LINE FOLLOWER WITH PID CONTROLLER USING MATLAB SIMULINK

# ABSTRACT

This documentation provides basic information about MATLAB and Simulink work for programming Arduino(UNO) for line follower with PID controller. This documentation includes problems that we faced during this project with their solutions found by us. There are some useful links related to this project at the end of this document which we refereed.

# PID CONTROLLER

DURING PID CONTROLLER ALGORITHM IT IS VERY IMPORTANT TO SET THE WEIGHTAGE OF THE INPUT PINS ACCORDINGLY,DURING OUR FIRST TEST WE SET THE PIN STATUS AS FOLLOWING

pin\_status[0]=-4;//0

pin\_status[1]=-3;//2

pin\_status[2]=-2;//4

pin\_status[3]=-1;//6

pin\_status[4]=1;//8

pin\_status[5]=2;//10

pin\_status[6]=3;//12

pin\_status[7]=4;//14

BUT THE PROBLEM WITH THIS SETTING WAS THAT WHEN ALL THE PINS WERE HIGH i.e. THE BOT IS OFF THE TRACK THEN ALSO THE ERROR WAS ZERO WHICH IS NOT CORRECT ,SO WE USED FOLLOWING PIN WEIGHTAGE

pin\_status[0]=0;//0

pin\_status[1]=2;//2

pin\_status[2]=4;//4

pin\_status[3]=6;//6

pin\_status[4]=8;//8

pin\_status[5]=10;//10

pin\_status[6]=12;//12

pin\_status[7]=14;//14

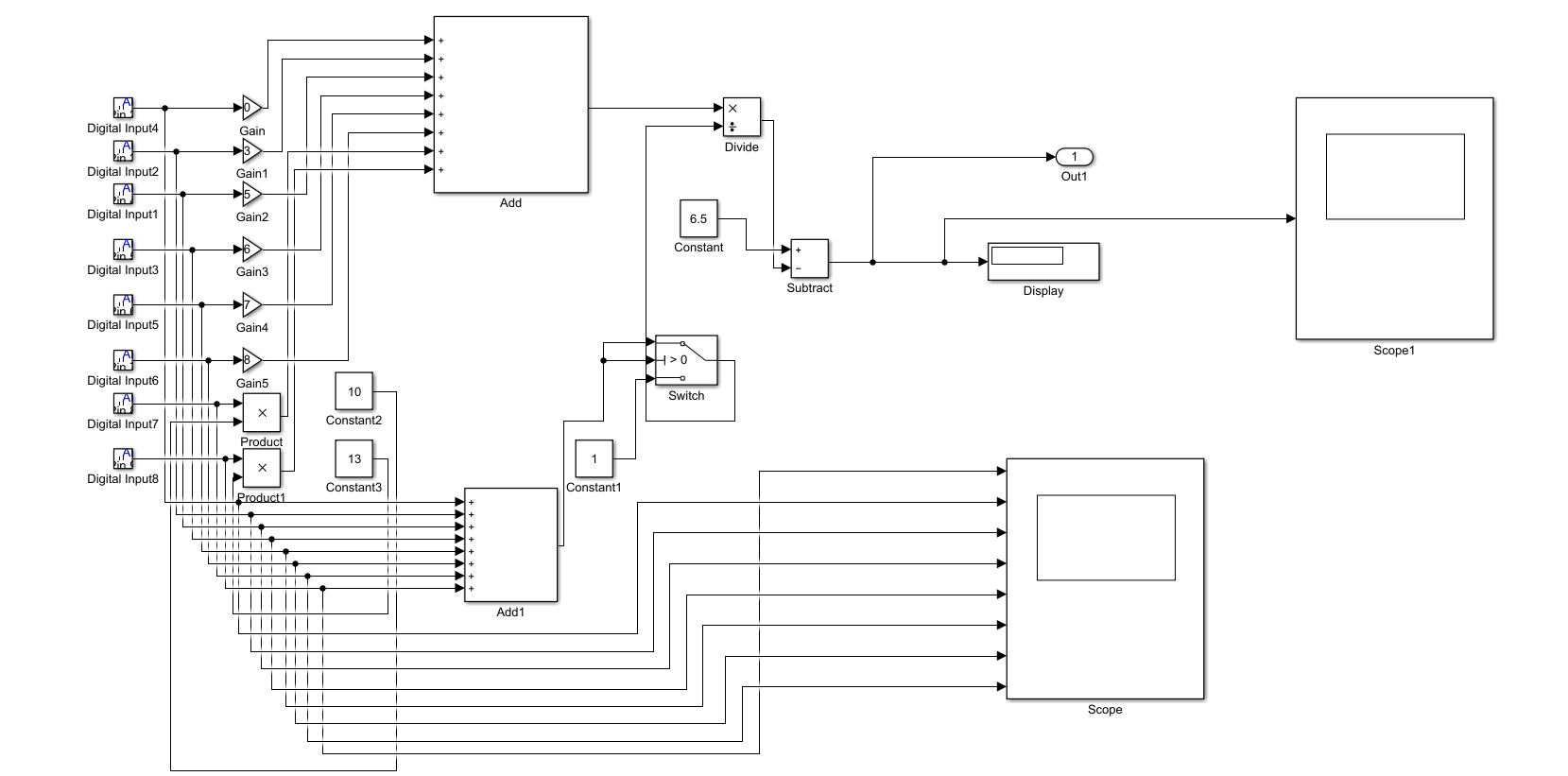
MOREOVER WHEN ALL THE PINS ARE LOW i.e. J=0 IN THE CALCULATION

error/=j;

THEN WRONG OUTPUT WILL BE GENERATED

SO ,AN EXTRA CONDITION IS REQUIRED i.e. WHEN J=0 THEN MAKE J=1 FOR CORRECT CALCULATION.

THEN AGAIN WE CHANGE THE PIN STATUS AND WE GAVE MORE WEIGHTAGE TO THE EXTREMUM IR SENSOR AS COMPARED TO THE MIDDLE IR SENSORS FOR OBTAINING A COORECT U TURN FOR LINE FOLLOWER.



* INPUT AND ERROR CALCULATION BLOCK

# INSTALLATION

### INSTALLATION OF MATLAB 2017a

MATLAB 2017 can be installed from its official site <http://www.mathworks.com> ,which provides 30 days free trial version of the MATLAB. Simulink is a part of MATLAB.

### INSTALLATION OF SUPPORT PACKAGES

To use Arduino blocks in Simulink, we have to install support packages. You should install “Simulink Support Package for Arduino Hardware”. You can install it from MATLAB>ADD ONs>GET HARDWARE SUPPORT PACKAGES. This support package will allow you to use basic Arduino blocks like Digital/analog input/output blocks in Simulink.

To interface Arduino to MATLAB you also have to install “MATLAB Support Package for Arduino Hardware”.

# SIMULINK MODEL

### INPUT AND ERROR CALCULATION SUBSYSTEM

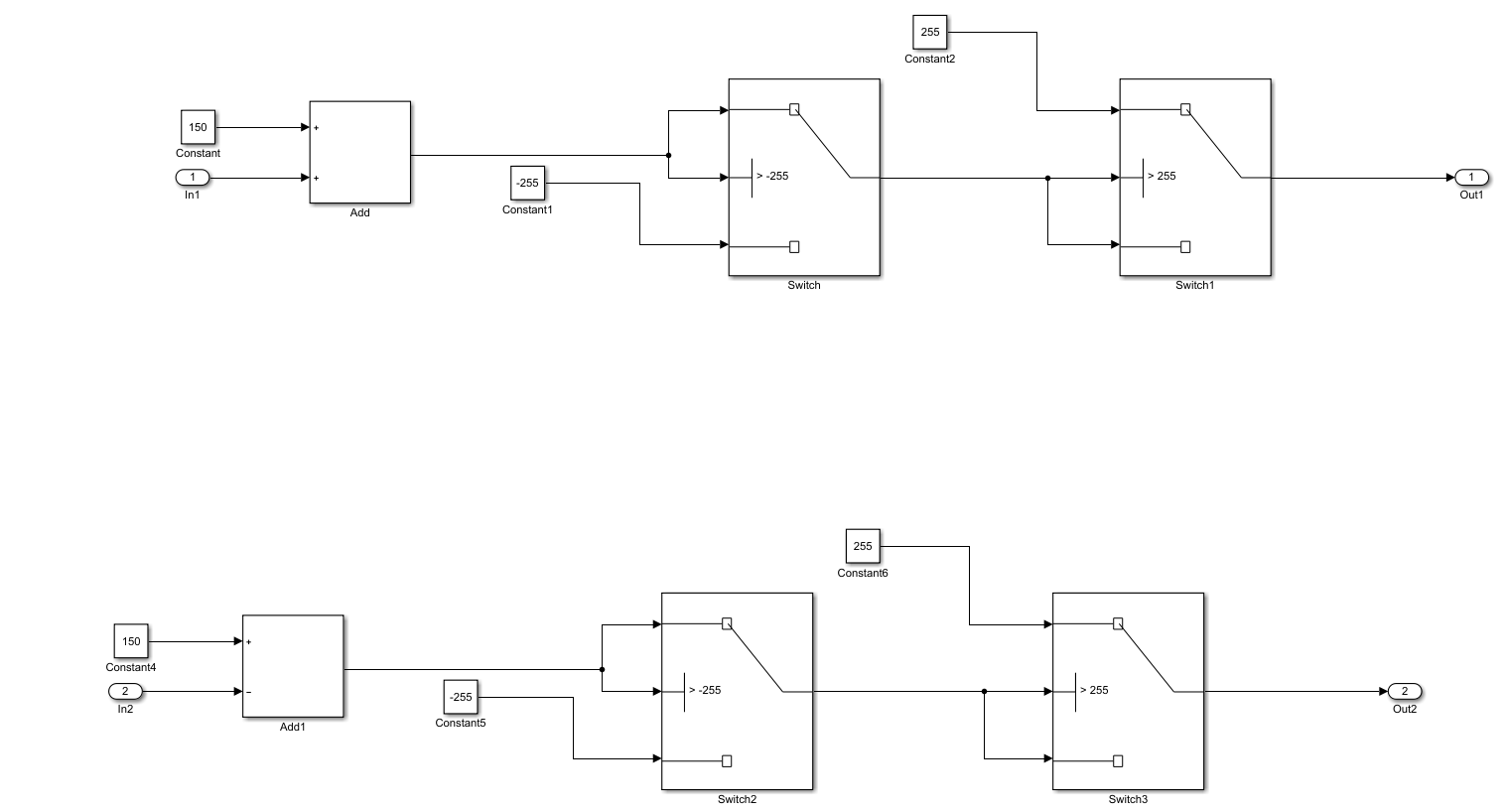
As we are using digital inputs from IR sensor array of 8 IR sensors, we should include 8 Digital Inputs blocks. You should also set Pin No which is used for input. You can use 0-13 in Arduino Uno and 0-53 in Arduino Mega 2560.But you should not use Rx-Tx pins which is 0&1. As explained in PID controller part we should multiply this input with its weightage for error calculation, which is done by gain(Multiply with 0-9 only) or multiplier block. This weighted input id added(By a single Adder block) and divided(By Division block) by no of pins which are on line explained in PID controller. This is subtracted from required value which will give final error at that time. This error is given as input to the PID block.

### PID BLOCK

Output of error calculation subsystem is the input of PID block. This block is given in library. We can set the value of Kp, Kd & Ki in properties of block. This block will give the value of total error based on input from the IR sensors. This is given as input to next PWM calculation Subsystem. As PWM have to be integer datatypes of all outputs of PID block have to be int16.

### PWM CALCULATION SUBSYSTEM

Final Error given by PID block is used as input for this subsystem. We have to add error to one motor’s base rpm while subtract from the other ones. For zero error both motors should be as same rpm for that we have given same base rpm to both motors. We are always rotating both motors in forward directions only & max value of PWM is 255. So motors PWM inputs should be between 0 to 255. To ensure this we have used two switch blocks for each motors which will work as if else statement.



### MOTOR DRIVER SUBSYSTEM

We are using Cytron for our project. We have to give 4 inputs from Arduino to it. Two inputs which are Dir1 & Dir2 will decide directions for motors forward or backword based on 1 or 0 as input. Which are given by digital output blocks. Other two are PWM inputs for both which are given by PWM blocks of Arduino library. For this we have to use analog pins only.

