

Automated Learning Robot



GROUP 20

Pradyumna Kumar

06D05015

Jayanth Tadinada

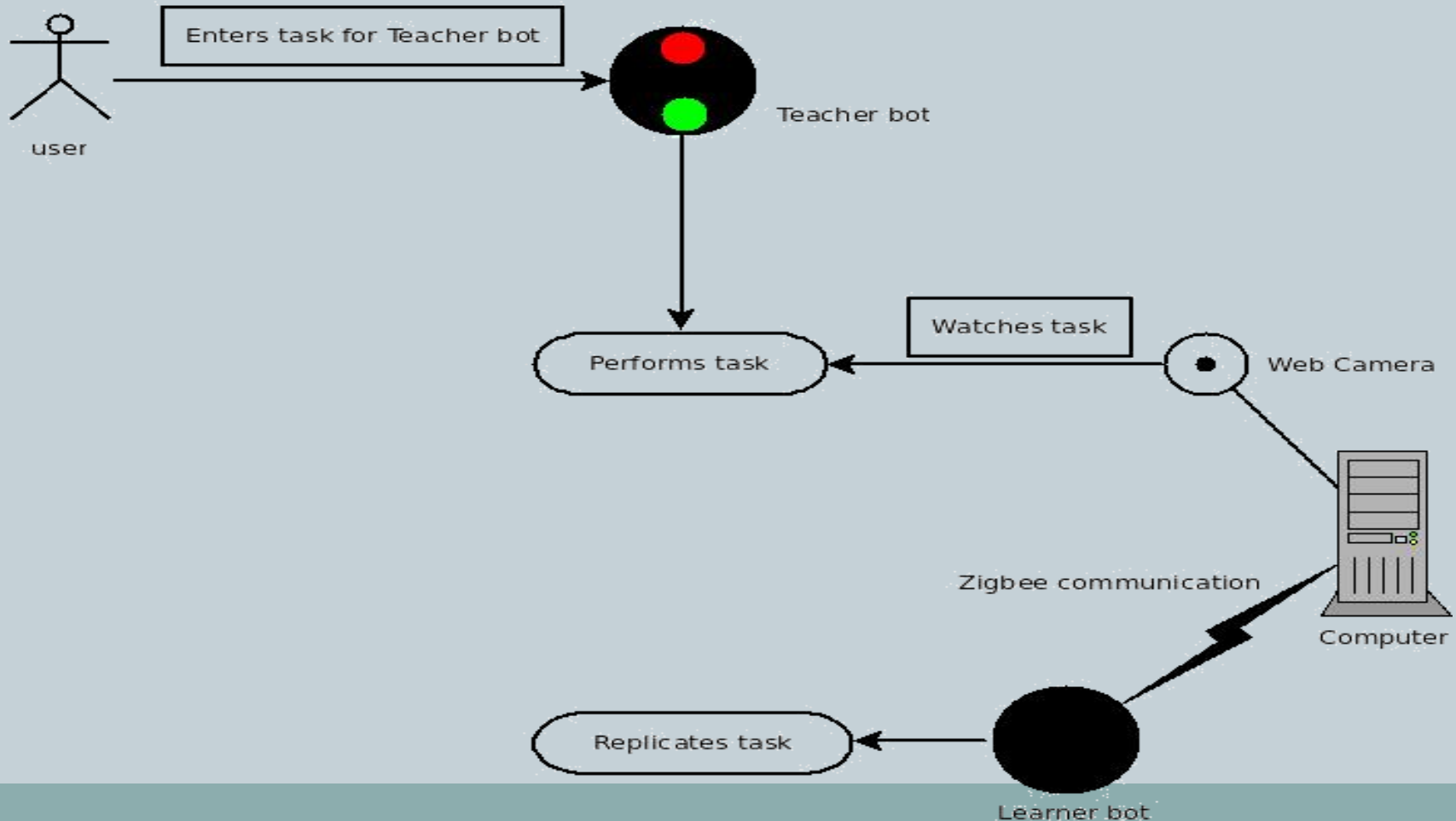
06D05016

Sharjeel Imam

06D05010

Problem Statement

Design and implementation of a self-learning robot



Task Specifications



Six major functional components

1. Manual bot control
2. Runtime capture of task
3. Task detection algorithm
4. Task encoding
5. Wireless transfer
6. Automated bot control

Task Specifications



Six major functional components

1. Manual bot control
 - Firebird V movement using Zigbee

Task Specifications



Six major functional components

1. Manual bot control
2. Runtime capture of task
 - Overhead camera connected to a computer through USB
 - Bot is covered with a black disc with distinct colour markings to denote front and back sides of the bot

Task Specifications



Six major functional components

1. Manual bot control
2. Runtime capture of task
3. Task detection algorithm
 - Bot detection and its movement measurement using Matlab
 - Convert distances and angles from camera frame to actual arena frame

