

CPTR-215
HW#19

Design and implement an assembly language program that will control both motors on the class vehicle and allow it to be “driven” using the joy stick to steer it. The vehicle should be capable of driving straight ahead or reverse in a straight line. It should be able to go right by slowing the right wheel while the left continues to turn at normal speed. Likewise it should be able to turn left by slowing the left wheel.

Satisfactory performance for HW#19 means that the vehicle can do the following:

- drive forward in a straight line at one speed
- drive forward but turning to the left (left wheel turns more slowly than the right)
- drive forward but turning to the right (right wheel turns more slowly than the left)
- drive backward in a straight line at one speed

Enhanced performance would be the ability to turn while going backward, moving forward/backward at more than one speed, or possibly turning at more than one rate.

Recognizing that the joy stick has only on/off switching and rough direction detection (only every 45 degrees), a suggested definition of joystick function is this:

- *Up* makes the vehicle go forward..
- *Down* makes the vehicle go backward.
- *Center* stops the vehicle
- Pressing *Right* while the vehicle is going forward causes the right wheel to slow down as long as *Right* is pressed. Releasing *Right* returns motion to straight forward.
- Pressing *Left* while the vehicle is going forward causes the left wheel to slow down as long as *Left* is pressed. Releasing *Left* returns motion to straight forward.

Please put a concise definition (such as that above) of how your joystick controls vehicle motion into the header of your program.

Recognizing that it may be difficult to find just the right pulse width values to make the motors rotate at exactly the same rate, the definition of going straight forward/backward means reasonably straight but there could be a little curvature.

Design your program before coding it. Use a flow chart, NS diagram, state diagram, or a combination thereof to document your design. These can be drawn neatly by hand. You don't need to spend time electronically drawing them.

To turn in:

- 1) Hard copy of your .s file. On this hard copy report your success or failure in getting the program to work. Hand written is ok. Or typed into a comment in the .s file is ok.
- 2) Hard copy of your design document(s)
- 3) Email me a copy of your .s file.

