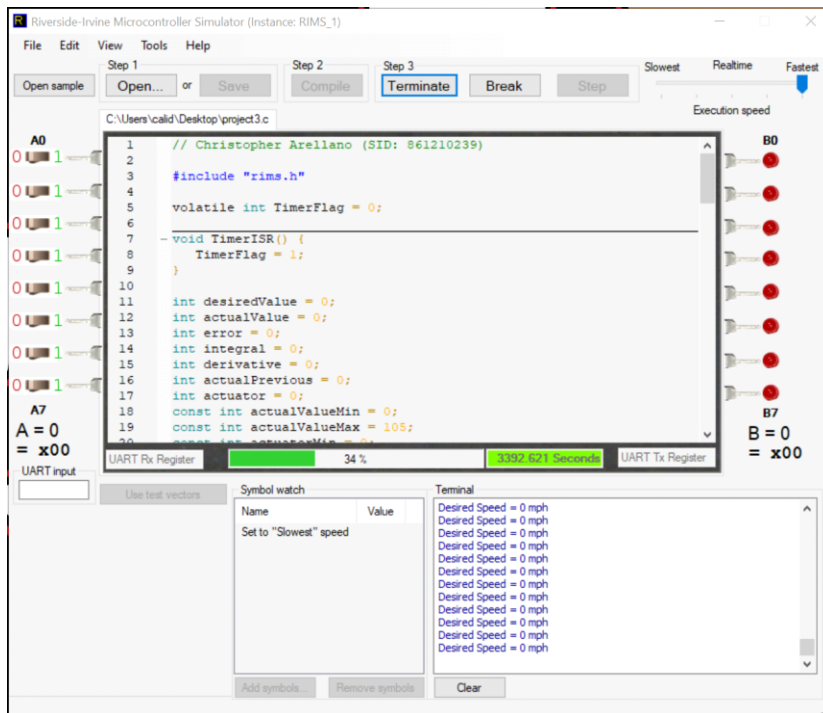


### Project #3 Summary

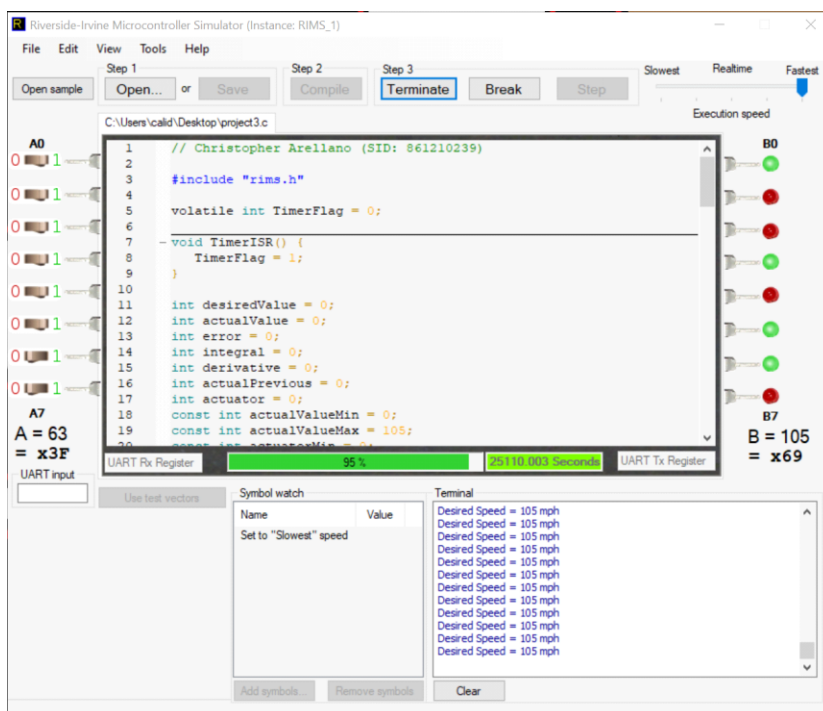
#### Project Description:

For project #3 I have made a car speed PID system in resemblance to the car speed PI system that Professor Frank Vahid made during one of our lectures in a google spreadsheet. For my system I have set A0 to the value of 5 mph, A1 to the value of 10 mph, A2 to the value of 15 mph, A3 to the value of 20 mph, A4 to the value of 25 mph and A5 to the value of 30 mph. Therefore when A0 through A5 are all pressed the speed is 105 mph and when none of A0 through A5 are pressed the speed is 0 mph. Thus the minimum speed possible is 0 mph and the maximum speed is 105 mph. I have set A6 as a selector between (!A6)seeing the current speed through the LEDs in binary representation or (A6)seeing the current speed through the terminal as text. I have set A7 as an emergency brake that resets all values and causes for the current speed to immediately change to 0 mph. When the emergency brake is set (A7) the system disregards any speed selections made by pressing any of the buttons A0 through A5. When the emergency brake is released (!A7) the car resumes to speeding up to the desired speed set by pressing any of the buttons A0 through A5. The car speed PID system gradually speeds up to the desired speed inputted by the pressing of any button A0 through A5 and avoids going over the desired speed limit in most if not all cases. The Kp, Ki and Kd values I have used for this project are the same values I used for my project #2 of which are:  $K_p = 0.05$  ,  $K_i = 0.001$  and  $K_d = 2.5$  . Since float is not available in RIMS and in most cases is not a viable option, I have multiplied all decimal values by 1000 and I have divided certain calculations by 1000 in order to account for the adjustments made. My car speed PID system is very similar to the car speed PI system that Professor Frank Vahid made in a google spreadsheet with the only major changes being that I have included the derivative in my system, I have used project #2's Kp, Ki, and Kd values, and I have removed the decimal adjustments that were present in the spreadsheet calculations. Such decimal adjustments present in the spreadsheet calculations were:  $0.6 * \text{error} * P...$  and  $0.7 * v_t + 0.5 * u_t...$  .

Desired Speed 0 mph with LED binary representation:



Desired Speed 105 mph with LED binary representation:



[illegible]

The screenshot displays the Rideside-Irvine Microcontroller Simulator interface. The main window shows a C program for a microcontroller, likely an AVR, implementing a timer interrupt service routine (ISR). The code defines a timer flag, sets up an ISR, and calculates various values based on a desired speed.

```
1 // Christopher Arellano (SID: 861210239)
2
3 #include "rims.h"
4
5 volatile int TimerFlag = 0;
6
7 void TimerISR() {
8     TimerFlag = 1;
9 }
10
11 int desiredValue = 0;
12 int actualValue = 0;
13 int error = 0;
14 int integral = 0;
15 int derivative = 0;
16 int actualPrevious = 0;
17 int actuator = 0;
18 const int actualValueMin = 0;
19 const int actualValueMax = 105;
```

The simulator's top menu bar includes File, Edit, View, Tools, and Help. Below the menu are three tabs labeled Step 1, Step 2, and Step 3. Under Step 1, there are buttons for Open sample, Open..., or Save. Under Step 2, there is a Compile button. Under Step 3, there is a Terminate button (highlighted with a red box), Break, and Step buttons. To the right of these tabs are Execution speed controls: Slowest, Realtime, and Fastest (selected). A vertical slider below these controls allows adjusting the execution speed.

On the left side of the code editor, there are two registers displayed: A0 and A7. A0 is shown as 120 and A7 is shown as x78. On the right side, B0 is shown as 0 and B7 is shown as x00. Below the code editor, there are two status bars: UART Rx Register (showing 91%) and UART Tx Register (showing 31430 7600 Seconds).

At the bottom of the interface, there is a Symbol watch section with columns for Name and Value. It shows Set to "Slowest" speed. To the right of the Symbol watch is a Terminal section displaying a log of messages: Desired Speed = 75 mph, Current Speed = 75 mph, and Desired Speed = 75 mph, repeated multiple times.