

Instituto Tecnológico de Costa Rica
Área Académica de Ingeniería en Computadores
Introducción a los Sistemas Embebidos
Prof. Ing. Jeferson González Gómez
II Semestre 2018
Valor 50 puntos.
Fecha de entrega: miércoles 14 de noviembre de 2018.

Nombre: _____ Carné: _____

Nombre: _____ Carné: _____

Examen Final

Instrucciones Generales

- Trabaje individual o en parejas.
- La modalidad de entrega de este examen es a través del TecDigital.
- Cada estudiante debe subir una solución a este examen, aunque el trabajo se realice en parejas.
- Adjunte este enunciado como parte de su solución.

I Parte. Respuesta breve. 12 puntos

A continuación, se le presentan 12 preguntas de **respuesta corta**. Responda cada una de ellas de manera correcta. (1 punto c/u)

1. Mencione 3 características de los sistemas embebidos.
2. Mencione 2 etapas del codiseño de sistemas embebidos.
3. Lucía trata de portear el paquete opencv para raspberry pi zero, utilizando Yocto. Al realizar el comando `bitbake rpi-basic-image`, tiene el siguiente error: “Nothing provides opencv”. ¿Cómo podría solucionarle el error a Lucía?
4. ¿A qué comando, en espacio de usuario, corresponde el método `module_init(void)`?
5. Mencione 2 métodos de compilación en sistemas embebidos.
6. En el comando `$(CC) -o this iswrong.cpp -I../include -L../lib -lsomething` ¿Cual es el nombre de archivo completo de la biblioteca dinámica enlazada?
7. Escriba un comando (bash) que establezca el macro del compilador a `gcc`.
8. Mencione 3 niveles de abstracción en el diseño de sistemas embebidos.
9. Mencione 3 tipos de modelos en el diseño de sistemas embebidos
10. ¿Qué es un toolchain?

11. Al tratar de compilar un código fuente, Juanito tiene el siguiente error “undefined reference to ``sqrt'`”. Juanito dice que “*ya incluyó math.h*”, pero el error no se corrige. ¿Cómo solucionaría el problema que tiene Juanito?

12. ¿Cuál es la diferencia principal entre una biblioteca estática y una dinámica?

II parte. Desarrollo (33 puntos)

A continuación, se le presentan 2 problemas de desarrollo, los cuales se deben contestar con el mayor nivel de detalle posible, mostrando todos los pasos requeridos para llegar a la solución. Se deberá enviar un archivo con las respuestas de los ejercicios, así como el archivo de programación correspondiente. Puede trabajar individual o en parejas si así lo desea. De ser en parejas, debe asegurarse que el documento lleve el nombre y carné de ambos integrantes.

1. El código *exam.c* presenta una descripción del comportamiento de un sistema embebido a diseñar, para la adquisición, filtrado y procesamiento de una señal analógica de sonido. A partir de dicha descripción:
 - a) Realice **un modelo (gráfico) de alto nivel** del código utilizando algún modelo de computación (redes de procesos de Kahn, por ejemplo). No es necesario que muestre la descripción funcional de cada proceso. (3 puntos)
 - b) **Muestre detalladamente los pasos de una síntesis manual**, para obtener un modelo estructural de un sistema que pueda implementar dicha aplicación. Asuma que seguirá una metodología de diseño basado en plataforma, para un dispositivo Altera Cyclone® V SE 5CSEMA4U23C6N (ver Altera De0-Nano-SoC, como referencia). Puede utilizar componentes básicos de sistema, para dicha descripción (memorias, buses, procesadores, sistemas operativos, bloques de IP, HW específico, etc), siempre y cuando estén disponibles de una forma u otra en el dispositivo. Muestre el diagrama del modelo estructural del sistema. Nota: como restricción al diseño, para aumentar el desempeño, se deberá utilizar más de un elemento de procesamiento. (20 puntos)

2. Parte del proceso de encodificación JPEG para frames de imágenes en sistemas embebidos, corresponde al algoritmo de *run-zero coding* (RZE). En este tipo de encodificación, ante un arreglo de elementos discretos, proveniente de un FIFO, se deberá generar un arreglo salida que contenga, para cada número natural x (no incluye al 0), la cantidad de ceros que haya hasta x , seguido del número entero x correspondiente.

Por ejemplo,

$$\text{in} = [0, 0, 1, 0, 3] \rightarrow \text{out} = [2, 1, 1, 3]$$

$$\text{in} = [1, 2, 3] \rightarrow \text{out} = [0, 1, 0, 2, 0, 3]$$

$$\text{in} = [0, 0, 0, -1, 255, 3] \rightarrow \text{out} = [3, -1, 0, 255, 0, 3]$$

Realice un modelado del algoritmo de **encodificación** RZE (mostrado arriba), utilizando redes de procesos de Kahn. Incluya en su modelo una representación gráfica (5 puntos) , una descripción de cada proceso (5 puntos) y una representación ejecutable (descrita en Python) completamente funcional del algoritmo.(5 puntos) .