CECS 347   Project 2: A Following Robot

Preparation: You will need a LaunchPad, a robot car with the following components: 2 wheels and 2 DC motors, one motor driver with power supply circuit, a battery case with 6 to 8 batteries, and 3 sharp IR distance sensors.

Project Goals: understand basic microcontroller hardware components, including GPIO, interrupt, analog to digital conversion (ADC), and hardware PWM and be able to build an embedded control system that makes use of those components to interface with switches, LEDs, DC motors and sensors.

Starter Project: Project 2 Part1 or Romi_Car_Test.

Starter main: FollowingRobot.c

Reference Projects:

Romi_Car_Test: an example for DC motors.

ADCOneSample: an example for one sample using ADC0 Sample Sequencer 3.

WallFollower: an example for three samples using ADC0 Sample Sequencer 2.

FollowMe.c: main code for following object with one sensor connected to ADC Sample Sequencer 3 moving forward/backward/stop.

Project Description:

You will build a robot car that works on the following two modes:

1. Mode 1 - Object follower
2. Mode 2 - Wall follower

The robot car will start in an inactive state. Onboard switch 1 will be used to toggle activate/deactivate the robot. Once the robot is activated, it will default to be in mode 1.

The onboard switch 2 is used to toggle between the two modes when the robot car is active. When the robot car is inactive, press and release switch 2 has no effect to the car.

As soon as one mode is activated, the robot's sensors will do ten consecutive readings for obstacles around it and then start to perform functions defined for the current mode.

Mode 1 – Object Follower:
The robot car will follow an object in front of it in a fixed distance 20cm: when the object moves forward, the robot car will follow the object moving forward at a distance of 20cm; when the object moves backward, the robot car will follow the object moving backward at a distance of 20cm; when the object turn left, the robot car will follow the object turn left at a distance of 20cm; when the object turn right, the robot car will follow the object turn right at a distance of 20cm; when the object stops moving, the robot car will stop moving; when the object is out of range, the robot car stops moving. To stop the car: put an object <10cm in front of the car. Once the car is stopped, it will be in an inactive state.

Development steps:

1. Start with your part 1 project. Rename your project to Following_Robot. Add the following module to the project: ADC0SS3.c/h. You can find all this modules in project ADCOneSample.

2. Use Follow_Me.c as your main test file to test forward/backward following with one sensor.
3. Remove Follow_Me.c and add Following_Robot.c as main program. Replace ADC0SS3 module with ADC1SS1 module to take care of three sensor outputs. You can find an example on how to sample three analog signals in reference project WallFollower: ADCSWTrigger.c. Add two more sensors to your robot car and implement function object_follower() in Following_Robot.c. You can set mode = OBJECT_FOLLOWER for part 2 development and use function follow_me() in Follow_me.c as your start code for function object_follower().

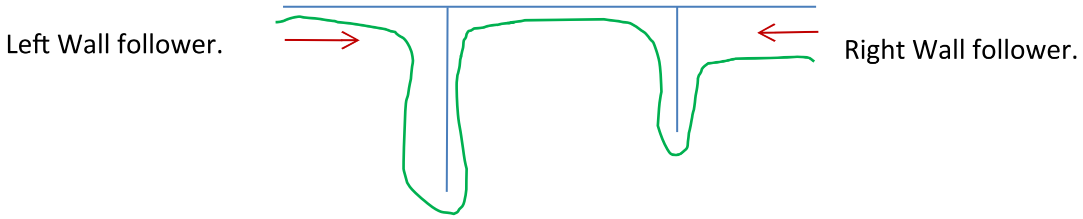
Mode 2 - Wall Follower:

The robot car will start by detecting a wall close to one side and forward the wall moving forward. The robot car should be able to do both right and left wall following depending on which wall is closer to it. The robot car is not allowed to touch any obstacles including a wall. **Optional feature**: To stop the car: put an object <10cm in front of the car. Once the car is stopped, it will be in an inactive state.

The onboard LEDs are used to indicate the status of the robot. Color descriptions are given in the following table:

Color	**Indication**
RED	Robot Car is inactive.
BLUE	Mode 1: Object Follower.
GREEN	Mode 2: Wall Follower.

Example Walls:



Hardware Specifications:

1. PLL is required to set the system clock to be 16MHz using external clock source.
2. Hardware PWM is required to drive the DC motors.
3. ADC Module 1 sample sequencer 1 is required for analog to digital conversion.
4. Three IR sensors are required for the robot car to detect obstacles around it.
5. GPIO pin assignments:

GPIO Pins	**Devices**
PE1	Left IR sensor output
PE0	Front IR sensor output
PE2	Right IR sensor output
PB4	Hardware PWM output for left motor
PB5	Hardware PWM output for right motor
PB32	Direction pins for left motor
PB76	Direction pins for right motor
PF123	Onboard color LEDs
PF0	Onboard switch 2
PF4	Onboard switch 1

Extra challenges:

1. Mode 2 (3 points): top three teams whose explorer time is the shortest: 1st: 3 points, 2nd: 2 points, 3rd: 1 points.
2. Hardware design (2 points): a hardware design that provide a stable system.

*** Please note when downloading code to your board or debug your code on board, you need to disconnect power supply to Launchpad vbus pin.**

Deliverable Requirements:

1. Project report: including all required sections listed in the report template. Special requirements for following sections:
   Software Design: include a flowchart is required to show your software solution.
   Hardware Design: include schematic and a picture of your robot car.
   Operation: Upload a video or a YouTube link for your demonstration. If it is a video link, please put it in "Operation" section of your project report.
2. Submit source code in .c/.h format.