Design Assignment 1

ELC 411

Jacob Levine

Matthew Strickland

1. Learn about PSoC registers
   1. 0xe000ed00 is the address of the nvic\_cpuid\_base register
   2. Sub-field name [startbit:endbit] of subfields of nvic\_cpuid\_base
      1. IMPLEMENTER [31:24]
      2. VARIANT [23:20]
      3. Constant [19:16]
      4. PARTNO [15:4]
      5. REVISION [3:0]
   3. 0x40005086 is the address of PRT6\_DR
2. Write C code to do two things – see main.c in da1
3. Single step through code – final values for string and nvic\_cpu\_base extractions:

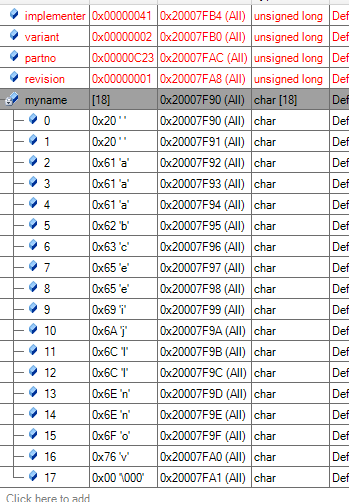


Figure 1- Jake Levine's sorted array

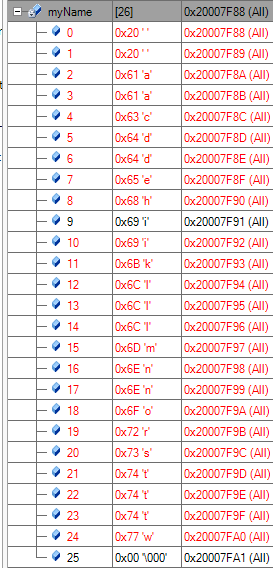


Figure 2- Matt Strickland's sorted array

All characters are correctly sorted, and variables have been correctly taken from the register.

1. Questions
   1. In C, how would you extract the value of a register field occupying bits [5:3] from a register stored in a variable ‘regval’?
      1. Create a new variable to store the field in
      2. Assign the new variable the value of regval shifted with the correct number of bits, in this case 4, bitmasked (AND operator) with the 3 bit value 0x7 (111)2.
   2. How would you modify regval… in the range 0..7?
      1. Left shift fieldval by 4 (to align with bits 5-3 of regval)
      2. Perform AND operator on fieldval AND 0xFFFFFFFF (making all bits 1 unless fieldval bits are 0)
      3. Set reg\_val equal to the bit wise AND operator of reg\_val and fieldval

Pseudo code:

//Occupy bits 5:3 of regval with integer stored in fieldval

uint32\_t fieldval = value between 0 and 7;

fieldval = (fieldval << 4); //aligns fieldval with 5:3 of regval

fieldval = fieldval & 0xFFFFFFFF //makes all bits 1 except original 3 bits

regval = regval & fieldval //bits 5:3 of field val now overwrite bits 5:3 of regval

* 1. Why do the space characters get sorted to the start of the array?
     1. Space characters have a lower ascii value than the alphanumeric characters.
  2. Briefly explain the function of the register PRT6\_DR
     1. PRT6\_DR is used to provide data output to the GPIO pins on the development board.

1. Code:

//#include "project.h"

#include <stdio.h>

#include <string.h>

#include <ctype.h>

#include <stdint.h>

int main(void)

{

// 4.1 ACCESS REGISTER VIA ITS ADDRESS

//reading value of nvic cpuid base addrs

#define NVIC\_CPUID\_BASE\_ADDR 0xE000ED00

//declaring variables

uint32\_t implementer;

uint32\_t variant;

uint32\_t partno;

uint32\_t revision;

uint32\_t reg\_val;

/\*declaring pointer to volatile unsigned int and setting it to point to nvic cpuid base addrs\*/

uint32\_t volatile \* my\_reg\_ptr = (unsigned int \*) NVIC\_CPUID\_BASE\_ADDR;

//getting value in register

reg\_val = \*my\_reg\_ptr;

//bitmasking variables

implementer = (reg\_val >> 24) & 0xFF;

variant = (reg\_val >> 20) & 0xF;

partno = (reg\_val >> 4) & 0xFFF;

revision = (reg\_val) & 0xF;

// 4.2 MANIPULATE A CHARACTER STRING

//declaring string titled ‘myname’

char myname[]= "jacob alan levine";

//sorting

int string\_length = strlen(myname);

char temp;

int i;

int j;

/\*nested for loop tests each character of myname[] and compares the ascii value to the next character, if greater than, char and char+1 are swapped\*/

for(j = 0; j<(string\_length-1); j++)

{

for(i = 0; i<(string\_length-1); i++)

{

if (myname[i]>myname[i+1])

{

temp = myname[i];

myname[i] = myname[i+1];

myname[i+1] = temp;

}

}

}

for(;;)

{

}

}