

Universidad Autónoma del Estado de México

Dirección General de Centros Universitarios
y Unidades Académicas Profesionales

Centro Universitario UAEM Zumpango

Ingeniería en Computación

Semestre: Octavo

Unidad de aprendizaje: Sistemas Digitales (L41037)

Unidad de Competencia: *Unidad 2*

TEMA: 2.1 *¿Qué es programar un dispositivo?*

2.2 *Diagramas de flujo*

2.2.1 *Diseño de un programa*

2.2.2 *Elementos de un programa (algoritmo y datos)*

2.2.3 *Herramientas de programación*

2.2.4 *Set de instrucciones*

2.3 *Modos de direccionamiento*

Docente: M. en T. I. Jorge Bautista López

Zumpango de Ocampo, Septiembre de 2019

Descripción del Material

Se presenta el material de proyección visual con la finalidad de reforzar la apropiación del conocimiento por parte del alumno, en la UA de **Sistemas Digitales**, impartida en el octavo semestre de la Licenciatura en Ingeniero en Computación.

La intención del material es que el alumno comprenda el **Modelo de programación** empleado en los sistemas digitales.

Justificación

La elaboración de este material es para apoyar en la recopilación de conceptos, características ejemplos sobre los temas concernientes al modelo de programación de la Unidad de Competencia 2, perteneciente a la Unidad de Aprendizaje de: **Sistemas Digitales.**

El presente material es de apoyo tanto para el profesor como para el alumno.

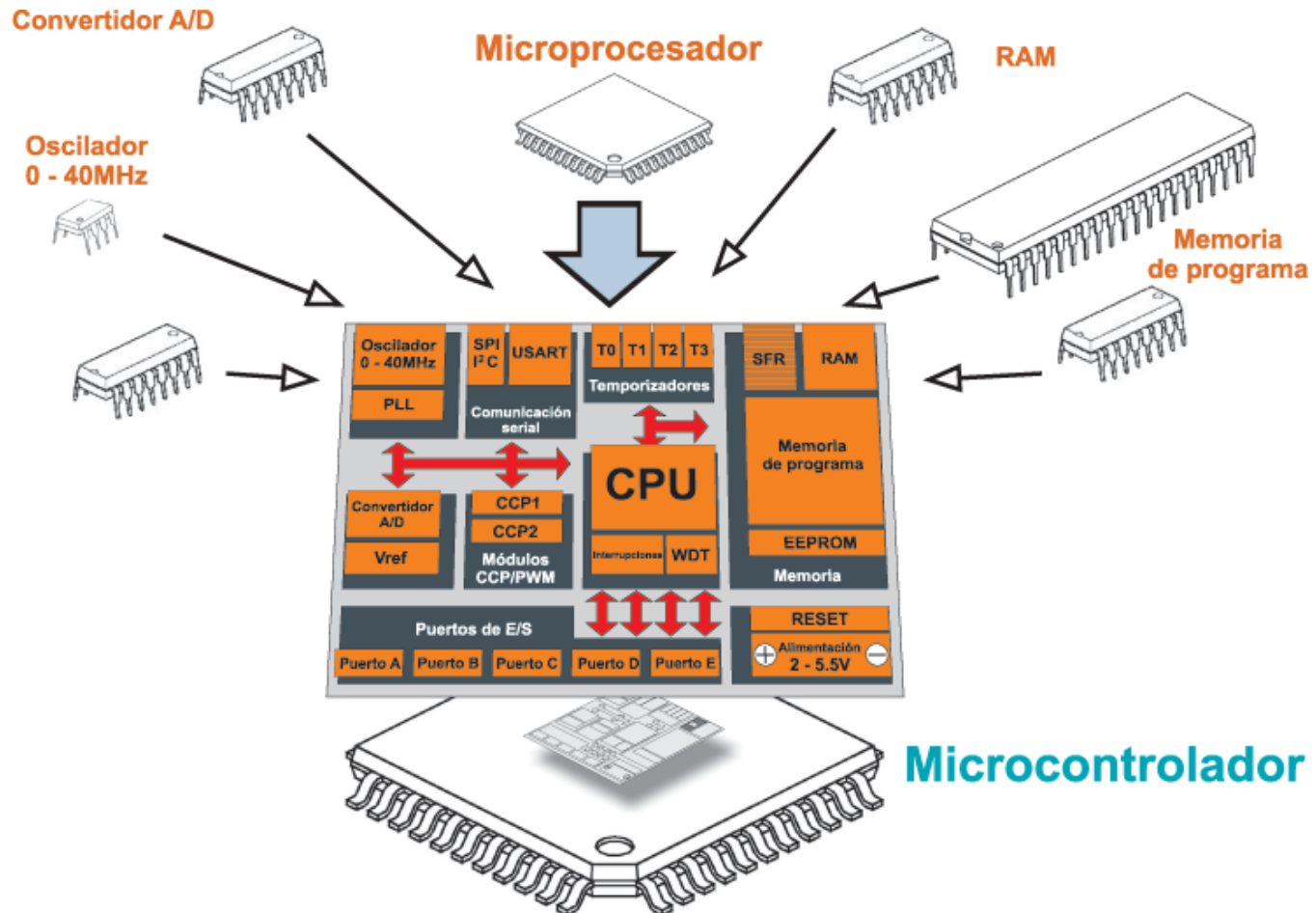
Propósito de la Unidad de Aprendizaje

Preparar al alumno para construir cualquier sistema mínimo y las diferentes aplicaciones de los microcontroladores y microprocesadores, con vistas a capacitar al estudiante a su egreso en el análisis, diseño, desarrollo y construcción de Hardware y sistemas de adquisición y distribución de señales.



Propósito de la Unidad de Competencia

Entender el modelo de programación



Estructura de la Unidad de Aprendizaje

Unidad de competencia 1. Comprender los concepto básico de los microprocesadores y microcontroladores lógica de transferencia entre registro y diseñar un procesador basado en la teoría de acumulador para usos específicos.

Unidad de competencia 2. Entender el modelo de programación

Unidad de competencia 3. Entender el modelo de programación

Unidad de competencia 4. Identificar las diferentes interfaces de los microcontroladores y microprocesadores.

Unidad de competencia 5. Diseñar una unidad central de procesamiento

Unidad de competencia 6. Diseñar un sistema mínimo.

Unidad de competencia 7. Comprender e identificar los diferentes tipos de aplicaciones.

