Bluegiga Forums / Community Forums / Bluetooth Smart

Port & Pin output assignments. PWM output to LED

Answered

Mike Smartt
asked this on Apr 8, 00:51

What would be the BGScript version of the following:

****

int ledPin = 9

pinMode(ledPin, OUTPUT)

for (int fadeValue = 0 ; fadeValue <= 255; fadeValue += 5) {
analogWrite(ledPin, fadeValue)
}

****

Trying to learn how the "bit mask formula" works to access each of the individual pins on the BLE113-256/121.

0 people would like this to be answered. Be the first!

Comments

Mike Smartt
Auto Flight Systems, LLC

Update: *for the record "dumbing it down" was for my benefit;) i'm not very well versed when it comes to hardware integration.*

So here's the kind of thing I'm talking about. In the BGScript dev guide page 34 (6.2.5) it shows an example of how to generate output of PWM signals. There are 4 outputs in the example which use channel 1, 2, 3, and 4. It also explains that it uses a 2 channel alternate configuration with pins "p1.1, p1.0, p0.7 and p0.6". Following the exercise I hooked up a Logic Analyzer to the 4 pins and they do in fact output at duty cycles corresponding to the value entered. (see attached image).

However, what it does not explain is how channels 1-4 equate to p1.1, p1.0, p0.7 and p0.6? I keep searching but cannot find a reference to pin assignments? Am I missing a document?

Thank you again.

Screen Shot 2015-04-07 at 10.49.41 PM.png (quick view)
Screen Shot 2015-04-07 at 10.49.52 PM.png (quick view)

April 8, 2015, 09:03

Jeff Rowberg
Bluegiga Technologies

Answer

Hello Mike,

These pin assignments come from the peripheral I/O pin reference table, which can be found in the BLE113 datasheet. See Table 3 on page 9 for detail. This same material (without hardware.xml examples) is also found in the CC254x Datasheet/User Guide.

The PWM output does not use the same kind of port/pin selection bitmasks that are used by the **hardware_io_port_...** commands dealing with digital I/O, due to the internal hardware functionality of the chipset. Those functions are more straightforward in terms of pin selection; if you require clarification on those, let me know.

April 8, 2015, 16:19

Mike Smartt
Auto Flight Systems, LLC

I think I'm better off simply using the outputs here that are working and not trying to understand the pin assignments too much. That being said hopefully they won't interfere with the I/O I still need for the MPU6050/9150.

I'm still trying to work through how to translate the incoming data to correspond to the comparator_value in real time *(which is why I used the fadeValue example above)*.

The block sending update values from the Watch is attached as reference. This is what worked in a *proof-of-concept* ... (using an ATTiny** to power the LED though). I'd like to eliminate the ATTiny all together and parse this directly in BGScript which I have NO IDEA how to do. Attaching the block used to parse the incoming RGB values on the ATTiny.

Thank you.

Support

Watch_Output.png (quick view)
ATTiny.png (quick view)

April 9, 2015, 02:37

Mike Smartt
Auto Flight Systems, LLC

Jeff please don't waste any time on this. I just about have it **_I think_** and I know you guys are dealing with a lot right now. BTW, I finally figured out (00000000 = 76543210) **_I think_**.

If I run in to any issues I'll post them here.

April 9, 2015, 15:09

Add a comment

Save comment