

Example converter I2C -> PWM

SPECIFICATIONS:

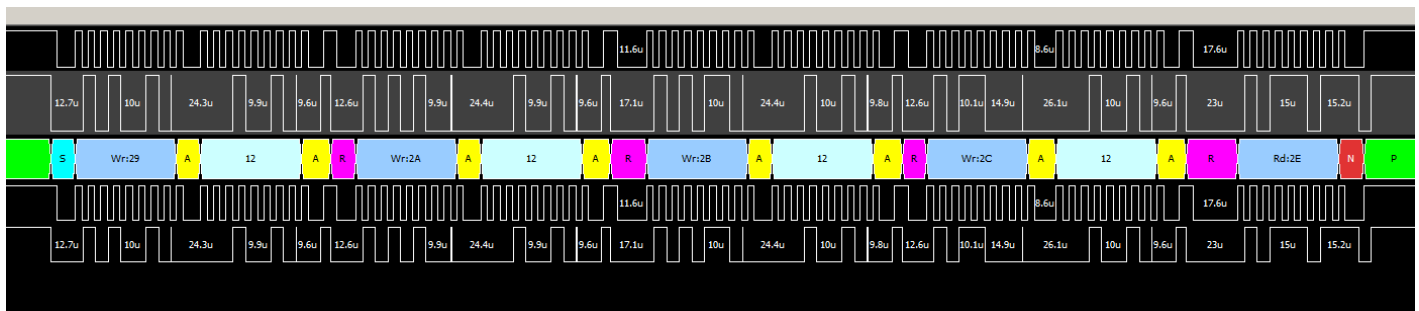
Address Sequence = { 0x52, 0x54, 0x56, 0x58, 0x5a, 0x5c, 0x5e, 0x60 };
throttle value = from 0x00 to 0xff

The sequence is important, because you can't write to 0x52, and then 0x56

SEQUENCE:

```
*"S, W52, 10, RS, W54, 12, RS, W56, 10, RS, W58, 11, P"*  
S = Start  
RS = Repeat Start (it is important, YOU DON'T SEND START)  
P = Stop.  
W = write to address XX, numbers = data following.
```

LOGIC ANALYZER CAPTURE



NOTE:

You should send data continuously (or at least faster than 100ms) even if data doesn't change

EXAMPLE FUNCTION FOR DSPIC30F3013

```
void update_motor_I2C(void)  
{  
    char no_ack = 1;  
  
    /* MOTOR 1 */  
    StartI2C(); while(I2CCONbits.SEN); // Wait for finish START  
    MasterWriteI2C(0x52); while(I2CSTATbits.ACKSTAT == no_ack); // Wait for ACK  
    MasterWriteI2C(data_throttle[0]); while(I2CSTATbits.ACKSTAT == no_ack);  
    /* MOTOR 2 */  
    I2CCONbits.RSEN = 1; while(I2CCONbits.RSEN); // Wait for finish REPEAT START  
    MasterWriteI2C(0x54); while(I2CSTATbits.ACKSTAT == no_ack);  
    MasterWriteI2C(data_throttle[1]); while(I2CSTATbits.ACKSTAT == no_ack);  
    /* MOTOR 3 */  
    I2CCONbits.RSEN = 1; while(I2CCONbits.RSEN);  
    MasterWriteI2C(0x56); while(I2CSTATbits.ACKSTAT == no_ack);  
    MasterWriteI2C(data_throttle[2]); while(I2CSTATbits.ACKSTAT == no_ack);  
    /* MOTOR 4 */  
    I2CCONbits.RSEN = 1; while(I2CCONbits.RSEN);  
    MasterWriteI2C(0x58); while(I2CSTATbits.ACKSTAT == no_ack);  
    MasterWriteI2C(data_throttle[3]); while(I2CSTATbits.ACKSTAT == no_ack);  
    StopI2C(); while(I2CCONbits.PEN); //Wait for finish STOP  
}
```