Bean (Hitt Consulting)
Registered Member

Date Joined Jul 2004
Total Posts : 3877

Posted 12/28/2006 1:09 PM (GMT -7)

If I have a program that doesn't use all of the 32K program space, can I store data in the upper end of the 32K EEPROM ?

I need about 1K on the Propeller demo board.

Can I just use the last 1K ?
Does the propeller have any checksum or anthing that will get screwed up ?
Does it store the program starting at address 0 ?

Here is the code I'm trying (Uses the i2cObject)

[edit] I have posted working code further down...

```
I2C.Init(29, 28, TRUE) ' Use Propeller Demo Board EEPROM
I2C.Start

I2C.i2cStart
I2C.i2cWrite($A0, 8)
I2C.i2cWrite(124, 8)
I2C.i2cWrite(0, 8)
I2C.i2cWrite("H", 8)
I2C.i2cWrite("E", 8)
I2C.i2cWrite("L", 8)
I2C.i2cWrite("L", 8)
I2C.i2cWrite("O", 8)
I2C.i2cStop
WAITCNT(4000000 + cnt)

I2C.i2cStart
I2C.i2cWrite($A0, 8)
I2C.i2cWrite(124, 8)
I2C.i2cWrite(0, 8)
I2C.i2cStart
I2C.i2cWrite($A1, 8)
Overlay.PutChar(I2C.i2cRead(1))
Overlay.PutChar(I2C.i2cRead(1))
Overlay.PutChar(I2C.i2cRead(1))
Overlay.PutChar(I2C.i2cRead(1))
Overlay.PutChar(I2C.i2cRead(1))
I2C.i2cStop
```

Bean.

Cheap used 4-digit LED display with driver IC www.hc4led.com
Low power SD Data Logger www.sddatalogger.com
SX-Video Display Modules www.sxvm.com
Stuff I'm selling on ebay http://search.ebay.com/_W0QQsassZhittconsultingQQhtZ-1

"USA Today has come out with a new survey - apparently, three out of every four people make up 75% of the population." - David Letterman

Post Edited (Bean (Hitt Consulting)) : 12/30/2006 5:25:22 PM GMT

Back to Top

Mike Green
Registered Member

Date Joined Oct 2004
Total Posts : 3928

Posted 12/28/2006 2:51 PM (GMT -7)

Bean,
Your basic concept is fine. The bootloader doesn't care what's beyond the end of your program in EEPROM. When you download a new program via the Propeller Tool and ask for it to be written to EEPROM, the whole 32K gets written and your saved data will get erased.

Your posted code doesn't wait for the data to be written to EEPROM before trying to write again. The EEPROM will not respond for about 5ms. Unless you use a paged write (depends on the EEPROM, but a 32 byte page size will usually work) to a page boundary and only one address select at the beginning of the page, then up to a page's worth of data, then a stop sequence. This will result in only one write cycle for the whole thing.

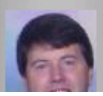
Easiest thing to do is just insert a 5ms delay after each byte written.

The program is loaded starting at EEPROM location zero and the checksum is only through the end of the program (and a few bytes beyond to the start of the stack space).

Use the "readLocation" and "writeLocation" routines in the I2C object. There's also a "writePage" routine. They do the start/address/data/stop stuff for you and are in the current version of the I2C object (1.4 or later). There's also a "devicePresent" routine that you can use instead of waiting for a write to finish. The EEPROM will appear as "present" when it's finished with the write cycle.
Mike

Back to Top

Bean (Hitt Consulting)
Registered Member



Posted 12/28/2006 5:21 PM (GMT -7)

Mike,
I downloaded version 1.3 from Parallax website. I can't seem to find 1.4 in the forums ???

Anyway, I'm still not having any luck.

Date Joined Jul 2004
Total Posts : 3877

I changed the code to this:

```
I2C.Init(29, 28, TRUE) ' Use Propeller Demo Board EEPROM
I2C.Start

I2C.WriteLocation($A0, 31744, "H", 16, 8)
WAITCNT(4000000 + cnt)
I2C.WriteLocation($A0, 31745, "E", 16, 8)
WAITCNT(4000000 + cnt)
I2C.WriteLocation($A0, 31746, "L", 16, 8)
WAITCNT(4000000 + cnt)
I2C.WriteLocation($A0, 31747, "L", 16, 8)
WAITCNT(4000000 + cnt)
I2C.WriteLocation($A0, 31748, "O", 16, 8)
WAITCNT(4000000 + cnt)

Overlay.PutChar(I2C.ReadLocation($A0, 31744, 16, 8))
Overlay.PutChar(I2C.ReadLocation($A0, 31745, 16, 8))
Overlay.PutChar(I2C.ReadLocation($A0, 31746, 16, 8))
Overlay.PutChar(I2C.ReadLocation($A0, 31747, 16, 8))
Overlay.PutChar(I2C.ReadLocation($A0, 31748, 16, 8))
```

P.S. I running at 40MHz if that matters...

