



CI-0119 Proyecto Integrador de Arquitectura y Ensamblador

Proyecto del curso
2-2020

Motivación

Podemos encontrar microprocesadores en casi cualquier dispositivo electrónico, desde un “lápiz inteligente” hasta un cohete especial (que tendrá más de uno por supuesto). Algunos de estos procesadores son de propósito general, como los que utilizamos en las computadoras diariamente. Otros, están diseñados para un fin específico. Independientemente de cuál sea el propósito del procesador, todos tienen una arquitectura y un conjunto básico de instrucciones. La arquitectura y el conjunto de instrucciones requieren un diseño tanto a nivel de hardware como de software.

Considerando lo anterior, este semestre simularemos un equipo dedicado a monitorear medidas de seguridad a la entrada de un edificio. Este equipo deberá verificar 3 situaciones: personas usan o no mascarillas, temperatura de cada persona y distancia entre personas. Todo será simulado, desde la entrada de datos (de las cámaras), hasta la arquitectura del equipo que será especialmente diseñado para este propósito.

Monitor de espacios públicos

En el contexto de la pandemia en la que vivimos hay medidas de seguridad sanitaria que deben extremarse con el objetivo de contener el virus y evitar así la transmisión en la medida de lo posible. Espacios públicos en ciertos escenarios (aeropuertos, hospitales, mercados, etc.) son especialmente vulnerables. Haciendo uso de estrategias de *deep learning*, cámaras infrarrojas y otro equipo especializado es posible monitorear las 3 situaciones que se mencionaron anteriormente.

Como proyecto de este curso, lo que se desea es:

1. Diseñar una solución tecnológica que permita determinar en un espacio público el uso de mascarillas, la temperatura por persona y la distancia entre personas. Esta solución debe incluir:
 - a) Cuáles son los componentes físicos para implementar la solución (¿cuántos registros, de qué tamaño?, ¿cuánta memoria y para qué se usará?, ¿habrá circuitos integrados involucrados?, ¿qué hacen esos circuitos y cuáles son las entradas y cuáles las salidas?, ¿cómo todos estos componentes interactúan entre sí?, etc.). Ejemplo: puede decidirse que hay un registro que almacena la cantidad de personas presentes en un momento dado que se llamado np (número de personas) con un tamaño de 4 bits (suponiendo un máximo de 16 personas por cuadro).





- b) Cuáles son los datos necesarios por persona para hacer el monitoreo respectivo (puede por ejemplo suponer que existen dispositivos que ya devuelven los datos que se necesitan para hacer los cálculos necesarios). Estos datos pueden venir de cámaras infrarrojas simuladas, de dispositivos de IoT simulados, etc.
 - c) Cómo se presentarán los datos (¿habrá una alarma sonora por cada problema detectado?, ¿o será presentado en pantalla?, ¿cómo saber cuál es la persona que está incumpliendo alguna de las medidas sanitarias?, etc.).
 - d) Cuáles son las limitaciones de la solución. Ejemplo: no habrá más de x cantidad de personas al mismo tiempo, las personas caminan en una sola dirección (¿o pueden caminar en ambas direcciones?), entre otros.
2. Implementar la solución diseñada en el punto anterior. Esta implementación debe considerar lo siguiente:
- a) Cualquier cálculo matemático requerido para calcular distancia entre personas y temperatura corporal deberá implementarse en lenguaje ensamblador.
 - b) Cualquier registro, componente, circuito integrado, cámara infrarroja, etc. que se contemple en el diseño, deberá ser simulado en lenguaje ensamblador. Por ejemplo, en la sección de datos (.data) deberá reservarse espacio para almacenar el registro np mencionado anteriormente.
 - c) La interacción directa con el usuario, incluyendo el obtener los datos para que sean usados por los componentes electrónicos y la salida del sistema (alarmas sonoras, imágenes, texto, etc.), será implementada en un lenguaje de alto nivel (C/C++).
 - d) Cualquier detalle de implementación no mencionado en los puntos 2a, 2b y 2c, podrá a programarse en alto o bajo nivel.

Detalles de logística

Considerando lo anterior, los puntos 1a y 1b deberán ser determinados en consenso por todo el grupo (no por equipo). Todo el resto de los puntos (1c, 1d y todo el punto 2) deberán ser definidos e implementados por cada equipo.

Deberán hacerse 3 entregas mayores por equipo a lo largo del semestre:

1. Propuesta de diseño
2. Avance de prototipo
3. Prototipo completo

En cada una de las entregas habrá 2 evaluaciones: 1 evaluación grupal y 1 evaluación individual. Las evaluaciones grupales corresponden a la completitud y calidad de lo presentado. Las evaluaciones individuales corresponderán a una serie de preguntas que se realizarán en horas de clase





con el objetivo de que cada miembro del equipo tenga la oportunidad de demostrar su participación durante el desarrollo de cada etapa.

Adicionalmente, se asignarán tareas cortas durante el transcurso del semestre que permitan la completitud de las entregas.

Con cierta regularidad se revisará y calificará el uso adecuado del repositorio en github. Todos los documentos, código y demás materiales asociados al proyecto deberán almacenarse en ese repositorio.

Las fechas de entrega y los alcances del avance serán definidos por el grupo el día de la entrega de este enunciado. Fechas que se deben definir son:

- Entrega de diseño por equipos (evaluación grupal 1 y evaluación individual 3)
- Entrega de diseño consensuado (tarea corta)
- Entrega de avance (evaluación grupal 2 y evaluación individual 3)
- Entrega de prototipo (evaluación grupal 3 y evaluación individual 3)

Fecha recomendada de diseño por equipos: no más allá de un mes después de discutido este enunciado.

