

CPE 301

Embedded Systems Design Lab - Fall 2018

LAB 6 - UART

Objective:

To understand the four UART functions and to perform simple character manipulation in ANSI C. Start with the Arduino sketch code contained in Echo2C.txt. You will write the UART functions which are below. Make sure you understand and verify all bit settings in the U0init function. At this point in the course you will be using the serial port in "polled" mode and not using interrupts.

1. U0init(int ubaud) –
 - will initialize the serial port. (You are expected to fully understand what this function does and be able to initialize the U(S)ART in different modes.)
2. U0kbhit(void)
 - Check the RDA status bit, and return True if the bit is set, return False if the bit is clear.
3. U0getchar(void)
 - Return the character which has been received by the UART.
4. U0putchar(unsigned char U0pdata)
 - Wait until the serial port TBE status bit is high
 - Then take the character U0pdata and write the character to the transmit buffer

Procedure:

1. Write, compile, download to the Arduino SBC and test the keyboard program starting with the Echo2C.txt file, adding code you write for the three functions described above. **Thoroughly** test this portion of your code before moving onto step 2.
2. Copy your tested and working Echo2.ino text to a new sketch Echo3C.ino with the following changes.
 - For each key pressed the program will send back a total of five ASCII characters which will print on the terminal the hexadecimal value of the ASCII code for the key pressed formatted as the string "0xXX\n". Where XX are the two ASCII characters which represent the digits of the hexadecimal value of the input character.
 - For example, if the user presses enters the character "1" the program will send back "0x31\n"

NOTE: ALL programs written for this course need to be documented in a complete manner which will make sense to you if you need to go back and review the program a year or two from now. You need to include your NAME and a revision number on ALL programs as well.