Hexapod Locomotion

Team Members:

09005021 : Alankar Saxena
09005023 : Shivam Agarwal
09005025 : Rahul Varshneya
09005030 : Anvit Singh Tawar

Problem Statement

- Moving the hexapod fast and smoothly in all the directions.
- Turning the hexapod in both clockwise and anticlockwise direction.
- Controlling the locomotion with keyboard using ZigBee module for wireless communication.
- Implementing various gaits which are suited for different terrain and situations.

Task Specification

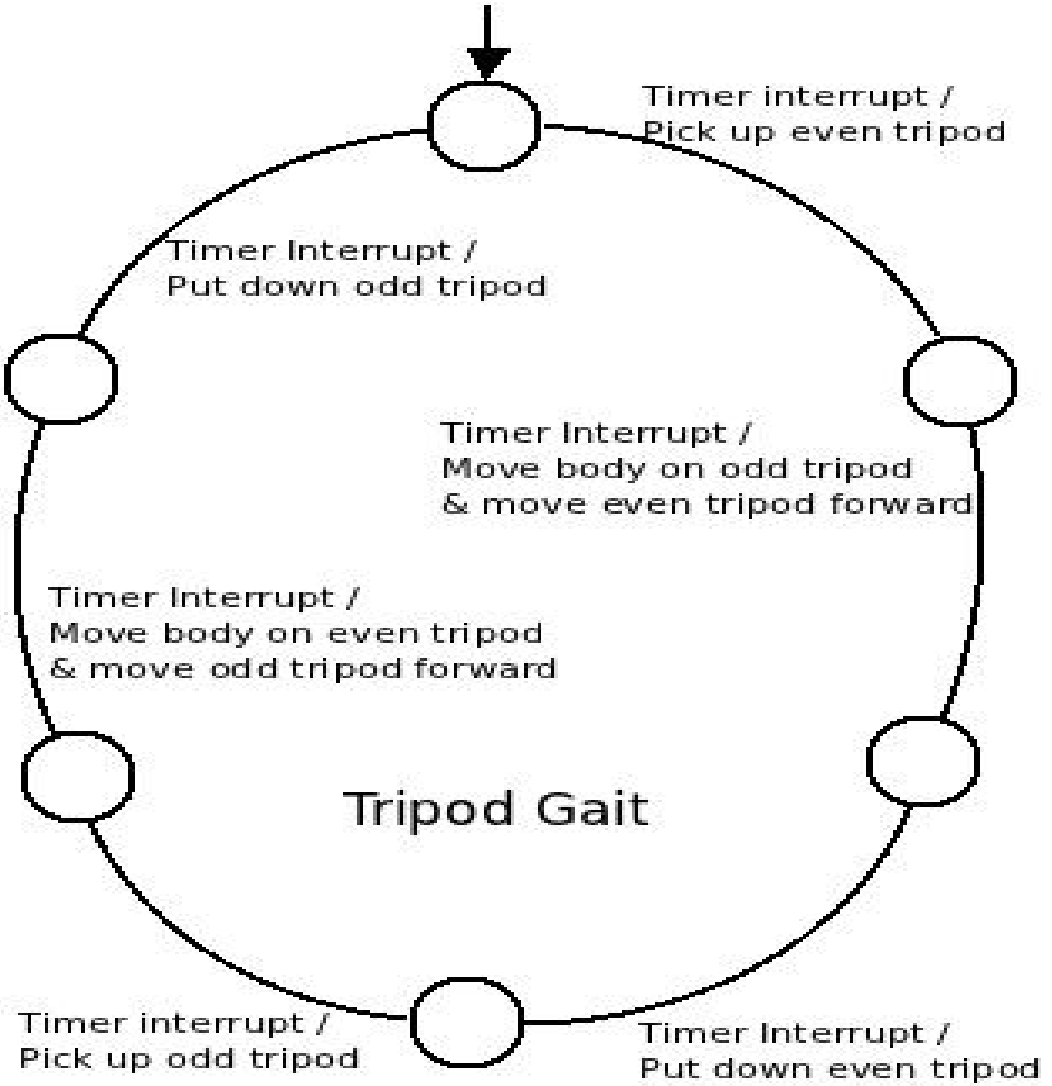
- There are 3 major tasks handled
  - Motion:
    - There are 12 different directions of motion (6 corresponding to its 6 legs and the other 6 for movement in direction between two legs).
    - Gaits implemented for motion: Tripod, Tripod continuous, Tripod insect, Two legged, Wave and Ripple.
  - Turning:
    - Hexapod can turn either clockwise or anticlockwise.
    - Gaits implemented for turning: Tripod turn, Tripod turn continuous and Two legged turn.
  - Zigbee Communication:
    - Motion of the bot can be controlled by a keyboard through ZigBee module for wireless communication.

