

Robotics & AI Program

PROJECT EXECUTION GUIDE – LEVEL 3

Project 1: Building Height Measurement using IR Sensors and Time-of-Fall Method

Theory:

- This project uses the second equation of motion: $S = ut + (1/2)gt^2$
- Here, $u = 0$ (free fall), $g = 9.8 \text{ m/s}^2$, t = time measured using Arduino's `millis()` function
- Two IR sensors detect the start (top) and end (bottom) of the fall
- Arduino calculates and prints the height using the measured time interval

Execution Steps:

- Mount two IR sensors inside a vertical pipe, aligned properly
- Top sensor OUT → Digital Pin X, Bottom sensor OUT → Digital Pin Y
- Connect both sensors to 5V and GND
- Use `millis()` to track time from top to bottom trigger
- Calculate height: $S = 0.5 * g * t * t$, display it via Serial Monitor

Common Errors:

- Sensors misaligned → missed detection
- Ball not suitable (size, reflection) → inconsistent results
- Lack of debounce → multiple triggers
- Improper reset of `millis()` → inaccurate readings
- Sensor wiring issues (D0/A0 confusion)

Project 2: Human Height Measurement using Ultrasonic Sensor

Theory:

- Ultrasonic sensor (HC-SR04) measures distance from head to sensor
- Subtracting this from known height gives person's height
- Sensor sends pulses and receives echo to calculate time and distance

Execution Steps:

- Mount sensor above standing area (on a frame)
- Connect TRIG → Pin 9, ECHO → Pin 10, VCC → 5V, GND → GND

- Upload provided code and observe the distance
- Set constant for sensor height in code (e.g. 180 cm)
- Display result as Person Height = Known Height - Measured Distance

Common Errors:

- Wrong TRIG/ECHO pins → no signal
 - Sensor not facing straight → wrong readings
 - Obstructions or uneven head surface → affects accuracy
 - Delay missing → noisy values
 - Height constant not set properly → wrong height
-

Project 3: TSOP Relay Control System

Theory:

- TSOP1738 receives IR signals from a remote
- Specific IR code triggers relay ON/OFF via Arduino
- Used to control appliances like LED, fan etc. wirelessly

Execution Steps:

- Connect TSOP OUT → Pin 11, Relay IN → Pin 6, VCC & GND
- Upload provided code (use Serial Monitor to detect IR codes if needed)
- Match IR code in sketch with remote button
- Toggle relay ON/OFF with remote input
- Test circuit with LED or small appliance

Common Errors:

- TSOP not aligned with remote → no detection
 - Wrong IR code → no response
 - Relay draws too much power → avoid 3.3V pin
 - Code debounce missing → double toggle
 - Wrong jumper setting on relay board
-

Project 4: Intelligent Car Parking Management System

Theory:

- IR sensors at entry and exit track vehicle count
- Count increases at entry and decreases at exit
- Arduino displays available parking slots and triggers alerts

Execution Steps:

- Connect Entry Sensor → Pin 3, Exit Sensor → Pin 4
- Outputs: Green LED → Pin 7, Red LED → Pin 8, Buzzer → Pin 6
- Upload code and simulate car movements by blocking sensors
- Observe count changes and overflow alert (when full)
- Optional: Show count on Serial Monitor

Common Errors:

- Sensors not properly triggered → count mismatch
 - Too fast hand movement → missed reading
 - Count goes negative → fix with logic condition
 - Two objects passing together → debounce needed
 - Power issues with multiple components
-

Project 5: Temperature Measurement using Thermistor

Theory:

- NTC thermistor decreases resistance with rise in temperature
- Steinhart-Hart equation is used to convert resistance to temperature
- Voltage divider circuit used to measure thermistor voltage drop

Execution Steps:

- Build voltage divider: Thermistor → 5V and 10k → GND, mid-point to A0
- Upload code using Steinhart-Hart constants
- Touch or heat thermistor to see temperature rise
- Observe values on Serial Monitor
- Optional: Display temperature on screen

Common Errors:

- Thermistor and resistor swapped → wrong values
- Wrong resistance constant in code
- Floating analog input → connect GND properly
- Wrong Steinhart constants for your thermistor
- Pin A0 conflict with other sensors