

Hello all, i am trying to do a simple pid simulation with discovery board and matlab.

However i dont think that my data is runing as it should, what i want to say is that i dont get the usualy rising spike edge from the pid.

The pid calculation is done in the microcontroler, matlab only reads the two serial datas separated by @ character.

```
int main(void){

SystemInit();
cycleCounterInit();
SysTick_Config(SystemCoreClock / 1000);
lcd_init();
init_leds();
serial_init();

RCC_AHB1PeriphClockCmd(RCC_AHB1Periph_GPIOA, ENABLE);
RCC_APB2PeriphClockCmd(RCC_APB2Periph_ADC1, ENABLE);
GPIO_InitTypeDef GPIO_InitStructure;
GPIO_InitStructure.GPIO_Pin = GPIO_Pin_1;
GPIO_InitStructure.GPIO_Mode = GPIO_Mode_AN;
GPIO_InitStructure.GPIO_PuPd = GPIO_PuPd_NOPULL ;
GPIO_Init(GPIOA, &GPIO_InitStructure);

ADC_CommonInitTypeDef ADC_CommonInitStructure;
ADC_InitTypeDef ADC_InitStructure;

/* ADC Common Init */
ADC_CommonInitStructure.ADC_Mode = ADC_Mode_Independent;
ADC_CommonInitStructure.ADC_Prescaler = ADC_Prescaler_Div2;
ADC_CommonInitStructure.ADC_DMAAccessMode = ADC_DMAAccessMode_Disabled;
ADC_CommonInitStructure.ADC_TwoSamplingDelay = ADC_TwoSamplingDelay_5Cycles;
ADC_CommonInit(&ADC_CommonInitStructure);

ADC_InitStructure.ADC_Resolution = ADC_Resolution_12b;
ADC_InitStructure.ADC_ScanConvMode = DISABLE; // 1 Channel
ADC_InitStructure.ADC_ContinuousConvMode = DISABLE; // Conversions Triggered
ADC_InitStructure.ADC_ExternalTrigConvEdge = ADC_ExternalTrigConvEdge_None; // Manual
ADC_InitStructure.ADC_ExternalTrigConv = ADC_ExternalTrigConv_T2_TRGO;
