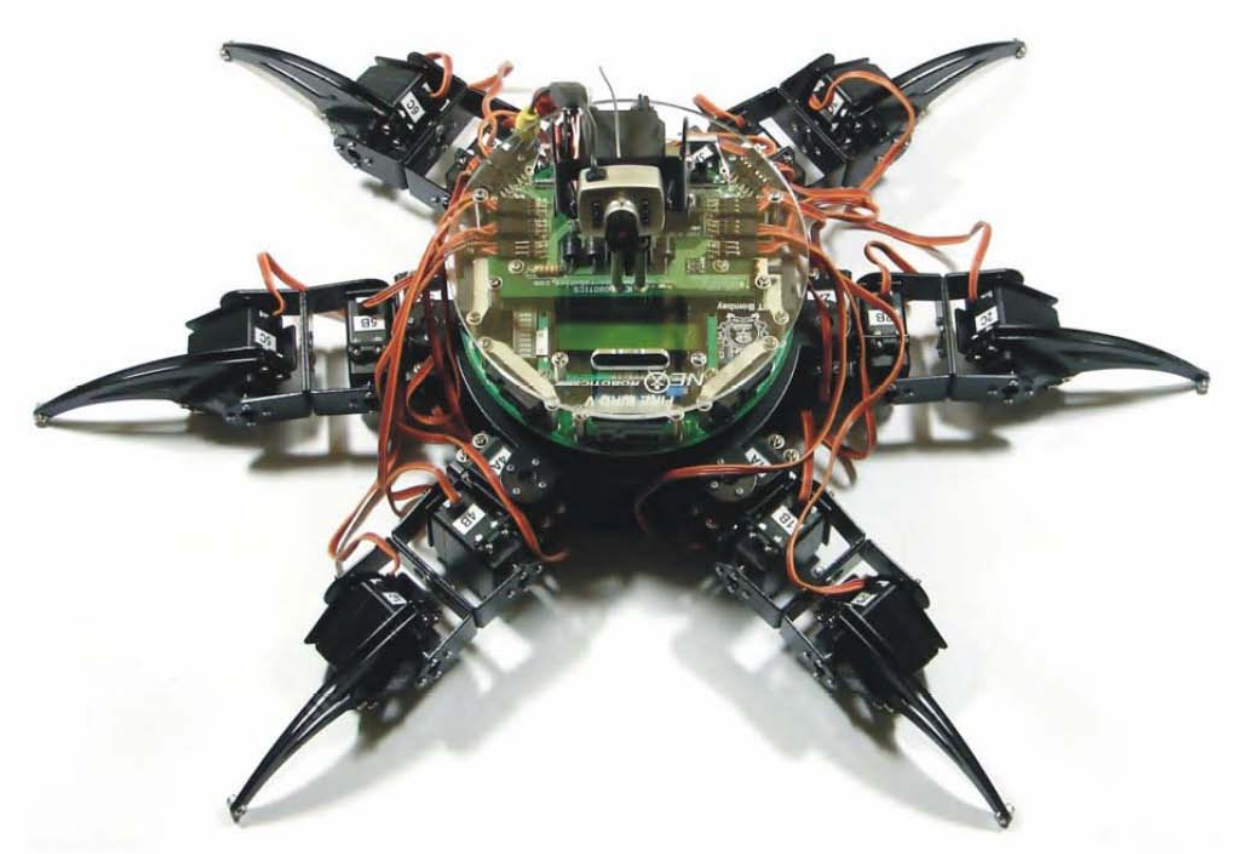
Building a Locomotion based Hexapod Application



Team 6
Srijit Dutt (10305056)
A Udaya Kumar (09305073)
Venkatesh Velaga (09305063)

Under the guidance of Prof. Kavi Arya

1. Introduction

1.1 Problem Statement

The objective of the project is to build locomotion based Hexapod Application. The main task is to control the locomotion of the hexapod, to make it climb the obstacle and to make it dance.

The project demonstrates all three tasks sequentially. It walks some distance and then climbs obstacle and it starts dancing for some time. Apart from this we demonstrate controlling of the hexapod locomotion with key board using zigbee module for wireless communication.

