Embended systems
Microcontrollers
Fall 2016
Laboratory Work 4

Vasile Schidu

January 22, 2017

Date Performed:	October 02, 2016
Partners:	Vasile Schidu
Instructor:	Bragrarenco Andrei

Introduction

Topic: Pulse width modulation. Controlling motor with H-Bridge

Objectives:

1. Implement H-Bridge
2. Create a basic car using elements described higher

Tasks: Write a C program and schematics for a car using Universal asynchronous receiver/transmitter, h-bridge, pulse width modulation. Use keyboard as control for wheels. Car should be able to steer, increase velocitym decrease velocity, stop or free wheeling.

PWM

Definition Pulse width modulation (PWM) is a fancy term for describing a type of digital signal. Pulse width modulation is used in a variety of applications including sophisticated control circuitry. A common way we use them here at SparkFun is to control dimming of RGB LEDs or to control the direction of a servo motor. We can accomplish a range of results in both applications because pulse width modulation allows us to vary how much time the signal is high in an analog fashion. While the signal can only be high (usually 5V) or low (ground) at any time, we can change the proportion of time the signal is high compared to when it is low over a consistent time interval.

Duty Cycle When the signal is high, we call this "on time". To describe the amount of "on time" , we use the concept of duty cycle. Duty cycle is measured in percentage. The percentage duty cycle specifically describes the percentage of time a digital signal is on over an interval or period of time. This period is the inverse of the frequency of the waveform. If a digital signal spends half of the time on and the other half off, we would say the digital signal has a duty cycle of 50 procente and resembles an ideal square wave. If the percentage is higher than 50 procente, the digital signal spends more time in the high state than the low state and vice-versa if the duty cycle is less than 50 procente. Here is a graph that illustrates these three scenarios

H-Bridge

Definition An H bridge is an electronic circuit that enables a voltage to be applied across a load in either direction. These circuits are often used in robotics and other applications to allow DC motors to run forwards or backwards. Most DCto-AC converters (power inverters), most AC/AC converters, the DC-to-DC push–pull converter, most motor controllers, and many other kinds of power electronics use H bridges. In particular, a bipolar stepper motor is almost invariably driven by a motor controller containing two H bridges.

Resourses

Short Theory: Proteus developed by Labcenter Electronics, is a software with which you can easily generate schematic captures, develop PCB and simulate microprocessor. It has such a simple yet effective interface that it simplifies the task required to be performed. This one aspect has attracted many users to select this tool amongst many others offering the same services. Atmel® Studio 6 is the integrated development platform (IDP) for developing and debugging Atmel ARM® Cortex®-M and Atmel AVR® microcontroller (MCU) based applications. The Atmel Studio 6 IDP gives you a seamless and easy-to-use environment to write, build and debug your applications written in C/C++ or assembly code. A

microcontroller (sometimes abbreviated μC, uC or MCU) is a small computer on a single integrated circuit containing a processor core, memory, and programmable input/output peripherals. Microcontrollers are designed for embedded applications, in contrast to the microprocessors used in personal computers or other general purpose applications. In computing, a device driver (commonly referred to as a driver) is a computer program that operates or controls a particular type of device that is attached to a computer. A driver provides a software interface to hardware devices, enabling operating systems and other computer programs to access hardware functions without needing to know precise details of the hardware being used.

About stdio library: Input and Output operations can be performed using theCStandardInput andOutput Library (cstdio, known asstdio.hin the C language). This library uses what are calledstreamsto operate with physical devices such as keyboards, printers, terminals or with any other type of files supported by the system. Streams are an abstraction to interact with these in an uniform way. All streams have similar properties independently of the individual characteristics of the
physical media they are associated with. Streams are handled in thecstdiolibrary as pointers toFILEobjects. A pointer to aFILEobject uniquely identifies a stream, and is used as a parameter in the operations involving that stream.

Atmel® microcontrollers: Atmel® microcontrollers (MCUs) deliver a rich blend of efficient integrated designs, proven technology, and groundbreaking innovation that is ideal for today's smart, connected products. In this era of the Internet of Things (IoT), microcontrollers comprise a key technology that fuels machine-to-machine (M2M) communications. Building on decades of experience and industry leadership, Atmel offers proven architectures that are optimized for low power, high-speed connectivity, optimal data bandwidth, and rich interface support. By using our wide variety of configuration options, developers can devise complete system solutions for all kinds of applications. Atmel microcontrollers can also support seamless integration of capacitive touch technology to implement buttons, sliders, and wheels (BSW). In addition, Atmel MCUs deliver wireless and security support. No matter what your market or device, Atmel offers a compelling solution that is tailored to your needs—today and tomorrow. Atmel is a global industry leader in the design and manufacture of microcontrollers and related system solutions, including capacitive touch solutions, advanced logic, mixed-signal, nonvolatile memory, and radio frequency (RF) components. Leveraging one of the industry's broadest intellectual property technology portfolios and backed by a comprehensive ecosystem, Atmel MCU products enable designers to develop complete solutions for industrial, consumer, security, communications, computing, and automotive markets. Developers have the option of combining Atmel microcontrollers with industry-leading Atmel touch technology. Atmel technology for touchscreens and fixed-function buttons, sliders and wheels provides a rich user experience with unparalleled performance, while minimizing power consumption.

