Nathan Edwards

From: Nathan Edwards [Nathan.Edwards@asu.edu]
Sent: Tuesday, February 17, 2009 9:29 PM
To: 'Jacob Rosenthal'
Subject: RE: I2C and PWM Ruminations..

You bring up a great design question. We do have to account for the time delay to receive the Receiver ACK. I2C actually will have 18 clocks to transfer 8 bits of data plus a latency for the Receiver ACK. So maybe our best case is 20 clocks for 8bits. This mean throughput at 500KHz is 8bits/(20*2us) = 200Kbps.

Lets say our worst case for ACK latency is 9 clocks - that gives us 27 clock, round up to 30 clocks for an 8 bit transfer. Throughput would then be 8bits/(30*2us) = 133Kbps.

So yes, time to 1 vibrator ON would be approx 60us (using my worst case). Add some system delay and another 60us to turn OFF. That still means 1 Vibrator ON/OFF cycle is approx 120us or 0.12ms. This is great since we want minimum duty cycle (100%) to be at 50ms!!!

If we are controlling 16 vibrators at once we need to account for loss of arbitration and the ACKs, but as you see we have plenty of room to have at least 25 duty cycle settings (1 cycle per 1ms) or more. I think that 25 magnitude settings is probably too many, maybe we will need only 10 settings for the haptics research.

Maybe we need to run our own studies/tests and publish our findings on the ACK latencies through I2C with CAT5 distances. Another item I just read on limitation of I2C communication is the Max Bus Capacitance. Need to do some math on the capacitance of CAT5 with our planned setup, but I am thinking we should be fine since we are using relatively large conductors (32 AWG or 28AWG).

Good thoughts!!!

From: Jacob Rosenthal [mailto:jakerosenthal@gmail.com]
Sent: Tuesday, February 17, 2009 8:49 PM
To: nedward1@asu.edu
Subject: Fwd: I2C and PWM Ruminations..

Little more math

PLUS remember we have to split this across possibly all 16 motors at the same time

so turn on and turn off motor 1-16 takes 16*16*2 clocks=512 clocks

so we could turn on and then turn off a motor, assuming all other motors need to be touched in between in 512 clocks

our frequency=500000hz=

T=period=1/500000=2us

so we could turn on and off every motor in 2us*512 clocks
=1.024ms

we were talking about splitting our duty, 50ms bursts 25% 75% 100%.. so

it could work at about 1 ms bursts?

---------- Forwarded message ----------
From: **Jacob Rosenthal** <jakerosenthal@gmail.com>
Date: Tue, Feb 17, 2009 at 8:34 PM
Subject: Re: I2C and PWM Ruminations..
To: nedward1@asu.edu

Check some of my math. Im trying to see what the atmega168 were using can do over i2c

scl freq = cpu clock/(16 + 2*TWBR*prescaler) (from the manual)

twbr is 8 bit register
prescalers are 1, 4, 16, 64
funnel runs at 8MHz
so fastest scl freq = 8MHz/16=500,000Hz=500Khz

consider wed be doing say 8bit start and address and 8 bit data to get something done
16 clocks to transfer 8 bit

So what is our throughput then?

On Mon, Feb 16, 2009 at 3:46 PM, <nedward1@asu.edu> wrote:

> Jacob,
>
> Can you see if you can find some of these pwm controllers with 3 or 4 addr pins. Alternately may need your mux idea.
>
> I am nearly done with our proj schedule, it is going to be tight - just thinking we need to place the PIC controller/mux parts in our first order for testing along with my I2C slave.
>
> Nathan Edwards
> ---
> message sent from PalmTX
>
> -----Original Message-----
>
> From: Jacob Rosenthal <jakerosenthal@gmail.com>
> Subj: Re: I2C and PWM Ruminations..
> Date: Sat Feb 14, 2009 12:17 am
> Size: 1K
> To: Nathan Edwards <Nathan.Edwards@asu.edu>
> cc: Jon Lindsay <jonathan.lindsay@asu.edu>, Daniel Moberly <Daniel.Moberly@gmail.com>, Kris Blair <Kristopher.blair@asu.edu>
>
> Oh I agree, we certainly dont need an h bridge, just the simplest pic that includes both a pwm and i2c. Flashing would only ever need occur once for each module. Like you say, well see if the expander works.

Somewhat off topic,I suggest jst connectors or something similar to connect each vibrator unit to the belt, and the microcontroller to the belt.

JST are the same tiny molex power connector used by our battery

On Fri, Feb 13, 2009 at 11:54 PM, Nathan Edwards <Nathan.Edwards@asu.edu> wrote:
An H-bridge controller basically controls a motors speed androtational direction. They are used in almost any servo type application or robotic drive systems where you want variable speed and direction. Not sure if our motor could handle the variable speed/dircetion, but our application is much simpler in that we only need to turn on and off at certain times.