

CS684 Documentation



CS684 – 2010 Project

Project: μ Scribe

The objective of this document is to help someone else run the code that is delivered as part of this project.

Project Title: μ Scribe

Students:

Name	Roll No.	Email
Abhinav Maurya	10305016	ahmaurya@cse.iitb.ac.in
Yogesh Kakde	10305039	yogesh@cse.iitb.ac.in
Lokesh Rajwani	10305066	rajwani@cse.iitb.ac.in

Project Objective

The project aims to program the Firebird V Hexapod to scribe out shapes, letters and numbers. Such an automated scribing robot can be used in many industrial applications.

Hardware Platform

Hexapod is a standalone robot, which is composed of the following:

1. Firebird V ATMEGA2560
2. Six limbs attached to the chassis, with Firebird V mounted and fixed over it.
3. 18 servo motors (3 servo motors per limb).
4. External 7.4V, 1800mAh Lithium Polymer Battery.

Software

1. ICC AVR 7 (to compile the project files and build the Hex code)
2. AVR Studio 4 (to burn the Hex code into Hexapod)

Code Description

Code Files.

Filename	Purpose	Executes on
main.c	Main Program	Robot
hexapod_firebirdv.h	Contains the abstractions of major operations, like forward(), rotate_clockwise(), etc.	Robot
hexapod.h	Consists of code initializing the ports, all basic operations like angle_1A(), etc.	Robot

Deliverables

Filename	Contains
C-code.tar.gz	SourceCode of programs to be burnt on Robot. Contains documentation of the code as well.
Documents.tar.gz	Contains Project related doc files.

Execution Instructions

The given code is developed under ICC AVR7 IDE under the project name “ICC_AVR_HEXAPOD”.

This project is compiled and build (using ICC AVR7) to generate ICC_AVR_HEXAPOD.HEX file.

Now, this ICC_AVR_HEXAPOD.HEX file is burnt into microcontroller using AVR Studio 4 with the help of AVR MK II USB Connector.

Now switch on the hexapod. It should be noted that the hexapod given consists of two switches, one which starts the FirebirdV, hence starts executing the code been burnt on it. And second switch to turn on the servo motors. Hence a sufficient delay at the beginning is given so that one can place the hexapod easily at start.

Coding Guidelines

1. We have used the standard copyright statement in our code declaring the code to be open-source and property of ERTS Lab.
2. We have used the standard header file “hexapod_firbirdv.h” which will contain all the implementations of core actuations and sensing.
3. We have also developed and used another header file “hexapod.h”, containing description of basic motions.
4. We have commented the code appropriately and also given explanations at places as to what the functions are targeted to do.