1.2. Requirement Specification

1.2.1. Making the hexapod move smoothly in all directions.
1.2.2. Obstacle climbing by Hexapod
1.2.3. Making Hexapod to dance

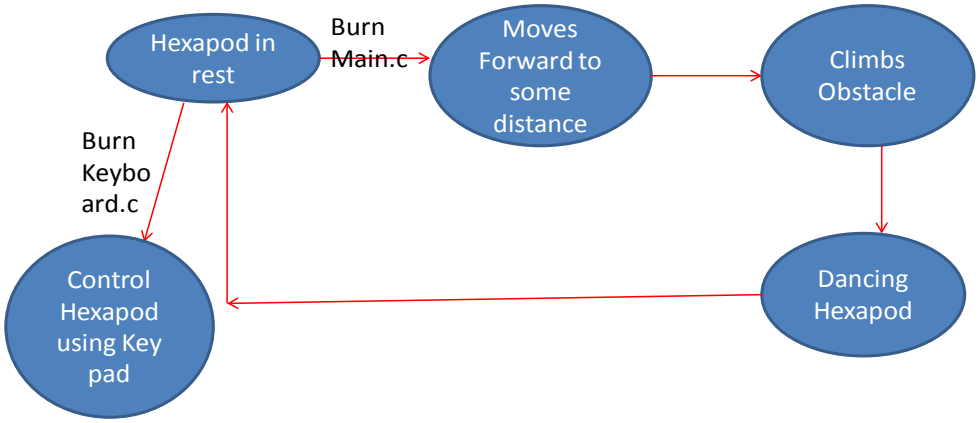
1.3. System Design (Finite State Machine)

The Finite state machine in the next page explains the flow of the work that we have done. The description of the states follows

State 1 represents the Resting state of the hexapod When we burn the code(For making it climb the obstacles[main.c]) we can see that the hexapod starts moving forward that is represented by **state 2** and transition happens to the next state i.e. **state 3** where the hexapod climbs the obstacle and then we transition to next state i.e. **State 4** Where the hexapod dances some time and then returns to state1 that is resting state.

When we burn code(Keyboard.c) when the hexapod is in state 2 hexapod will make transition to **state 5** where we can control the hexapod motion with keyboard.

FSM



Moving forward works as follows (Explained as comments in gait.c file)

1. Keeps the odd tripod down
2. Lifts the even tripod up
3. Moves its body based on odd tripod
4. Keeps the even tripod down
5. Lifts the odd tripod up
6. Moves the body with even tripod

Climbing Obstacle is as follows

1. Increase the bot height
2. Trust the body of the robot forward
   a. Pick up leg 3 and 4
   b. Move the legs 3 and 4 in air
   c. Put legs 3 and 4 down
   d. Move legs 1 and 2
   e. Pickup legs 1 and 2

f. Move legs 1 and 2 in air

3. Bot pushes itself on to the platform
   a. move body on leg 3 & 4
   b. move leg 1 & 2
   c. pick up 3&4
   d. moving leg 3&4 in air
   e. make leg 3 &4 touch ground
   f. move body on leg 3 & 4
   g. drag body forward on legs 1 & 2
   h. pick up 3 & 4
   i. move forward the legs 3 and 4 in air
   j. put 3 & 4 down
   k. push body on 3 & 4
4. Make hexapod sit
5. pick up legs 5 & 6
6. move legs 5 & 6 in air
7. crawl on 1, 2 , 3 ,4
8. put down 1, 2, 3, 4
9. crawl on 1, 2 , 3 ,4
10. put down 5 & 6

Similar steps for dancing is provided in Hexapod_motion.c file with the function named dance() Every instruction used is commented out for readability

Similar steps for controlling the hexapod in all directions is provided in keyboard.c file with the function keyboard_control().

1.4. Assumptions and Limitations

1.4.1. Not using any IR Sensor for detecting the obstacle so initial state of the hexapod and distance of obstacle from the hexapod's initial state is assumed initially.

1.6. External Hardware used

Zigbee module for Wireless Communication

Here we use zigbee module for controlling the hexapod locomotion through keyboard and the function which does this task is keyboard_control() present in the file called keyboard.c.

