.75	12.00	12.25	
D1	9.90	10.00	10.10	Note 2
E	11.75	12.00	12.25	
E1	9.90	10.00	10.10	Note 2
B	0.30	–	0.45	
C	0.09	–	0.20	
L	0.45	–	0.75	
e	0.80 TYP			

Notes:
1. This package conforms to JEDEC reference MS-026, Variation ACB.
2. Dimensions D1 and E1 do not include mold protrusion. Allowable protrusion is 0.25 mm per side. Dimensions D1 and E1 are maximum plastic body size dimensions including mold mismatch.
3. Lead coplanarity is 0.10 mm maximum.

10/5/2001

	TITLE	DRAWING NO.	REV.
2325 Orchard Parkway San Jose, CA 95131	**44A**, 44-lead, 10 x 10 mm Body Size, 1.0 mm Body Thickness, 0.8 mm Lead Pitch, Thin Profile Plastic Quad Flat Package (TQFP)	44A	B

40P6



COMMON DIMENSIONS
(Unit of Measure = mm)

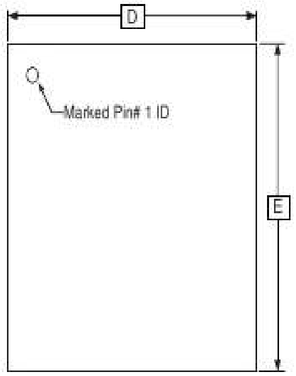
SYMBOL	MIN	NOM	MAX	NOTE
A	–	–	4.826	
A1	0.381	–	–	
D	52.070	–	52.578	Note 2
E	15.240	–	15.875	
E1	13.462	–	13.970	Note 2
B	0.356	–	0.559	
B1	1.041	–	1.651	
L	3.048	–	3.556	
C	0.203	–	0.381	
eB	15.494	–	17.526	
e	2.540 TYP			

Notes:
1. This package conforms to JEDEC reference MS-011, Variation AC.
2. Dimensions D and E1 do not include mold Flash or Protrusion. Mold Flash or Protrusion shall not exceed 0.25 mm (0.010").

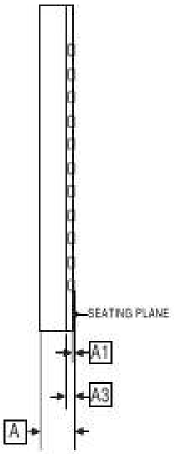
09/28/01

	TITLE	DRAWING NO.	REV.
2325 Orchard Parkway San Jose, CA 95131	**40P6**, 40-lead (0.600"/15.24 mm Wide) Plastic Dual Inline Package (PDIP)	40P6	B

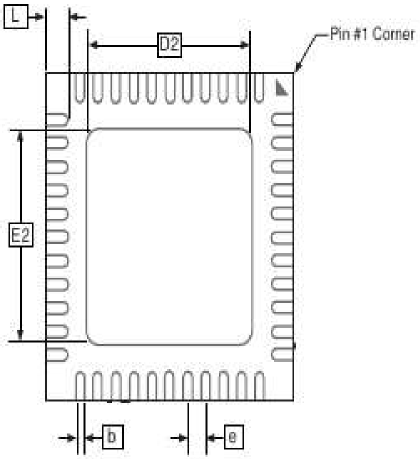
44M1



TOP VIEW



SIDE VIEW



BOTTOM VIEW

COMMON DIMENSIONS
(Unit of Measure = mm)

SYMBOL	MIN	NOM	MAX	NOTE
A	0.80	0.90	1.00	
A1	–	0.02	0.05	
A3		0.25 REF		
b	0.18	0.23	0.30	
D		7.00 BSC		
D2	5.00	5.20	5.40	
E		7.00 BSC		
E2	5.00	5.20	5.40	
e		0.50 BSC		
L	0.35	0.55	0.75	

Note: JEDEC Standard MO-220, Fig. 1 (SAW Singulation) VKKD-1.

01/15/03

	TITLE	DRAWING NO.	REV.
2325 Orchard Parkway San Jose, CA 95131	**44M1**, 44-pad, 7 x 7 x 1.0 mm Body, Lead Pitch 0.50 mm Micro Lead Frame Package (MLF)	44M1	C

Data Sheet Change Log for ATmega16

This section contains a log on the changes made to the data sheet for ATmega16.

Changes from Rev. 2466B-09/01 to Rev. 2466C-03/02

All page numbers refer to this document.

1. **Updated typical EEPROM programming time, Table 1 on page 18.**
2. **Updated typical start-up time in the following tables:**

   Table 3 on page 23, Table 5 on page 25, Table 6 on page 26, Table 8 on page 27, Table 9 on page 27, and Table 10 on page 28.
3. **Updated Table 17 on page 41 with typical WDT Time-out.**
4. **Added Some Preliminary Test Limits and Characterization Data.**

   Removed some of the TBD's in the following tables and pages:

   Table 15 on page 36, Table 16 on page 40, Table 116 on page 272 (table removed in document review #D), "Electrical Characteristics" on page 290, Table 119 on page 292, Table 121 on page 294, and Table 122 on page 296.
5. **Updated TWI Chapter.**

   Added the note at the end of the "Bit Rate Generator Unit" on page 177.
6. **Corrected description of ADSC bit in "ADC Control and Status Register A – ADCSRA" on page 218.**
7. **Improved description on how to do a polarity check of the ADC diff results in "ADC Conversion Result" on page 215.**
8. **Added JTAG version number for rev. H in Table 87 on page 228.**
9. **Added not regarding OCDEN Fuse below Table 105 on page 260.**
10. **Updated Programming Figures:**

    Figure 127 on page 262 and Figure 136 on page 273 are updated to also reflect that AVCC must be connected during Programming mode. Figure 131 on page 269 added to illustrate how to program the fuses.
11. **Added a note regarding usage of the "PROG_PAGELOAD ($6)" on page 279 and "PROG_PAGEREAD ($7)" on page 279.**
12. **Removed alternative algortihm for leaving JTAG Programming mode.**

    See "Leaving Programming Mode" on page 287.
13. **Added Calibrated RC Oscillator characterization curves in section "ATmega16 Typical Characteristics – Preliminary Data" on page 298.**
14. **Corrected ordering code for MLF package (16MHz) in "Ordering Information" on page 11.**
15. **Corrected Table 90, "Scan Signals for the Oscillators(1)(2)(3)," on page 234.**

Changes from Rev. 2466C-03/02 to Rev. 2466D-09/02

All page numbers refer to this document.

1. **Changed all Flash write/erase cycles from 1,000 to 10,000.**
2. **Updated the following tables: Table 4 on page 24, Table 15 on page 36, Table 42 on page 82, Table 45 on page 109, Table 46 on page 109, Table 59 on page 141, Table 67 on page 166, Table 90 on page 234, Table 102 on page 258, "DC Characteristics" on page 290, Table 119 on page 292, Table 121 on page 294, and Table 122 on page 296.**
3. **Updated "Erratas" on page 15.**

Changes from Rev. 2466D-09/02 to Rev. 2466E-10/02

All page numbers refer to this document.

1. **Updated "DC Characteristics" on page 290.**

Changes from Rev. 2466E-10/02 to Rev. 2466F-02/03

All page numbers refer to this document.

1. **Added note about masking out unused bits when reading the Program Counter in "Stack Pointer" on page 10.**
2. **Added Chip Erase as a first step in "Programming the Flash" on page 287 and "Programming the EEPROM" on page 288.**
3. **Added the section "Unconnected pins" on page 53.**
4. **Added tips on how to disable the OCD system in "On-chip Debug System" on page 34.**
5. **Removed reference to the "Multi-purpose Oscillator" application note and "32 kHz Crystal Oscillator" application note, which do not exist.**
6. **Added information about PWM symmetry for Timer0 and Timer2.**
7. **Added note in "Filling the Temporary Buffer (Page Loading)" on page 253 about writing to the EEPROM during an SPM Page Load.**
8. **Removed ADHSM completely.**
9. **Added Table 73, "TWI Bit Rate Prescaler," on page 181 to describe the TWPS bits in the "TWI Status Register – TWSR" on page 180.**
10. **Added section "Default Clock Source" on page 23.**
11. **Added note about frequency variation when using an external clock. Note added in "External Clock" on page 29. An extra row and a note added in Table 118 on page 292.**
12. **Various minor TWI corrections.**
13. **Added "Power Consumption" data in "Features" on page 1.**
14. **Added section "EEPROM Write During Power-down Sleep Mode" on page 20 .**

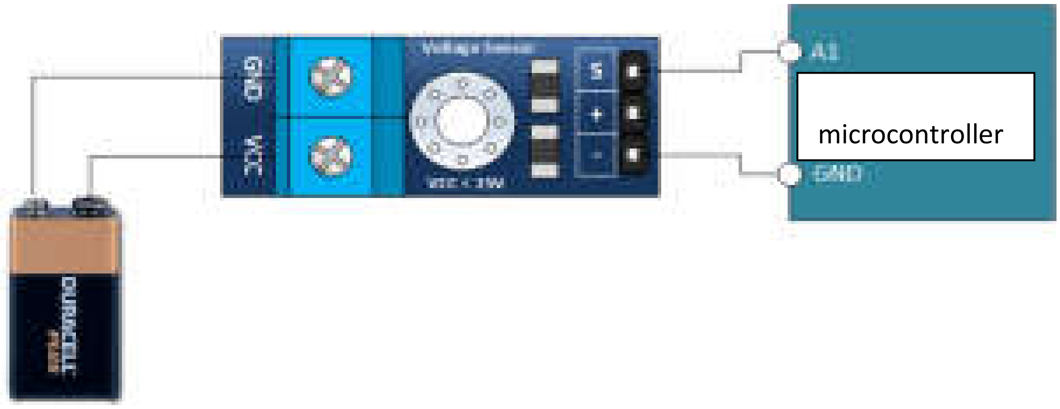
APPENDIX C

Voltage Detection Sensor Module 25V:

module is based on the principle of resistive voltage divider design, it can make the red terminal connector input voltage to 5 times smaller.Arduino analog input voltages up to 5 v. The voltage detection module input voltage not greater than 5Vx5=25V (if using 3.3V systems, input voltage not greater than 3.3Vx5=16.5V).It has the limit of microcontroller analog input 5 VDC only. So if you wish to measure higher voltages, you will need to resort to another means. One way is to use a voltage divider. The one discussed here is found all over Amazon and eBay. It is fundamentally a 5:1 voltage divider using a 30K and a 7.5K Ohm resistor.

Note: Keep in mind, you are restricted to voltages that are less than 25 volts. More than that and you will exceed the voltage limit of your microcontroller input.

Connection Diagrame:



Package Includes :

1×Voltage Detection Sensor Module.

Specifications And Features :

1. Dimensions: 28 x 14 x 13 mm(LxWxH).
2. Weight: 4 gm.
3. Input Voltage range: DC0 to 25 V
4. Voltage detection range: DC 0.02445 V to 25 V
5. Analog Voltage resolution : 0.00489 V
6. Output Interface : “+ ” connected 5/3.3V, “-” connected GND, “s” connected microcontroller AD pins
7. DC input interface: red terminal positive with VCC, negative with GND
8. You can also use the IICLCD1602 LCD to display voltage.
9. By 3P connector, connect this module with the expansion of board microcontroller, not only makes it easier for you to detect voltage battery.