

Fernando Mendoza



TECNOLÓGICO NACIONAL DE MÉXICO.

INSTITUTO TECNOLOGICO DE OAXACA



Alumnos:

Mendoza Martínez Ángel Fernando

Ruiz Páez Roxana

Núñez Pérez Adriana

Cisneros Hernández Axel Heydrich

Cortés Guzmán Vanessa

López López César

***32bits y su relación con Arduino, cómo se ha desplazado, diferencias con 64bits y su salto a ARM.***

Materia: Lenguajes de Interfaz

Carrera: Ingeniería En Sistemas Computacionales

Docente:  García Martínez Raúl

***Definición.***

Arduino es una plataforma de electrónica «open-source» o de código abierto cuyos principios son contar con software y hardware fáciles de usar. Básicamente lo que permite esta herramienta es la generación de infinidad de tipos de microordenadores de una sola placa, que luego pueden tener una amplia variedad de usos según la necesidad de la persona que lo cree. Es decir, una forma sencilla de realizar proyectos interactivos para cualquier persona.

***Arquitectura.***

La arquitectura de un procesador está realizada para leer una cantidad determinada de unidades de datos. En este caso, dicha CPU puede operar como máximo con 32 bits de datos a la vez (8×4).

Los sistemas operativos de 32 bits utilizaron desde un inicio Windows 95. Un procesador de 32 bits soporta como máximo 4 GB de Memoria RAM. Dependiendo del procesador, un programa puede extenderse hasta los 64 GB.

El rango de valores con los que puede operar un procesador de 32 bits va desde 0 hasta 4.294.967.295.

La arquitectura x86 es en origen una arquitectura de 16 bits que se inicio con la serie 8086/8088. La primera vez que Intel implemento una versión de 32 bits de la arquitectura fue en los 80386. Precisamente Intel tomo una decisión de diseño de cara a los registros de su procesador bastante ingeniosa: en vez de crear nuevos registros para el modo de 32 bits extendieron los de 16 bits, algo que luego AMD repitió al desarrollar la extensión a 64 bits de la arquitectura

***Aplicación.***

Arduino se puede utilizar para desarrollar elementos autónomos, o bien conectarse a otros dispositivos o interactuar con otros programas, para interactuar tanto con el hardware como con el software.

Actualmente, el uso de Arduino puede catalogarse en dos grandes grupos:

1. Arduino se utilizado como un microcontrolador, cuando tiene un programa descargado desde un ordenador y funciona de forma independiente de éste, y controla y alimenta determinados dispositivos y toma decisiones de acuerdo al programa descargado e interactúa con el mundo físico gracias a sensores y actuadores.

2. Arduino hace de interfaz entre un ordenador u otro dispositivo, que ejecuta una determinada tarea, para traducir dicha tarea en el mundo físico a una acción. Y viceversa, gracias a sensores que están conectados a la placa Arduino podemos hacer que el ordenador ejecute determinada acción.

***Características.***

Las placas de Arduino se caracterizan por leer entradas – la luz de un sensor, pulsar un botón, o un mensaje de texto enviado a una Red Social – para convertirla en una salida – activando un motor, encendiendo un LED, publicando algo on-line.

La placa Arduino se conecta a un ordenador a través de un USB, donde se conecta con el entorno de desarrollo Arduino (IDE). El usuario escribe el código de Arduino en el IDE, y luego lo sube al microcontrolador que ejecuta el código, interactuando con las entradas y salidas como sensores, motores y luces.

***Relación con Arduino.***

Una placa de Arduino de 32 bits al momento de ingresar un código para realizar cierta tarea o función ya sea pulsar un botón, encender un led etc, el código no es tan extenso ni muy complicado. Realiza el reconocimiento de los puertos sin mucho esfuerzo, en cambio con el lenguaje ensamblador debemos conocer los diferentes saltos mecanismos para lograr reconocer un puerto y realizar una función.

***Ensamblador.***

Ensamblador es un lenguaje de bajo nivel. Consiste en una lista de instrucciones que no son de ninguna manera comparable a cualquier otro lenguaje como C, Basic o Pascal. El AVR tiene alrededor de 130 instrucciones, dependiendo del tipo, periféricos y tamaño del microcontrolador. En este link está el manual de las instrucciones con su respectiva explicación, y en este otro link se encuentra la guía de usuario de AVR Assembler.

Como ya hemos visto Arduino podemos programarlo en C puro, pero también podríamos programarlo en ensamblador, aunque no es una opción recomendable. Sin embargo, como veremos podremos obtener el código en ensamblador de nuestro sketch.

Sin embargo, podría ser útil en determinadas situaciones insertar código en ensamblador dentro de nuestro sketch de Arduino. El compilador GNU C para procesadores Atmel AVR RISC procesadores permite incrustar el código de lenguaje ensamblador en programas C. Esta característica se puede utilizar para optimizar manualmente las partes críticas de tiempo del software o utilizar una instrucción de procesador específico, que no están disponibles en el lenguaje C.

***ARM***

Es una arquitectura de 32 bits desarrollada en 1983 por la empresa Acorn Computers Ltd para usarse en computadoras personales que maneja un sistema de instrucciones realmente simple lo que le permite ejecutar tareas con un mínimo consumo de energía.El ARM2 fue un microprocesador de 32 bits muy sencillo con solo 30,000 transistores, El ARM2 tampoco contaba caché, siendo ésta simplicidad lo que le permitió un menor uso de energía. En 1991 después de varios años de trabajo entre Apple y Acorn presentaron un nuevo modelo del núcleo de ARM llamado ARM6 con 35,000 transistores, usándolo principalmente para la primer PDA de Apple, el Apple Newton. Fue a partir del desarrollo del ARM8 cuando comenzó a usarse dentro de calculadoras, GPS y dispositivos móviles. Actualmente nos encontramos que la mayoría de los dispositivos usan la familia Cortex de ARM.