

**UNC Charlotte, Department of Electrical and Computer Engineering
ECGR 4101/5101, Fall 2008, Homework #4**

Due: 10/6/2008, at the beginning of class (20 points)

Assignment should be typed, a hard copy turned in to instructor

1. 5 pts: Read the article “Content Collection and Display” from the September issue of Circuit Cellar Ink.
 - a. How many and what types processors are used in the device.
 - b. What programming language is used.
 - c. What operating system(s) is(are) used to run the device.
2. 15 pts: Consider the following listing of a program compiled from C. It includes the original C source code as comments for clarity. Show the contents of the stack and registers R0, R1, SP and FB just before the `exitd` in `$f2` executes. Also identify what each byte represents (e.g. LSB (least significant byte) of argument, MSB of variable, dynamic link, etc.). Assume that execution starts at the beginning of `f1`, at which point `FB` is `1104h` and `SP` is `1100h`. Assume that the address of the `mov.w` instruction following `jsr $f2` is `ff018h`. Use `??` to indicate values which are unknown or in boxes which are not used.

```
### # FUNCTION f2
### # FRAME      AUTO (      arg)      size  2, offset -2
### # REGISTER ARG  (      arg)      size  2, REGISTER R1
### # ARG Size(0)  Auto Size(2)      Context Size(5)
### # C_SRC :    int f2(int arg) {
$f2:
    enter      #02H
    mov.w      R1,-2[FB] ; arg arg
### # C_SRC :    return arg+5;
    mov.w      -2[FB],R0 ; arg
    add.w      #0005H,R0
    exitd

### # FUNCTION f1
### # FRAME      AUTO (      b)      size  2, offset -2
### # FRAME      AUTO (      a)      size  2, offset -2
### # ARG Size(0)  Auto Size(2)      Context Size(5)
### # C_SRC :    void f1(void) {
_f1:
    enter      #02H
### # C_SRC :    int a=10, b;
    mov.w      #000aH,-2[FB] ; a
### # C_SRC :    b = f2(a);
    mov.w      -2[FB],R1 ; a
    jsr        $f2
    mov.w      R0,-2[FB] ; b
### # C_SRC :    }
    exitd
```

Address	Contents	Description
10f3		
10f4		
10f5		
10f6		
10f7		
10f8		
10f9		
10fa		
10fb		
10fc		
10fd		
10fe		
10ff		
1100		

Register	Contents
R0	
R1	
SP	
FB	

- 3.
- 4.