PSoC 4 Pioneer Kit Community Project#023 – tinyprintf Example

We are beginning this week with a base example that will be used by our XBee projects this week. This example is called our TinyPrintf example. In this example we include a custom component called "tinyprintf" to enable simple API calls to transmit UART data up to a PC. This building block will be used by the XBee examples posted this week. The tinyprintf code was developed by SpareTimeLabs and all credits and links are included in the main.c file.

Forum Post Attachments:

At the bottom of this post we are including the following items:

- Example Project Zip File
- Zip File of Images
- Project Schematic
- Component Configurations

Components Used:

The user can download the example project at the bottom of this post. The project uses the following list of Creator Components:

- UART
- Tinyprintf

The components are configured by right clicking on the component in your Top Design schematic view and selecting ***Configure***. Please enable the following selections in the Configuration windows for the listed components above.

Firmware Description:

The main.c firmware is included in the example project. Please review the commented sections for more details.

The tinyprintf component was built off of the driver module from SpareTimeLabs. Their source code is credited in the main.c file. We used their code in the custom component.

The custom component is only a firmware component and does not use any of the hardware resources. If you remember our LED Memory example, example #014, we saw the use of a custom driver file. In that example we could have bundled the code into a component for later use. In this example we have done just that, our tinyprintf code has been added to a custom component that we can use in later examples.

In our code we see a number of API's open to use:

NewLine() – Defined in the main.c header to allow for easy newline additions in the hyperterminal window

printf() – associated to UARTPutString API for UART communication

sprint() – sends a string to a buffer and appends the string with a variable.

These API's will ease data transfer over the UART to a hyperterminal window. This custom component will be used to display data traffic over our XBee modules in our coming examples. The custom component can also be used in any example using UART communication or hyperterminal debugging.

Hardware Connections:

The only hardware connection for this example will to connect the PSoC 4 UART connections to the PSoC 5LP.

PSoC 4 **P4[1] ->** P5LP **P12[6]**

PSoC 4 **P4[0] ->** P5LP **P12[7],**

Test Your Project:

Program the Pioneer kit and then launch the hyperterminal. Once the hyperterminal is running click the Reset button on the Pioneer kit and see the displayed data on the hyperterminal window. Check the UART Configuration for the baud rate. For this example the baud rate has been set to 9600 baud.

I hope this example can help you out in your design.

http://www.element14.com/community/message/78621