

Physical Construction

MPU6050
Accelerometer/Gyro
scope (GY-521)



This side faces
the front of robot

11.1V 3S
LiPo
Battery

dsPIC33EP256MU806
Board

HC-05
Bluetooth
Module

Right
Wheel

2x RepRap
StepStick
Stepper Motor
Drivers (A4988)

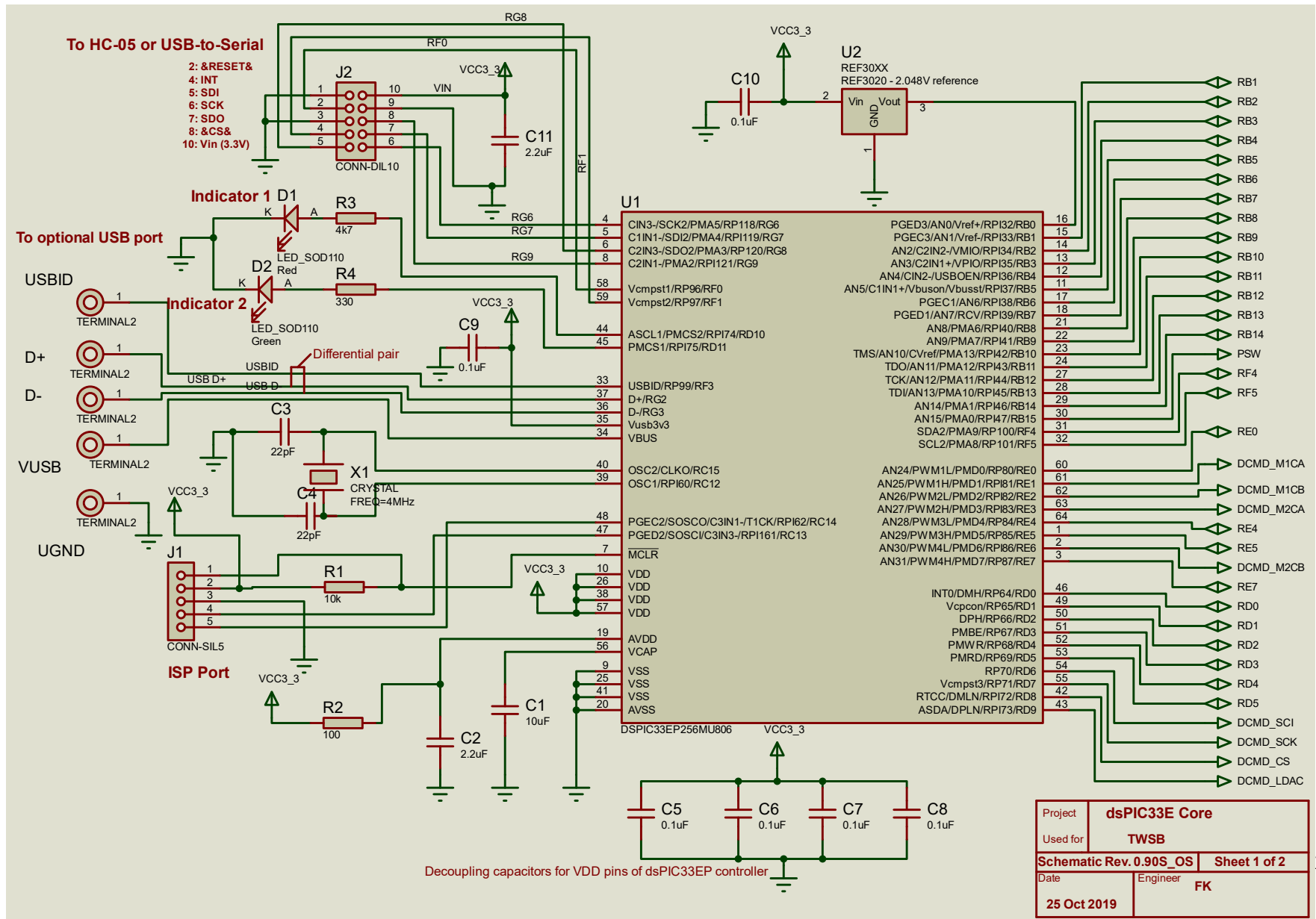
Left
Wheel

2x NEMA 17
Stepper Motor

FRONT VIEW

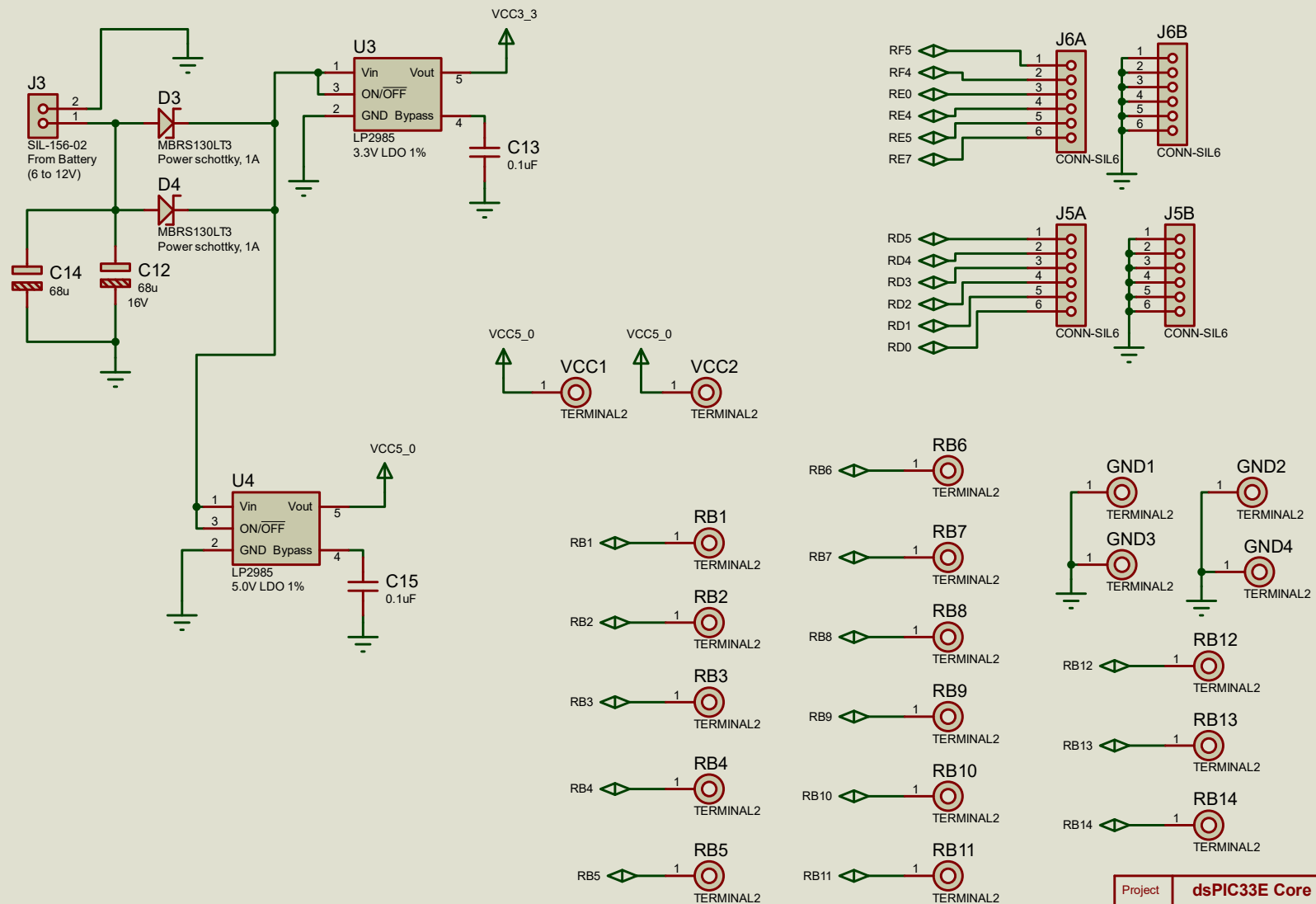
BACK VIEW

dsPIC33EP256MU806 Board (1)