Bean.

Cheap used 4-digit LED display with driver IC www.hc4led.com
Low power SD Data Logger www.sddatalogger.com
SX-Video Display Modules www.sxvm.com
Stuff I'm selling on ebay http://search.ebay.com/_W0QQsassZhittconsultingQQhtZ-1

"USA Today has come out with a new survey - apparently, three out of every four people make up 75% of the population." - David Letterman

Back to Top

Bean (Hitt Consulting)
Registered Member

Date Joined Jul 2004
Total Posts : 3877

Posted 12/29/2006 10:13 AM (GMT -7)

It seems the i2cobject version 1.3 doesn't work correctly with the propeller eeprom.
I changed the i2cStart and i2cStop code and got it working partially.

Mike, can you post a link to version 1.4. I'll try that.

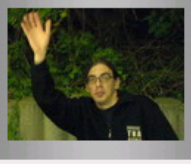
Bean.

Cheap used 4-digit LED display with driver IC www.hc4led.com
Low power SD Data Logger www.sddatalogger.com
SX-Video Display Modules www.sxvm.com
Stuff I'm selling on ebay http://search.ebay.com/_W0QQsassZhittconsultingQQhtZ-1

"USA Today has come out with a new survey - apparently, three out of every four people make up 75% of the population." - David Letterman

Back to Top

Catweazle
Registered Member



Date Joined Dec 2006
Total Posts : 17

Posted 12/29/2006 1:08 PM (GMT -7)

Hello,

I've tried the I2Cobject out. I connected a external EERPOM (24LC512) but should be the same as the 24LC256.
EEPROM adresspins A0..A2 and WP(writeprotect) are tied to GND

Port A7 = SDA (Pullup 10k to 3v3)
Port A6 = SCL (Pullup 10k to 3v3)

The Code... (using i2cObject version 1.3)

```
'-------------------------------------------------------------------------------------------------------------------------------
' init i2c driver
i2cObject.init(7,6,false) ' SDA=A7, SCL=A6 , (24LC512)

' write an "x" to adress 0
i2cObject.writeLocation($A0, 0, "x", 16, 8) ' $A0+i2cDeviceAdress, EEPROMadress, EEPROMdata, Adressbits, Databits
waitcnt(400_000 + cnt) 'wait 5ms @ 5MHz - this is very important !!!

' read adress 0
eepromdata:=i2cObject.readLocation($A0,0,16,8) ' $A0+i2cDeviceAdress, EEPROMadress, Adressbits, Databits

' eepromdata holds the data ("x") now
'-------------------------------------------------------------------------------------------------------------------------------
```

i2cDeviceAdress = 0 (A0..A2 are tied to GND, so the EEPROM device adress is 0)

- Eric

Back to Top

Mike Cook
Registered Member

Posted 12/29/2006 1:14 PM (GMT -7)

Don't know if this helps or not...

But I'm using v1.3 of the i2cObject.spin, talking to a DS1307 hanging off the same pins as the Propeller's eeprom (PINS 28 & 29). I'm using a 'Home built' demo board that has the I2C lines pulled up to +3.3VDC

So far I have not noticed any issue with the I2C driver, but like I stated I'm communicating with a RTC chip and not the Propeller's eeprom.

<EDIT>

If your using the Propeller Demo Board, it only has SDA pulled up, that might be the problem, here's a link the the Parallax Demo Board's schematic:

http://www.parallax.com/dl/docs/prod/prop/PropDemoDschem.pdf

Just an ASSUMPTION here but I think the I2C spec wants both SDA & SCL pulled up.

</EDIT>

Mike

Post Edited (Mike Cook) : 12/29/2006 8:28:01 PM GMT

Back to Top

---

sharpie
Registered Member

Date Joined May 2006
Total Posts : 144

Posted 12/29/2006 1:35 PM (GMT -7)

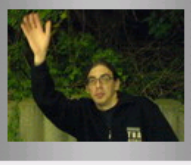
I have also had much luck with using the same eeprom my code is stored in with the folllowing...

```
rom.Init(i2cSDA, i2cSCL, false)

  if rom.isStarted == true
    txt.out($0D)
    txt.str(string("Started OK"))
  else
    txt.out($00)
    txt.str(string("Start Failed"))
  txt.out($0D)
  txt.str(string("i2c Message:"))
  txt.dec(rom.getError)
  waitcnt(5_000_000+cnt)
  if rom.devicePresent(EEPROM_Addr) == true
    txt.out($0D)
    txt.str(string("Selftest PASSED"))
    waitcnt(50_000_000+cnt)
    txt.out($00)
    outa[led] := low
    cognew(DoStuff,@stack1)
    cognew(DoMoreStuff,@stack2)
  else
    txt.out($00)
    txt.str(string("Selftest FAILED"))


Pub DoStuff
    rom.writeLocation(EEPROM_ADDR, eepromLocation, data, 16, 8)
    waitcnt(50_000 + cnt)
    eepromlocation++
```

Wouldn't work on the demo board due to the pullups, but works great every time with them on my custom boards(with the pullups).