Unidad de Competencia 2

Habilidades

- Conocer el modelo de programación para los microcontroladores y los microprocesadores.
- Elaborará diagramas de flujo.
- Conocer los diferentes modos de direccionamiento para microcontroladores y microprocesadores.

Actitudes / Valores

- Asistir puntualmente a clases
- Cumplir con las actividades y las tareas asignadas
- Mostrar disposición para el trabajo en equipo
- Mostrar tolerancia con las opiniones diversas
- Adoptar una actitud ética, crítica y comprometida con la aplicación de los conocimientos adquiridos en beneficio de la sociedad

Conocimientos

2.1 ¿Qué es programar un dispositivo?

2.2 Diagramas de flujo

2.2.1 Diseño de un programa

2.2.2 Elementos de un programa (algoritmo y datos)

2.2.3 Herramientas de programación

2.2.4 Set de instrucciones

2.3 Modos de direccionamiento

2.1 ¿Qué es programar un dispositivo?

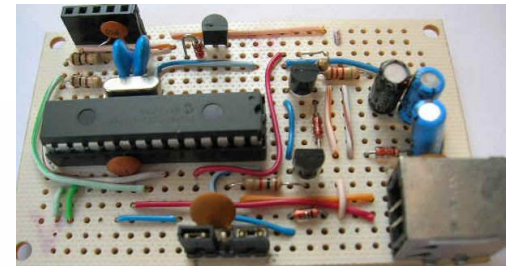
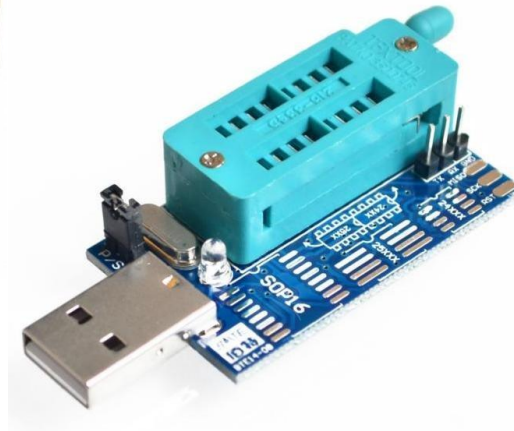
El término **programar** se utiliza en sistemas digitales para definir al proceso de guardar el archivo .hex en un microcontrolador, o de forma análoga el archivo .jedec o .jed en un Dispositivo Lógico programable (PLD).

Existen otros términos coloquiales que son empleados para referirse al mismo proceso, tales como:

- ✓ Grabar
- ✓ Quemar

2.1 ¿Qué es programar un dispositivo?

En la actualidad se pueden encontrar diferentes dispositivos para realizar esta tarea, desde grabadores o programadores genéricos (solo para una familia específica de uC) hasta los universales que soportan cientos de miles de circuitos.



Tipos de programadores

2.1 ¿Qué es programar un dispositivo?

En conjunto con el programador debe de contar con:

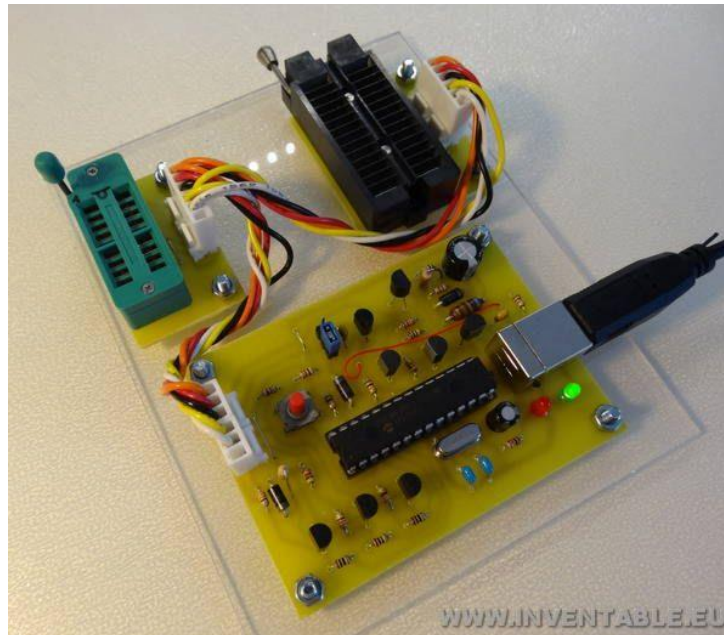
- Compilador instalado en una PC.
- Programador
- Software de aplicación para el programador



Equipo requerido para programar un dispositivo

2.1 ¿Qué es programar un dispositivo?

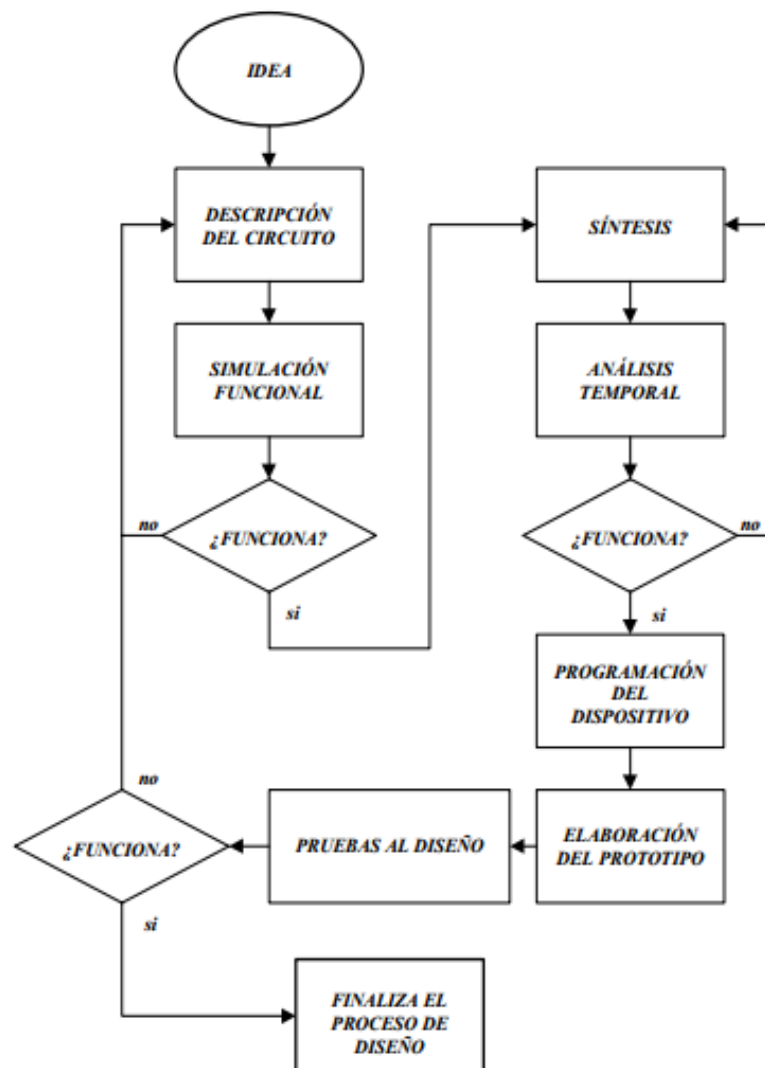
Los compiladores lógicos, tienen como función procesar y sintetizar el diseño lógico que se va a introducir en un microcontrolador o PLD, mediante un método específico (código en lenguaje bajo o alto nivel, ecuaciones booleanas, diagramas de estado, tablas de verdad)