2. Present Status

2.1.1. Project Timeline

Task	Date
Survey, Project Topic Selection	16th September
Design Basic Hexapod Servo Control Routines	3rd October
Different Hexapod Gaits	10th October
Advanced Hexapod Motions and Zigbee Module	17th October
Obstacle Climbing	24th October
Documentation	7th November

2.1.2. Current Status and requirements completed

Done with all the tasks specified in the above table

2.2 Delays Occurred

Some delay occurred in the initial stages when we faced a problem where all the legs are not behaving alike due to the frictional effect. We with the help of other team working on hexapod solved this problem by getting rubber material attached to all the legs and thus reducing the frictional effect to some extent.

2.3 Critical Steps in our Project

Getting the hexapod to move smoothly in all directions as desired is really a hard task and we spent lot of time in deciding the good algorithm suitable for the same. Here the good algorithm means that, if we want to climb the obstacle we need to use different gait (Dual gait) where we work with two legs at a time as specified in the previous section. And also if we want to move on the level ground we used tripod gait motion where three legs are moved at a time naming odd tripod and even tripod.

2.4 Individual Roles and Contributions

Srijit Dutt (10305056) : Coding.

Venkatesh Velaga (09305063): Participated in selection of algorithms and finalizing them for implementation and prepared documentation and presentation.

Udaya Kumar (09305073): Initial participation for topic selection.

2.5 Time Devoted to project (Man-Days)

Total of 6 man days are dedicated to complete this project.

3. Final Road Map of the Project

Task	Deliverables	Date
Survey, Project Topic Selection		16th September
Design Basic Hexapod Servo Control Routines		3rd October
Different Hexapod Gaits		10th October
Advanced Hexapod Motions and Zigbee Module		17th October
Obstacle Climbing	Code Completion	24th October
Documentation	API + Project Report	7th November

4. Innovation, Creativity and Reusability Index of your Project

4.1. Innovations and Creativity

- 4.1.1 Obstacle Climbing

The height of the obstacle climbed by the hexapod is comparable to the maximum height the robot can lift its body to

- Different gaits for different terrain types

  As explained clearly in previous sections different gait motions are been given for different situations (Moving while climbing, Moving on even surface)

4.2. Reusability

- Every task implemented is modular and independent of other tasks. Here modular means that every gait motions and zigbee control are written as different modules so that can be modified independently.
- Common functionalities are defined as macros in the header files(Hexapod.h, Hexapod_basic.h, keyboard.h)
- Code is made as modular as possible so that one module can be edited without effecting the other and each module is reusable.

5. Suggestions for improving the process

Mare lab workshops have to be conducted on hexapod to get better understanding on the working of hexapod. That helps in devoting time for creative work rather than spending time for understanding the basic terminology.

6. Bug Report

1. There is a bug in the program hexapod_motion.c , in the function walk_on_four() which, sometimes is not behaving as desired. Here we are trying to walk on 4 legs (2, 3, 4, 5) and lifting up front and back legs in order to take small displacements fast. Frictional effect is not letting the small displacements to happen.

2. There is a bug in the file hexapod_basic.c, in the function angle_change() which, sometimes is not delivering the required functionality. Angle_change is not making the same angle change to all the legs for the same argument. This bug occurs due to the manufacturing fault of hexapod.

7. Test Plan

As we have integrated the motion of the hexapod with climbing and dancing, our task is automated. Just on invoking the task we can see the hexapod performing the three tasks sequentially.

On firing other file keyboard.c on to the hexapod we can see the hexapod moving as per the instructions given by the keyboard. Instructions are specified in the file keyboard.c

8. Future Scope

1. Any new applications on hexapod like obstacle avoidance using web cam
2. Making Hexapod dance according to tunes.
3. Better Locomotion algorithms including study of crab and insect gaits.

9. Learning

1. Using Zigbee module for wireless communication.
2. Hexapod Calibrations
3. How to overcome real time problems while integration
4. Got Hands on experience to work on an application by using the knowledge gained from the lectures.
5. Style of Documentation
6. Style of coding