Back to Top

Catweazle
Registered Member



Date Joined Dec 2006
Total Posts : 17

Posted 12/29/2006 1:35 PM (GMT -7)

Bean, after having a second look at your code

you have written:

I2C.Init(29, 28, TRUE) ' Use Propeller Demo Board EEPROM
I2C.Start

instead you should try:

I2C.Init(29, 28, **FALSE**) ' Use Propeller Demo Board EEPROM
~~I2C.Start~~ ' you don't need this

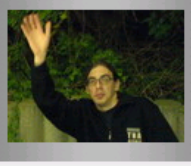
hope that work ...

- Eric

Post Edited (Catweazle) : 12/29/2006 8:41:08 PM GMT

Back to Top

Catweazle
Registered Member



Date Joined Dec 2006
Total Posts : 17

Posted 12/29/2006 1:52 PM (GMT -7)

EDIT:
the I2C.Init(29, 28, **FALSE**) works only with 10k Pullup resistor on SCL !!
At 40Mhz you need a waitcnt(200_000 + cnt)

- Eric

Back to Top

Bean (Hitt Consulting)
Registered Member

Date Joined Jul 2004
Total Posts : 3877

Posted 12/30/2006 10:18 AM (GMT -7)

I have modified some code that Beau sent me to work with the Propeller Boot EEPROM.

The attached code drives the SCL line, so don't use it for general I2C stuff. It is written specifically to read/write from/to the Propeller Boot EEPROM.

I'm using the code to read/write from/to the last 6K (26624 thru 32767) of the 32KB EEPROM on the Propeller demo board.

Don't forget that when you save code to the Propeller EEPROM (F11) it ALL gets erased.

Comments welcome, (Beau?)

Bean.

Cheap used 4-digit LED display with driver IC www.hc4led.com
Low power SD Data Logger www.sddatalogger.com
SX-Video Display Modules www.sxvm.com
Stuff I'm selling on ebay http://search.ebay.com/_W0QQsassZhittconsultingQQhtZ-1

"USA Today has come out with a new survey - apparently, three out of every four people make up 75% of the population." - David Letterman

Post Edited (Bean (Hitt Consulting)) : 12/30/2006 11:55:44 PM GMT

File Attachment :
HITT_Boot_EEPROM_010.spin 5KB (application/octet-stream)
This file has been downloaded 54 time(s).

Back to Top

Javalin
Got a Propeller, need some SPIN?

Date Joined Jul 2004
Total Posts : 402

Posted 1/2/2007 2:46 AM (GMT -7)

Chaps,

Just to clarify the earlier versions of the i2cObject did expect (following the i2c spec) that both sca/scl would be pulled up. But the current version allows you to specify when you init the object a true/false to indicate whether the scl will be pulled up.

This is explained in detail within the i2cDemo.spin & i2cObject.spin code documentation.

(Note the SDA HAS to be pulled up)

I've attached version 1.4 - its "beta" as I've been really lazy recently and haven't 100% tested it to my satisfaction. I'll get off my backside and sort it out soon. Version 1.4 does "page read" and "page write" on eeproms.

James

File Attachment :
i2cLibrary_Version1_4_beta.zip 24KB (application/x-zip-compressed)
This file has been downloaded 68 time(s).

Back to Top

djh82uk
Registered Member

Date Joined Jun 2007
Total Posts : 61

Posted Today 9:56 AM (GMT -7)

Ok Sorry if I am thick, but can someone help me use hitt's code?

I want to write 2 numbers to storage, but I don't know what variable to pass to the write and read methods?

as I how do I work them out?

