

The CoinForth Documentation

Arduino BLE HostTest Release

Version 0.1

Copyright © 2015 Dennis Ruffer

Table of Contents

[BLE-STACK 1.4.1](#)
[Draft Areas](#)
[Basic editing](#)
[Images](#)
[Program listings](#)
[Lists and tables](#)
[Localization](#)

Abstract

This started out as a Docbook example within Syntext Serna Free 4.3.0-20110207.0, which is the latest **FREE** version I found at <http://syntext-sera-free.software.informer.com/download/?ca336a2>. I've started adding my coinForth notes, with the eventual goal of becoming some useful documentation.

BLE-STACK 1.4.1

This software used to be available at <http://www.ti.com/tool/ble-stack>, but I only see **BLE-STACK-2-1** right now, since the CC2540 is pretty old at this point. I have installed it here, so all other file paths will be relative to this:

[C:\Texas Instruments\BLE-CC254x-1.4.1.43908\](#)

It requires IAR's EW8051 v9.10.3 with a full license, but all I did was change the device from **CC2540F256** to **CC2540F128** and compiled the **CC2540EM** configuration for using the P0 serial port. I saved the [HostTestReleaseCC2540.hex](#) output file, which should work better than the [CC2540_SmartRF_HostTestRelease_All.hex](#) file TI included.

TI's SmartRF Flash Programmer Ver. 1.12.7 can program those files, but both files give the following error message:

```
CC2540 - ID1188: HEX file content at address 0x26CEE exceeds chip's 128 kB flash size
```

Even IAR, when debugging, gives the following warning:

```
Warning: Possible IDATA stack overflow detected.  
To see the instruction that caused the possible overflow, choose Debug>Break and close this message box. To continue execution, just close
```

The Disassembly shows:

```
?BANKED_ENTER_XDATA:  
001F20 65 0C      XRL  A, V4  
>001F22 45 0D      ORL  A, V5
```

That is somewhere in TI's library (e.g. no source code).

I've also change the baud rate in [Projects\ble\common\npi\npi_np\npi.h](#) from 115,200 to 19,200 to fit with 328eForth v2.20's existing serial port driver.

```
#define NPI_UART_BR          HAL_UART_BR_19200
```

I have also found a "better" Project -> Options -> Linker -> Linker configuration file from:

[\Projects\ble\common\cc2540\ti_51ew_cc2540b.xcl](#) to [\Projects\ble\common\cc2540\ti_51ew_cc2540f128b.xcl](#)

However, now I get the following error when compiling:

```
Error[e16]: Segment BLENV_ADDRESS_SPACE (size: 0x1000 align: 0) is too long for segment definition. At least 0x1000 more bytes needed. The  
problem occurred while processing the segment placement command  
"-Z(CODE)BLENV_ADDRESS_SPACE= BLENV_ADDRESS_SPACE_START- BLENV_ADDRESS_SPACE_END", where at the moment of placement  
the available memory ranges were "-none-"  
Reserved ranges relevant to this placement:  
CODE:3de6d-3fb30      BANKED_CODE  
BIT:0-7              BREG  
BIT:80-97            SFR_AN  
BIT:a0-af            SFR_AN  
BIT:b8-c7            SFR_AN  
BIT:e8-ef            SFR_AN  
BIT:f8-ff            SFR_AN  
Error while running Linker
```

[\Projects\ble\SimpleBLEPeripheral\CC2540DB\SimpleBLEPeripheral.eww](#) compiles and

[\Projects\ble\SimpleBLEPeripheral\CC2540DB\CC2540F128DK-MINI Keyfob\Exe\SimpleBLEPeripheral.hex](#) can be flashed, so let's start there.

The CC2540's serial port is connected to the ATA6614Q pins PD2 (RX) and PD3 (TX), which require a "soft" serial port, like what Arduino provides in their **SoftwareSerial**. <https://www.arduino.cc/en/Reference/SoftwareSerial> and <http://arduiniana.org/libraries/newsoftserial/>.

<http://ross-arduino-projects.blogspot.com/2014/04/setting-up-coin-ble-dev-kit.html> has: Tool: Select AVRISP mkII and Device: ATA6614Q for Atmel Studio setup, but can't get the mkII to show up there yet. <https://atmel-support.force.com/customers/500G000000ohYDZ>. Fixed!

```
C:\Users\Dennis\Documents>atprogram -t avrispmk2 selftest
Firmware check OK
[ERROR] No self tests to perform for this tool. (TCF Error code: 1)
```

Under the Arduino source, burn the following bootloader:

[./hardware/arduino/avr/bootloaders/atmega/ATmegaBOOT_168_atmega328_pro_8MHz.hex](#)

Change the fuse: HIGH: 0XDA from 0XD9, which is used by eForth.

Inside Arduino 1.6.5, File -> Examples -> 01.Basics -> Blink, then Sketch -> Upload and the LED under the reset button starts blinking.

<https://github.com/sdwood68/YAFFA> Yet Another Forth For Arduino

Using Arduino's Serial Monitor @ 19200 baud, I get:

```
YAFFA - Yet Another Forth For Arduino, Version 0.6
Copyright (C) 2012 Stuart Wood
This program comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to
redistribute it under certain conditions.
```

```
Terminal Echo is On
Pre-Defined Words : 157
Input Buffer: Starts at $6AB, Ends at $70A
Token Buffer: Starts at $68B, Ends at $6AA
Forth Space: Starts at $140, Ends at $63F
315 ($13B) bytes free
>>
>>
>> words5 .
```

Looks like there is some work to do.

Then again, using my 328eForth setup in HyperAccess (or whatever you favorite terminal emulator is, setup for 19200 @ 8-None-1), it's working fine!

```
>> 5 .
5 OK
>> : junk 5 0 do [char] * emit loop ;
OK
>> junk
***** OK
```

Warning

The rest of this doc are the original contents, which I'll delete from the final doc, but for now, provide easy reference for things I want to do in my content.