



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

IIC2343 – Arquitectura de Computadores (I/2018)

Proyecto Semestral

Implementación y programación de un computador básico

1. Motivación

El objetivo principal de este curso es que el alumno entienda qué es un computador, conozca cuáles son sus partes y sus funciones, y realizar programas para el computador. Para lograr adquirir estos conocimientos y habilidades se requiere por una parte adquirir el conocimiento teórico y conceptual de lo que es un computador, pero también ser capaz de llevarlo a la práctica. El proyecto semestral corresponde a la parte práctica del curso, en la cual los alumnos deberán implementar un computador básico completo y desarrollar programas ocupando su computador.

2. Descripción

El proyecto consiste, en su primera etapa, en implementar un computador básico a nivel de hardware, utilizando los conocimientos vistos en clases. Dado que implementar todos los componentes del computador directamente con componentes físicos supera las habilidades enseñadas en el curso, se utilizarán FPGAs, que son componentes de hardware que permiten programar en software los circuitos para luego traspasarlos automáticamente al hardware y emularlos.

En particular, se ocupará para desarrollar el proyecto la placa de desarrollo Basys3 de Digilent, la cual cuenta con un FPGA Xilinx Artix-7, una pantalla LED, botones e interruptores. Esta placa puede ser programada mediante el ambiente de desarrollo Vivado WebPack de Xilinx usando el lenguaje VHDL, a través del cual se deberán programar los componentes del computador, para luego enviarlos a la placa, que se encargará automáticamente de generar las conexiones necesarias para construir la representación física de dichos componentes. De esta forma, los alumnos trabajarán a nivel lógico sólo con elementos de software, pero podrán probar sus componentes con hardware real.

El proyecto consistirá en entregas incrementales y tareas que permitirán a los alumnos construir poco a poco el computador, a la par con la materia vista en el curso y permitirán comprender otros conceptos no evaluados en las entregas incrementales. Una vez que se complete un computador funcional se agregará el elemento de programación del computador, para lo cual habrá su disposición programas que puedan ser ejecutados en este, cumpliendo distintas funciones según sean solicitadas. Estos programas irán aumentando en dificultad a través de las entregas, según las nuevas capacidades que se vayan agregando en paralelo a el computador. Al llegar a la última entrega, se añadirá un presentación que tendrá el objetivo de mostrar lo que han aprendido.

Al finalizar el semestre los alumnos habrán desarrollado un computador enteramente funcional, con la mayor parte de los elementos fundamentales vistos en el curso.

3. Metodología

El proyecto será realizado en grupos de **4 a 5 alumnos** seleccionados de manera aleatoria, y estará dividido en **3 entregas** prácticas incrementales y **2 tareas**, que son **obligatorias**. Además, ambas notas (tanto proyecto como tareas) son **reprobatorias** cada una por separado, por lo que es fundamental que se avance lo solicitado en todas las entregas para no quedarse atrás. En cada entrega se solicitará enviar los archivos de los componentes de hardware implementados o archivos de la tarea, y un informe evaluado. Todos los archivos solicitados deben estar presentes en la rama Master de un repositorio Git que se les asignará en cada entrega.

Al comenzar el semestre se le entregará a cada grupo una placa de desarrollo, que podrán usar fuera de la universidad. **La placa debe ser tratada con sumo cuidado para evitar dañarla**, y deberán devolverla al final del semestre en las mismas condiciones en que fue entregada.

A cada grupo se le asignará un repositorio privado de Github que usarán para entregar sus proyectos. Por eso, es necesario que envíen sus usuarios de Github a los ayudantes (<https://goo.gl/forms/ESEUWx3WrejEt6p32>). En cada entrega, se descargará la versión más cercana a la hora de entrega que no pase la fecha límite. Si algún grupo quiere corregir una versión que supere esta fecha, deberá indicarlo con un mail o en un issue del repositorio aceptando el descuento por atraso (5 décimas por hora o fracción).

También se asignará a cada grupo un ayudante de proyecto que se encargará del seguimiento del grupo, responder dudas, evaluación, correcciones y asignación de bonus o penalizaciones.

La corrección del proyecto **NO** será presencial, sino que consistirá en que cada grupo entregue en su repositorio los archivos `.bit` requeridos en cada entrega. Las tareas también deberán ser entregadas de la misma forma.

En algunas entregas deben lograr correr distintos algoritmos en su placa; en esos casos, algunas horas antes de la entrega se subirán archivos al Siding para que generen los archivos `.bit` asociados. No se corregirán entregas fuera de plazo, pero se puede optar a una corrección en la semana siguiente con un descuento asociado.

