Engineering 102 Spring 2016

Team Project (complete by April 14)

Project Statement:

Design, simulate, build and test a digital controller for an overdamped second order LTI system to satisfy the following specifications for the response to a step reference input of 2.5V:

- **zero overshoot**
- **zero steady state error**
- **minimize the control input while keeping the 2% settling time under 4s.**
- **the sample rate is 10 Hz**

Objectives:

After completing the project, you should be able to:

a) Design and build an implementation of an overdamped second order system using a Sallen-Key circuit.

b) Design and simulate a discrete-time state space control loop for the system with state feedback and an observer to satisfy the design specifications given in the project statement.

c) Implement your digital control design on an Arduino Uno microprocessor and measure the experimental step response of the closed loop to a 2.5V step reference input.

d) Discuss the practical considerations of sampling rate, aliasing, control input saturation and the limitations of the hardware capabilities.

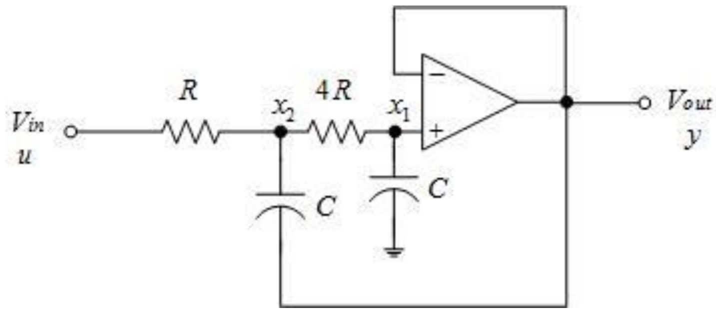
e) Discuss your design methods and rationale for the choice of the control loop parameters

f) Discuss the sources of error between the simulated and measured step response of the digital control loop.

Project Tasks

- Sign out a Big Stems Kit from the instructor and complete the "Using the Arduino Uno as a digital controller" tutorial attached as appendix to this document

- Build the following circuit with the LMC6484 op amp (data sheet in class directory).



where $C = 10\mu F; R = 50k\Omega$
Show that the continuous-time state equation is

$$\dot{\mathbf{x}} = \begin{bmatrix} -\frac{1}{2} & \frac{1}{2} \\ 0 & -2 \end{bmatrix}\mathbf{x} + \begin{bmatrix} 0 \\ 2 \end{bmatrix}u; \quad y = \begin{bmatrix} 1 & 0 \end{bmatrix}\mathbf{x}$$

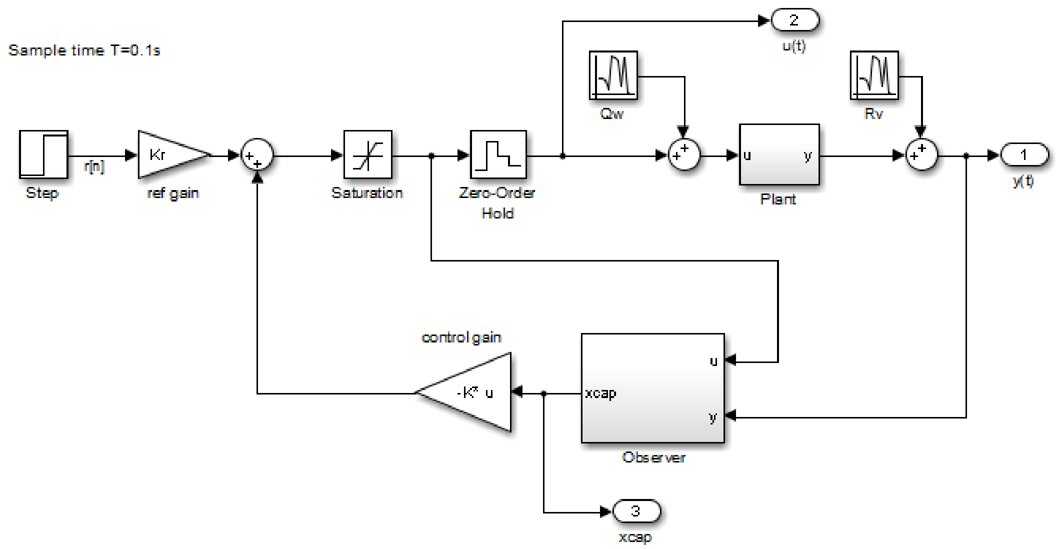
for $u = V_{in}; y = V_{out}$

- Determine the equivalent discrete-time state equation for the circuit with a sample time $T = 0.1\,s$

