ECE 540 Project 2 List of Files
(Last updated 17-Oct-2016)

Documentation files	
Name	**Description**
docs\BotSim Functional Spec.pdf	Functional specification of the BotSim external interfaces
docs\BotSim Theory of Ops.pdf	Internal theory of operation for the BotSim Simulator. You do not need to understand this material for Project 2 but you may find it interesting.
docs\project2.pdf	The Project write-up
docs\Proj2Demo Example Design Description.pdf	Theory of operation for the demo example. Includes description of the user interface.
docs\RojoBot World Video Controller.pdf	Theory of operation and task list for adding the video controller to your Rojobot system. Your demo will be based on this system coupled with the map that includes left and right turns.
docs\Proj2 Bot tracker.pdf	File showing the simple right-turn-only track for the Rojobot. You may use this file to check that your Rojobot system is running correctly before you have the video controller.
docs\ECE 540 Project 2 List of Files.pdf	This document

| Verilog and constraints files for Part 1 (demo and no video) ||
Name	Description
debounce.v	Not included in the release. Use the same modules as Project 1. Debounces pushbuttons and switches.
sevensegment.v	Not included in the release. Use the same modules as Project 1. Seven segment display interface.
kcpsm6.v	Not included with the release. You should download the latest Picoblaze from the Xilinx web site. Xilinx PicoBlaze for Artix Series 7 FPGA on the Nexys4 board.
nexys4fpga.v	Not included with the release – you need to create it. You may find the nexys4fpga.v file from Project 1 to be a good starting point.
nexys4_bot_if.v	Not included with the release- you need to create it You may find the file *kcpsm6_design_template.v* which is part of the Picoblaze release helpful in creating this. You can get the I/O port address map from some of the constants in proj2demo.psm included with the release. I/O interface between the application CPU and the Nexys 4 board periperhals and the Botsim.
hdl_part1\proj2demo.v	Program file for the application Picroblaze CPU. Produced by the kcpsm6.exe assembler from proj2demo.psm
hdl_part1\bot.v	BotSim top level module. Instantiates a PicoBlaze and the Rojobot simulator program, world_if.v, and map.v
hdl_part1\bot_pgm.v	Program file for the BotSim simulator. Produced by the kcpsm6.exe assembler.
hdl_part1\map.v	
hdl_part1\ world_if.v	The register-based interface to the BotSim Simulator .
hdl_part1\world_map.v	Instantiates a 16Kx2 bit dual-port ROM, produced by Xilinx *Core Generator*, which holds a map of the RojoBot's virtual world.
constraints\n4DDRfpga_novideo.xdc	Vivado Constraints file for proj2demo. There are no VGA signals included.

Verilog and constraints files for Part 2 (with video)	
Name	**Description**
nexys4fpga.v	Not included with the release. Top level Verilog file for the demo. You will have to modify your nexys4fpga.v from the Part 1 of the project to include the VGA signals
nexys4_bot_if.v	Not included with the release. I/O interface between the application CPU and the Nexys4 board periperhals and the Botsim. You may have to modify you nexys4_bot_if.v from the Part 1 of the project if you add additional registers to the interface.
hdl_part2\dtg.v	Generates the video raster timing signals vert_sync, horiz_sync, video_on, and pixel_row and pixel_column, which indicate the current vertical and horizontal pixel position on the screen.
hdl_part2\bot_pgm.v	Program file for the BotSim simulator. Produced by the kcpsm6.exe assembler. The Rojobot moves more quickly through the output track if you include this file in your project instead of the bot_pgm_v from Part 1
constraints\n4DDRfpga_withvideo.xdc	Vivado Constraints file for your demo. This file includes the VGA signals.

World Maps	
Name	**Description**
world_maps/world_map_part1/world_map.ngc	This is a simple world map that includes only right turns. You can use this file to check and/or debug your Rojobot implementation. Copy world_map.ngc to your synthesis directory for the project. The world_map_part1/map directory contains a file called world_map_part1.doc which shows the layout of the track.
world_maps/world_map_lr/world_map.ngc	This is the world map you should use for your demo. It contains both left and right turns. Copy and overwrite world_map.ngc in your synthesis directory for the project. The world_map_lr/map directory contains a file called worldmap_lr.doc which shows the layout of the track.
world_maps/world_map_loop/world_map.ngc	This is a fun map that contains loops but only right turns. You can use it to debug your video logic before you add the video controller to the project. Copy and overwrite world_map.ngc in your synthesis directory for the project. The world_map_loop/map directory contains a file called worldmap_loop.doc which shows the layout of the track
world_maps/world_map_part1/map, world_maps/world_map_loop/map, world_maps/world_map_lr/map	Each of the world map directories contains a directory called map. The map directory contains the text used to generate a track, a .coe (Xilinx coefficients file) that the Core Generator uses to initialize the Block RAM and a perl script that can be used to convert the .txt file to a .coe file. Perhaps the most useful file in the directory is a .doc file which shows the layout of the virtual world.
Firmware for the BotSim	
Name	**Description**
firmware_part1\proj2demo\proj2demo.psm	PicoBlaze Assembly language source code for the Proj2Demo program
firmware_part1\proj2demo\proj2demo.v	Copy of the file hdl_part1\proj2demo.v
Firmware_part1\proj2demo\ROM_form.v	Template used by the Picoblaze assembler.