Task Specifications



Six major functional components

1. Manual bot control
2. Runtime capture of task
3. Task detection algorithm
4. Task encoding
 - Specific encoding for each task, as described below
 - *20F30R25B90L* means: 20cm forward, 30° right turn, 25cm back and 90° left turn
 - All distances in actual arena frame

Task Specifications



Six major functional components

1. Manual bot control
2. Runtime capture of task
3. Task detection algorithm
4. Task encoding
5. Wireless transfer
 - Computer transfers encoded task to automatic bot using Zigbee

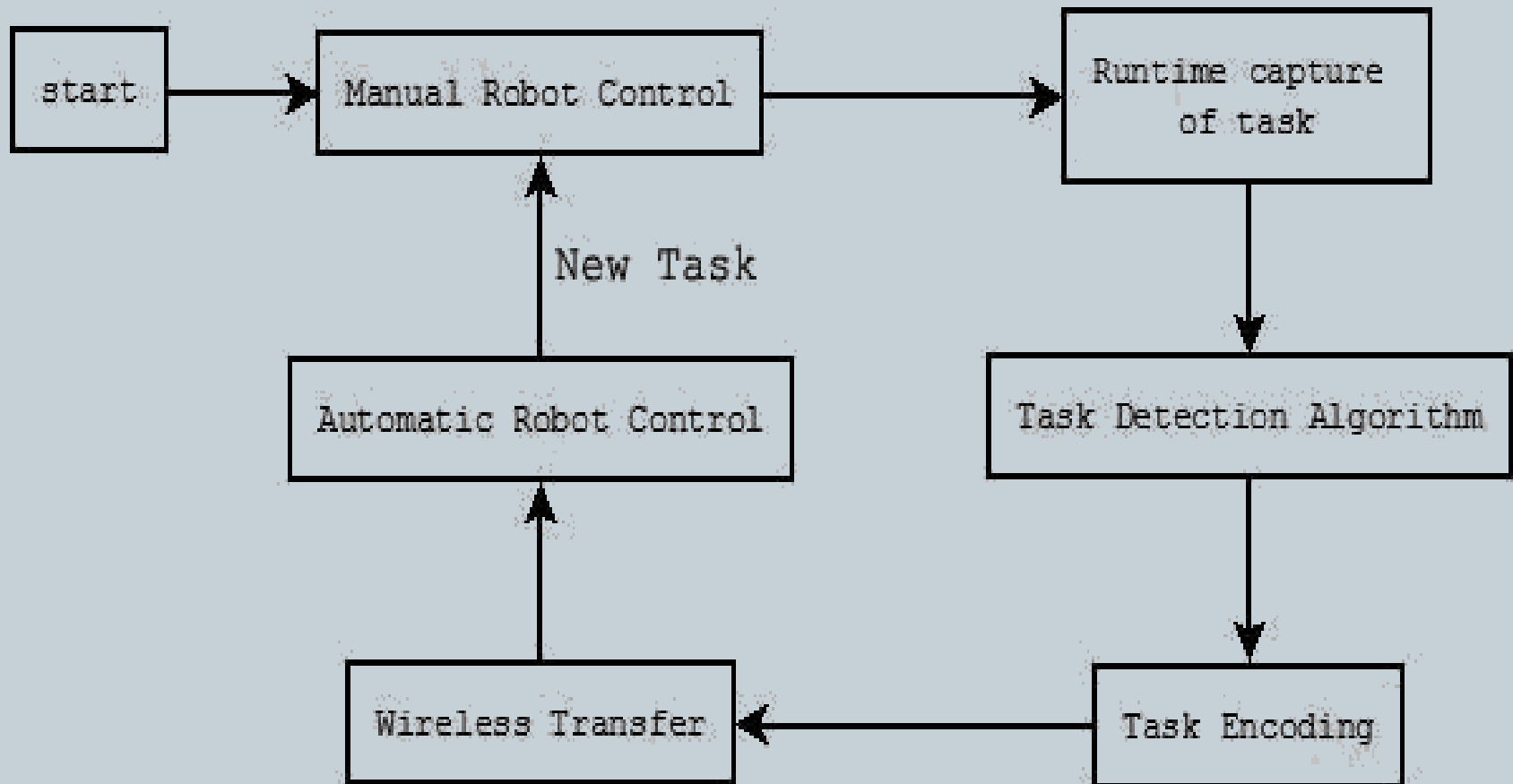
Task Specifications



Six major functional components

1. Manual bot control
2. Runtime capture of task
3. Task detection algorithm
4. Task encoding
5. Wireless transfer
6. Automated bot control
 - Decode the received task and perform it