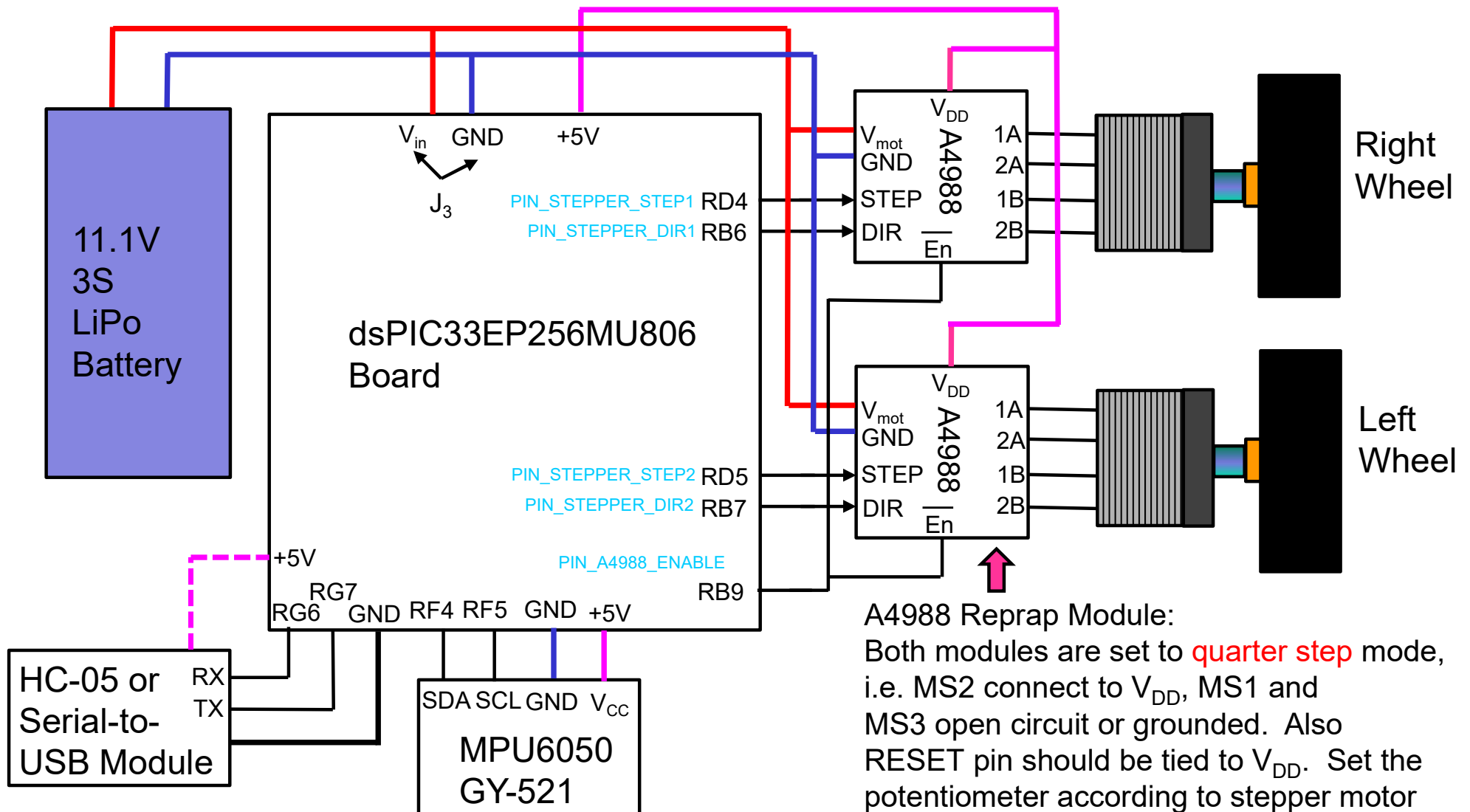
Project	dsPIC33E Core
Used for	TWSB
Schematic Rev.	0.90S_OS
Sheet	1 of 2
Date	25 Oct 2019
Engineer	FK

dsPIC33EP256MU806 Board (2)