<https://www.inventable.eu/2013/03/27/062-programador-de-pics/>

2.2 Diagramas de flujo

Representación gráfica que describe un proceso, sistema o algoritmo, la cual permite comprender el funcionamiento lógico del sistema propuesto.

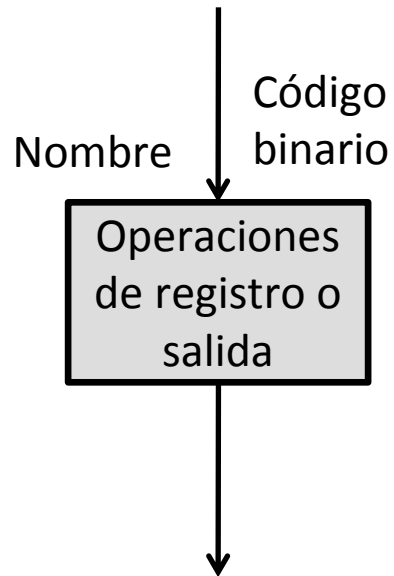


2.2 Diagramas de flujo

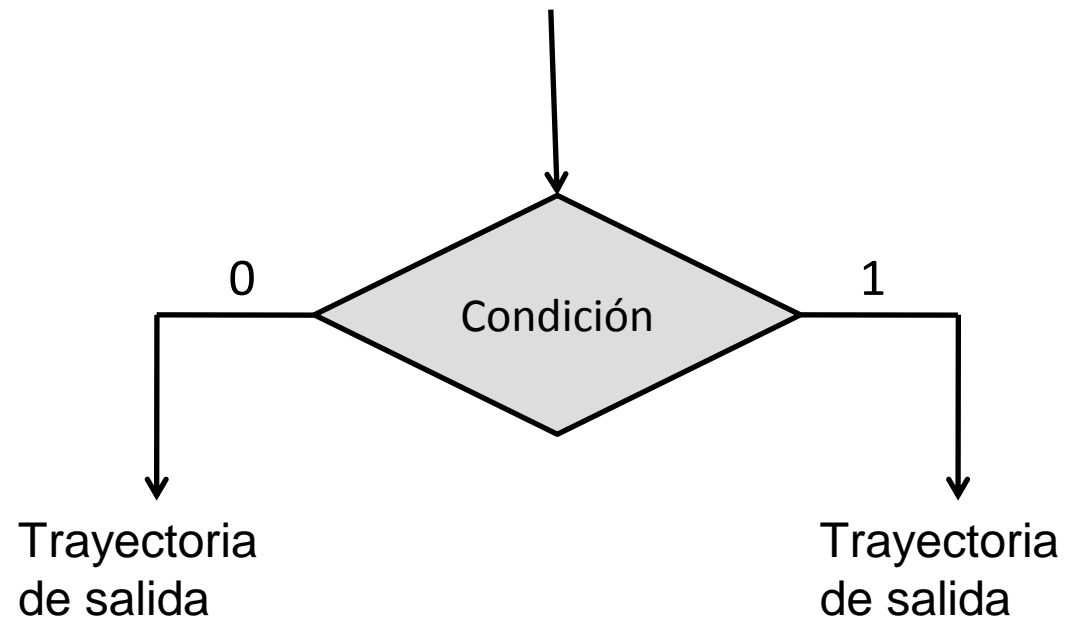
Diagrama de control ASM: Diagrama de flujo apropiado para describir las operaciones secuenciales de un sistema digital el cual cuenta con componentes básicos:

- a) Cuadro de estado
- b) Cuadro de decisión
- c) Cuadro condicional

2.2 Diagramas de flujo

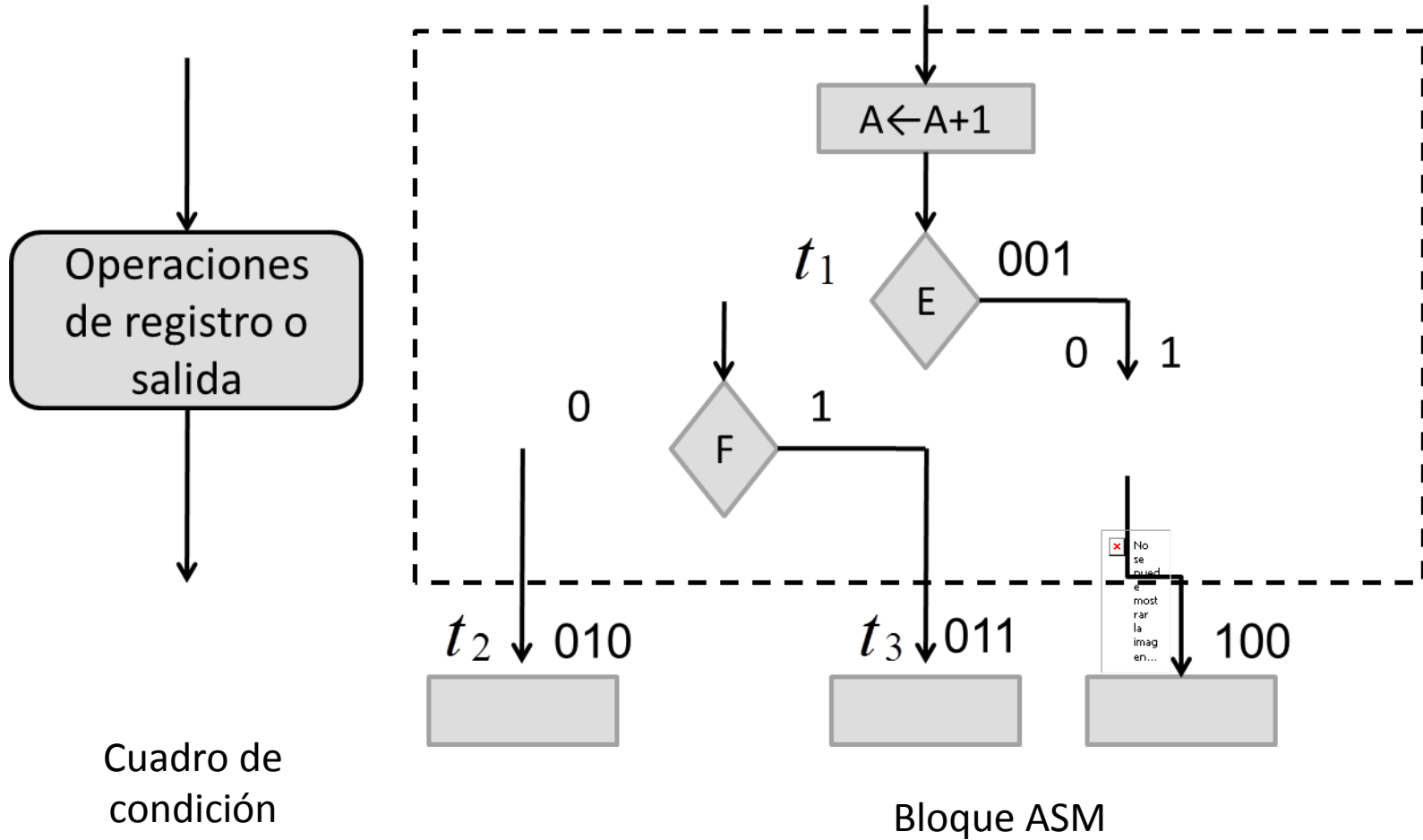


Cuadro de estado



Cuadro de decisión

2.2 Diagramas de flujo

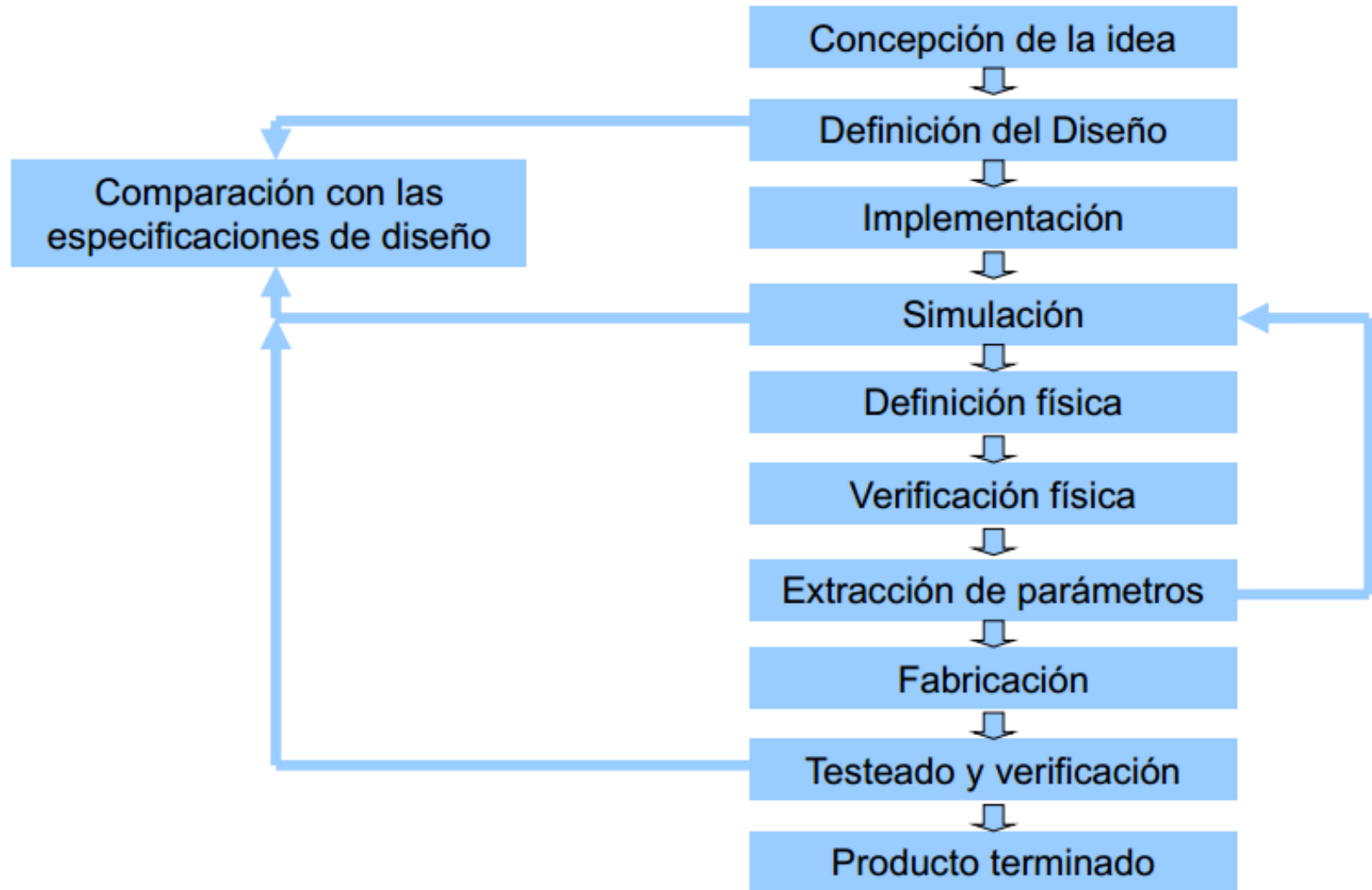


2.2.1 Diseño de un programa

Existen diferentes formas de diseñar un programa, por ejemplo para el caso de los PLD's:

- ✓ **Flujo de datos.** Se describen como fluyen los datos de las entradas hacia las salidas.
- ✓ **Comportamental.** Requiere de varias decisiones antes de definir los datos de salida correctos, por lo que se requiere de una descripción algorítmica del funcionamiento del circuito para facilitar el diseño del sistema.
- ✓ **Estructural.** Se emplea en circuitos que requieren de más de una función.

2.2.1 Diseño de un programa



2.2.2 Elementos de un programa (algoritmo y datos)

Algoritmo

- Secuencia finita de instrucciones, reglas o pasos que describen de forma precisa las operaciones que un ordenador debe realizar para llevar a cabo un tarea en un tiempo más finito. (Knuth, D. 1968)
- Descripción de un esquema de comportamiento expresado mediante un repertorio finito de acciones y de informaciones elementales, identificadas, bien comprendidas y realizables a priori (Scholl, P. 1988).
- Conjunto finito de pasos definidos, estructurados en el tiempo y formulados con base a un conjunto finito de reglas no ambiguas, que proveen un procedimiento para dar la solución o indicar la falta de esta a un problema en un tiempo determinado (Quispe-Otazu, 2004).

2.2.2 Elementos de un programa (algoritmo y datos)

		Elementos de un programa	
		Algoritmo	Datos
Microcontrolador	Configuración del dispositivo	B'10101011' H'F5' 0X8A	Binario Hexadecimal
	Programa principal	D'123' .123 'G'	Decimal ASCII
PLD	Entidad	Enumerados, Enteros, Punto flotante, físicos	
	Arquitectura		

2.2.2 Elementos de un programa (algoritmo y datos)

Línea N°.	Sumador-completo de dos datos con longitudes de 1-bit (Declaración de Entidad)
1	--Declaración de la entidad de un circuito sumador
2	entity sumador is
3	port (A, B, Cin: in bit;
4	SUMA, Cout: out bit);
5	end sumador;

(entity) Inicia declaración de la entidad

(--) Indica Comentario

Identificador de la entidad

Nombres de los puertos

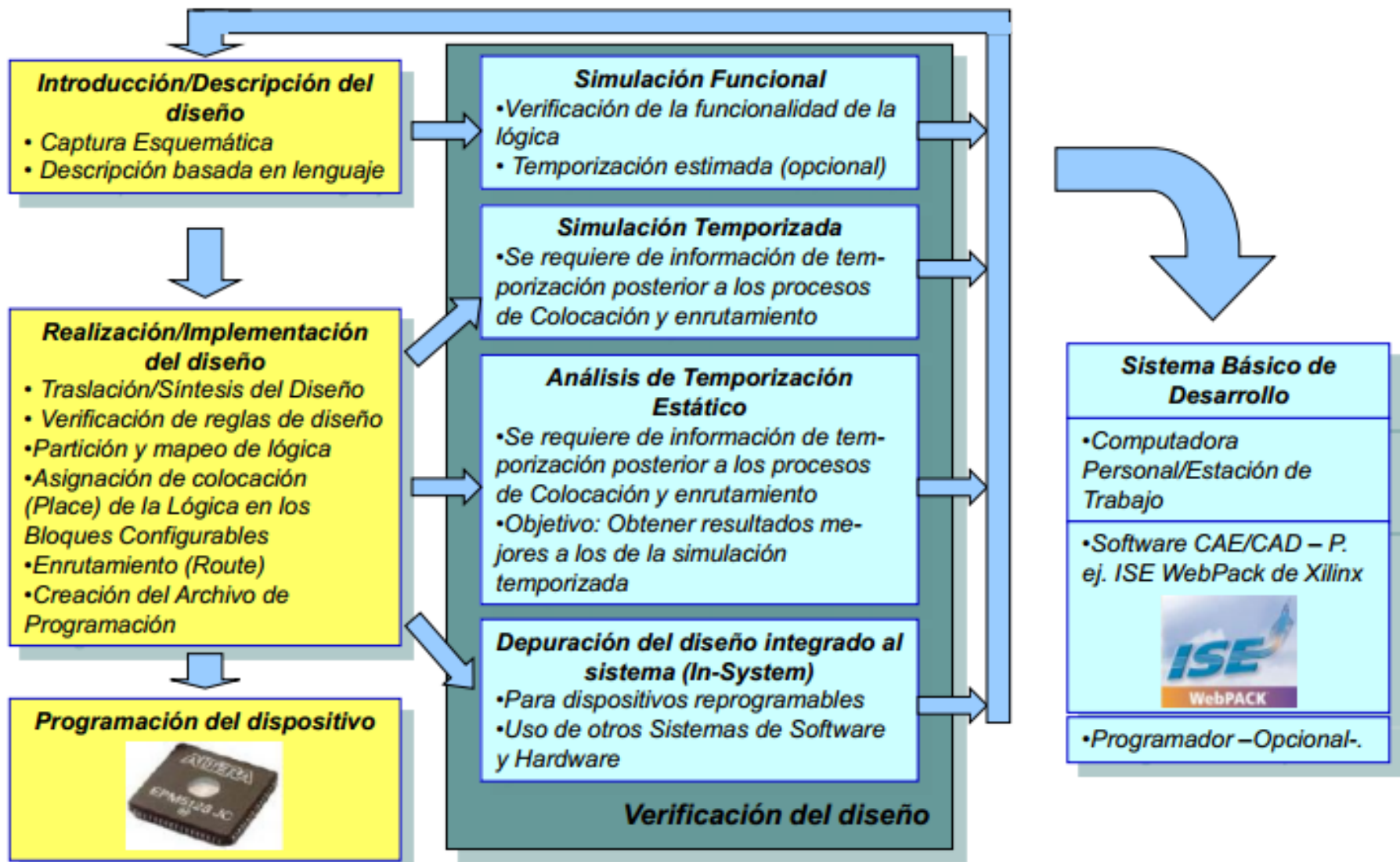
Modo de Operación

Tipo de Dato

(end) Finaliza declaración de la entidad

(;) Finaliza declaración o subdeclaración

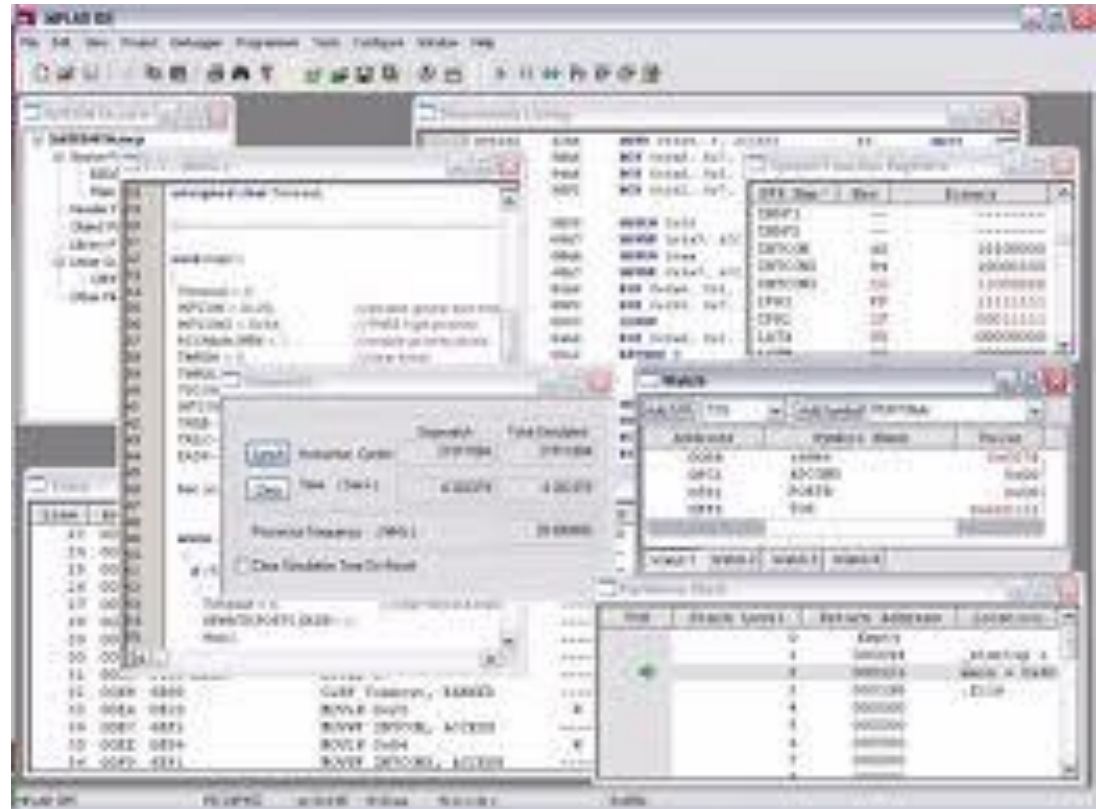
2.2.2 Elementos de un programa (algoritmo y datos)



2.2.3 Herramientas de programación

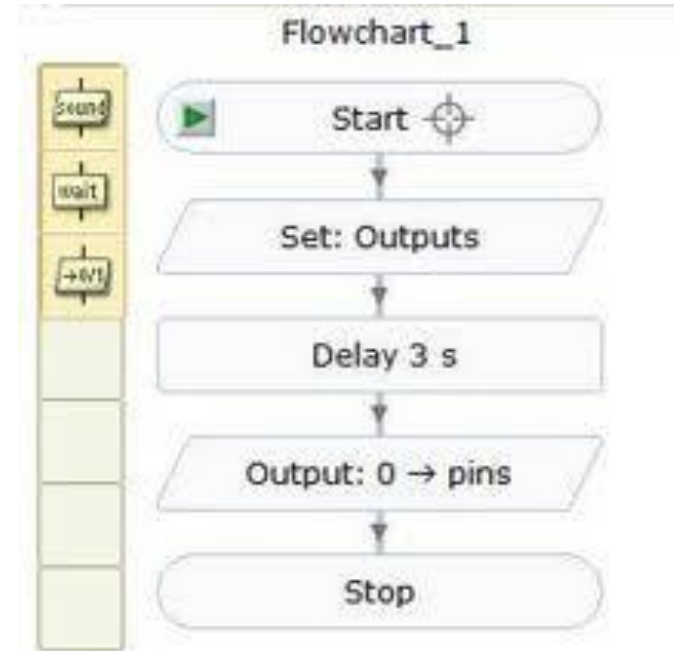
- Herramientas CAD – EDA
 - ❖ CAD (Computer Aided Design). Emplea técnicas de análisis basado en grafos, las cuales forman parte de todo el proceso de diseño
 - ❖ EDA (Electronic Design Automation). Son herramientas de software y hardware utilizados para el diseño de sistemas electrónicos.
- Lenguajes de Descripción de Hardware
- Lenguajes de bajo y alto nivel