ADC_InitStructure.ADC_DataAlign = ADC_DataAlign_Right;
ADC_InitStructure.ADC_NbrOfConversion = 1;
ADC_Init(ADC1, &ADC_InitStructure);

/* ADC1 regular channel 11 configuration */
ADC_RegularChannelConfig(ADC1, ADC_Channel_1, 1, ADC_SampleTime_144Cycles); // PC1

/* Enable ADC1 */
ADC_Cmd(ADC1, ENABLE);

char adctxt[10];
char pidtxt[10];
float Kp=1;
float Ki=0;
float Kd=0;

float windupGuard=2;
float SetPoint=2.0;
float input=0;
float error=0;
float output=0;
float current_time=0;
float last_time=0;
```

```
float deltaT=0;
float integral=0;
float derivative=0;
float prevErr=0;

while(1){
loop_20hz(){ // a 20hz interrupt every 20000 us systick count

ADC_SoftwareStartConv(ADC1);
while(ADC_GetFlagStatus(ADC1, ADC_FLAG_EOC) == RESET);
float bbb = ADC_GetConversionValue(ADC1);
        bbb=bbb*5/4096;

input=bbb;
current_time = micros(); // returns curent time from systick in uS
deltaT = (current_time - last_time)/1000000.0;
error=SetPoint-input;
integral=constrain((integral+error*deltaT),0,windupGuard);
derivative = (input-prevErr)/deltaT;
output = (Kp*error)+(Ki*integral)+(Kd*derivative);
prevErr =input;
last_time = micros(); // returns curent time from systick in uS

if(output<0){ output=0; }

sprintf(txtadc,"%0.1f",input);
sprintf(pidtxt,"%0.1f",output);

USART_Puts(txtadc);
USART_Puts("@"); // send to matlab curent adc value
USART_Puts(pidtxt); // sends to matlab curent pid output
USART_Puts("\r\n"); // sends the terminator chars

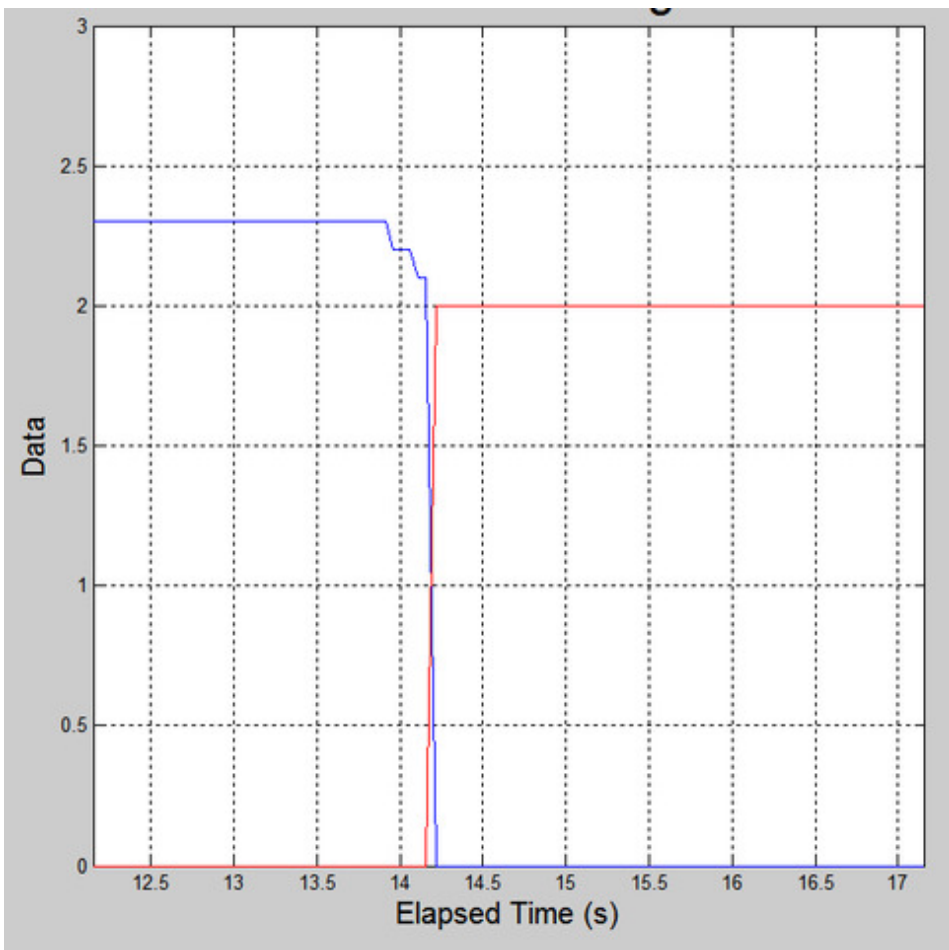
}

}

}
```

This is my pid plot.

blue line - is adc read from pot ( when potentiometer is idle it measures about 2.3 volts)  
red line - is the pid output



I will post the matlab code also,

```

clc
clear all
if ~isempty(instrfind)
    fclose(instrfind);
    delete(instrfind);
end
close all
clc
disp('Serial Port Closed')
clear
clc
serialPort = 'COM7';           % define COM port #

plotTitle = 'Serial Data Log'; % plot title
xlabel = 'Elapsed Time (s)';   % x-axis label
ylabel = 'Data';              % y-axis label
plotGrid = 'on';              % 'off' to turn off grid
min = 0;                       % set y-min
max = 3;                       % set y-max
scrollWidth = 5;               % display period in plot, plot entire data log if <= 0
delay = .01;                   % make sure sample faster than resolution

%Define Function Variables
time = 0;
data = 0;
count = 0;
bdata = 0;
btime = 0;

%Set up Plot
plotGraph = plot(time,data,'b');

```

```

title(plotTitle,'FontSize',25);
xlabel(xLabel,'FontSize',15);
ylabel(yLabel,'FontSize',15);
axis([0 10 min max]);
grid(plotGrid);

%Open Serial COM Port
s = serial(serialPort)
set(s,'DataBits',8);
set(s,'StopBits',1);
set(s,'BaudRate',9600 );
set(s,'Parity','none');

disp('Close Plot to End Session');
fopen(s);

tic
while ishandle(plotGraph) %Loop when Plot is Active
    zz=fscanf(s); % - HERE SCANS THE SERIAL INPUT FROM STM
    uu=regexp(zz,'@', 'split');
    dat =sscanf(uu{1},'%f');
    bdat =sscanf(uu{2},'%f');

    if(~isempty(dat) && isfloat(dat)&& isfloat(bdat)) %Make sure Data Type is Correct
        count = count + 1;
        time(count) = toc; %Extract Elapsed Time
        data(count) = dat(1); %Extract 1st Data Element
        bdata(count) = bdat(1);

        set(plotGraph,'XData',time(time > time(count)-scrollWidth),'YData',data(time > time(count))-
scrollWidth));hold on
        btime=time;

        plot(btime,bdata,'r');hold off

        axis([time(count)-scrollWidth time(count) min max]);
        else

            set(bplotGraph,'XData',time,'YData',bdata);
            axis([0 time(count) min max]);

        end

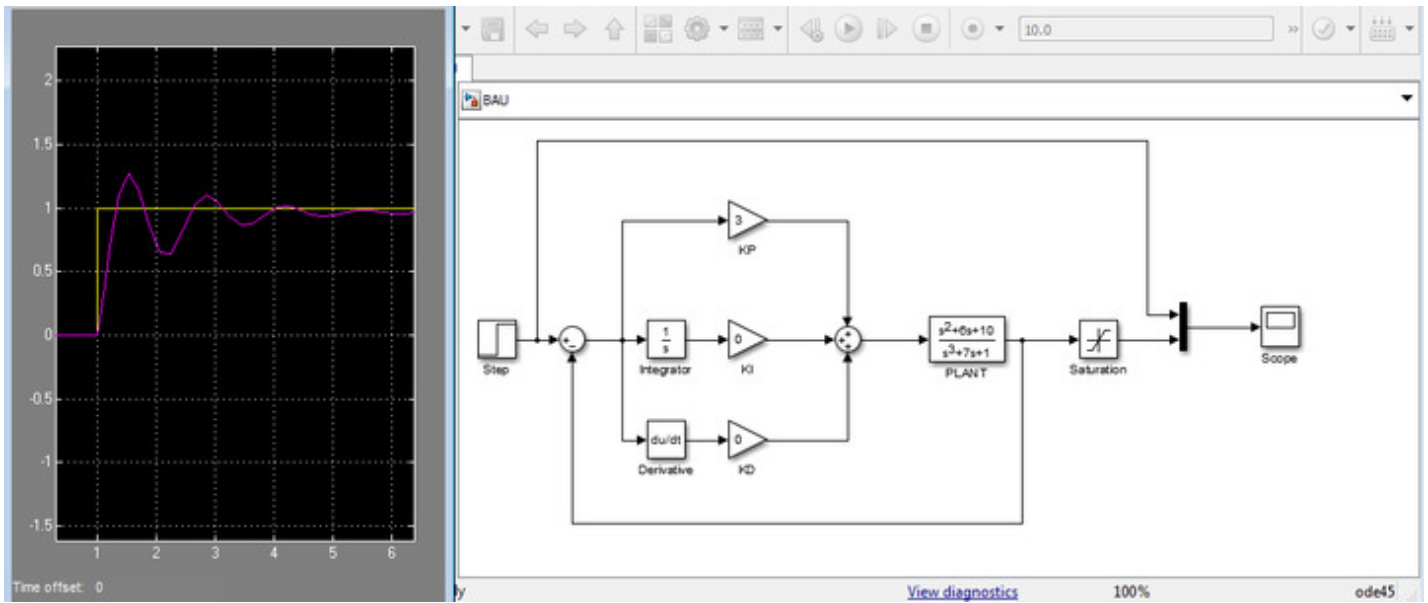
        drawnow;
end

%Close Serial COM Port and Delete useless Variables
fclose(s);
clear count dat delay max min plotGraph plotGrid plotTitle s ...
scrollWidth serialPort xLabel yLabel;

disp('Session Terminated...');

```

And here is a simulation graph done only in matlab environment wich shows me very nicely the glitch of pid response at the rising edge



So on my stm code and plot why i dont see the glitch? Is it because i sample so fast before the serial output? Did anyone else encounterd such issue? This setup is done in the absence of all hardware setup, its used for pid control on a quadcopter platform, so i have to wait for mottors and esc to arrive. I just wanted to test my pid engine using a simple adc input from a potentiometer, but not seeing the pid glitch i cant form an idea about seting the relative values for the KP,KI and KD parameters.