**Gameboy Printer with Arduino – an example by Jonathan Pittaway**

**The connection:**

Refer to Miles Burton’s documentation page for how to connect them together:

<http://milesburton.com/Gameboy_Printer_with_Arduino>

(You can use a GBA link cable like I did if you have one spare or it is easier to get hold of, you just need to file the plastic projection down so it will fit in the port.)

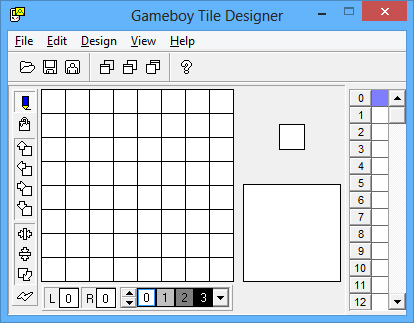
**The protocol:**

The data to be printed is sent as 16-bit tiles which fit together to form an image. The function in the example GBSerialOut() is called to send one byte at a time, where the argument is the byte in either hex or binary. Although binary format is easier to understand, the example uses hex because I used GBTD to create all of my tiles and it exports in hex by default. (<http://www.devrs.com/gb/hmgd/gbtd.html>)

Each pair of bytes defines a line:

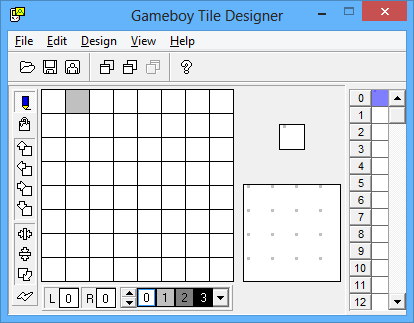
00000000

00000000



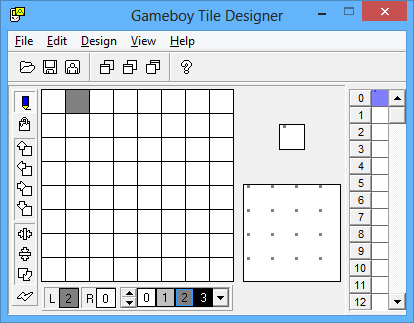
01000000

00000000



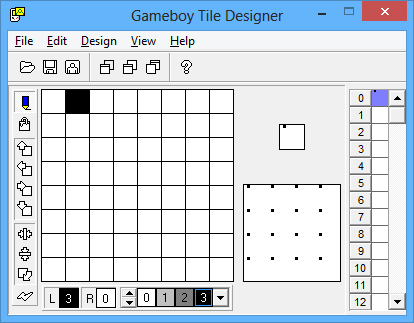
00000000

01000000



01000000

01000000



So read from top-to-bottom, the 2 red bits define a pixel, where:

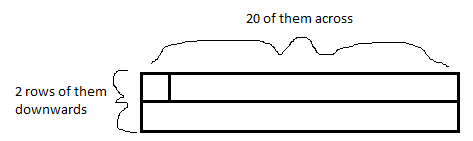
00 = white

10 = light grey

01 = dark grey

11 = black

A full tile is made up of 8 of these lines, so 16 bytes. A full line is made up of 40 tiles arranged like so:



The GB printer can hold 9 of these lines for printing. A total of 16 \* 40 \* n bytes must be sent to print where n is the number of lines. There is code in the example which keeps track of how complete a line is and pads it with zeros accordingly to make a complete line for printing.

**The sample Arduino program:**

The sample program is an adaptation of the example provided by Miles Burton on his page. It expands the range of objects you can print by making the format of the data being sent easier to understand. Tiles can be predefined as a byte array and sent to the printer when required. Further expansion of the program could allow the tiles themselves to be sent over serial, allowing for some sort of custom image printing program as opposed to simply sending text and printing that. This would also help to conserve what little RAM the Arduino has! Text to be printed should be typed into the serial monitor, prefixed with a lowercase ‘t’ and terminated with a ‘/’. So to print the word “hello” you send “thello/”. This program was tested on an Arduino Uno.

**The .exe program:**

I have written a simple program using Visual Basic 2012 that sends text over serial to the Arduino. It adds the ‘t’ and ‘/’ behind the scenes and lets you choose the COM port to use by changing the text in the lower text box, for me it was COM4.

**Credits:**

Furrtek – for his extensive documentation of the Gameboy Printer protocol which can be found here: <http://furrtek.free.fr/index.php?p=crea&a=gbprinter>

Miles Burton – for the sample code which wraps some key commands into functions and demonstrates printing: <http://milesburton.com/Gameboy_Printer_with_Arduino>