Liz Crittenden

Final Project User Manual

*Setting Up The Board*

Parts Needed (Besides the HCS12 Microcontroller):

* Ultrasonic Range Sensor HC-SR04
* Tower Pro Servo Motor SG90
* Freenove DC Motor
* External LED
* One 1k resistor
* 7 male-to-female jumper wires
* Two male-t-male jumper wires
* Additional USB cable and wall piece
* Small flathead screwdriver

Installing the External Components (Before You Connect the Board to Power):

1. Tower Pro Servo Motor:

The Servo SG90 has 3 built-in wires: brown, red and orange. Connect these wires to the J7 PP5 pins at the top center-right of the board (diagram 1.) You will be using the lower 3 pins in order yellow-red-brown from left to right.

2. External Range Sensor:

The HC-SR04 will need 4 male-to-female jumper wires. Connect the female ends to the sensor. The pins on the sensor are labeled Vcc, Trig, Echo, and Gnd. We will be using the PORTT connections. These are directly to the left of the breadboard, third from the bottom. The VCC wire will go to the +5V connection. The trig wire will go to the PT4 connection. The echo wire will go to the PT2 connection. The Gnd wire will go to the GND connection. (diagram 2)

3. DC Motor:

The DC Motor will need 2 male-to-female jumper wires. Connect the female ends to the motor. The male ends will use the blue screw down terminals labeled M1 and M2 on the bottom left side of the board. Use your screwdriver to open the screws, and connect the wires to the front side of the terminal (where the lettering is.) Close them securely with the screwdriver. (diagram 3) Make sure the red jumper on J25 (just above and left of the screw down terminals) is connected to VCC. (diagram 4)

4. External LED:

We will be using the built in breadboard to wire the external LED. Use the remaining male-to-female jumper wire on the ground connection on the bottom left corner of the board. The male end will go in the second column of the breadboard, labeled “-” next to the blue line. In that same second column, connect one end of a male-to-male wire. Connect the other end to any column, same row, to the right side of this blue line. Also in the same row, connect one end of the 1k resistor. Connect the other end of the resistor to a different row. Connect one side of the LED to the same row as this end of the resistor. The other side of the LED should be in the row below. Lastly, connect a jumper wire in the same row as this side of the LED, and connect the other end of the wire to PM0 in J90. J90 is located just to the left of the breadboard at the top. (diagram 5)

Now we are ready to connect the board to the computer. The USB cord already connected to the upper port on the left side of the board goes to the computer. The extra USB cable will connect to the lower port on the left side, and plug into the outlet. This serves as the additional power source needed to power the motors (diagram 6). Ignore the beeping from the board while only a single cable is connected.

*Setting Up The Computer*

With both USB cables now connected, power on the computer and open CodeWarrior IDE. Select Load Example Project → HCS12X → LBD\_Dragon12\_Plus. Choose a project name and location and select Create Project.

When the IDE opens, on the left side click the + sign next to “Sources” and double click the file main.c You will see an example program. Delete this and paste in the program for Blackjack. Above Sources, there will be a drop-down arrow next to “Full Chip Simulation”. Click the arrow and change this to HCS12 Serial Monitor.

The program is ready to use! There are two switches on the right side of the board labeled Run/Load and SW7 (middle area). Make sure the left switch is in the Load position. To run the program, there is a green arrow to the right of the words HCS12 Serial Monitor in the CodeWarrior IDE. Click this, and a window will pop up with another green arrow under the word Component. This is the run button. When you click this for the first time, the program will ask you which COM port of your computer the board is connected to, and to choose the model (MC9S12DG256B). Now Blackjack should be running on your board!

*Running the Program*

Once the game has started, you will see a welcome message on the LCD. The bet to play is $50. While the cards are being dealt, this will be indicated by the LEDs coming on one at a time. When dealing is finished, the player’s hand and the dealer’s hand are displayed on the LCD. Now the game will wait for the player to make a choice of hit or stay.

To hit, press the SW5 button one time. The cards will be dealt again, and if the player’s hand remains below 21, the new total will show on the LCD. This process will continue until you choose to stay. When you’re ready to end the game, hold your hand face down over the right side of the board. Now the dealer’s second card will be revealed, your cards will show again, and you can see who won.

While gameplay is going on, be careful not to move too close to the dealer (the distance sensor)! If you do, there will be a warning displayed on the LCD and the game won’t continue until you back away.

If you won: Congratulations! You beat the dealer. You’ll know you won because the external LED will flash on and off 3 times, and the DC motor will run at full speed. There will also be a message saying “You won!” on the LCD.

If you lost: Too bad. You’ll know you lost because sad music will play from the speaker, and the servo motor will move once in each direction. There will also be a message saying “You lost!” on the LCD.

After the game is over, you may see the amount of money won and lost by pressing 1 on the hex keypad. If you would like to see the score, exit the True Time Simulator window that popped up when you clicked the run button *before* pressing 1. Navigate to the MGTEK miniIDE and open it. If this is the first time playing the game, you will need to select Terminal → Options and choose the correct COM port for the board. Each time you open the miniIDE, you will need to select Terminal → Connect. Now press 1 on the hex keypad, and the scores will print to the terminal.

If you don’t want to see the scores, press 2 on the keypad and a simple “Goodbye” will show on the LCD, so that you know the program has ended. If you choose any other key, the program will wait for you to make a valid choice.

*Outputs*

* LEDS- The LEDs light one at a time to indicate the cards being dealt (as well as the 7 segment displays)
* LCD- The LCD is used to write multiple messages to the player including a welcome message, dealing, hit or stay, too close to dealer, you won, and you lost.
* Timer/Speaker – the timer channel channel 5 outputs sad music through the on-board speaker upon losing
* DC Motor- the DC motor spins at full speed for 3 seconds upon winning
* Servo Motor- the servo motor turns once fully in each direction upon losing
* External LED- the external LED wired to PORTM bit 0 flashes on and off 3 times upon winning
* MiniIDE- if chosen, the player may print the scores to the MiniIDE

*Inputs*

* SW5 – While gameplay is active, push-button 5 is used to indicate being dealt another card
* Hex Keypad- Once the game has ended and win or lose has been displayed, the program waits for either 1 or 2 to be pressed as a choice to view the scores on the MiniIDE.
* Photosensor- While gameplay is active, the user may hold their hand over the on-board photosensor Q1 to indicate staying
* Distance Sensor- While gameplay is active, the external distance sensor wired to PORTT is monitoring every .25 seconds to make sure no object comes too close to the board

*Flow Chart for Code – See Attachment*



