Reg. No.										

MANIPAL INSTITUTE OF TECHNOLOGY
MANIPAL
A Constituent Institution of Manipal University

SEVENTH SEMESTER B.Tech. (E & C) DEGREE END SEMESTER EXAMINATION
NOV 2017
SUBJECT: ADVANCED EMBEDDED SYSTEM DESIGN (ECE - 4001)

TIME: 3 HOURS **MAX. MARKS: 50**

Instructions to candidates
- Answer **ALL** questions.
- Missing data may be suitably assumed.

1A. Assume the features of an embedded system are given in the form of technical tasks *T0, T1, T2, T3, T4* and *T5*.
   i. Explain the concept of application mapping and activity scheduling in a typical embedded system design flow.
   ii. Map the application shown as task graph in **Fig.1.1** to suitable processing elements given in **Table.1.1** to finish the whole execution before 15 msec. (multiple instances of same processing element is not allowed).
   iii. Draw the activity scheduling graph for the application mapped in above question.

1B. Write a minimum of two differences among the following
   i. Hard processor and soft processor
   ii. Zynq SoC and Discrete FPGA-Processor combination
   iii. General Purpose processor and Digital Signal processor

1C. Out of PS and PL of Zynq SoC, which one suits best to implement the 2-D Discrete Cosine Transform (DCT) function as shown below, give and explain the reason.

$$F(u,v) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)\cos\frac{\pi(2x+1)u}{2M}\cos\frac{\pi(2y+1)v}{2N}$$

(5+3+2)

2A. A system whose functionality is as shown in **Fig.2.1** is considered to implement using CY8CKIT-044.
   i. List all the necessary components to place in *.cysch* file and explain the configuration of each.
   ii. Explain the configuration that need to be done on *.cydwr*
   iii. Write a C code to fill *main.c* file with given functionality (A few PSoC Macros for WDT are also given in **Fig.2.1**, use if required).

2B. Assume the Railway reservation system has developed using PSoC, its software contains two infinite loop tasks named *Ticket_Book_Task*( ) and *Ticket_Cancel_Task( )* for booking and cancelling the tickets respectively. Consider that a global variable *Tickets* of int type is already declared to hold the number of available tickets. Write possible C code for each task and explain the importance of using CyEnterCriticalSection and CyExitCriticalSection APIs in your code.

2C. What is dynamic reconfiguration of PSoC? Explain with an example.

(5+3+2)

3A. From the component catalogue of PSoC Creator, assume we have selected one *DigitalIn*, two *DigitalOut* components and named them as SW, LED_GREEN, LED_RED respectively. The two *DigitalOut* pins are configured in *Strong drive mode* with initial drive state as *LOGIC ZERO* as shown in **Fig.3.1**. When working with CY8CKIT-044, if the code in *main.c* is as shown in **Fig.3.2** then write and explain in detail the status of LED_GREEN and LED_RED pins under the following drive conditions of SW pin when it is not connected to any digital value externally

i. Drive mode: *Resistive pull up*, Initial drive: *'1'*.
ii. Drive mode: *Resistive pull down*, Initial drive: *'0'*.
iii. Drive mode: *Strong*, Initial drive: *'1'*.
iv. Drive mode: *Open drain drives high*, Initial drive: *'1'*.
v. Drive mode: *Open drain drives low*, Initial drive: *'0'*.

3B. What is Bootloader? Draw and explain Bootloader function flowchart.

3C. What is the maximum size of WDT1 and WDT2 counters of PSoC 4200M device's watch dog timer and explain difference in their interrupt generation mechanism.

(5+3+2)

4A. List all the power saving modes of PSoC and write the status of CPU, SRAM, UDBs, GPIOs in each mode. With transition diagram, explain how to enter into each power mode along with their wakeup sources.

4B. Write Verilog description to create a 4-bit counter custom component (as shown in **Fig. 4.1**) using PSoC PLDs.

4C. Assume a Bootloader is loaded into flash memory of PSoC 4200M. Explain two ways to make CPU execute it.

(5+3+2)

5A. Assume the schematic in *.cysch* file of PSoC Creator is as shown in **Fig.5.1.**

i. Choose values of R1 and R2 to give an opamp gain of 11
ii. Write C code to convert analog input to digital and send the result to PC using UART.

5B. Explain the configuration of Instruction, Registers, Input and Output windows of DP element of PSoC Creator's UDB Editor while implementing the state diagram shown in **Fig.5.2**.

5C. Draw the following

i. PSoC PLD structure and
ii. UDB Datapath

(5+3+2)

Task	**CPU1**	**CPU2**	**ASIC**
	T(ms)	*T(ms)*	*T(ms)*
T0	2	3	1
T1	2	4	6
T2	4	3	8
T3	4	9	5
T4	6	8	5
T5	8	6	9
$\Upsilon_{0,1} = \Upsilon_{0,2} = \Upsilon_{2,4} = \Upsilon_{3,5}$= 1 ms			

