I2C Transmit

My name is Alan Hawse and this is PSoC 101. We just did an I2C receive project. Now let's reverse that and transmit data from the slave to the Control Panel. I do not want to spend a lot of unrelated effort creating data to transmit but the example is more interesting if the data changes value. So I will simply run the PWM slowly and transmit the counter value.

Make a copy of the receive project. The only change we need here is to slow down the clock so that the PWM does not run faster than the I2C bridge can send data. Open the clock customizer and set it to 1kHz. The I2C defaults to 100kHz so that leaves plenty of time to update the master.

In C we already have most of the code in place. For safety we should make the buffer un-writable from the master so I will change the second argument in the SetBuffer1 function. In the main loop I just need to read the counter value and copy it into the I2C buffer. The EZI2C interrupt handlers do all the rest of the work.

Back in the Bridge Control Panel we connect as normal. Now I need a new command. Click in the editing panel and type control-enter to create a new line. The new command starts with an 'r' – for read. Then the I2C address. There is no need to set the offset for a read command – you just write "x" for as many bytes as you want. Then close the command with a 'p' again. When you run the command it will respond with the value of the PWM counter. Every time you run the command you will get a new value. That proves that the data is getting transmitted but it is hard to really see how it is changing.

Let's plot the data on a graph to help with that. Go to the Chart menu and select Variable Settings. In the dialog make one of the entries active by checking the box. Give it a memorable name, like counter, and make sure it is a one-byte variable.

Change the command to replace the "x" with an at-sign and the name you used for the variable. Now, instead of pressing enter to run the command, press the repeat button. Click on the chart tab and you'll see the PWM counter incrementing up to 255 then dropping back to zero in the classic saw-tooth pattern.

To extend this program modify the PWM to be center-aligned. What kind of waveform do you get in the Control Panel now?

As always you are welcome to email me at alan_hawse@cypress.com