Block Diagram



Project Plan



work Division



Pradyumna



Sharjeel



Jayanth

Task	27 th Sep	4 th Oct	11 th oct	18 th Oct	25 th Oct	1 st Nov	8 th Nov
Requirement Analysis							
Project Scope Analysis							
SRS							
Run time capture of task							
Task Detection Algorithm							
Task Endocing							
Task Parsing							
Task Decoding							
Manual Bot Control							
Finetuning task detection algorithm							
Wireless Transfer							
Integration							
Testing							
Documentation and Report Writing							

Problems & Solutions



1. Light Conditions

- Glare due to direct light from tube lights (black arena didn't help)
- Orange and green colours looked white
- **Solution:** Turn off tube lights. Generate diffused light using a table lamp facing the roof

Problems & Solutions



1. Light Conditions

2. Robot Movement Control

- Wheel shaft interrupts are getting disabled after Zigbee reception
- **Solution:** Re-enable the interrupts after every reception

Problems & Solutions



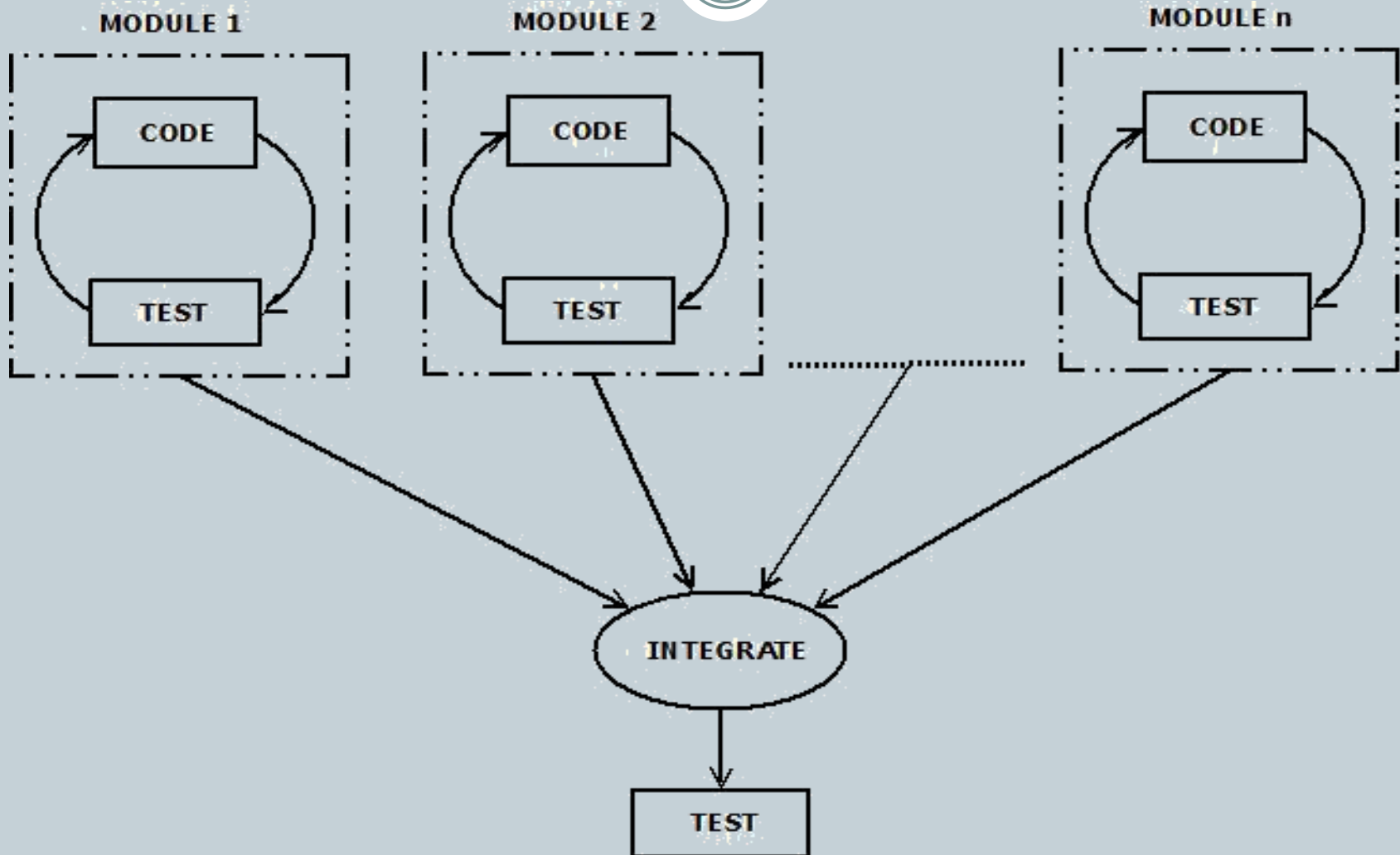
1. Light Conditions
2. Robot Movement Control
3. Path detection
 - Intermediate error points due to random objects in the arena
 - **Solution:** The angle-gap between successive points cannot be greater than 70 degrees