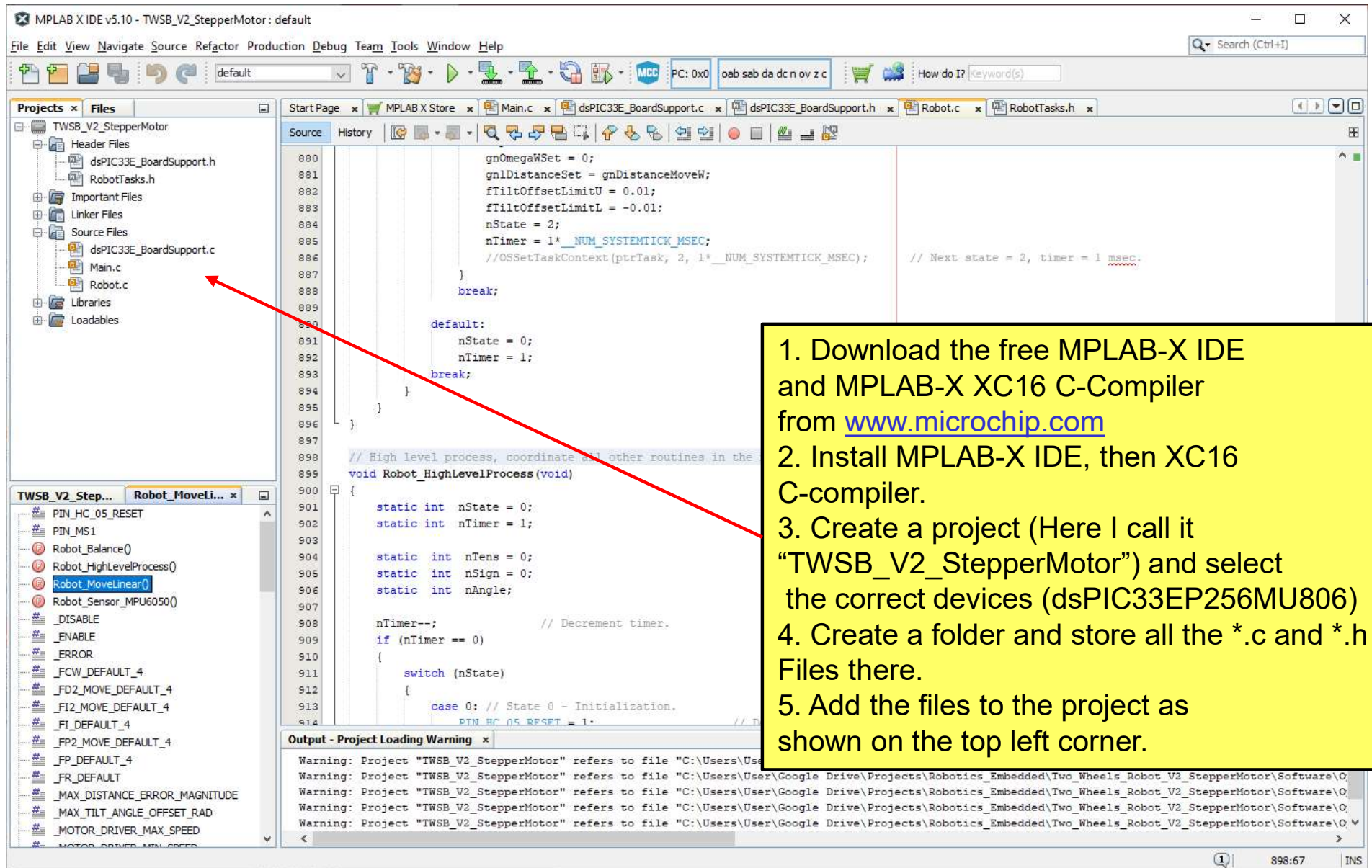


Project	dsPIC33E Core
Used for	TWSB
Schematic Rev. 0.90S_OS	Sheet 2 of 2
Date	25 Oct 2019
Engineer	FK