2.2.3 Herramientas de programación



Interfaz de MPLab para lenguaje ensamblador

2.2.3 Herramientas de programación



Interfaz para lenguaje de alto nivel y lenguaje gráfico

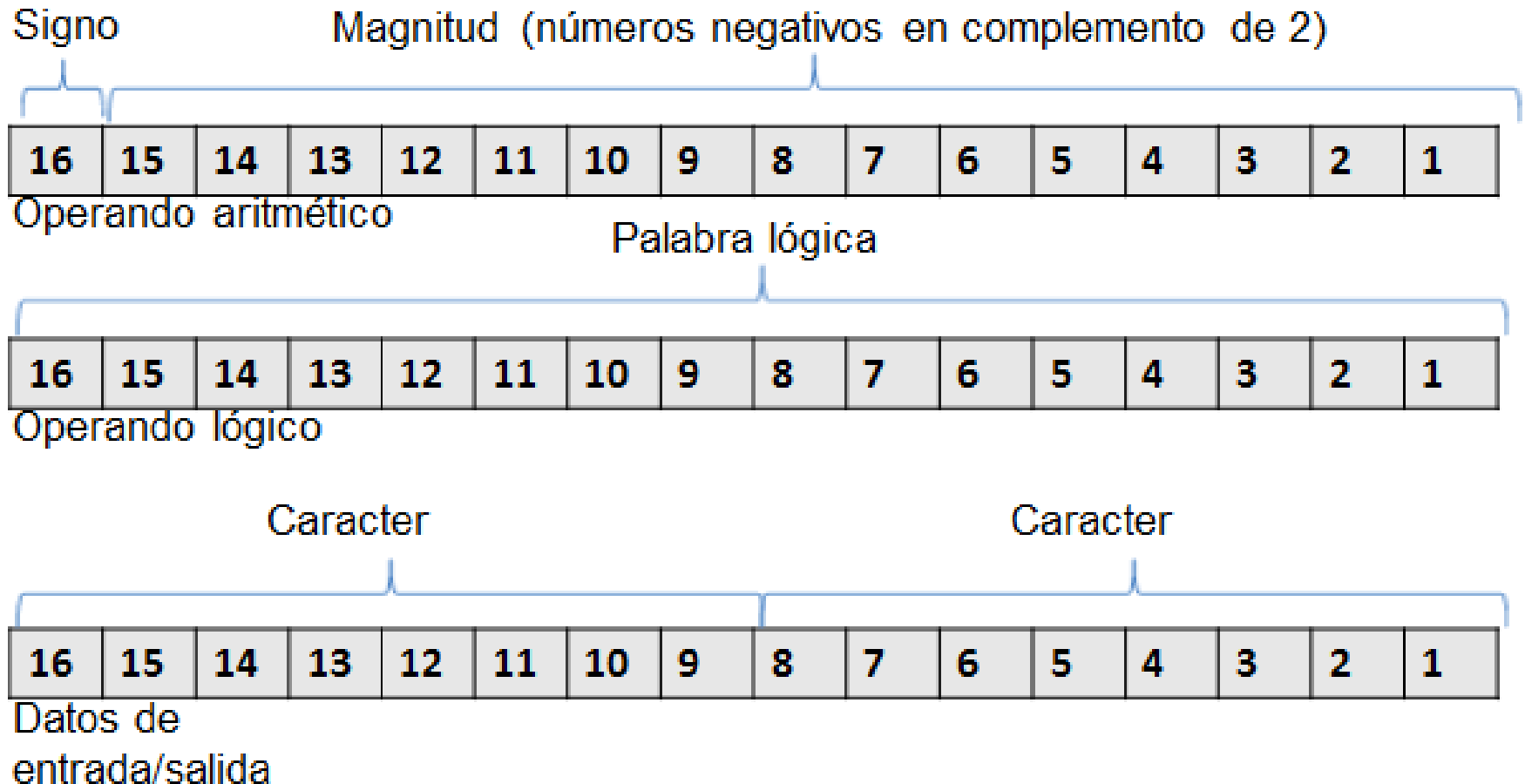
2.2.4 Set de instrucciones

- Conjunto de instrucciones para programar el microcontrolador y/o microprocesador
- Depende de las características de la arquitectura
- El set de instrucciones puede ser:
 - RISC (reduced instruction set computer)
 - CISC (complex instruction set computer)
- Clasificación del set de instrucciones
 - Orientadas a bytes
 - Orientadas a literal y control
 - Orientadas a bits

2.2.4 Set de instrucciones

- Una palabra de memoria consiste de 16 bits. Una palabra puede representar una unidad de datos o una instrucción.
- Los datos para las operaciones aritméticas se presentan por un número binario de 15 bits con un signo en la posición del bit decimo sexto.
- Se asume que los números están en su equivalente de complemento de 2.

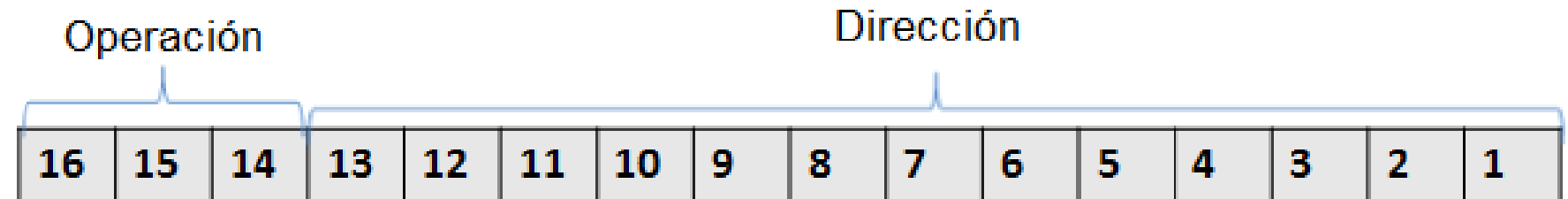
2.2.4 Set de instrucciones



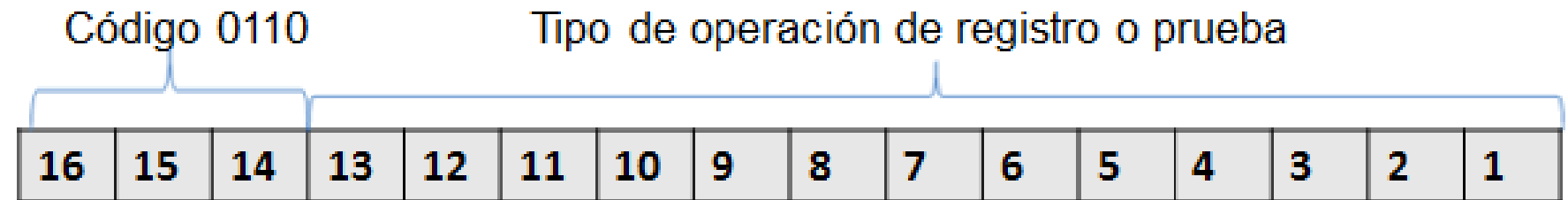
2.2.4 Set de instrucciones

- Las operaciones lógicas se realizan con bits individuales de la palabra con el bit 16.
- Cuando la computadora se comunica con un dispositivo I/O, la información transferida se considera compuesta de caracteres alfanuméricos de 8 bits.