Problems & Solutions



1. Light Conditions
2. Robot Movement Control
3. Path detection
4. Wireless Transfer
 - Matlab sends an extra line-feed termination character after every serial transmission
 - Characters get repeated at the receiver
 - **Solution:** Ignore the extra character and include a delay after transmitting every character to prevent repetitions

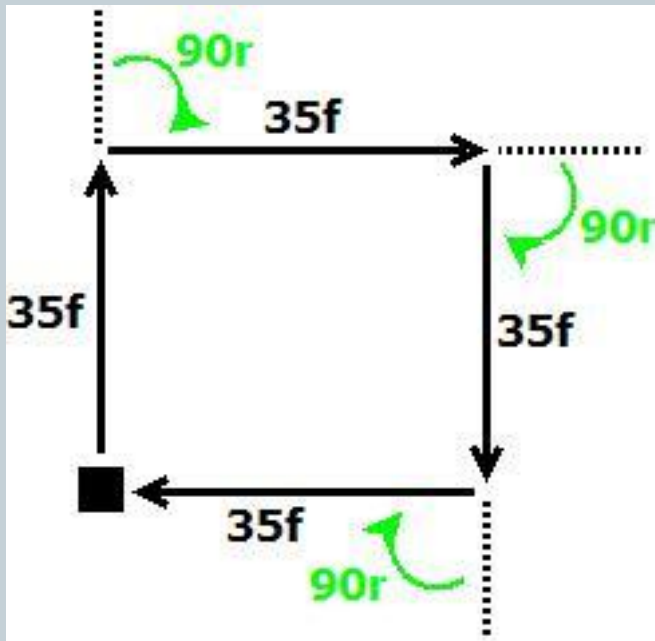
Testing



Test Cases



1. Square with right turns



Encoded Path: 38f 90r 40f 95r 32f 95r 41f

Test Cases



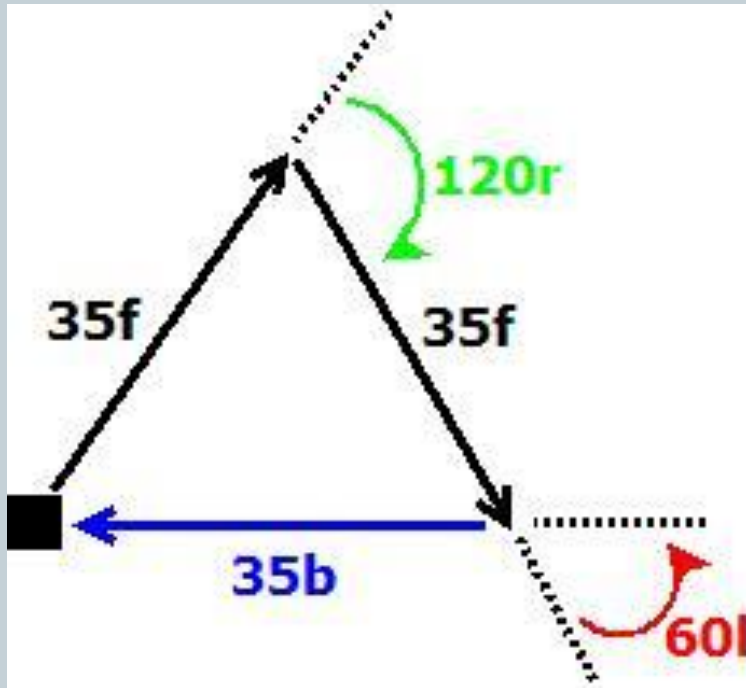
2. Motion in 'Z'



Encoded Path: 32f 125r 45f 113l 40f

Test Cases

3. Equilateral Triangle



Encoded Path: 40f 127r 41f 66l 40b

Performance Metrics



Real time performance

Path is generated within 2 seconds of the teacher robot completing the task

Error

Error in distance: 20 – 25 %

Error in angle: 10 – 15 %

Larger the distance or angle, lower the error

Reusability



1. Separate functions/files for independent code modules
2. Intuitive function and variable naming
3. Fully documented code (both C and Matlab) using Doxygen
4. Inline comments in the code to explain the algorithm

Future Enhancements



1. Correcting the learner bot while it is performing the task
2. Better noise reduction for robust performance
3. Transmitting the command just after the teacher bot has finished performing a segment of the task
4. Detecting and encoding paths with arcs (like circles)
5. Extend this for tasks which cannot be captured by camera (like sensor readings) using coloured LEDs

Thank You