T(ms)*-Time taken by each task to execute(in milli seconds).*

Table 1.1

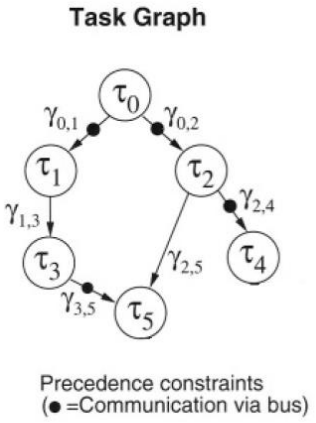

Fig.1.1

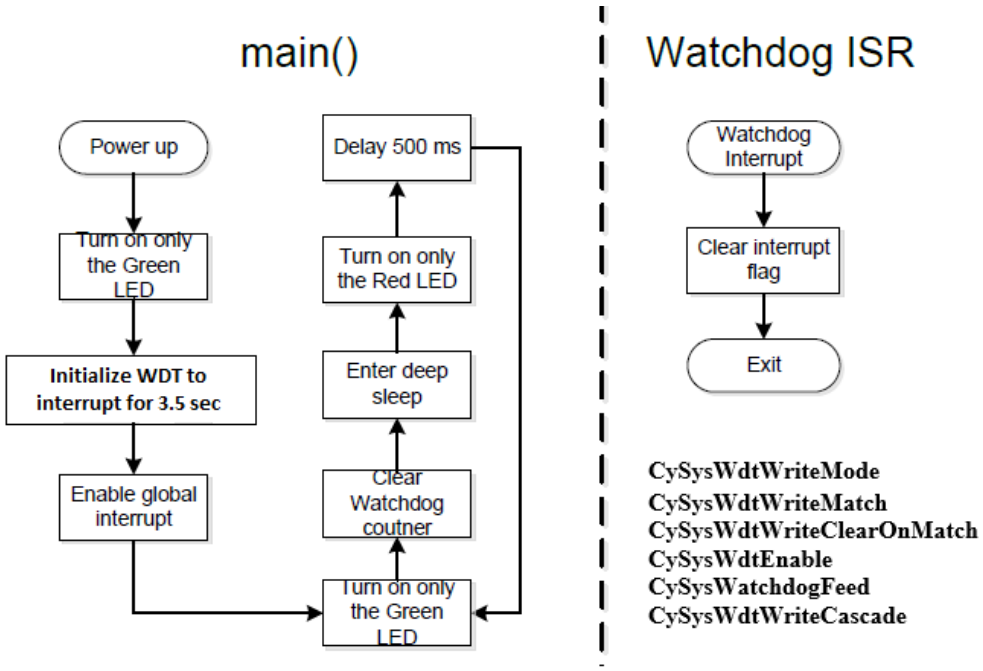

Fig.2.1

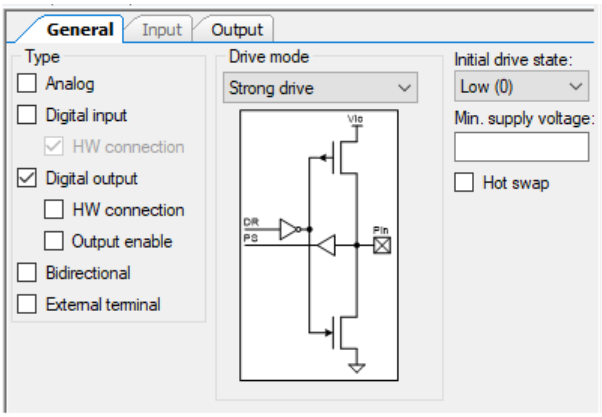

Fig.3.1

```
for(;;)
  {
    if (SW_Read()==1)
      {
      LED_GREEN_Write(~LED_GREEN_Read());
      CyDelay(500);
      }
    else if(SW_Read()==0)
      {
      LED_RED_Write(~LED_RED_Read());
      CyDelay(500);
      }
  }
```

Fig.3.2

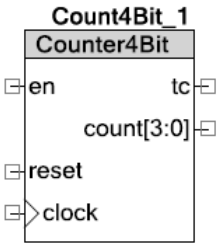

Fig.4.1

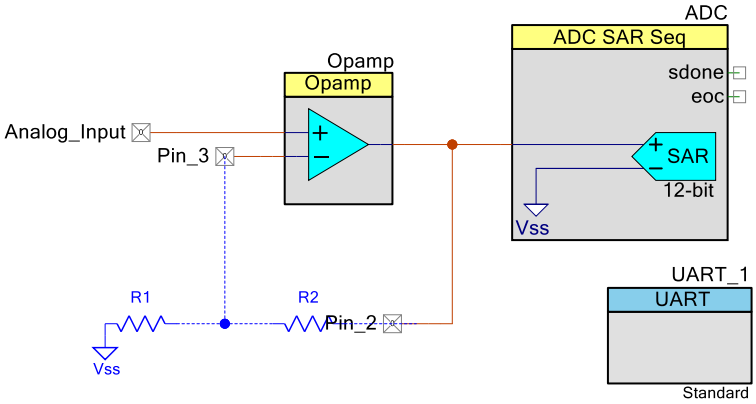

Fig.5.1

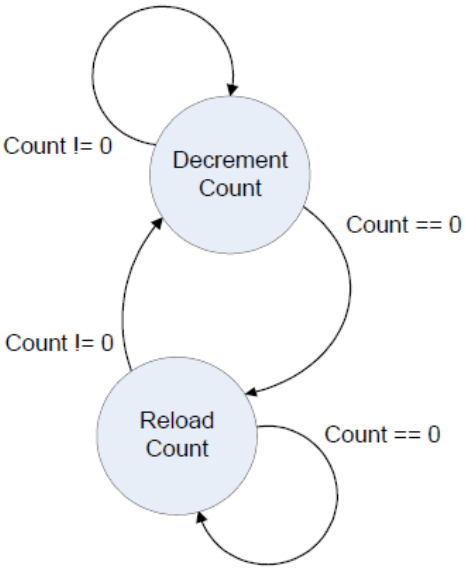

Fig.5.2