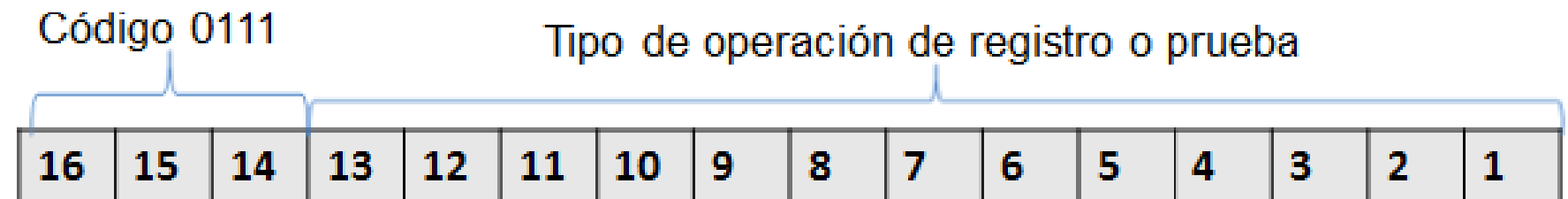
2.2.4 Set de instrucciones



Instrucción de referencia de memoria



Instrucción de referencia



Datos de
entrada/salida

2.3 Modos de direccionamiento

- ❑ El código de operación de una instrucción específica, la operación que va a ser ejecutada después de haberse leído en la memoria y lo ubica en la unidad de control del CPU.
- ❑ Los operandos pueden estar localizados en los registros de proceso, en las palabras de memoria o en los registros de interconexión.
- ❑ La forma como son determinados los operandos durante la ejecución del programa se determina a partir del modo de direccionamiento de la instrucción.

2.3 Modos de direccionamiento

- ❑ El modo de direccionamiento de una instrucción se especifica con un código binario de la misma forma como se especifica el código de operación.
- ❑ En los microprocesadores de 8 bits, el primer byte de una instrucción es un código binario combinado que especifica la operación y el modo de la instrucción.

2.3 Modos de direccionamiento

addressing mode	example	result
immediate	ADD R1, #5	$R1 \leftarrow R1 + 5$
register	ADD R1, R2	$R1 \leftarrow R1 + R2$
direct	ADD R1, 100	$R1 \leftarrow R1 + M[100]$
register indirect	ADD R1, (R2)	$R1 \leftarrow R1 + M[R2]$
post-increment	ADD R1, (R2)+	$R1 \leftarrow R1 + M[R2]$ $R2 \leftarrow R2 + d$
pre-decrement	ADD R1, -(R2)	$R2 \leftarrow R2 - d$ $R1 \leftarrow R1 + M[R2]$
displacement	ADD R1, 100(R2)	$R1 \leftarrow R1 + M[100 + R2]$
indexed	ADD R1, (R2+R3)	$R1 \leftarrow R1 + M[R2+R3]$
memory indirect	ADD R1, @(R2)	$R1 \leftarrow R1 + M[M[R2]]$

2.3 Modos de direccionamiento

- ❑ Los llamados modos de direccionamiento son las diferentes maneras de especificar en informática un operando dentro de una instrucción.
- ❑ Un modo de direccionamiento especifica la forma de calcular la dirección de memoria efectiva de un operando mediante el uso de la información contenida en registros y/o constantes, contenida dentro de una instrucción de la máquina o en otra parte.²

² <http://tutorialensamblador.galeon.com/unidad3.html>

2.3 Modos de direccionamiento

- **Implícito:** este tipo de instrucción son de 1 byte.
- **De registro:** los operandos están en los registros que residen dentro del CPU.
- **Indirecto de registros:** especifica un registro o un par de registros en el procesador.
- **Inmediato:** el microprocesador de 8 bits se coloca el operando en la memoria.
- **Directo:** el operando reside en la memoria.
- **De la pagina cero:** es similar al directo excepto que la parte de la dirección contiene solo un 1 byte.

2.3 Modos de direccionamiento

- **De pagina presente:** reside en la memoria.
- **Indexado:** las instrucciones contiene 3 bytes con dos en una dirección de 16 bits.
- **De registro de base:** es similar al modo indexado, excepto que la parte de dirección consiste de un número de bits.
- **Indirecto:** la parte de la dirección especifica donde se almacena la dirección efectiva.
- **Indirecto indexado:** la instrucción se agrega al contenido del registro índice para determinar la dirección.

2.3 Modos de direccionamiento

Modo de direccionamiento	Dirección efectiva	comentarios
Directo	AD16	Parte de la dirección de 16 bits de la instrucción.
Pagina cero	AD8	Parte de la dirección de 8 bits de la instrucción.
Pagina presente	PC (H) + AD8	Los 8 bits de mayor orden del PC encadenadas.
Relativo	PC + AD8	Contenido del PC más AD8 con signo.
Indexado	XR + AD16	Contenido del XR mas AD16
Registro base	XR + AD8	Contenido de XR mas AD8
Indirecto	M [AD16]	Dirección almacenada en el lugar dado por AD16
Indirecto-indexado	M [XR + AD8]	Dirección almacenada en el lugar (XR+AD8)
Indirecto-indirecto	M [AD8] + XR	Dirección almacenada en el lugar AD8 más el contenido de XR.

Conclusiones

Resulta de suma importancia el conocer y comprender aspectos y temas relacionados a los microcontroladores y microprocesadores, tales como la forma de programarlos, los algoritmos empleados para su programación así como también las herramientas de hardware y software empleados en el diseño e implementación de los sistemas digitales.

Resalta el hecho de que en la actualidad se pueden emplear lenguajes de bajo y alto nivel así como los lenguajes gráficos para diseñar sistemas acorde a las necesidades del usuario.



UAEM

Referencias

1. Morris, M. Mano “Lógica digital y diseño de computadores” Ed. Prentice Hall (1989) ISBN 9688800163
2. Mano Morris. (2003) “Diseño Digital”. Ed. Prentice Hall. 3ra edición.
3. Tocci Ronald J. (2003). “Sistemas Digitales: principios y aplicaciones”. Editorial Pearson Educación. 6ta edición.
4. Mandado, Enrique “Sistemas electrónicos digitales” 7ª Edición Ed. Marcombo (1992) Barcelona ISBN 8426711707
5. Hamburg VHDL archive. <http://tech-www.informatik.uni-hamburg.de/vhdl/>
6. IEEE Standard 1076 “IEEE Standard VHDL Language Reference Manual ” (2000) ISBN (Edición Impresa) 0738119482 ISBN (del PDF) 0738119482 <http://www.cs.indiana.edu/classes/p442/reading/VHDLref.pdf> o (2002) http://standards.ieee.org/catalog/olis/arch_dasc.html
8. TTL Data Book <http://upgrade.cntc.ac.kr/data/ttl>
9. Blakeslee, Thomas “Digital design with standard MSI & LSI: Design techniques for the microcomputer” Ed. John Wiley& sons (1979) 2a Edición New York ISBN 0471052221