Thanks

DJH

Back to Top

Mike Green
Registered Member

Date Joined Oct 2004
Total Posts : 3928

Posted Today 10:37 AM (GMT -7)

The Object Exchange has an entry for "Basic I2C Driver". In the comments at the beginning, there's a simple example of writing a small (32 byte) buffer to EEPROM and reading the same area back. If you want to write two numbers (both LONGs), then read them back, you could use the ReadLong and WriteLong routines in the object to do this. If you declared the object with 'OBJ I2C : "Basic I2C Driver"' and your variables are XXX and YYY:

```
I2C.WriteLong(I2C#BootPin,I2C#EEPROM,@XXX,@XXX)
waitcnt(clkfreq/100 + cnt)  ' Wait 10ms for write to finish
I2C.WriteLong(I2C#BootPin,I2C#EEPROM,@YYY,@YYY)
waitcnt(clkfreq/100 + cnt)  ' Wait 10ms for write to finish
```

What this does is to store XXX and YYY in the EEPROM locations where they are initialized from during a reset or power on. Normally these are cleared to zero when the program is downloaded by the Propeller Tool and you've overwritten these zero values in the EEPROM. If you don't want to use these locations, you can choose any others that are not occupied by your program and its data. The easiest thing to do is to start at the end of the EEPROM ($7FF4) and work downwards since the Propeller Tool fills the EEPROM from location zero upwards. This would look like:

```
I2C.WriteLong(I2C#BootPin,I2C#EEPROM,$7FF4,@XXX)
waitcnt(clkfreq/100 + cnt)  ' Wait 10ms for write to finish
```

```
I2C.WriteLong(I2C#BootPin,I2C#EEPROM,$7FF0,@YYY)
waitcnt(clkfreq/100 + cnt)  ' Wait 10ms for write to finish
```

To read this back in you'd need to do: XXX := I2C.ReadLong(I2C#BootPin,I2C#EEPROM,$7FF4)
or YYY := I2C.ReadLong(I2C#BootPin,I2C#EEPROM,$7FF0)

Post Edited (Mike Green) : 7/4/2007 5:41:57 PM GMT

Back to Top

djh82uk
Registered Member
Date Joined Jun 2007
Total Posts : 61

Posted Today 12:19 PM (GMT -7)

Thank you

I was hoping to write to the onboard eeprom on the proto board, will that object do it? I did not think it would. (hence the reasons for beans addition?)

I guess by the look of it beans version does not support longs? just bits? So if I used that one I would have the split the long into bits?

basically I am trying to store a max speed, say for instance 17.21MPH is made up of 2 variables, 17 and 21.

Just to make my intentions clearer

Thanks for your help so far :)

DJH

Back to Top

Mike Green
Registered Member

Date Joined Oct 2004
Total Posts : 3928

Posted Today 12:35 PM (GMT -7)

Both of the objects in the Object Exchange will work with the extra EEPROM on the Protoboard or the Hydra. On the Protoboard, the valid address range is $0000 through $FFFF. On the Hydra, the address range is $00000 through $1FFFF for the on-board EEPROM. If there's a memory expansion card installed on the Hydra, the on-board EEPROM becomes $20000 through $3FFFF and the expansion card then occupies $00000 through $1FFFF.

Bean's ReadLocation and WriteLocation routines handle single bytes only. If you want to handle words or longs, you have to do these one byte at a time.

From your description, I'm not sure how you're storing your speed value. 17.21MPH is a single value of some sort and there are all sorts of ways to store this. You could have 4 digits as ASCII characters. That still takes only one long variable (for the 4 bytes). A floating point value also takes 4 bytes. If you used a scaled fixed point value, it's still only 2 bytes (as 1721 hundreths of a mile per hour).

Back to Top

djh82uk
Registered Member
Date Joined Jun 2007
Total Posts : 61

Posted Today 2:27 PM (GMT -7)

well I do not have the cogs to do the Floating point I think, so I simply made the value (a four digit number) into a most significant bit (var1) and least significant bit (var2), each time there is a new speed I will just compare the first parts, if equal then also compare 2nd parts. But I guess i could just do all that before converting to MSB and LSB.

Thanks for the info. Do I need any pullups on the proto board?

DJH

Back to Top

Mike Green
Registered Member

Date Joined Oct 2004
Total Posts : 3928

Posted Today 3:42 PM (GMT -7)

You can easily store a 4 digit number in a 16 bit (word) variable, then do arithmetic and comparisons more naturally and efficiently.

Whether you need pullups depends on what you want to connect. The built-in stuff (and the optional I/O in the Accessory Kit) are complete as is. The EEPROM even has pullups on both the clock and data lines so it's easy to connect additional I2C devices. Do keep in mind that the Propeller is a 3.3V device and does require at least a series protective resistor (at least 220 ohm) on an input when connected to a 5V output device.

Back to Top

[<< Previous Thread | Next Thread >>]

New Topic | Post Reply | Printable

Forum Information

Currently it is Wednesday, July 04, 2007 6:01 PM (GMT -7)
There are a total of 195,855 posts in 33,367 threads.
In the last 3 days there were 84 new threads and 489 reply posts. View Active Threads

Who's Online

This forum has 9748 registered members. Please welcome our newest member, Skippy.
134 Guest(s), 6 Registered Member(s) are currently online. Details
Phil Pilgrim (PhiPi), lairdt, Basil, Diesel Stamp, rokicki, grindel