Wiring Diagram



Building the Firmware with MPLAB-X IDE



MPLAB X IDE v5.10 - TWSB_V2_StepperMotor : default

File Edit View Navigate Source Refactor Production Debug Team Tools Window Help

Search (Ctrl+F)

Projects * Files

- TWSB_V2_StepperMotor
 - Header Files
 - dsPIC33E_BoardSupport.h
 - RobotTasks.h
 - Important Files
 - Linker Files
 - Source Files
 - dsPIC33E_BoardSupport.c
 - Main.c
 - Robot.c
 - Libraries
 - Loadables

```
880 gnOmegaWSet = 0;
881 gnIDistanceSet = gnDistanceMoveW;
882 fTiltOffsetLimitU = 0.01;
883 fTiltOffsetLimitL = -0.01;
884 nState = 2;
885 nTimer = 1 * _NUM_SYSTEMTICK_MSEC;
886 //OSSetTaskContext(ptrTask, 2, 1 * _NUM_SYSTEMTICK_MSEC); // Next state = 2, timer = 1 msec.
887 }
888 break;
889
890 default:
891     nState = 0;
892     nTimer = 1;
893     break;
894 }
895 }
896 }
897
898 // High level process, coordinate all other routines in the
899 void Robot_HighLevelProcess(void)
900 {
901     static int nState = 0;
902     static int nTimer = 1;
903
904     static int nTens = 0;
905     static int nSign = 0;
906     static int nAngle;
907
908     nTimer--; // Decrement timer.
909     if (nTimer == 0)
910     {
911         switch (nState)
912         {
913             case 0: // State 0 - Initialization.
914                 PIN_HC_05_RESET = 1;
```

Output - Project Loading Warning

Warning: Project "TWSB_V2_StepperMotor" refers to file "C:\Users\User\Google Drive\Projects\Robotics_Embedded\Two_Wheels_Robot_V2_StepperMotor\Software\O...

Warning: Project "TWSB_V2_StepperMotor" refers to file "C:\Users\User\Google Drive\Projects\Robotics_Embedded\Two_Wheels_Robot_V2_StepperMotor\Software\O...

Warning: Project "TWSB_V2_StepperMotor" refers to file "C:\Users\User\Google Drive\Projects\Robotics_Embedded\Two_Wheels_Robot_V2_StepperMotor\Software\O...

Warning: Project "TWSB_V2_StepperMotor" refers to file "C:\Users\User\Google Drive\Projects\Robotics_Embedded\Two_Wheels_Robot_V2_StepperMotor\Software\O...

Warning: Project "TWSB_V2_StepperMotor" refers to file "C:\Users\User\Google Drive\Projects\Robotics_Embedded\Two_Wheels_Robot_V2_StepperMotor\Software\O...

1. Download the free MPLAB-X IDE and MPLAB-X XC16 C-Compiler from www.microchip.com
2. Install MPLAB-X IDE, then XC16 C-compiler.
3. Create a project (Here I call it "TWSB_V2_StepperMotor") and select the correct devices (dsPIC33EP256MU806)
4. Create a folder and store all the *.c and *.h Files there.
5. Add the files to the project as shown on the top left corner.

Loading the Program

- Build the project in MPLAB-IDE, then load the code into the micro-controller.
- In the photo is an example of loading the program using MPLAB-SNAP in-circuit debugger. Any programmer/debugger compatible with MPLAB-X IDE should be fine.

Custom designed
PCB for dsPIC33EP256MU806