4. Evaluación

Cada entrega del proyecto se evaluará de forma grupal y se ponderará por un porcentaje de coevaluación para calcular la nota de cada alumno.

Dado lo anterior, dentro de las primeras **24 horas** posteriores a cada entrega, **todos los alumnos** deberán completar de forma **individual y obligatoria** el formulario web que los ayudantes pondrán a su disposición, repartiendo un máximo de 4 puntos (aunque cambia según la cantidad de estudiantes en el grupo; el máximo corresponde al número de compañeros), con hasta un decimal, entre sus compañeros y una diferencia máxima entre el mayor y menor puntaje de 1. La suma de todos los puntos obtenidos por el integrante, *sp*, será utilizada para el cálculo de la nota de cada entrega, lo que puede hacer que este repruebe el curso.

La nota de cada entrega se calcula de la siguiente forma (con un máximo de 7,5):

$$NotaEntrega_{individual} = \min(k_g \times NotaEntrega_{grupal}, NotaEntrega_{grupal} + 0,5)$$

donde,

$$k_g = \frac{sp+3}{7}$$

y sp puede variar de acuerdo a la cantidad de estudiantes que conforman el grupo.

Los alumnos que no cumplan con enviar la coevaluación en el plazo asignado tendrán un **descuento de 0.5 puntos** en su nota de la entrega correspondiente.

La nota individual del proyecto se calcula de la siguiente forma:

$$NotaProyecto_{individual} = E_1 \times 0,2 + E_2 \times 0,4 + E_3 \times 0,4 \quad NotaTarea_{individual} = T_1 \times 0,5 + T_2 \times 0,5$$

5. Planificación

El calendario tentativo de las entregas y sus ponderaciones se presentan a continuación:

- Entrega 1: Entrega: Viernes 13 de Abril a las 23:59 horas. (**20 %**)
- Entrega 2: Entrega: Viernes 04 de Mayo a las 23:59 horas. (**40 %**)
- Entrega 3: Entrega: Viernes 25 de Mayo a las 23:59 horas. (**40 %**)
- Tarea 1: Entrega: Dos semanas despues de pasar en la cátedra subrutinas en Assembly básico.
- Tarea 2: Entrega: Dos semanas despues de pasar en la cátedra memoria virtual y caché.

6. Contacto

Cualquier pregunta sobre el proyecto, ya sean de enunciado, contenido o sobre aspectos administrativos deben comunicarse con los ayudantes creando issues en el Syllabus del Github del curso o directamente con los ayudantes:

- Francesca Lucchini: flucchini@uc.cl
- Felipe Pezoa: fipezoa@uc.cl
- Hernán Valdivieso: hfvaldivieso@uc.cl
- Luis Leiva: lileiva@uc.cl

7. Integridad académica

Los alumnos de la Escuela de Ingeniería de la Pontificia Universidad Católica de Chile deben mantener un comportamiento acorde a la Declaración de Principios de la Universidad. En particular, se espera que mantengan altos estándares de honestidad académica. Cualquier acto deshonesto o fraude académico está prohibido; los alumnos que incurran en este tipo de acciones se exponen a un Procedimiento Sumario. Es responsabilidad de cada alumno conocer y respetar el documento sobre Integridad Académica publicado por la Dirección de Docencia de la Escuela de Ingeniería.

Específicamente, para los cursos del Departamento de Ciencia de la Computación, rige obligatoriamente la siguiente política de integridad académica. Todo trabajo presentado por un alumno para los efectos de la evaluación de un curso debe ser hecho individualmente por el alumno, sin apoyo en material de terceros. Por

“trabajo” se entiende en general las interrogaciones escritas, las tareas de programación u otras, los trabajos de laboratorio, los proyectos, el examen, entre otros. Si un alumno copia un trabajo, obtendrá nota final 1,1 en el curso y se solicitará a la Dirección de Docencia de la Escuela de Ingeniería que no le permita retirar el curso de la carga académica semestral. Por “copia” se entiende incluir en el trabajo presentado como propio partes hechas por otra persona.

Obviamente, está permitido usar material disponible públicamente, por ejemplo, libros o contenidos tomados de Internet, siempre y cuando se incluya la referencia correspondiente.

Lo anterior se entiende como complemento al Reglamento del Alumno de la Pontificia Universidad Católica de Chile. Por ello, es posible pedir a la Universidad la aplicación de sanciones adicionales especificadas en dicho reglamento.