$$\mathbf{x}[n+1] = \mathbf{A_d}\mathbf{x}[n] + \mathbf{B_d}u[n]; \quad y[n] = \mathbf{C}\mathbf{x}[n]$$

- Design an observer-based discrete-time state feedback control system for the circuit to satisfy the design specifications given in the project statement.

- Simulate the discrete-time control loop for a step reference input of 2.5V.



Simulink => Discontinuities => Saturation

- Plot the simulated step response $y_{sim}(t)$, the estimated system state $\hat{x}(t)$ and the control input $u_{sim}(t)$. Tabulate the 2% settling time, overshoot, steady state error and integral absolute value of the deviation of the control input from the final steady state control input $\int_0^{t_{sim}} |u_{sim}(\tau) - u_{ss}| d\tau$

- Implement the digital control loop by coding the Arduino Uno microprocessor to interface with the circuit. Experimentally measure the closed loop step response $y(t)$ and the control input $u(t)$ of the system and compare with your simulated results.

Project Instructions

Please submit a report describing the design, simulation and testing of a discrete-time state space digital controller for a second order circuit to satisfy the design specifications given in the Project Statement. Please try to keep the report to a reasonable length (~10 pages, excluding Appendices)

The Project Report (hard copy) will be handed in to Prof Bright (P2386) by **noon Thursday April 14**. The report will be typed (double spaced) with the following format:

I. **Cover Page.** Team Members

II. **Background** (Brief description of project tasks. Short summary of methods used for control loop design)

III. **Simulations** (Step response simulation plots showing control inputs, estimated state variables and outputs for the circuit)

IV. **Experimental Results** (Experimental step response plots showing control inputs and outputs for the circuit)

V. **Conclusions** (Interpretations of results and comparison of simulated and experimental step response performance)

All relevant figures and tables should be labeled, referenced and included in the body of the report. Mathematical equations and drawings can be included freehand in the body of the report. MATLAB code, Simulink block diagrams and Arduino code should be included in separate Appendices to the report.

The assessment rubric to be used for the project is attached.

Please return the experimental kit to Prof Bright at the completion of the project. Please disassemble your circuits before returning the kit.

Appendix: Using the Arduino Uno as a digital controller

Instructions for downloading software for your Arduino Uno can be found at
arduino.cc/en/Guide/HomePage

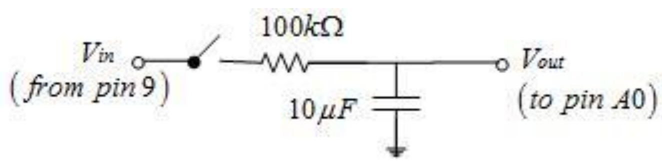
Arduino Exercise

First, determine the COM port your Arduino is attached to on your computer. **Control Panel =>Hardware and Sound/Devices and Printers/Device Manager**. You'll need to enter this when you start your "sketch" (program).