Project Plan

Person	Work Completed
Alankar Saxena	Turn gaits
Shivam Agarwal	ZigBee, experimentation with RTOS
Rahul Varshneya	Wave, Ripple, Two legged gaits
Anvit Singh Tawar	Tripod gaits and their modifications
Common work	Project selection, Discussion, SRS/Report/Presentation formulation, Understanding previously written code and Documentation

Task	Date
Project selection	17th Feb
Understanding/adding basic hexapod servo control routines	10th - 20th March
*Implementation of various hexapod gaits (motion and turn)	20th - 5th April
*Working and testing in RTOS environment	1st - 7th April
*Testing various gaits and further improvement	5th - 9th April
*User interface through ZigBee communication	10th April
Documentation	15th - 17th April

State chart – Tripod Gait



Innovation and Challenges

- Innovations
  - Speed of hexapod is increased by increasing the synchronization in the servos of hexapod.
  - Stability has also been taken into account by developing innovative gaits (wave, ripple, etc).
- Challenges
  - RTOS: Difficulty in defining interrupt service routines and handling inbuilt timer interrupts within the code. So the code didn't work.
  - Synchronization of servos: Coordinating the motion of 18 servos is a challenging task.
  - Trade-off between speed and stability.
  - Low battery life and long charging time.
  - Hardware challenges: Different legs move by different angles when given the same movement command.

Tasks Completed

(Problems faced with solutions)

- Motion – Fast and stable
  - Developing coordination between 18 servos: This was handled by implementing various gaits for movement of hexapod.
  - Slow movement of hexapod: By varying the time delay and angular extremities of servos, we were able to achieve greater speed.
  - Instability in hexapod movement: This was handled by using specific gaits like ripple and wave.
- Rotation – Fast and smooth
  - Slow rotation speed: We used variations of tripod gait to increase the rotational speed.
  - Collision of neighboring legs: We carefully studied the motion of servos and decided on the angular motion to be imparted.

Continued ...

- Zigbee – User friendly interface
  - Incapability of calling movement modules from ISR for Zigbee communication: Since all movement modules work using timer interrupts, calling them from the ISR for Zigbee communication stops their execution. To handle this, we defined different modes corresponding to different states of hexapod. These modes could be changed by the user through keyboard and communicated via Zigbee module.
- RTOS
  - Non working ISRs in RTOS: Due to unavailability of resources, we were unable to define ISRs in RTOS which made the code non-functional.

Testing

- Test criteria
  - Speed (quantitative)
  - Stability (qualitative)
  - Smooth motion (qualitative)
- Test description
  - Measured time taken to travel a fixed distance by the bot for various gaits.
  - Measured the angular distance traveled for measuring rotational speed.
  - Other criteria were qualitatively judged from tests done.

Performance metrics

Gait	Speed	Stability	Smoothness
Tripod (simple)	6.22 cm/s	Moderate	High
Tripod continuous	10.02 cm/s	Low	Moderate
Tripod insect	8.54 cm/s	Low	Moderate
Two legged	3.96 cm/s	Moderate	Moderate
Wave	1.33 cm/s	High	Low
Ripple	1.88 cm/s	High	Low

Gait	Speed	Stability	Smoothness
Tripod (simple)	1 rotation in 8.6 s	High	High
Tripod continuous	1 rotation in 5.2 s	Moderate	Moderate
Two legged	1 rotation in 9.5 s	High	Moderate

Stability increases with the number of legs supporting the bot during its motion
Smoothness increases with simultaneous movements of servos in various legs.

Test Results

- In motion we inferred the following
  - Tripod continuous gait is fastest but suffers from low stability because of the large step size.
  - Both ripple and wave gaits are the most stable gaits inspite of being slow. They use one leg at a time.
  - Other gaits lie in the intermediate range.
- In rotation we inferred the following
  - Tripod continuous gait is fastest due to large angular motion of the motors.
  - Two legged gait is slowest due to greater cycle length.

Re-usability

- Every task implemented is modular and independent of other tasks
- Common functionalities are defined as macros in the header files
- Code is made as modular as possible so that one module can be edited without effecting the other and each module is reusable.
- Code for all the gaits is well parametrized which can be easily edited to suit the user requirement.
- Code has been implemented in IAR embedded workbench. For executing the code in a different environment (which support C), minimal changes have to be done (specifically for ISRs which are environment dependent).
- Besides, all the functions coded have been built upon lower order functions which, in turn, depend on the hardware. On changing the hardware only those lower order functions need to be redefined.

Future Enhancements

- Possible extensions
  - Implementation in RTOS to aid multi-tasking
  - Obstacle avoidance
  - White line follower etc.
- There are a lot of applications of hexapod which involve its speedy and sturdy motion. Using our modules, user can select the type of motion he wants. For example
  - All terrain bot.
  - Military hexapods for unmanned missions.