ATmega328

The high-performance Atmel 8-bit AVR RISC-based microcontroller combines 32KB ISP flash memory with read-while-write capabilities, 1KB EEPROM, 2KB SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible timer/counters with compare modes, internal and external interrupts,serial programmable USART, a byte-oriented 2-wire serial interface, SPI serial port, 6-channel 10-bit A/D converter (8-channels in TQFP and QFN/MLF packages), programmable watchdog timer with internal oscillator, and five software selectable power saving modes. The device operates between 1.8-5.5 volts. By executing powerful instructions in a single clock cycle, the device achieves throughputs approaching 1 MIPS per MHz, balancing power consumption and processing speed.

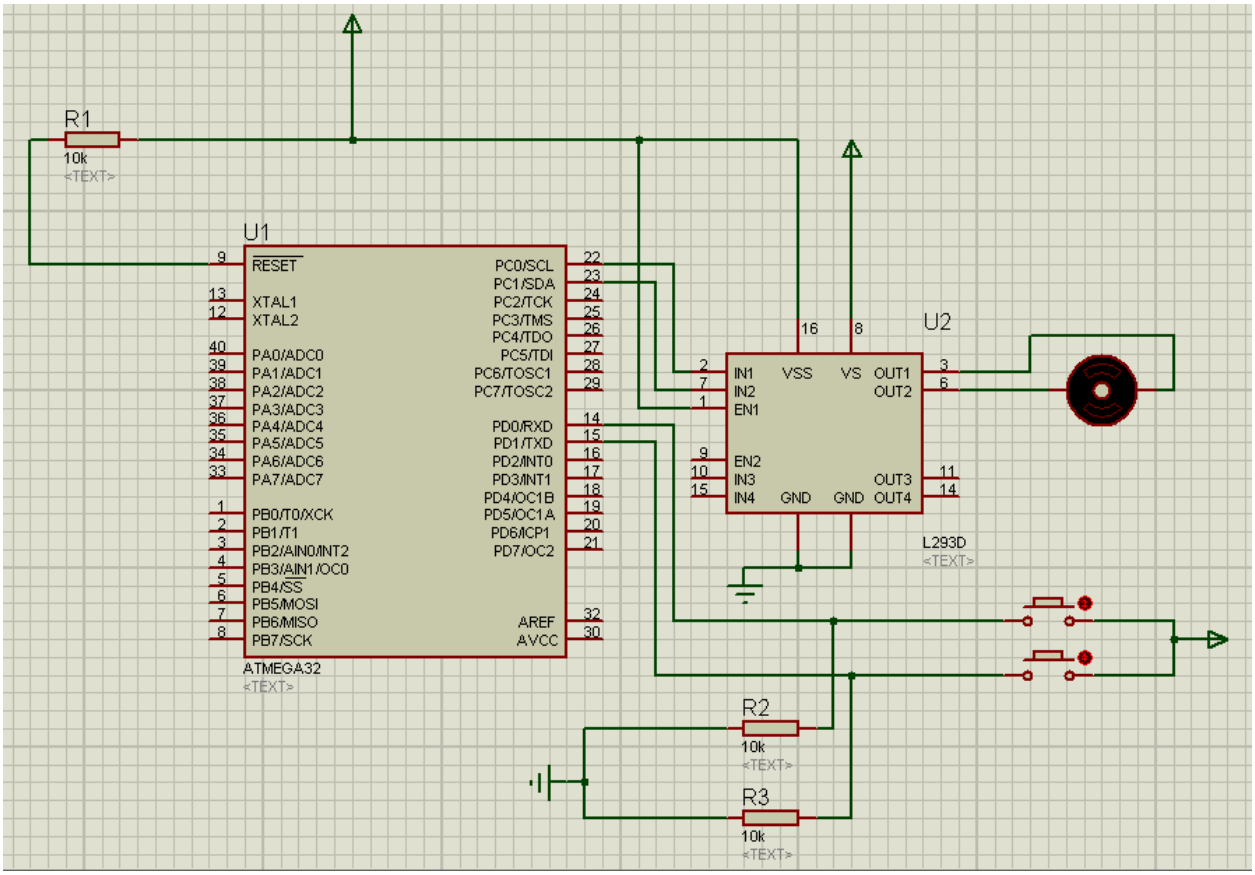
Atmel Studio: Atmel Studio 7 is the integrated development platform (IDP) for developing and debugging Atmel® SMART ARM®-based and Atmel AVR® microcontroller (MCU) applications. Studio 7 supports all AVR and Atmel SMART MCUs. The Atmel Studio 7 IDP gives you a seamless and easy-to-use environment to write, build and debug your applications written in C/C++ or assembly code. It also connects seamlessly to Atmel debuggers and development kits. Additionally, Atmel Studio includes Atmel Gallery,

an online apps store that allows you to extend your development environment with plug-ins developed by Atmel as well as by third-party tool and embedded software vendors. Atmel Studio 7 can also able seamlessly import your Arduino sketches as C++ projects, providing a simple transition path from Makerspace to Marketplace.

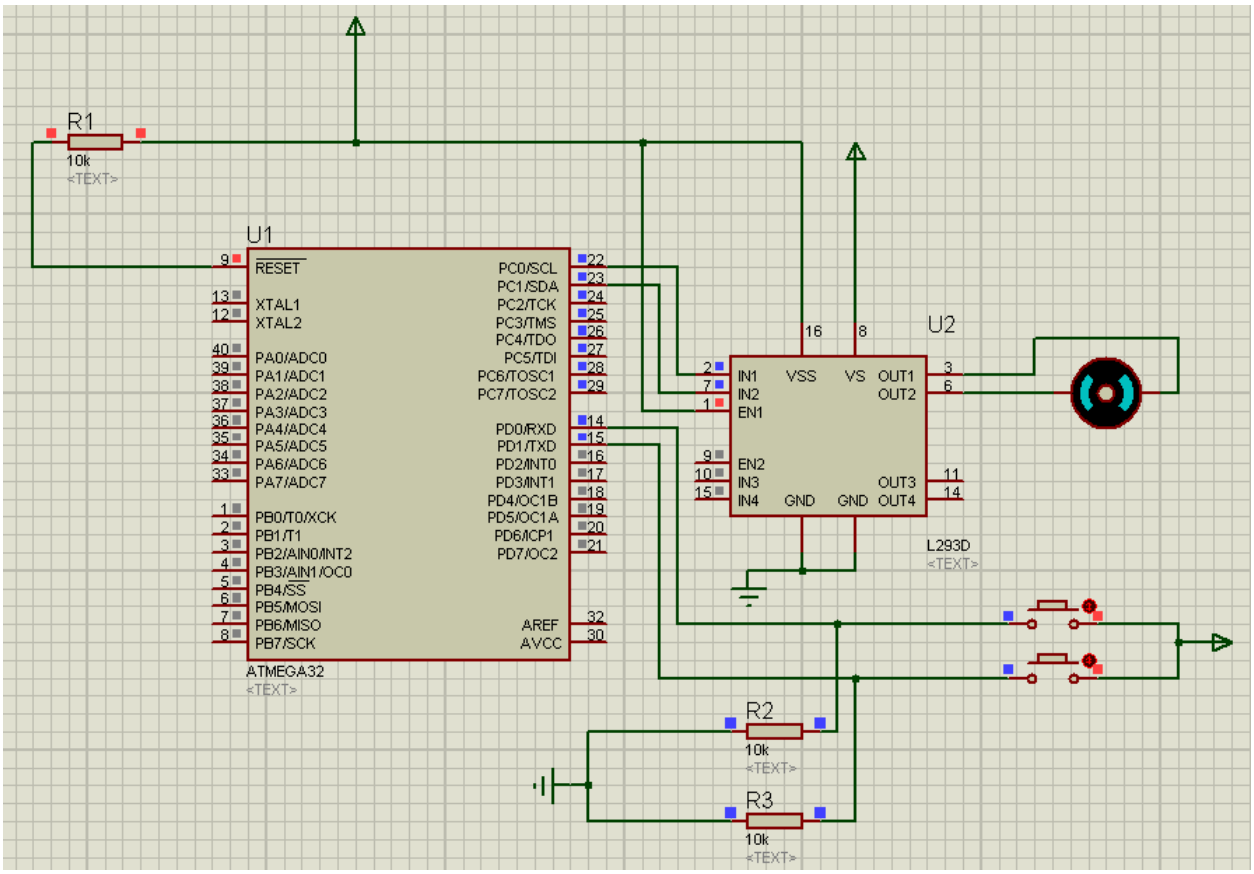
Solution

Proteus Scheme Let's take a look on this scheme. So we have Microcontroler **ATMEGA32** This microcontroler is composed from 4 ports each of thie have 8 pins.

So here is our project in Proteus.



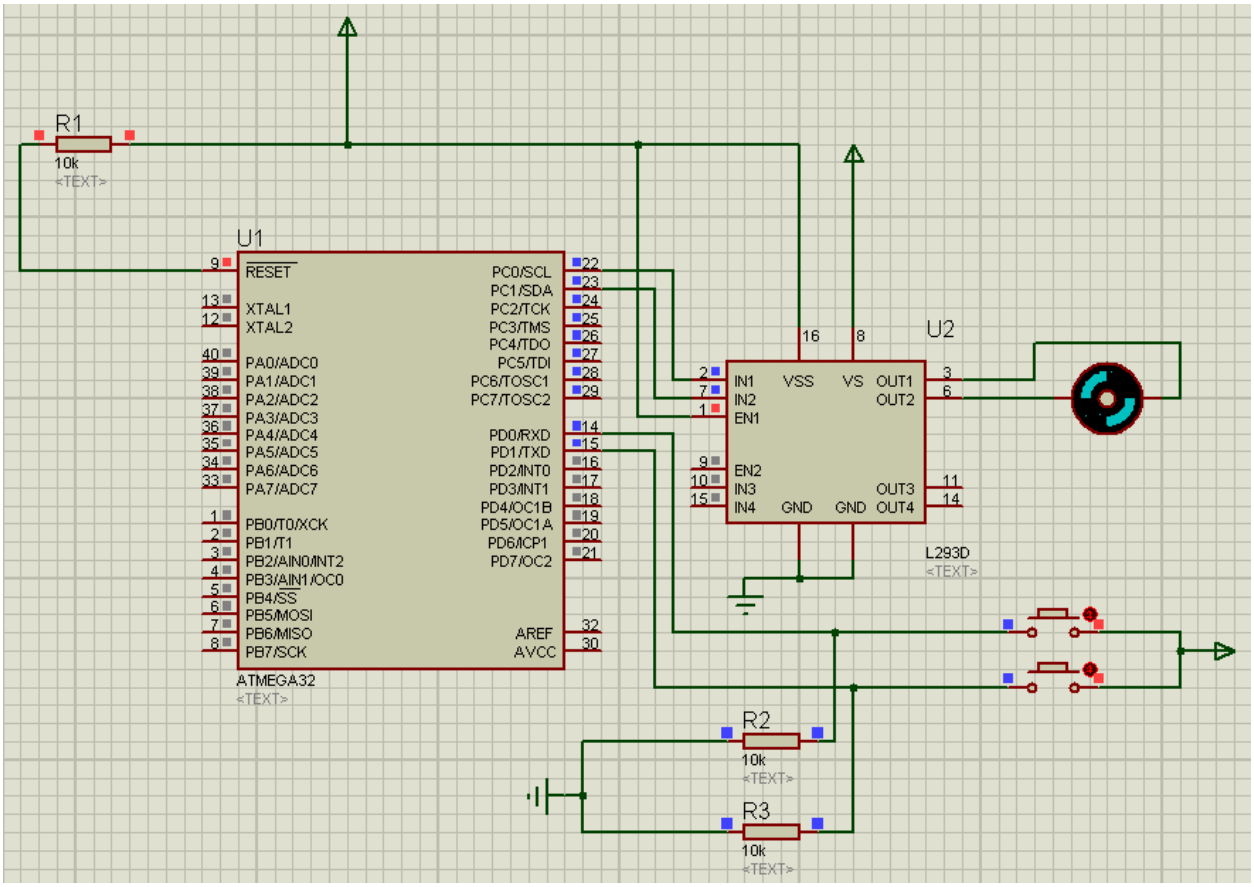
In the picture we cant see, but when I push the upper button motor is rotating to the left.



Here we pushed the lower button and motor keeps rotating to the right

Conclusion

In this laboratory work I've learned basic concepts of MCU programming in C language and building a simple printed circuit using Proteus. I've implemented a project that simulates the car behavior using a H-Bridge circuit and a motor.


R1
10k
<TEXT>
U1
RESET
XTAL1
XTAL2
PA0/ADC0
PA1/ADC1
PA2/ADC2
PA3/ADC3
PA4/ADC4
PA5/ADC5
PA6/ADC6
PA7/ADC7
PB0/T0/XCK
PB1/T1
PB2/AIN0/INT2
PB3/AIN1/OC0
PB4/SS
PB5/MOSI
PB6/MISO
PB7/SCK
PC0/SCL
PC1/SDA
PC2/TCK
PC3/TMS
PC4/TDO
PC5/TDI
PC6/TOSC1
PC7/TOSC2
PD0/RXD
PD1/TXD
PD2/INT0
PD3/INT1
PD4/OC1B
PD5/OC1A
PD6/ICP1
PD7/OC2
AREF
AVCC
ATMEGA32
<TEXT>
U2
IN1
IN2
EN1
EN2
IN3
IN4
VSS
VS
GND
GND
OUT1
OUT2
OUT3
OUT4
L293D
<TEXT>
R2
10k
<TEXT>
R3
10k
<TEXT>