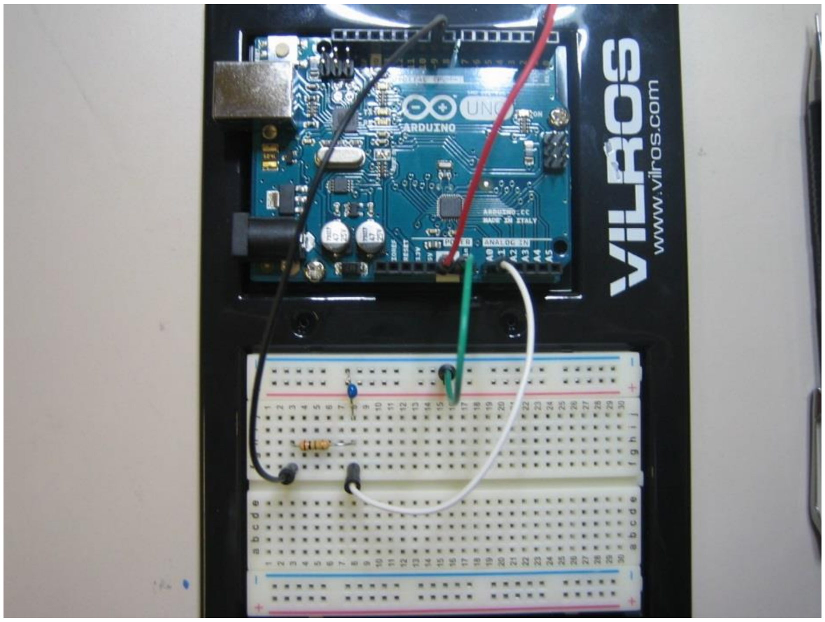
Arduino programs ("sketches") are saved as plain text files with the extension *.ino* . The following exercise demonstrates the input and output capabilities of the Uno controlling an external circuit. The Uno contains 6 analog input channels A0 to A5, 10 bit analog to digital conversion. Acquisition time is about 0.1ms. The analog output on pins 3, 5, 6, 9, 10 and 11 is actually a pulse-width-modulated (PWM) digital signal. The average value of the PWM represents the "analog" output, 0 to 5V range, 8 bit resolution. The frequency of the output PWM is approximately 490Hz on most pins, 980 Hz on pins 5 and 6

Proportional control of an RC circuit

- Connect a switch (wire) between pin2 and GND on the Arduino
- Connect GND on the Arduino to GND on the protoboard
- Connect Pin 9 to Vin of an RC circuit on the protoboard



- Connect the output of the RC circuit to analog pin A0 on the Arduino



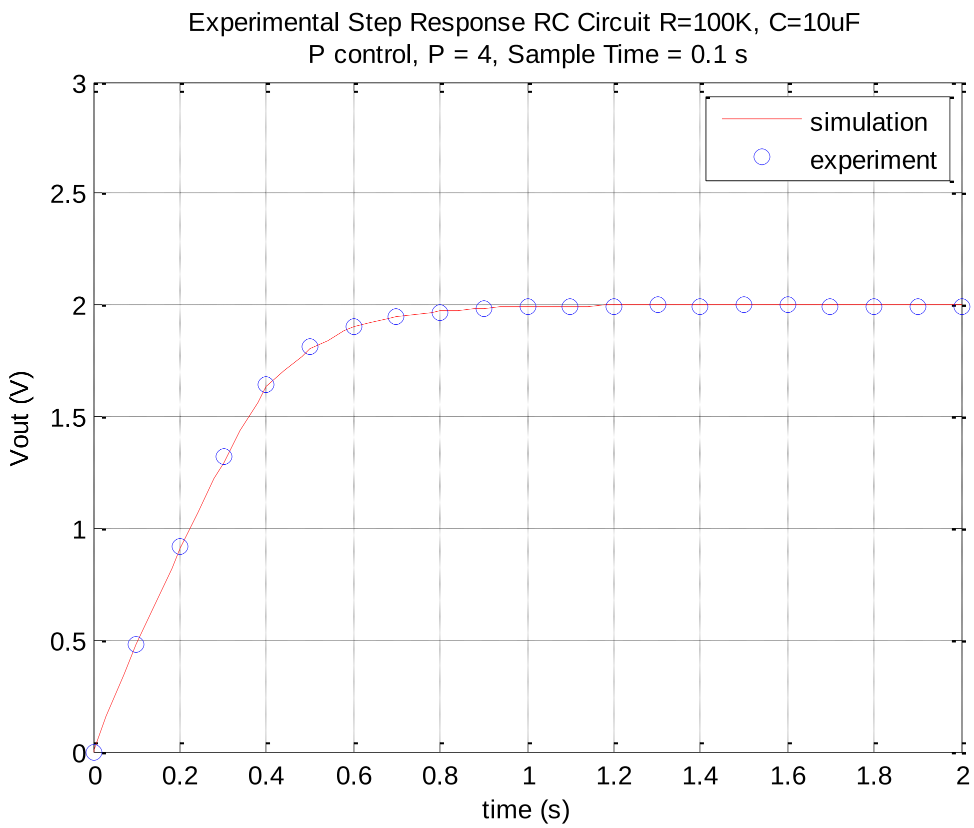
- File => New
- Paste the following code:

```
/*
 Step response of P control RC circuit after
 a switched 2.5V step reference input.

 The circuit:
 * switch connected from pin 2 to GND
 * input Vin from pin 9
 * output of circuit read at analog pin A0
*/

//PIN SETTINGS
const int yPin = A0;  // Analog input pin
const int uPin = 9;    // Analog output pin
const int switchPin = 2;     // input pin for the switch
boolean switchVal = HIGH;    // declare initial switch pin state

// Parameter Settings
int time = 0;  // initialize time
int uVal = 0;  // initialize control input
const float P = 4.0;
const float ref = 2.5; //2.5V reference input

// Initial Values for internal signals
float y=0;
float u=0;
float e=0;

void setup() {
  pinMode(switchPin, INPUT);  //set switch pin to input mode
  digitalWrite(switchPin,HIGH); //initialize to start with pull up voltage
  Serial.begin(9600);
}

void loop() {
  //WAIT FOR SWITCH
   while(switchVal == HIGH)  // repeat this loop until switch is turned on
                            // (switchVal will go LOW when switch is turned on)
  {
    analogWrite(uPin,0);
    switchVal = digitalRead(switchPin);  // read switch state
  }


 // READ CIRCUIT OUTPUT
  int sensorVal = analogRead(yPin);
      // convert to volts
      y=sensorVal*(5.0/1023.0);
      // calculate error
      e=ref-y;

  // CONTROLLER
     u=P*e;

     // check that control signal is in range
     if (u>5)
     {
       u=5;
     }
     if (u<0)
     {
     u=0;
     }

  // WRITE CIRCUIT INPUT
     uVal=u*(255/5);
     analogWrite(uPin,uVal);
```

```
 // print the results to the serial monitor:
  Serial.print(time++);
  Serial.print (" ");
  Serial.print (y);
  Serial.print (" ");
  Serial.println(u);

  // WAIT FOR NEXT SAMPLE
  delay(100); //sample frequency 10Hz
  switchVal = digitalRead(switchPin);  // read switch state
}
```

For datalogging you can download the serial terminal program CoolTerm[1] to your computer.

- Open CoolTerm, and choose your serial port in the Options menu.
- Click the Connect icon
- From the Connection Menu, choose Capture to TextFile… > Start. Name your file and save.
- To stop capture, choose Connect > Capture to TextFile… > Stop.
- Open your file in MATLAB for plotting and analysis



[1] http://freeware.the-meiers.org

Student Team:

Skills Demonstrated	None (0)	Not Adequate (1)	Marginal (2)	Adequate (3)	Good (4)	Superior (5)	Multiplier	Points
Design and build an implementation of an overdamped second order system using a Sallen-Key circuit.							3	
Design and simulate a discrete-time state space control loop for the system with state feedback and an observer to satisfy the design specifications given in the project statement							4	
Implement your digital control design on an Arduino Uno microprocessor and measure the experimental step response of the closed loop to a 2.5V step reference input.							4	
Discuss the practical considerations of sampling rate, aliasing, control input saturation and the limitations of the hardware capabilities.							3	
Discuss your design methods and rationale for the choice of the control loop parameters							3	
Discuss the sources of error between the simulated and measured step response of the digital control loop.							3	
							Final Grade	

Comments: