

1AEL0256

Microcontroladores

Sesión 01

Semestre 2026-0

Profesor: Kalun José Lau Gan

Agenda 22/01/2026 13:00

- Designación de delegado
- Lectura de sílabo
- Aspectos normativos del curso
- Requerimientos de software/hardware
- Lista de materiales para el laboratorio
- Recomendaciones del curso
- Repaso de conocimientos previos

Importantísimo

- Las clases a distancia son grabadas.
- Obligatorio encender sus cámaras durante el desarrollo de cada una de las sesiones de clase.
- Priorizar el uso de elementos multimedia antes del chat para las consultas.

Informativo:

- Sección: TE NRC1029 Jueves 13:00 – 14:59 (a distancia)
Viernes 10:00 – 11:59 (a distancia)
 - LB NRC1037 Jueves 15:00 – 18:59 (a distancia)
Viernes 15:00 – 18:59 (a distancia)
- Delegado/a: Ruben Sebastián Silva Ramos

Profesor a cargo:

- Kalun José Lau Gan (pcelklau@upc.edu.pe)

Elección del delegado

- Voluntarios
- Votación
- Resultados

Función del delegado:

- Representar a sus compañeros
- Recoger información del aula y entregar informe en la reunión de delegados (sobre el dictado de clases, sobre las competencias, sobre los servicios, etc.)
- Apoyar la participación en encuestas (invocar la máxima participación)
- Colaborar con el profesor en las actividades del curso

Organigrama Ingeniería UPC Campus Monterrico

Organigrama de la Escuela de Ingeniería Electrónica (Monterrico)



Patricia Fujishima Salinas
Asistente del Decano
Email: patricia.fujishima@upc.pe



DECANA DE INGENIERÍA
Rosario Villalta Riega
Email: rosario.villalta@upc.pe

Escuela de Ingeniería Electrónica

Jacqueline Prieto Quiroz
Asistente de la Escuela de Ingeniería Electrónica
Email: jacqueline.prieto@upc.pe



DIRECTOR DE ESCUELA
Carlos Valdez Velásquez-López
Director de las Carreras de Ingeniería Electrónica, Ingeniería Mecatrónica, e Ingeniería de Redes y Comunicaciones
Email: carlos.valdez@upc.pe



Luis Dávila
PTC Electrónica
Campus Monterrico
luis.davila@upc.pe



Carlos Perea
PTC Mecatrónica
Campus Monterrico
carlos.perea@upc.pe



Renzo Albatrino
Coordinador
Ing. Mecatrónica
renzo.albatrino@upc.pe



Erwin Dianderas
Coordinador
Ing. Electrónica
erwin.dianderas@upc.pe



Renato Gonzales
Coordinador Ing. de Redes y Comunicaciones
renato.gonzales@upc.pe



José Feliciano
Supervisor de Laboratorio
jose.feliciano@upc.pe



Henry Figueroa
Supervisor de Laboratorio
henry.figueroa@upc.pe

Sílabo

Disponible en el AV

- Objetivos
- Competencias
- Contenido
- Evaluaciones
- Fórmula de evaluación

III. INFORMACIÓN GENERAL

CURSO: Microcontroladores

CODIGO: EI-174

CICLO: 2021B

CUERPO ACADÉMICO: Luis Gao, Kalan / Salas Arriarán, Sergio

CRÉDITOS: 4

SEMANAS: 5

HORAS: 4 II (Laboratorio) Semanal - 4 II (Teorico) Semanal

ÁREA O CARRERA: Ingeniería Electrónica

III. MISIÓN Y VISIÓN DE LA UPC

Misión: Formar líderes integros e innovadores con visión global para que transformen al Perú. Visión: Ser líder en la educación superior por su excelencia académica y su capacidad de innovación.

III. INTRODUCCIÓN

El curso Microcontroladores explora el área de la Ingeniería Electrónica donde se integran las disciplinas de desarrollo integrado de hardware y software. Los microcontroladores son la base fundamental en el desarrollo electrónico hoy en día, todos los dispositivos electrónicos que consisten de numerosos circuitos poseen uno. Muchas aplicaciones en el campo de automotriz, industrial, robótica, biomédicas, domótica, electrónica de consumo, telefonía, electrónica de potencia y otros más requieren del uso de microcontroladores capaces de funcionar en base a un programa o Firmware eficiente. El estudiante desarrollará la capacidad de formular un proyecto sobre el diseño de un equipo electrónico digital basado en microcontroladores y aplicará técnicas de programación para implementar una solución a un problema real formulado.

El curso promueve el desarrollo de la competencia general de "Pensamiento Innovador" en el nivel medio y la competencia específica a nivel medio de "La capacidad de desarrollar y llevar a cabo la experimentación adecuada, analizar e interpretar datos, y usar el juicio de ingeniería para sacar conclusiones".

Respaldo: Sistema Digital

Competencias asignadas al curso:

- Competencia general UPC N°6 - Pensamiento innovador
 - Capacidad de generar propuestas sostenibles, creativas e inspiradoras de mejora o creación de un producto, servicio o proceso que impactan positivamente en un determinado contexto incorporando el ensayo y error como parte del proceso
- Competencia ABET N°6
 - La capacidad de desarrollar y llevar a cabo la experimentación adecuada, analizar e interpretar datos, y usar el juicio de ingeniería para sacar conclusiones.

Evaluación de las competencias

- La evaluación de las competencias asignadas al curso será en la DD con la siguiente rúbrica:

Rúbrica de evaluación de competencias asignadas al curso EL256 Microcontroladores			
Nombres completos del estudiante:	Código:	Carrera:	
Competencia general UPC N°6 - Pensamiento innovador Capacidad de generar propuestas sostenibles, creativas e inspiradoras de mejora o creación de un producto, servicio o proceso que impactan positivamente en un determinado contexto incorporando el ensayo y error como parte del proceso			
Dimensiones Nivel 2 Intermedio	Cumple con las expectativas (C)	En desarrollo (D)	Insatisfactorio (I)
Elabora una propuesta considerando distintas características únicas y distintivas de lo conocido.	Elabora satisfactoriamente una propuesta teniendo en cuenta características únicas y distintivas	Elabora parcialmente una propuesta teniendo en cuenta características únicas y distintivas	Elabora insatisfactoriamente una propuesta teniendo en cuenta características únicas y distintivas
	18 a 20	13 a 17	0 a 12
Diseña una propuesta usando los conceptos de diferentes áreas de conocimiento aplicando metodologías de innovación o metodologías de análisis orientadas a la creación o mejora de un producto, servicio o proceso	Diseña una propuesta teniendo en cuenta las metodologías propuestas en el curso y aplicando innovación para mejorar dicha propuesta	Diseña una propuesta de acuerdo a las metodologías en cuenta las metodologías propuestas en el curso y aplicando innovación para mejorar dicha propuesta	Diseña una propuesta de acuerdo a las metodologías en teniendo en cuenta las metodologías propuestas en el curso y aplicando innovación para mejorar dicha propuesta
	18 a 20	13 a 17	0 a 12
Elabora una propuesta cuyo valor proyecta un impacto positivo en un determinado contexto de manera argumentada	Elabora una propuesta que genera un impacto positivo en el desarrollo de aplicaciones con microcontroladores	Elabora parcialmente una propuesta que genera un impacto positivo en el desarrollo de aplicaciones con microcontroladores	Elabora insatisfactoriamente una propuesta que genera un impacto positivo en el desarrollo de aplicaciones con microcontroladores
	18 a 20	13 a 17	0 a 12
Competencia ABET N°6 La capacidad de desarrollar y llevar a cabo la experimentación adecuada, analizar e interpretar datos, y usar el juicio de ingeniería para sacar conclusiones.			
Dimensiones Nivel Intermedio	Cumple con las expectativas (C)	En desarrollo (D)	Insatisfactorio (I)
6.1. Desarrolla y realiza experimentos	Realiza la experimentación siguiendo estrictamente el procedimiento y registrando de manera correcta sus resultados	Realiza la experimentación siguiendo parcialmente el procedimiento y registrando de manera parcial sus resultados	Realiza la experimentación de manera inadecuada y no registra adecuadamente sus resultados
	18 a 20	13 a 17	0 a 12
6.2. Analiza, interpreta datos y usa el juicio de ingeniería en las conclusiones	Analiza e interpreta los resultados obtenidos en forma escrita y siguiendo criterios técnicos e ingenieriles correctos	Analiza e interpreta los resultados obtenidos en forma parcial y clara, sin seguir criterios técnicos ni ingenieriles	Analiza e interpreta los resultados de manera ineficiente y no clara, sin seguir criterios técnicos ni ingenieriles
	18 a 20	13 a 17	0 a 12
Calificación DD Fórmula: (D1+D2+D3+D4+D5) / 5			0

Fórmula PF

- Según sílabo:
- $$\text{PF} = 8\% (\text{LB1}) + 10\% (\text{PC1}) + 8\% (\text{LB2}) + 15\% (\text{TP1}) + 8\% (\text{LB3}) + 10\% (\text{PC2}) + 15\% (\text{DD1}) + 26\% (\text{TF1})$$

Cronograma de evaluaciones:

• EE	Semana 1	Sesión de laboratorio
• LB1	Semana 2	Sesión de laboratorio
• PC1	Semana 3	Sesión de teoría
• LB2	Semana 3	Sesión de laboratorio
• TP1	Semana 4	Según cronograma de la universidad
• LB3	Semana 5	Sesión de laboratorio
• PC2	Semana 6	Sesión de teoría
• DD1	Semana 7	Sesión de laboratorio
• TF1	Semana 7	Según cronograma de la universidad

Reglamento del curso y sus evaluaciones

- Además de lo estipulado en el sílabo se tienen los siguientes documentos normativos:
 - SICA-REG-05 REGLAMENTO DE ESTUDIOS DE PREGRADO:
<https://sica.upc.edu.pe/categoría/normalización/sica-reg-05-reglamento-de-estudios-de-pregrado>
 - REGLAMENTO DE DISCIPLINA DE ALUMNOS:
<https://sica.upc.edu.pe/categoría/reglamentos-upc/sica-reg-26-reglamento-de-disciplina-de-alumnos>
 - REGLAMENTO PARA LA PREVENCIÓN E INTERVENCIÓN EN CASOS DE HOSTIGAMIENTO SEXUAL- UPC:
<https://sica.upc.edu.pe/categoría/normalización/sica-reg-31-reglamento-para-la-prevencion-e-intervencion-en-casos-de-hostiga>
- Indicaciones generales para las evaluaciones en línea (disponible en AV)

Reglamento del curso y sus evaluaciones

- El curso es de naturaleza teórico-práctico con implementación en físico de prototipos basados en circuitos electrónicos.
- Se recomienda la toma de apuntes durante las sesiones de clase.
- **Asistencia y participación como mínimo al 75% de sesiones de clase.**
- Sesiones virtuales:
 - Tener sistemas multimedia (micrófono y webcam) en correcto funcionamiento y durante toda la sesión la cámara web deberá estar activada.
 - Se priorizará la atención mediante audio y video antes del chat.
- Sesiones presenciales:
 - Restricción de tomas fotográficas y/o registro de video durante las sesiones.
- Revisar la lista de materiales que se emplearán en las sesiones de laboratorio.

Reglamento del curso y sus evaluaciones

- Reglamento de asistencia a las sesiones y evaluaciones del curso:

a. La asistencia a las actividades programadas, sean clases teóricas, prácticas, evaluaciones o cualquier otra actividad programada en la asignatura, es obligatoria. En toda actividad programada se generará un registro de asistencia como máximo 15 minutos luego de la hora programada para el inicio de la actividad. El alumno que no responda a la solicitud de registro de asistencia será considerado como ausente. El alumno tiene el deber de permanecer durante la sesión de clase; en caso deje de participar en la sesión, el docente podrá modificar el registro de asistencia y considerar al alumno como ausente.

La realización de las actividades previstas en el Sistema de Gestión de Aprendizaje (Aula virtual), en seguimiento de las horas lectivas y no lectivas de las asignaturas presenciales, blended y virtuales, son de cumplimiento obligatorio por parte del alumno.

Reglamento del curso y sus evaluaciones

- Reglamento de asistencia a las sesiones y evaluaciones del curso:

- b. (i) Para que el alumno pueda rendir la evaluación final de una asignatura debe asistir -al menos- al 75% de las sesiones teóricas y prácticas programadas, sean estas presenciales o a distancia. El alumno que -por cualquier causa- supere el 25% de inasistencias en una asignatura, tendrá cero en la evaluación final de la asignatura.
- (ii) Si un alumno rinde la evaluación final de una asignatura, obteniendo una nota en la misma; y luego excede el límite máximo de inasistencias, le corresponderá la nota cero en dicha evaluación.
- (iii) Para el cálculo del promedio final de una asignatura en la cual el alumno haya superado el 25% de inasistencias, se promedian las notas que obtuvo en la misma, incluyendo la nota cero de la evaluación final. Los tipos de notas en los que se aplica esta disposición, en orden de prioridad, se detallan en el Anexo 6 del presente Reglamento.

Reglamento del curso y sus evaluaciones

- Reglamento de asistencia a las sesiones y evaluaciones del curso:

c. El alumno que no rinda una evaluación o no cumpla con la entrega de un trabajo académico dentro del plazo establecido en la asignatura y sin que medie una justificación excepcional presentada al docente de la asignatura y aprobada por el mismo, recibe una calificación equivalente a cero **y se refleja en el sistema como "No Rindió" (NR).**

Reglamento del curso y sus evaluaciones

- Reglamento de asistencia a las sesiones y evaluaciones del curso:

d. El alumno que no rinda una evaluación recuperable prevista en el sílabo de una asignatura, a excepción de la evaluación final, puede realizar el trámite de Subsanación de evaluación no rendida. Dicho trámite consiste en subsanar una de las notas de las evaluaciones no rendidas con la nota obtenida en la evaluación final (EB).

El plazo para la presentación de la solicitud de subsanación y la formalización del pago del trámite lo fija la Secretaría Académica. El alumno que haya realizado este trámite no podrá acceder al trámite de Recuperación de evaluaciones estipulado en el literal "e" del presente artículo.

El trámite de subsanación solo es posible si la asignatura consigna el tipo de nota EB en su fórmula de evaluación. Para aquellas asignaturas que no tuvieran el tipo de nota EB en su fórmula de evaluación, procederá el trámite de recuperación.

Reglamento del curso y sus evaluaciones

- Reglamento de asistencia a las sesiones y evaluaciones del curso:

e. Si un alumno no rinde la evaluación final (EB), cabe su recuperación en fecha posterior a la inicialmente programada. La evaluación de recuperación abarca todos los temas de la asignatura. El alumno que no se presente a la misma obtiene cero de calificación.

El plazo para la presentación de la solicitud y la formalización del pago del trámite lo fija la Secretaría Académica, que programa en una sola oportunidad la fecha para la recuperación de la evaluación y la publica. La evaluación de recuperación sólo reemplaza la nota de la evaluación final que el alumno no rindió en la fecha originalmente programada, salvo el caso mencionado en el literal anterior (de asignaturas que no tuvieran el tipo de nota EB en su fórmula de evaluación).

El alumno que haya realizado este trámite no podrá acceder al trámite de Subsanación de evaluación no rendida estipulado en el literal "d" del presente artículo.

Reglamento del curso y sus evaluaciones

- Reglamento de asistencia a las sesiones y evaluaciones del curso:

f. El alumno que haya faltado a más del 25% de las horas programadas en una asignatura podrá solicitar la recuperación de una evaluación recuperable no rendida en su oportunidad. Ello no aplica para la evaluación final de la asignatura; a la cual -de acuerdo con lo indicado en el artículo 2.1.b del presente Reglamento- se le registrará como nota cero, por haber superado el límite máximo de faltas en la asignatura.

g. Las fechas de entrega de las evaluaciones estarán establecidas en el silabo y/o en el Aula Virtual de la asignatura. Si se entrega después de la fecha y hora límite, la evaluación se dará por No Rendida.

Indicaciones de evaluaciones escritas:

- Dar lectura a las indicaciones antes de proceder a desarrollar la evaluación escrita.
- Son evaluaciones escritas:
EE, PC1, PC2, EREC

• La evaluación es de naturaleza individual y empieza a la hora establecida. Estudiante que llegue después de 15 minutos de inicio será marcado como ausente a la evaluación corroborado con el reporte de asistencia de participantes del AV. Tener en cuenta que pasado los 15 minutos iniciales se bloqueará el acceso al documento enunciado.

• El desarrollo será de **manera manual** escrito con lapicero azul o negro en hoja bond A4 limpia, deberá **membratar** cada hoja usada con el nombre de la evaluación, su código UPC, nombre completo, carrera profesional cursando y número de página. No utilizar retozos de hojas y/o hacer collage en la hoja principal. De no seguir esta indicación primaria automáticamente recibirá la calificación de cero en cada hoja sin identificación, no permitiendo reclamos posteriores.

• El desarrollo deberá ser de manera **clara, detallada, ordenada y secuencial**. Se anulará toda solución sin desarrollo y/o justificación, en desorden, con respuesta ambigua, doble, sin unidades o usando tipografías diferentes.

• El uso excesivo de **líquido corrector**, así como **borrones**, dibujos de diagramas y circuitos en **garabato**, el evaluador podrá anular la resolución de la pregunta sin posibilidad a reclamos posteriores.

• Luego del desarrollo se tomarán **fotografías bien enfocadas, iluminadas, en el sentido correcto (orientación) y sin sombras, o empleando scanner**, de lo desarrollado colocando en la parte inferior derecha el documento de identidad (TIU, DNI, licencia de conducir, pasaporte o carnet universitario), revisarse si la resolución de la toma de imagen sea la suficiente para ver los detalles de la resolución. Cada imagen deberá de tener adjunto el documento de identidad, luego empieza el Microsoft Word para colocar toda la evidencia y posteriormente en formato PDF (opción 'grabar como' y seleccionar el formato de exportación) con el siguiente formato de **nombre de archivo**:

EL256_[tu sección]_[tu apellido]_[tu nombre]_[código]_2022_EREC.pdf

Ejemplo: EL256_EL59_Perez_Carlos_u200222838_2022_EREC.pdf

Se obtendrá la calificación mínima en la evaluación de no cumplir esta indicación o hacer ediciones de imagen post toma de la fotografía, sin posibilidad a reclamos posteriores.

• Tener cuidado que solo hay un intento de subida de archivo, pasado el tiempo se bloqueará el sistema de subida. Se recomienda subir la documentación **20 minutos antes del término del periodo de tiempo** al Aula Virtual. Verificar si se subió correctamente (abrir el archivo subido en la sección de calificaciones) y no aparezca el símbolo de "subida en proceso" (). Envíos de los archivos vía correo electrónico u otro medio que no sea el Aula NO serán tomados en cuenta y no se considerarán en la evaluación.

• Envíos monogramas con **TARDANZA** por el AV tendrán una penalización de -10p.

• Mantener iniciada la sesión de videoconferencia mientras dure la evaluación. **No emplear el chat global o audio.** De desconectarse por más de 5 minutos deberá reportarlo al profesor encargado de la evaluación ó delegado caso contrario se anulará la evaluación sin posibilidad a reclamo posterior.

• No habrá consultas relacionadas con el contenido de las preguntas. Deberán de emplear el criterio ingenieril para dar solución al problema.

• Tener en consideración el reglamento de evaluaciones en línea publicado en el AV.

• Se tendrá **visitas inopinadas** de supervisores de la universidad por lo que deberán tener el móvil apagado y en modo silencioso. En caso de una eventual solicitud de activación de dichos medios, El supervisor reportará al profesor para la **suspensión de la evaluación** en caso el alumno haga caso omiso o encuentre **faltas a la probidad** de la evaluación tal como lo estipula el reglamento de disciplina de alumnos SICA-REG-26.

• Luego de haber cumplido con la carga de la documentación al Aula Virtual dejar un mensaje de constancia en el **chat** de la videoconferencia antes de retirarse. La omisión a esta indicación representará una penalidad de -5p en su calificación obtenida sin lugar a reclamo posterior.

Sobre la actitud frente al estudio

- La importancia de la actitud positiva frente al estudio y el tiempo de dedicación (Nº de horas semanales de estudio = Nº de créditos del curso)
- Que para aprender es muy importante preguntar, cuestionar, participar, no importa equivocarse.
- La importancia de organizarse y habituarse desde la primera semana
- Tener presente y acudir a los talleres (gratuitos) del Área de Orientación Psicopedagógica: Organización del Tiempo, Manejo del Estrés, Manejo de la Ansiedad, etc. (consultar con profesor de campus)

Requerimientos generales del curso:

- Software:
 - Suite Microsoft Office (para la elaboración de informes)
 - Microchip MPLABX IDE v6.25
 - Microchip XC8 v3.00
 - PuTTY v0.83
 - Labcenter Proteus v8.17 en adelante
 - EaseEDA
 - Autodesk Fusion
- Hardware:
 - Computador Workstation de buen desempeño con Windows 10 ó 11 y elementos multimedia y acceso a Internet
 - Lo detallado en la lista de materiales
- Documentación:
 - Manejo de las hojas técnicas de dispositivos electrónicos y demás herramientas (requisito idioma inglés)

Requerimientos generales del curso:

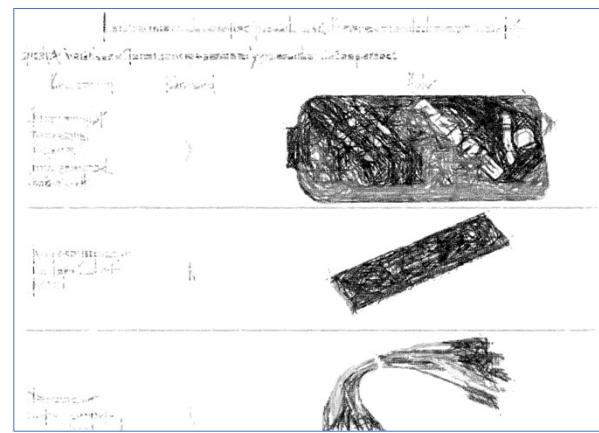
- Conocimientos previos:
 - Teoría de circuitos eléctricos (ley de Ohm)
 - Teoría de circuitos lógicos digitales (álgebra de Boole)
 - Desarrollo de algoritmos en diagramas de flujo
 - Programación de computadoras (lenguaje C)
 - Implementación de prototipos de circuitos electrónicos en protoboard
 - Uso correcto de instrumentos de medición y herramientas de laboratorio (multímetro, osciloscopio, etc)
 - Dibujo CAD (schematic diagram)

Detalle de software y documentación :

- Software:
 - Microchip MPLAB X v6.25 (solo soporte a PICKIT4 en adelante, no soporta MPASM, se puede emplear el PICKIT3 pero se necesita instalar manualmente los drivers)
 - Microchip XC8 v3.00 (la más actual)
 - Proteus soporta microcontroladores PIC18 Q10
- Documentación inicial:
 - Hoja técnica del PIC18F57Q43:
 - <https://ww1.microchip.com/downloads/aemDocuments/documents/MCU08/ProductDocuments/DataSheets/PIC18F27-47-57Q43-Data-Sheet-40002147F.pdf>
 - PIC18F57Q43 Curiosity Nano Hardware User Guide
 - <https://ww1.microchip.com/downloads/en/DeviceDoc/PIC18F57Q43-Curiosity-Nano-HW-UserGuide-DS40002186B.pdf>
 - Microchip PIC18F47Q10 Datasheet rev.F:
 - <https://ww1.microchip.com/downloads/aemDocuments/documents/MCU08/ProductDocuments/DataSheets/PIC18F27-47Q10-Microcontroller-Datasheet-DS40002043.pdf>
 - PIC18F47Q10 Curiosity Nano Users Guide:
 - <https://ww1.microchip.com/downloads/aemDocuments/documents/MCU08/ProductDocuments/UserGuides/PIC18F47Q10-Curiosity-Nano-Hardware-User-Guide-40002103.pdf>

Lista de materiales

- Disponible en el AV
- Mínimo de materiales para el desarrollo de los laboratorios.
- Se emplearán tanto para las sesiones de laboratorio como para las asignaciones y prácticas.



Grupo público de Microcontroladores

Nota: No es empleado para comunicaciones oficiales de la asignatura



Introducción a los microcontroladores

Dar respuesta a estas cuatro interrogantes:

- ¿Qué es un microcontrolador?
- ¿Cuáles son sus principales características?
- ¿Qué componentes lo conforman?
- ¿En qué equipos electrónicos los puedes encontrar?

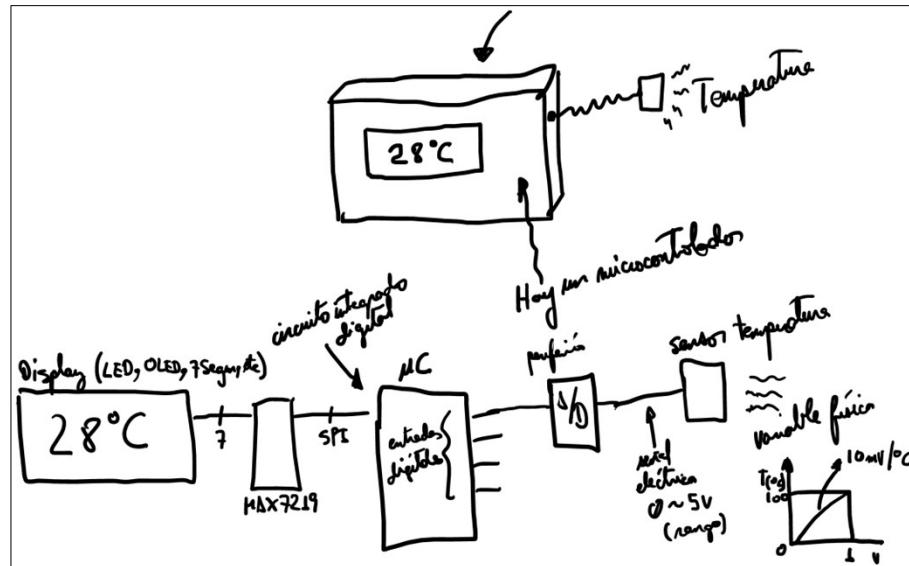
A lo largo del tiempo...

- Computadoras a escala compacta (Intel 1971 – 4004)
- Microcontroladores (computadora compacta)
- Sistemas Embebidos
- IoT (Internet of Things)

Respuestas:

- El microcontrolador es un dispositivo electrónico basado en semiconductor de naturaleza secuencial, programable, que integra los principales componentes de un sistema de procesamiento de datos (CPU, Memoria y Puertos de E/S) junto a periféricos de soporte (gestión energética, protección ante fallos, driver de oscilador, temporizadores, A/D, PWM, comunicación serial, etc) y que junto a un firmware desarrollado en un entorno de desarrollo permitirá desempeñar una tarea personalizada.

Caso: Un termómetro



¿Qué es un microcontrolador?

- Dispositivo electrónico programable, un circuito integrado microelectrónico basado en semiconductores.
- Posee casi todos los componentes necesarios para que funcione de manera autónoma.
- Presente en casi todos los equipos electrónicos utilizados en la actualidad
- Los microcontroladores son para aplicaciones específicas, quiere decir que la elección del microcontrolador será en base a las especificaciones de la aplicación.

¿Qué es un microcontrolador?

- Diego Estrella: es un dispositivo electrónico capaz de llevar a cabo procesos lógicos para desempeñar una tarea específica.
- Frank Panduro: Es un circuito integrado programable
- Giancarlo León: es un tipo de circuito que permite realizar distintas tareas específicas dentro de un sistema electrónico
- Alvaro Pumasupa: Un microcontrolador es un dispositivo en un único circuito integrado capaz de ser programado

¿Cuáles son sus principales características?

- Bajo consumo energético
- Pequeño
- Integra varios componentes para un funcionamiento autónomo
- Orientado exclusivamente a aplicaciones de control, no necesariamente podrá realizar procesamiento de señales.

¿Qué componentes lo conforman?

- La unidad central de proceso, procesador ó microprocesador
- Memoria: Espacio para los datos, espacio para el programa
- Periféricos de E/S: Interacción con el mundo exterior..
 - ADC, DAC
 - PWM
 - Temporizadores
 - Módulos de comunicación serial
 - I2C
 - UART
 - SPI
 - 1-wire
 - CAN

¿En qué equipos electrónicos lo puedes encontrar?

- Lavadora de prendas de vestir
- Televisores
- Computadoras
- Cámaras de vigilancia
- Aire Acondicionado
- Motos
- Celulares
- Scooters eléctricos
- Calculadora
- Relojes digitales
- Termas de baño, etc, etc

Brainstorm

- Circuito integrado programable
- Realiza una tarea
- Permite el manejo de circuitos a otros circuitos
- Realiza tareas específicas y ejecuta programas almacenados en su memoria
- Soporta voltajes bajos
- Automatizador de funciones
- Permite controlar periféricos como sensores
- Reprogramable

Brainstorm

- Circuito integrado programable
- Sistema digital
- Puertos de E/S, buses, CPU, memoria
- Controlar un circuito eléctrico
- Chip que ejecuta órdenes establecidas en memoria
- Controlar elementos de E/S
- Componente de bajo consumo
- Tiene un procesador
- Es una computadora de tamaño reducido
- Orientado a aplicaciones muy específicas

Brainstorm 2

<p>¿Microcontrolador?</p> <ul style="list-style-type: none"> Circuito integrado ✓ Dispositivo ✓ Uau Sistema electrónico Circuito programable Chipsito ✓ PIC Cerebro del circuito Procesador con módulos ✗ Dispositivo lógico programable Tiene sistema de entrada y salida ✗ Estómago porque procesa info ✗ Convertidor analógico a digital Medición de intervalos y sincronización 	<p>Integrado programado por registros</p> <ul style="list-style-type: none"> Pequeña computadora Poco consumo de energía ✗ Chip de control Se usa en sistemas embebidos Control de procesos en el sistema Industria automotriz Ejecutar ordenes grabadas desde la memoria Inyección de combustible y sistema de frenado antibloqueo Controlar periféricos externos ✗ Proyectos mas óptimos y de uso mas específico En nuestros televisores
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Brainstorm

- ¿Microcontrolador?
 - Circuito pequeño que se puede programar
 - Circuito electrónico basado en semiconductor
 - Nos ayuda a automatizar procesos
 - Ejecuta un conjunto de reglas
 - Bajo consumo eléctrico
 - Presentes en la mayoría de equipos electrónicos con cierto grado de automatización.
 - Tiene todos los componentes funcionales para que opere de manera independiente.

¿Qué es un microcontrolador?

¿Qué es un microcontrolador? Version 2026-0

- Integra todos los componentes para un funcionamiento autónomo.
 - No necesitas memoria externa, no necesitas fuente de reloj, tiene periféricos internos como temporizadores, pwm, comunicación, etc que ayudan a que no se requieran cosas adicionales para hacer una aplicación de sistema embebido.
- Circuito integrado programable que funciona como una computadora
 - Los microcontroladores no vienen programados de fábrica.
- Sirve para controlar o realizar tareas pre-programadas.

¿Para qué se usan los microcontroladores? Version 2026-0

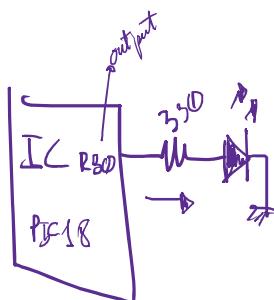
- Para que funcione mi cargador de celular
- Semáforos, sistemas de control de tracción (EBD) / frenado (ABS)
- Refrigeradores modernos, freidoras de aire (cocina)
- Drones, robots (sistemas no tripulados)
- Smartwatches (wearables)
- Monitores cardiacos, espirómetros, pulsioxímetros, tensómetros (biomédica)
- Etc etc etc etc, **TODO EQUIPO ELECTRÓNICO MODERNO TENDRÁ UN MICROCONTROLADOR**

¿Qué es un microcontrolador?

- Es un circuito integrado fabricado con tecnología microelectrónica (basado en semiconductor).
 - ARM Cortex-m (m0, m3, m33+, m4, m7)
 - ESP32, STM32F103, RPI Pico
 - RISC-V Hazard-3
 - RPI Pico 2 (RP2350)
- Contiene todos los componentes necesarios para funcionar como una pequeña computadora en un solo chip (componentes como CPU, memoria, E/S y demás periféricos).
- Es un dispositivo programable (necesitas un IDE, lenguaje de programación)
- Sirve para darle vida a los aparatos electrónicos mediante la ejecución de un programa.

¿Qué es un microcontrolador?

- Es un circuito electrónico basado en semiconductor programable (viene sin programar desde fábrica), posee todos los elementos para un funcionamiento autónomo.
- Internamente contiene un CPU (microprocesador), Memoria (programa y de datos), puertos de E/S (interface con el mundo exterior) y periféricos adicionales como temporizadores, pwm, comunicación serial, etc.

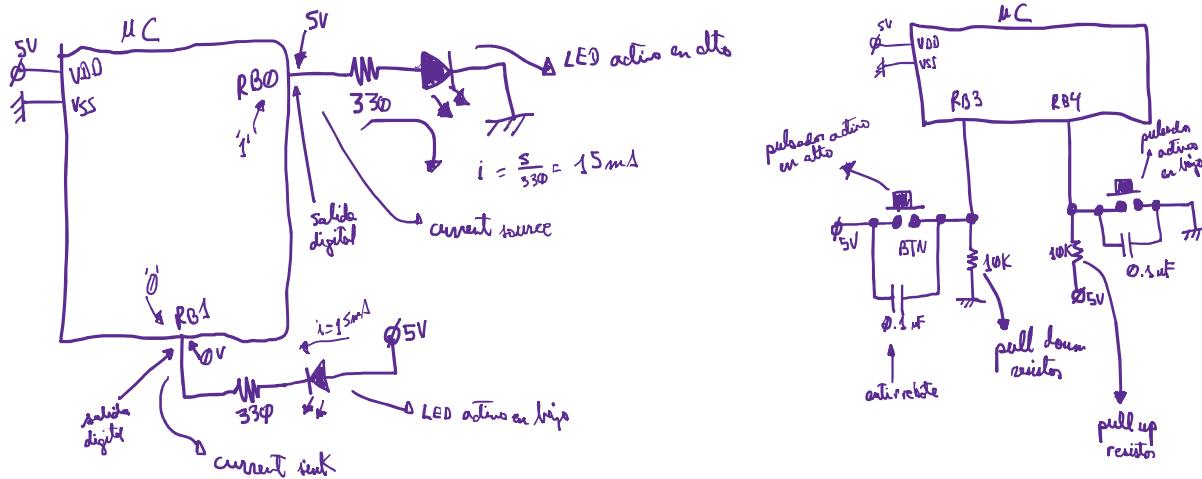


Sobre los puertos de E/S:
Entrada -> Alta Impedancia
Salida -> Baja Impedancia

¿Qué es un microcontrolador?

- Para programar un microcontrolador, en la mayoría de casos se requiere de una interface de programación (un programador). En el caso de Microchip se usa el PICKit5. El programador pasará el código desarrollado en una PC hacia la memoria de programa del microcontrolador.
- En el caso del Curiosity Nano PIC18F57Q43, éste tiene ya integrado la función de programador (Curiosity Debugger).
- Para hacer un programa que el microcontrolador haga una tarea, se debe de utilizar una IDE (entorno de desarrollo) que puede ser proporcionada por el fabricante. En nuestro caso usaremos el Microchip MPLAB X IDE.
- Se tiene que entrenar para poder hacer un programa en un lenguaje, los lenguajes mas utilizados para microcontroladores son: C y Assembler.

Revisión de interfaces de E/S



Analizando lo siguiente:

The Wokwi simulation interface shows an ESP32-S3-WROOM-1 board. Pin 40 (GND) is connected to the negative terminal of a red LED. Pin 41 (Digital 0) is connected to the positive terminal of the LED. A 33Ω resistor is connected between Pin 41 and the positive terminal of the LED. A current source is connected between Pin 41 and ground. The LED is labeled "LED activo en alto". A 4Ω resistor is also connected between Pin 41 and ground. The code in the sketch is as follows:

```

void setup() {
    // put your setup code here, to run once:
    Serial.begin(115200);
    Serial.println("Hello, ESP32-S3!");
    pinMode(40, OUTPUT);
}

void loop() {
    // put your main code here, to run repeatedly:
    //delay(10); // this speeds up the simulation
    digitalWrite(40, HIGH);
    delay(500);
    digitalWrite(40, LOW);
    delay(500);
}

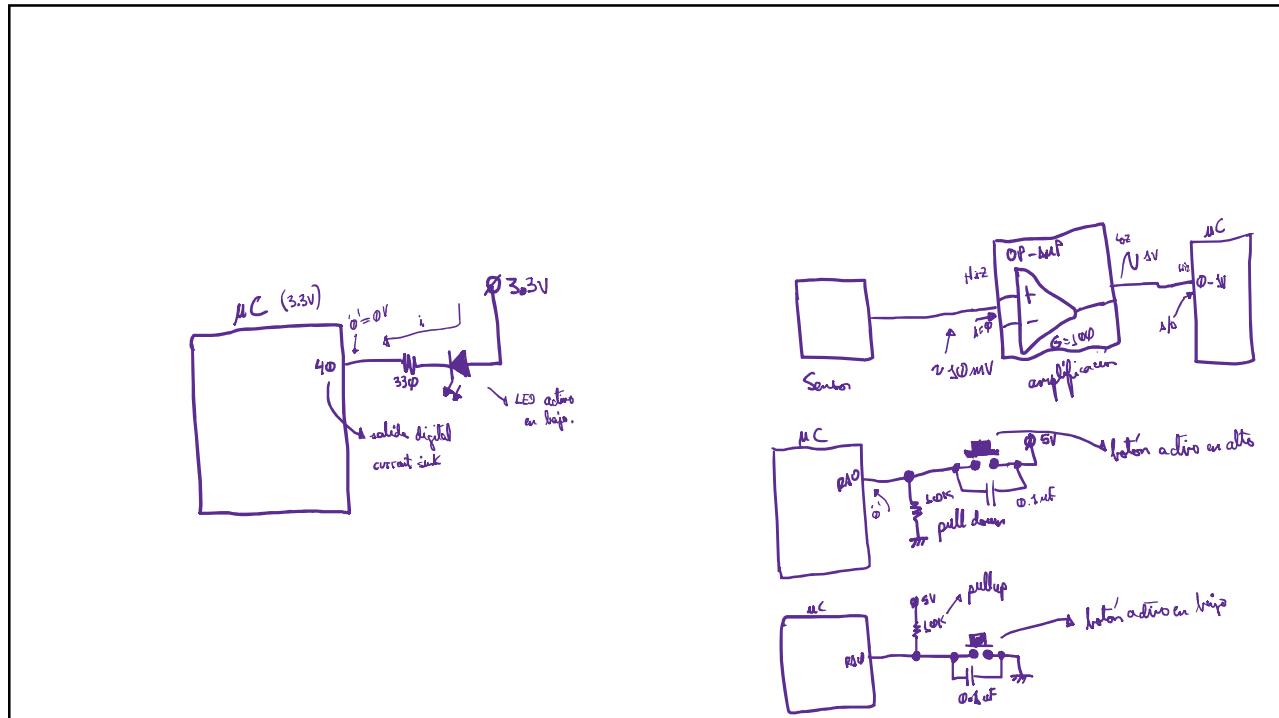
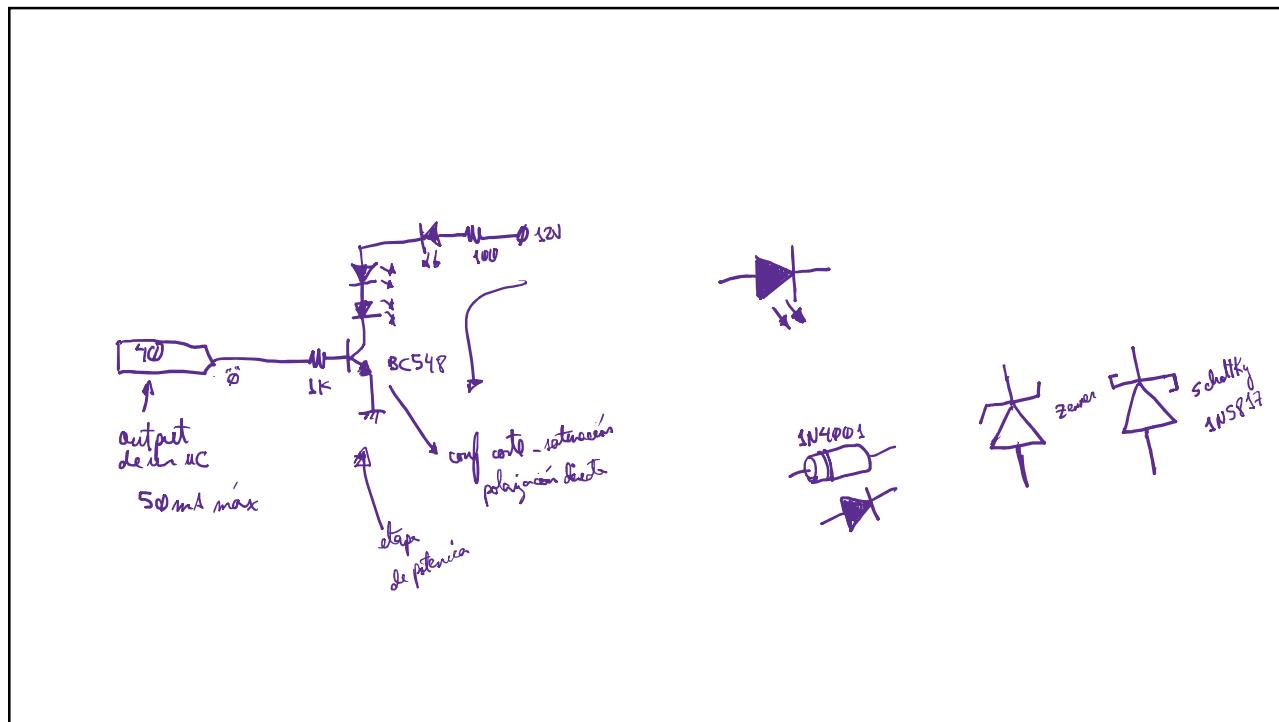
```

$$\text{LV-TTL} \rightarrow \begin{cases} 1 & -3.3V \\ 0 & 0V \end{cases}$$

$$V = RI$$

$$I = \frac{3.3}{33\Omega} = 10\text{mA}$$

LED activo en alto



¿Qué es un microcontrolador?

- Es un dispositivo microelectrónico (basado en tecnología de semiconductores) programable que posee casi todos los componentes para un funcionamiento autónomo.
- Componentes: CPU, memorias de programa y de datos (RAM y ROM), periféricos (E/S, temporizadores, A/D, etc), gestión energética, fuentes de reloj
- Se requiere de un programa (hecho con un lenguaje de programación desde un entorno de desarrollo en una PC) y de un programador para que transporte el código compilado hacia la memoria de programa del microcontrolador antes de iniciar su operación
- Su funcionamiento es de manera secuencial (necesita de una fuente de reloj).
- Para dar soluciones “compactas” (embedded) a determinado problema.
 - Portátil (autonomía, tamaño, consumo, etc)

¿Esto es un microcontrolador?



- Es una plataforma de desarrollo electrónico basado en un microcontrolador (ATMEL ATMega 328P)

¿Esto es un microcontrolador?



- Es una plataforma de desarrollo electrónico basado en un microcontrolador (Renesas RA4M1 ARM Cortex-M4)

¿Esto es un microcontrolador?



- Es una computadora de placa reducida basado en el procesador (SBC) Qualcomm Dragonwing QRB2210 interconectado con un microcontrolador STM32U585

¿Por qué no enseña Arduino en lugar de PIC? ¿Arduino no es más fácil?

- Arduino consume mas energía
- Arduino se programa a un nivel mas alto y por ende consume mas recursos de procesamiento.
- Arduino lo trabaja como si fuera una “caja negra”.
- Trabajar con PIC puedes alcanzar mayores niveles de eficiencia en términos de desempeño, costo, consumo energético, uso de memoria.
- Arduino es un entorno de desarrollo open-source el cuál el microcontrolador destino posee un firmware inicial para la interacción con el software IDE en la PC. Como consecuencia de esto el microcontrolador tendrá menor desempeño frente a usar lenguaje Assembler.
- Cuando se tiene que atender aplicaciones o procesos críticos, en Arduino no tenemos velocidad de respuesta a menos que se emplee microcontroladores de mayor desempeño. Esto no representaría problema alguno si se desarrolla en Assembler.

Entonces si usar Arduino presenta tantas desventajas. ¿Por qué se usa extensivamente?

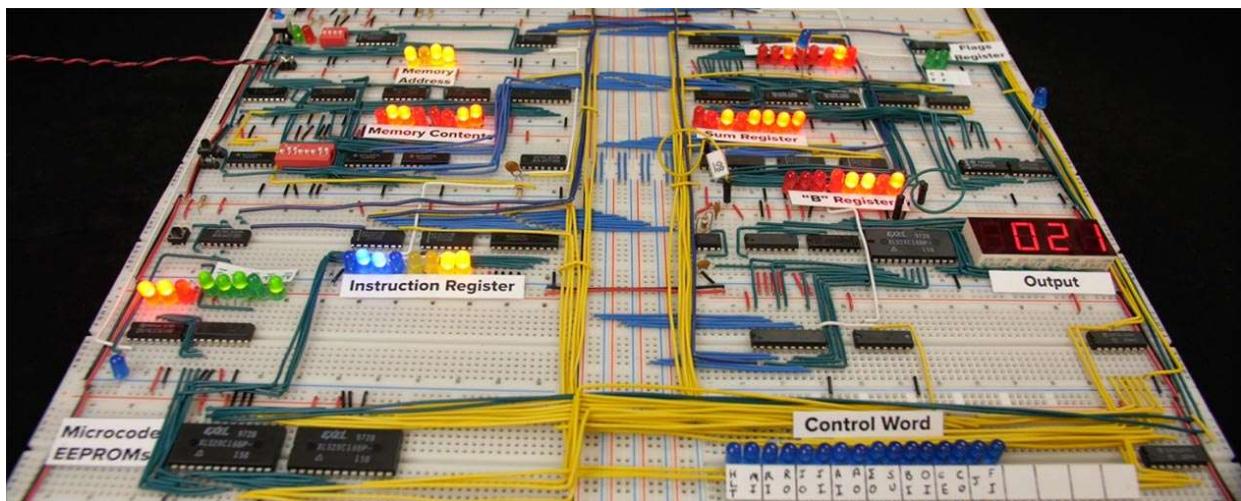
- Tendencia open source, open hardware.
- Por el poco tiempo que requieres para hacer una solución electrónica, sin tener prioridad en la optimización de recursos.
- Mucha experiencia y muchos usuarios (técnicos y no técnicos) usando esta plataforma.
- Bastante documentación de ejemplos y notas de aplicación basado en esta plataforma.
- Fabricantes OEM desarrollan variedad de sensores y actuadores en forma de módulos plug-in para ser usados en esta plataforma.

He visto que en la industria predomina el uso de PLCs. ¿Y los microcontroladores no son usados en esa área?

- Los PLCs son “computadoras” orientadas al sector industrial. En su interior encontrarás microcontroladores!
- Puedes encontrar plataformas de Arduino para la industria al mismo estilo que un PLC:

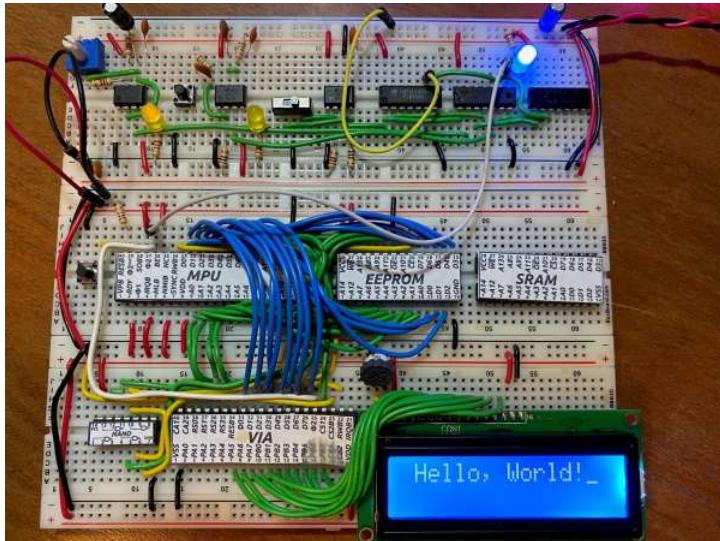


Evolución de los microcontroladores



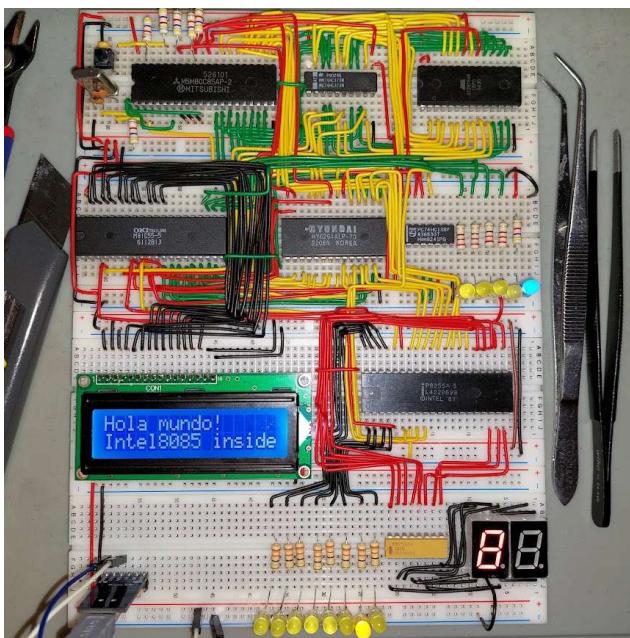
CPU de 8 bits con elementos digitales discretos. Desarrollado por Ben Eater

Evolución de los microcontroladores:



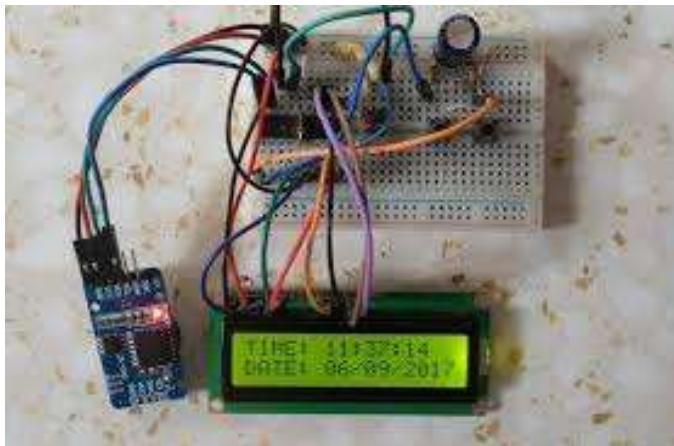
Plataforma de desarrollo para el 6502. Desarrollado por Ben Eater

Evolución de los microcontroladores:



Plataforma de desarrollo para el procesador Intel 8085. Desarrollado por Kalun Lau

Evolución de los microcontroladores



Prototipo basado en un microcontrolador moderno (mínima cantidad de componentes externos) conectado a un RTC DS3231 para una aplicación de reloj

Aplicaciones con microcontroladores



Aplicaciones con microcontroladores

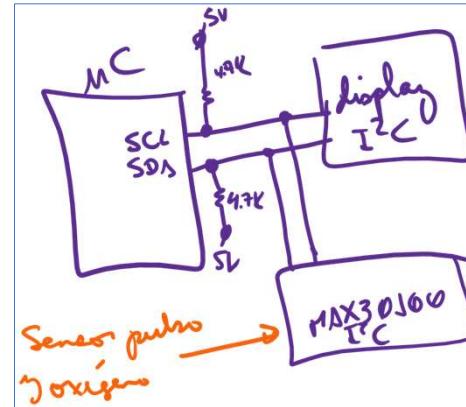


¿Y en dónde mas hay microcontroladores?

- Cargadores de celular (Quick charge, PD)
- Powerbanks (control de la sobrecarga/subdescarga de las celdas de Li-Ion)
- Tu cable USB-C (los que tienen PD)
- Linternas LED (las de mejor calidad)
- Ventiladores USB (los que tienen batería y control de intensidad)
- Refrigeradoras actuales
- Focos LED de iluminación de ambientes (de marcas como Philips, Osram, etc)
- Bicicletas eléctricas
- Scooters

COVID19

Ejemplo de aplicación empleando microcontroladores: Pulsioxímetro



COVID19

Pulsioxímetro
fabricado localmente



COVID19

Ejemplo de aplicación empleando microcontroladores: Proyecto UPC Phukuy

PROYECTO PHUKUY:

EL ESPIRÓMETRO QUE APOYA A PACIENTES CON COVID - 19

FUE ELABORADO POR INVESTIGADORES DE LA FACULTAD DE INGENIERÍA UPC.

CARACTERÍSTICAS:

- MIDE LA CANTIDAD DE OXÍGENO EN LOS PULMONES
- DE BAJO COSTO
- USO PERSONAL
- DE USO PORTABLE
- SEGUR

UPC exigete, innova

Microcontroladores

- Fabricantes
 - ¿Microchip?
 - #1 en ventas de microcontroladores de 8 bits
 - Portafolio inmenso de microcontroladores
 - PIC (10, 12, 16, 18, 24, 32)
 - AVR
 - Infineon (ex Cypress Semiconductor) – PSoC (PSoC 6)
 - NXP (ex Phillips)
 - TI (Texas Instruments)
 - Renesas
 - Freescale (ex Motorola Semiconductor)
 - Intel
 - ST Semiconductor
 - Líder en 32bits (STM32)
 - etc

Microcontroladores

- Herramientas de desarrollo
 - Software de Simulación: Proteus, Tina
 - Software de Desarrollo: Depende de la familia y fabricante
 - Microchip PIC: MPLABX (XC8 Assembler, XC8)
 - Microchip AVR: AVR Studio
 - ST Semiconductor STM32: STM32 CubeIDE
 - Cypress Semiconductor PSoC: PSoC Creator
 - Multiplataforma: Arduino
 - Micropython: Raspberry Pi Pico, ESP32, A9G

Importancia del algoritmo

- Los algoritmos son representaciones gráficas de una tarea que va a hacer el microcontrolador.
- Pueden ser representados en diagramas de flujo, NS, pseudocódigo.
- En el presente curso se hará uso de diagramas de flujo (flowchart)

A tomar en cuenta:

- El proceso de aprendizaje de un lenguaje de programación es lento y progresivo, **no hay atajos**, mientras mas practiques, mas experiencia acumulas.
- Las ayudas como ChatGPT o algún otro entorno de IA no te va a ayudar a implementar correctamente un circuito electrónico.

Repaso de conocimientos previos

- Álgebra de Boole, circuitos digitales (Fund. Sist. Digit. Thomas Floyd)
- Algoritmos, diagramación en diagrama de flujo
- Arquitectura de computadoras (Org. Y Arq de PCs de William Stallings)
- Circuitos eléctricos (interfaces de potencia, sensores)
 - Transistores en corte y saturación, diodos rectificadores, LEDs.
 - Optoacopladores
 - Relés
- Señales analógicas y digitales (señales y sistemas)
 - Op-Amp: Modos de trabajo (amplificador (noinv, inv), oscilador, comparador, sumador, integrador, diferencial, compresor, filtrado, etc)
 - Conversión A/D y D/A, Teorema de muestreo

Bases numéricicas

bcd

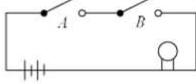
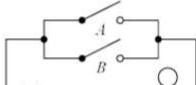
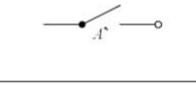
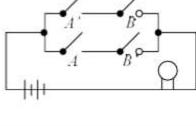
Decimal	Binario	Octal	Hexa
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

BIN = 11010100B
HEX = D4H

HEX = A5H
BIN = 10100101

HEX = FBH
BIN = 11111011

Álgebra de Boole

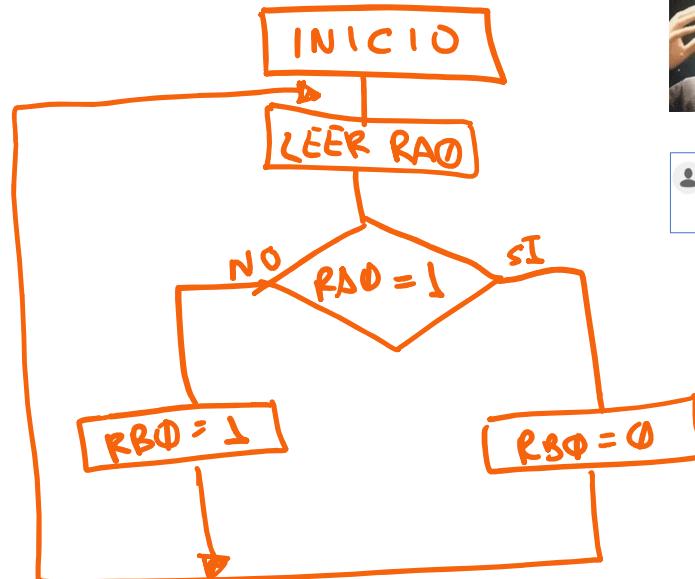
Expresión	Compuerta Lógica	Tabla de Verdad	Circuito de Interruptores															
$X = AB$	 AND	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>X</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table>	A	B	X	0	0	0	0	1	0	1	0	0	1	1	1	
A	B	X																
0	0	0																
0	1	0																
1	0	0																
1	1	1																
$X = A + B$	 OR	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>X</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table>	A	B	X	0	0	0	0	1	1	1	0	1	1	1	1	
A	B	X																
0	0	0																
0	1	1																
1	0	1																
1	1	1																
$X = A'$	 NOT	<table border="1"> <thead> <tr> <th>A</th> <th>X</th> </tr> </thead> <tbody> <tr><td>0</td><td>1</td></tr> <tr><td>1</td><td>0</td></tr> </tbody> </table>	A	X	0	1	1	0										
A	X																	
0	1																	
1	0																	
$X = A \oplus B$ $\Rightarrow X = A'B + AB'$	 XOR (OR exclusivo)	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>X</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table>	A	B	X	0	0	0	0	1	1	1	0	1	1	1	0	
A	B	X																
0	0	0																
0	1	1																
1	0	1																
1	1	0																

Axiomas del Álgebra de Boole	
Leyes Comutativas	$a + b = b + a$
	$a * b = b * a$
Leyes Distributivas	$a + (b * c) = (a + b) * (a + c)$
	$a * (b + c) = (a * b) + (a * c)$
Leyes de Identidad	$a + 0 = a$
	$a * 1 = a$
Leyes de Complemento	$a + a' = 1$
	$a * a' = 0$
Leyes de Idempotencia	$a + a = a$
	$a * a = a$
Leyes de Acotamiento	$a + 1 = 1$
	$a * 0 = 0$
Leyes de Absorción	$a + (a * b) = a$
	$a * (a + b) = a$
Leyes Asociativas	$(a + b) + c = a + (b + c)$
	$(a * b) * c = a * (b * c)$
Unidad del Complemento	Si $a + x = 1$ y $a * x = 0$, entonces $x = a'$
Ley de Involución	$(a')' = a$
Teoremas	$0' = 1$
	$1' = 0$
Leyes de DeMorgan	$(a + b)' = a' * b'$
	$(a * b)' = a' + b'$

Diagramas de flujo

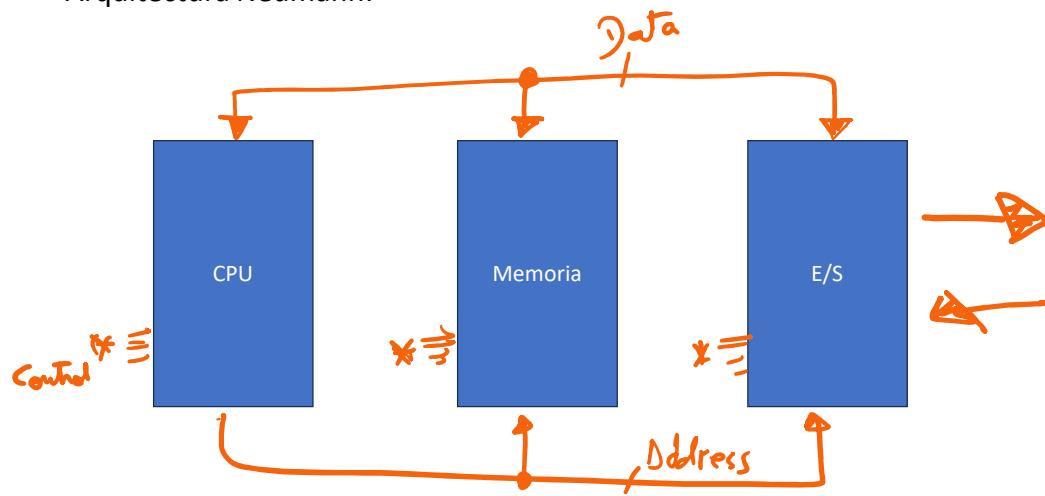


Josue Andre Perez Rodriguez 2:35 PM
un NOT secuencial
xd



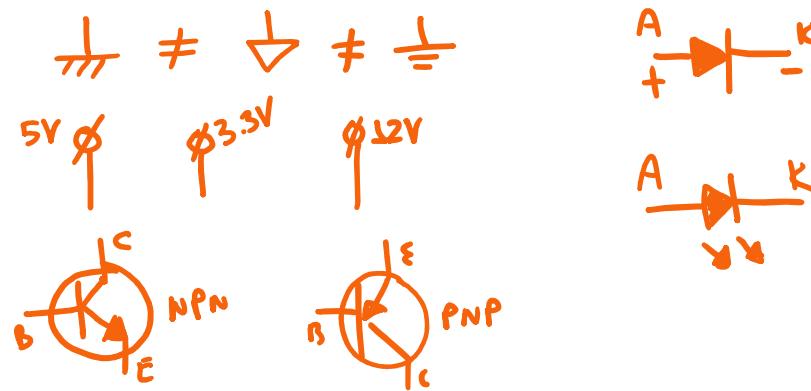
Arquitectura de Computadores

- Estructura de una computadora
 - Arquitectura Neumann:



Circuitos Eléctricos

Usar la simbología correcta al momento de desarrollar los diagramas esquemáticos de circuitos electrónicos



Agenda 22/01/2026 15:00

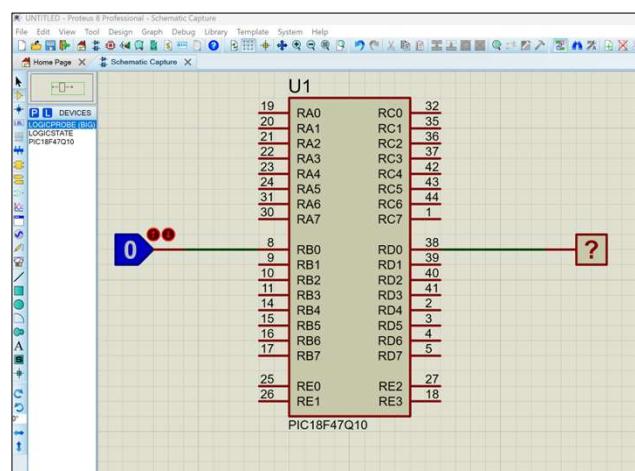
- El primer ejemplo empleando el módulo Curiosity Nano PIC18F47Q10 para que funcione un negador lógico de 1 bit

Ruta (workflow) para el desarrollo de prototipos de aplicaciones con microcontroladores:

1. Análisis de los requerimientos de la aplicación (prestaciones, consumo energético, puertos de E/S, funcionalidades, expandibilidad, actualizaciones a futuro, etc).
2. Desarrollo del hardware
 - a) Prototipado en físico usando protoboard
 - b) Prototipado en simulador (Proteus)
3. Desarrollo del algoritmo en diagrama de flujo
4. Codificación del algoritmo en un lenguaje de programación (XC8)
5. **Simulación (en lo posible) en un software como Proteus**
6. Pruebas en físico (previo grabado del firmware en la memoria del microcontrolador) como en simulación.
7. Elaboración de PCB (Autodesk Eagle).
8. Elaboración de carcasa (Autodesk Fusion) y su impresión en 3D.

Desarrollar un negador lógico en un microcontrolador PIC18 Q10 o Q43

- Si se desarrolla en el Curiosity Nano PIC18F47Q10 se podrá hacer la simulación en el Proteus

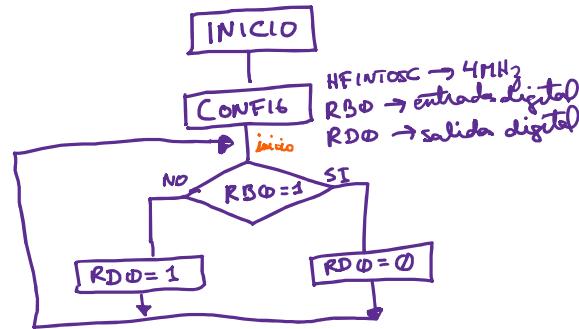


Desarrollar un negador lógico en un microcontrolador PIC18 Q10 o Q43

- Diagrama de flujo

Un negador lógico: 

$$\begin{array}{c|cc} & A & B \\ \hline 0 & 1 & 1 \\ 1 & 1 & 0 \end{array}$$



Desarrollar un negador lógico en un microcontrolador PIC18 Q10 o Q43

- Programa en el XC8 PIC Assembler

<pre> 1 PROCESSOR 18F47Q10 2 #include "cabecera.inc" 3 4 PSECT upcino, class=CODE, reloc=2, abs 5 upcino: 6 ORG 000000H ;Lo primero que el CPU ejecuta cuando se energiza, el vector de reset 7 bra configuro ;salta a etiqueta configuro 8 9 ORG 000100H ;Zona de programa de usuario 10 configuro: 11 ;Conf del módulo de oscilador 12 movlw 0EH ;Al Bank14 13 movlw 60H 14 movwf OSCCON1, 1 ;NOSC=HFINTOSC, NDIV 1:1 15 movlw 02H 16 movwf OSCFRQ, 1 ;HFINTOSC a 4MHz 17 movlw 40H 18 movwf OSCEN, 1 ;HFINTOSC enabled 19 ;Conf las E/S 20 movlw 0FH ;Al Bank 15 21 bsf TRISB, 0, 1 ;RB0 entrada 22 bcf ANSELB, 0, 1 ;RB0 digital 23 bcf TRISD, 0, 1 ;RD0 salida 24 bcf ANSELD, 0, 1 ;RD0 digital </pre>	<pre> 26 inicio: 27 btfss PORTB, 0, 1 ;Pregunto si RB0 es uno 28 bra noescierto ;Falso a si RB0=1 29 bra escierto ;Verdad a si RB0=0 30 31 noescierto: 32 bsf LATD, 0, 1 ;RD0 = 1 33 bra inicio ;salto a inicio 34 35 escierto: 36 bcf LATD, 0, 1 ;RD0 = 0 37 bra inicio ;salto a inicio 38 39 end upcino </pre>
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

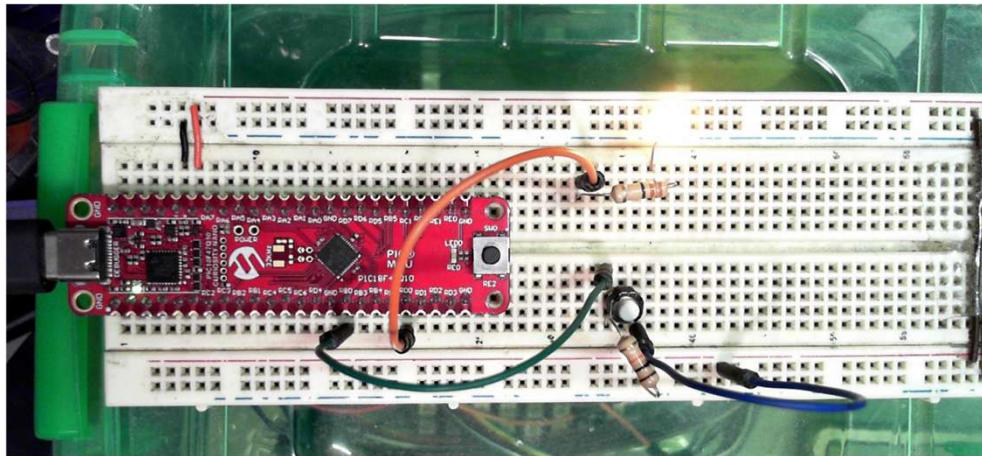
Desarrollar un negador lógico en un microcontrolador PIC18 Q10 o Q43

Comentarios:

- Tener en cuenta que los puertos de entrada/salida pueden ser entradas o salidas determinado por el registro TRISx, pueden ser analógicos o digitales determinado por el registro ANSELx. Por defecto las E/S son entradas analógicas.
- Según 11.3 de la hoja técnica, los registros de funciones especiales (SFR) se encuentran a partir de la dirección 0E1FH y se extiende hasta OFFFH, diferente a lo que posee los Q43. Por lo que hay que revisar a detalle en qué banco se encuentran los registros a modificar.
- Ese es el motivo por el cual para acceder a los registros de configuración del oscilador se pasó al Bank14 (BSR=0EH) y para acceder a la manipulación de las E/S se pasó al Bank15 (BSR=0FH)

Desarrollar un negador lógico en un microcontrolador PIC18 Q10 o Q43

- Implementación con el Curiosity Nano PIC18F47Q10



Desarrollar un negador lógico en un microcontrolador PIC18 Q10 o Q43

Preguntas:

- Al leer los comentarios del código desarrollado. ¿Tiene relación dicho código con el diagrama de flujo?
- Según lo experimentado. ¿Hay alguna diferencia funcional entre usar el Proteus y la implementación en la vida real?
- ¿A qué frecuencia está trabajando el CPU según lo estipulado en la configuración del módulo de oscilador?
- En el circuito implementado. ¿Por qué no se configuró el weak pull-up en RB0?

Retroalimentación

Preguntas:

- Se tiene el siguiente código:

```
unsigned int funcionaso(void){
    unsigned int salida = 3;
    for(unsigned int x = 0; x < 5; x++){
        if(x % 2 == 0){
            salida = salida + x;
        }
        else{
            salida = salida - x;
        }
    }
    return salida;
}
```

¿Cuál es el valor resultante?

- a) 3 b) 5 c) 0 d) 2

Fin de sesión

- Evaluación de entrada

Agenda 23/01/2026 11:00

- La familia PIC18 de Microchip
- El Curiosity Nano PIC18F57Q43
- El Curiosity Nano PIC18F47Q10
- Manejo de la documentación (hojas técnicas y repositorios)
- El MPLAB X IDE
- Segundo ejemplo simulado e implementado

Preguntas previas:

- ¿Debo de tener los materiales para esta semana?
 - Esta semana 2 iniciaremos con las experiencias de laboratorio con el PIC18F57Q43 de manera dirigida, es decir, el profesor hará toda la experiencia de implementación. Se recomienda tener los materiales para las experiencias de la semana 3.
- El lenguaje Assembler para el microcontrolador lo debemos de saber previamente o lo vamos a ver en las clases?
 - Se los va a atender durante las sesiones con el compromiso de que el alumno practique.
- ¿Cuáles son los softwares que necesito instalar en mi PC?
 - ~~MPLAB X versión 6.20 (la mas actual)~~
 - ~~XC8 versión 2.50 (la mas actual)~~
 - MPLABX v6.25, XC8 v3.10
- ¿Qué cable se usa para el Curiosity Nano?
 - Cable USB a MicroUSB de datos (versión antigua)
 - Cable USB a USC-C (versión 2024)

Preguntas previas:

- ¿Hay algún libro sobre este nuevo microcontrolador?
 - No
- Al conectar el Curiosity Nano a la PC solo se enciende, no lo detecta. ¿Qué puede ser?
 - El cable USB que has usado no transporta datos por ende no lo reconoce, cambia de cable.
- ¿Cómo puedo saber si mi PC ha detectado correctamente el Curiosity Nano?
 - Al conectar el Curiosity Nano a la PC, si es que es la primera vez, la PC instalará de manera automática los controladores (drivers) y aparecerá como popup una ventana de explorador de archivos de Windows y se visualizará una unidad de disco correspondiente al Curiosity Nano. (Verificado en Windows 10 y 11)
- ¿El microcontrolador se usa en la industria o solamente con fines académicos? ¿Se puede conectar otro microcontrolador al microcontrolador que estamos usando mediante pines?
 - Los microcontroladores los encuentras en casi todos los equipos electrónicos, desde un reloj hasta el equipo electrónico mas avanzado.
 - Es posible conectar dos microcontroladores y establecer un canal de comunicación entre ellos, mediante el uso de comunicación serial en sus diferentes tipos: I2C, SPI ó UART.

Preguntas previas:

- ¿Es necesario tener un osciloscopio en el curso? ¿Alguna referencia de modelos de scopes pequeños?
 - Es opcional y nos servirá para visualizar señales que cambian de forma en el tiempo, como el PWM.



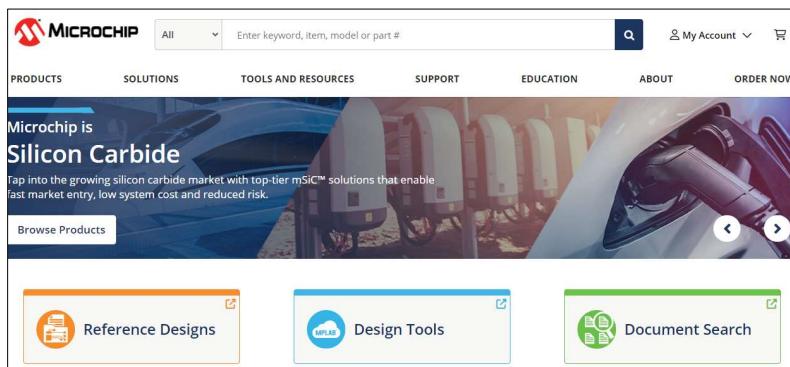
Preguntas previas

- ¿Pueden coexistir diferentes versiones de MPLABX y XC8 en una misma PC?
 - Si, pero de preferencia dejar la versión mas actual solamente a fin de evitar eventuales conflictos.
- Al terminar la instalación del MPLABX me aparece la opción de descargar e instalar el XC8, XC16 y XC32. ¿Cuál de todos selecciono?
 - Solo el XC8
- Los bits de configuración que se han modificado. ¿Serán los mismos para cada proyecto que vayamos a realizar?
 - Si, siempre y cuando sea el mismo modelo de microcontrolador, se configuran los mismos 5 bits de configuración para cada proyecto.

Mercado de los microcontroladores

- Actualmente hay preferencias en usar microcontroladores de 8 bits y de 32 bits:
 - Aplicaciones de baja a mediana escala de complejidad -> 8bits
 - Aplicaciones de mediana a alta escala de complejidad -> 32bits
- Se ha disminuido el uso de microcontroladores de 16 bits, muy posible debido a similitud de costos de manufactura y estrecha relación con los de 32 bits.
- Los precios entre micros de 32bits y 8 bits están muy similares.
- Los µC de 8 bits naturalmente consumen menos energía.

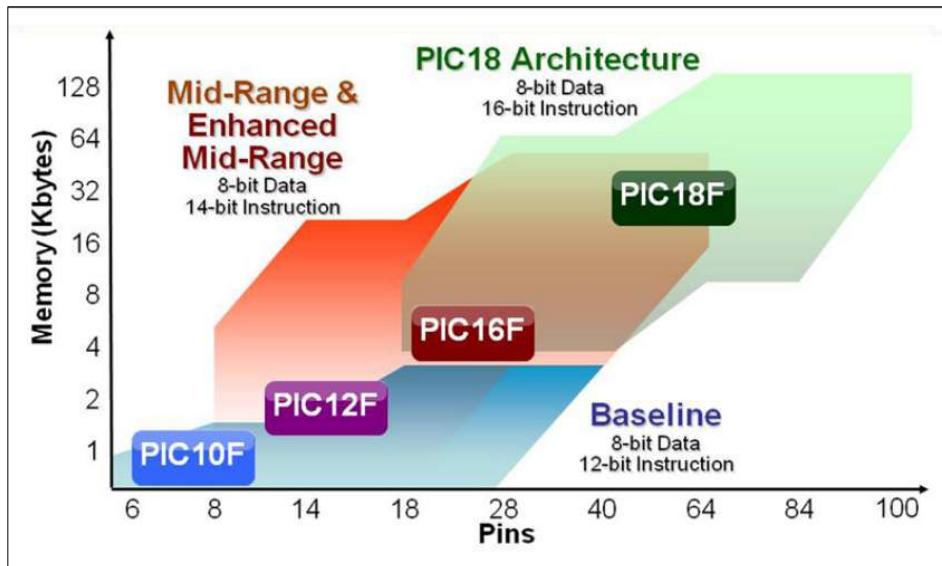
Microchip INC



- División de semiconductores de General Instrument, en 1989 se independizó y se formó Microchip INC.
- Actual líder en el mercado de microcontroladores de **8 bits**.
- En Julio del 2016 compró Atmel y todo su portafolio de productos.
- También adquirió Micrel, Microsemi, KeeLoq, HI-TECH Software, Novocell, ISSC, etc.

Microchip INC

- Familias de microcontroladores PIC de 8 bits



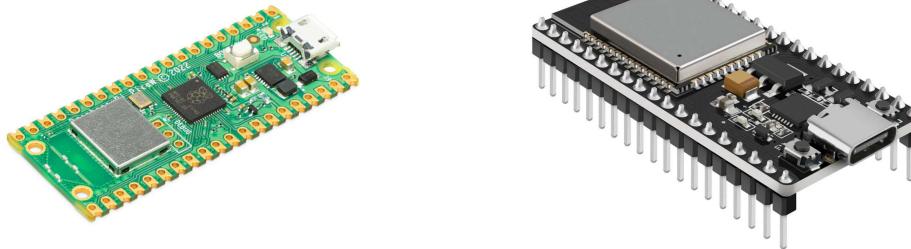
ST Microelectronics

- Líder en microcontroladores de 32bits (STM32)

STM32 MCUs 32-bit Arm® Cortex®-M			
	STM32F7	STM32H7	LONGEVITY • 10 YEARS • COMMITMENT
High Performance	1082 CoreMark 216 MHz Cortex-M7	Up to 3224 CoreMark Up to 550 MHz Cortex-M7 240 MHz Cortex-M4	
	STM32F2	STM32F4	STM32H5
	398 CoreMark 120 MHz Cortex-M3	608 CoreMark 180 MHz Cortex-M4	Up to 1023 CoreMark 250 MHz Cortex-M33
	STM32G0	STM32B4	
Mainstream	142 CoreMark 64 MHz Cortex-M0+	569 CoreMark 170 MHz Cortex-M4	
	STM32C0	STM32F0	
	114 CoreMark 48 MHz Cortex-M0+	106 CoreMark 48 MHz Cortex-M0	
	STM32F1	STM32F3	Optimized for mixed-signal applications
	177 CoreMark 72 MHz Cortex-M3	245 CoreMark 72 MHz Cortex-M4	
	STM32L4+	STM32U5	
Ultra-low-power	409 CoreMark 120 MHz Cortex-M4	651 CoreMark 160 MHz Cortex-M33	
	STM32L0	STM32L4	
	75 CoreMark 32 MHz Cortex-M0+	273 CoreMark 80 MHz Cortex-M4	
	STM32L5		
	443 CoreMark 110 MHz Cortex-M33		
	STM32WL	STM32WB	
Wireless	162 CoreMark 48 MHz Cortex-M4 48 MHz Cortex-M0+	216 CoreMark 64 MHz Cortex-M4 32 MHz Cortex-M0+	
	STM32WB	STM32WBA	
	407 CoreMark 100 MHz Cortex-M33		
		Cortex-M0+ Radio co-processor	

Nuevos contrincantes en los micros de 32 bits:

- Raspberry Pi Foundation: RP2040 y RP2350
 - Costo muy reducidos
 - Flexibilidad
- Xtensa Tensilica LX7: ESP32
 - Ampliamente utilizado para aplicaciones de IoT



Familia PIC18 de Microchip

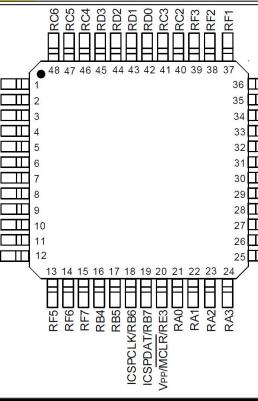
- Primeros dispositivos PIC18: ej. PIC18F452, PIC18F458
- Siguiente generación de PIC18: ej. PIC18F4550
- Actualización de generación de PIC18: Familia K ej. PIC18F45K50
- Familias “Q” de PIC18: Q10, Q40/41, Q43, Q71, Q83/84 ej. PIC18F57Q43
- Nuevas familias Q PIC18: Q20, Q24 (released 27/11/2023)

Product Family	Pin Count	Program Flash Memory (kB)	RAM (kB)	Data EEPROM (kB)	8-bit PIC® Microcontrollers												
					Intelligent Analog		Waveform Control ⁽¹⁾		Logic and Math		Safety and Monitoring		Communications		User Interface		Low Power and System Flexibility
PIC18-Q40/41	14-20	16-64	1-4	512	12 ⁽²⁾	✓	8	✓ ⁽³⁾	✓	✓	✓	✓	✓	✓	✓	✓	✓
PIC18-Q43	28-48	32-128	2-8	1024	12 ⁽²⁾	✓	8	✓	✓	✓	3	✓	✓	✓	✓	✓	✓
PIC18-Q83 ⁽²⁾ /84 ⁽²⁾	28-48	64-128	8-13	1024	12 ⁽²⁾	✓	8	✓	✓	3	✓	✓	✓	✓	✓	✓	✓
PIC18-Q10	28-40	16-128	1-3.6	256-1024	10 ⁽²⁾	✓	5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
PIC18-Q71	28-48	16-64	1-4	256	12 ⁽²⁾	✓	8 ⁽³⁾ /10	2	✓	✓	3	✓	✓	✓	✓	✓	✓

El Microcontrolador PIC18F57Q43

Device	Program Memory Flash (bytes)	Data SRAM (bytes)	Data EEPROM (bytes)	Memory Access Partition/ Device Information Area	I/O Pins/ Peripheral Pin Select	8-Bit Timer with HLT/ 16-Bit Timers	16-Bit Dual PWM/ CCP	Complementary Waveform Generator	Signal Measurement Timer	Numerically Controlled Oscillator	Configurable Logic Cell	12-Bit ADC/C (channels)	8-Bit DAC	Comparator/ Zero-Cross Detect	High-Low Voltage Detect	UART/ SPI/2_C	UART with Protocol Support	Direct Memory Access (DMA)	Windowed Watchdog Timer	16-Bit CRC with Scanner	Vectorized Interrupts	Peripheral Module Disable	Temperature Indicator
PIC18F27Q43	128k	8192	1024	Y/Y	25/Y	3/4	3/3	3	1	3	8	24	1	2/1	1	2/1	4/1	6	Y	Y	Y	Y	Y
PIC18F47Q43	128k	8192	1024	Y/Y	36/Y	3/4	3/3	3	1	3	8	35	1	2/1	1	2/1	4/1	6	Y	Y	Y	Y	Y
PIC18F57Q43	128k	8192	1024	YY/	44/Y	3/4	3/3	3	1	3	8	43	1	2/1	1	2/1	4/1	6	Y	Y	Y	Y	Y
PIC18F57Q43	128k	8192	1024	YY/	44/Y	3/4	3/3	3	1	3	8	43	1	2/1	1	2/1	4/1	6	Y	Y	Y	Y	Y

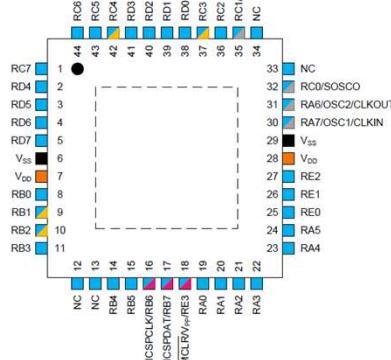
- Encapsulado de 48 pines
 - Amplio rango de voltaje de alimentación: 1.8V – 5.5V DC
 - Bajo consumo energético (menor a 800nA en modo SLEEP y alrededor de 1mA en operación con 4MHz, 5V)
 - Amplio rango de temperatura: -40°C – +85°C (grado industrial)
 - Hasta 64MHz de frecuencia en CPU
 - Memoria de programa de 128Kbytes y memoria de datos de 8Kbytes
 - Periféricos nuevos (detector de cruce por cero, CLC, etc)
 - Interrupciones vectorizadas
 - Módulo PPS para asignación de funciones en los pines



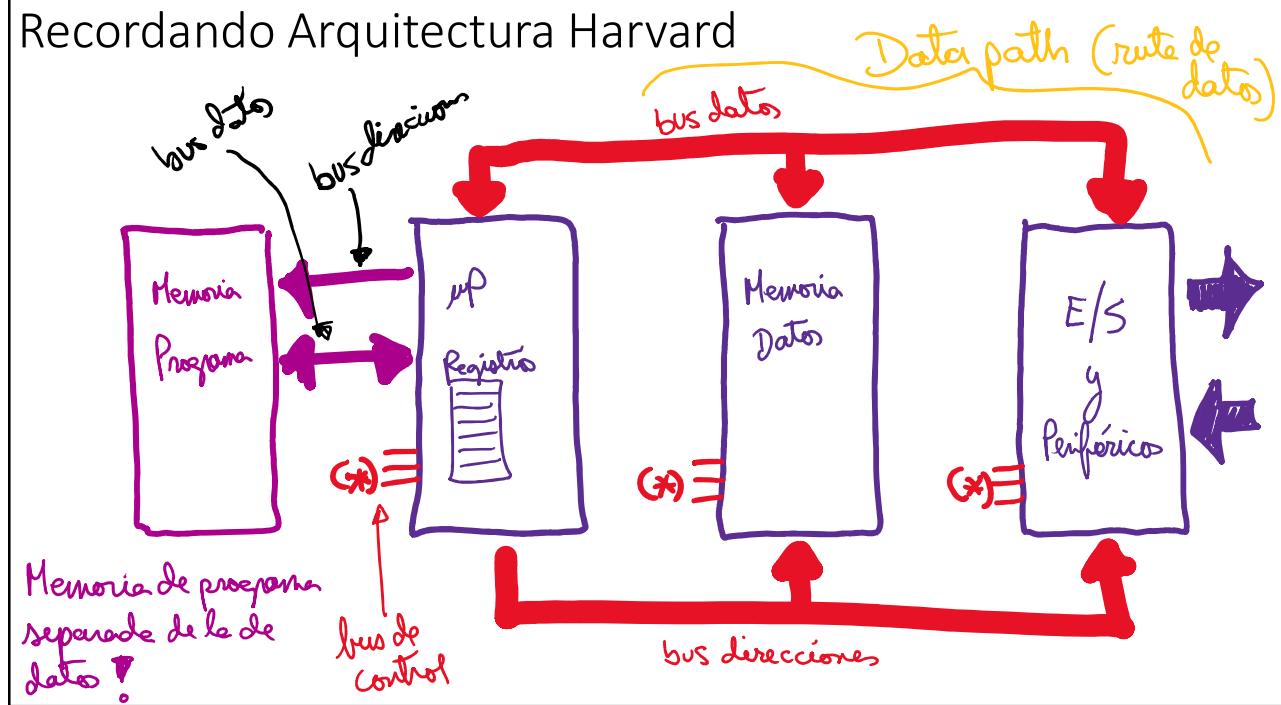
El Microcontrolador PIC18F47Q10

Device	Program Memory Flash (bytes)	Data SRAM (bytes)(2)	Data EEPROM (bytes)	I/O Pins	16-bit Timers	Comparators	10-bit ADC ² with Comparator(ch) Shift DAC	Zero-Cross Detect	CCP/10-bit PWM	CWG	CLC	Low Voltage Detect (LVD)	8-bit TMR with HLT	Windowed Watchdog Timer	CRC with Memory Scan	ELASRT	P2x/SPI	PPS	Peripheral Module Disable	Temperature Indicator	Debug(1)
PIC18F27Q10	128k	3615	1024	25	4	2	24	1	1	2/2	1	8	1	3	Y	Y	2	2	Y	Y	I
PIC18F47Q10	128k	3615	1024	36	4	2	35	1	1	2/2	1	8	1	3	Y	Y	2	2	Y	Y	I

- Encapsulado de 44 pines
 - Amplio rango de voltaje de alimentación: 1.8V – 5.5V DC
 - Bajo consumo energético (menor a 800nA en modo SLEEP y alrededor de 1mA en operación con 4MHz, 5V)
 - Amplio rango de temperatura: -40°C – +85°C (grado industrial)
 - Hasta 64MHz de frecuencia en CPU
 - Memoria de programa de 128Kbytes y memoria de datos de 3Kbytes
 - Periféricos nuevos (detector de cruce por cero, CLC, etc)
 - Interrupciones vectorizadas
 - Módulo PPS para asignación de funciones en los pines

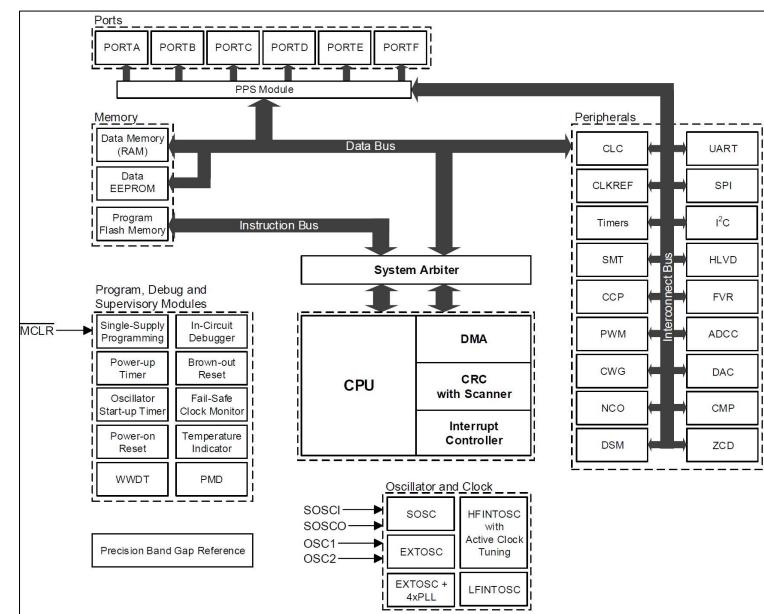


Recordando Arquitectura Harvard



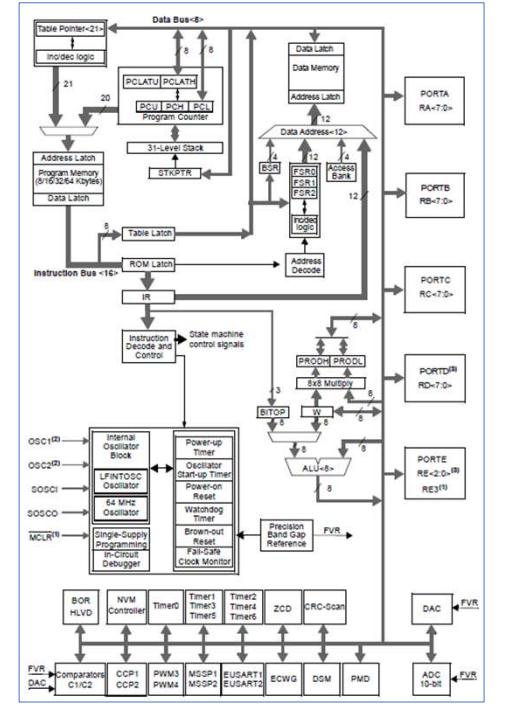
El Microcontrolador PIC18F57Q43

- Arquitectura Harvard:
 - Memoria de programa separada de la memoria de datos y con buses independientes
- Cambio en organización de la memoria de datos
- Módulo Timer0 mejorado
- ADC mejorado de 12bits
- Módulo generador de voltaje de referencia para el ADC



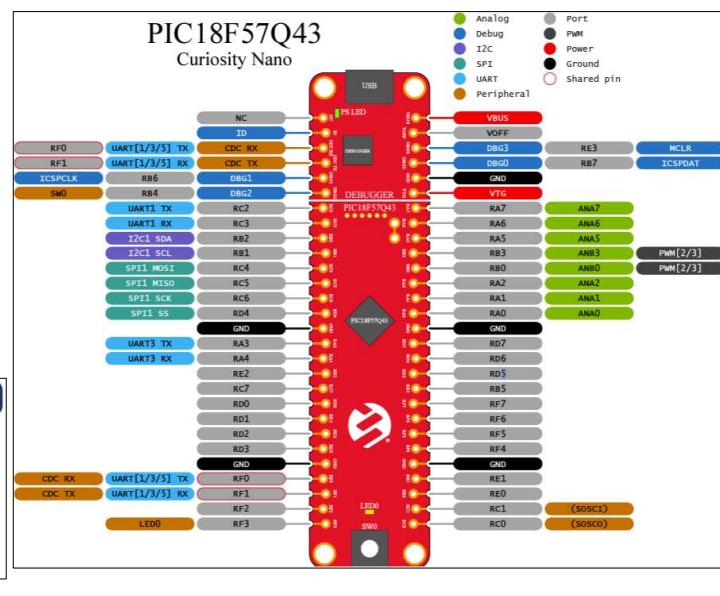
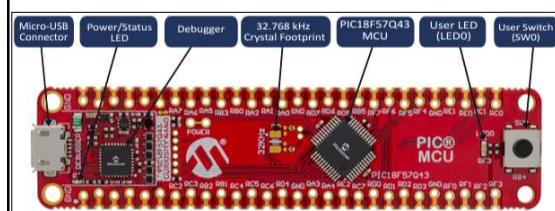
El Microcontrolador PIC18F57Q10

- Arquitectura Harvard:
 - Memoria de programa separada de la memoria de datos y con buses independientes
 - Memoria de programa 128Kb, memoria de datos 3615bytes
 - Módulo Timer0 mejorado
 - ADC de 10 bits
 - Oscilador interno de hasta 64MHz



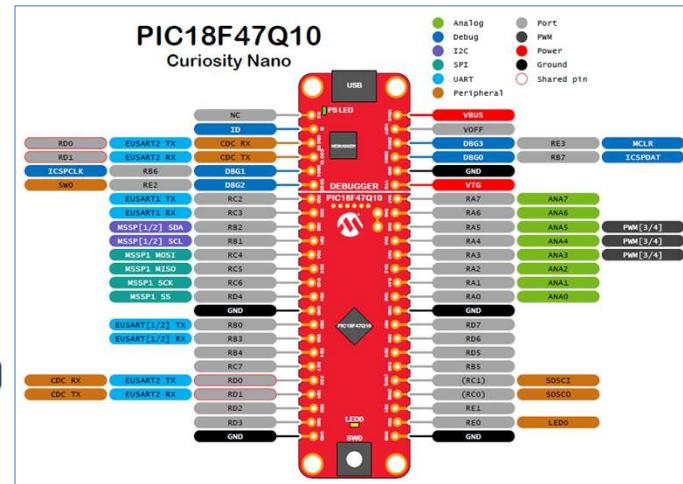
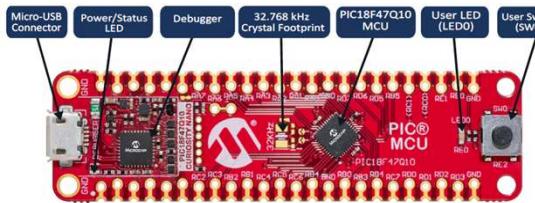
El módulo Curiosity Nano PIC18F57Q43

- Plataforma de desarrollo que integra programador/depurador y microcontrolador PIC18F57Q43.
 - Integra un LED (RF3) y un pulsador (RB4), ambos activos en bajo, el pulsador requiere resistencia pull-up.



El módulo Curiosity Nano PIC18F57Q10

- Plataforma de desarrollo que integra programador/depurador y microcontrolador PIC18F57Q10.
- Integra un LED (RE0) y un pulsador (RE2), ambos activos en bajo, el pulsador requiere resistencia pull-up.



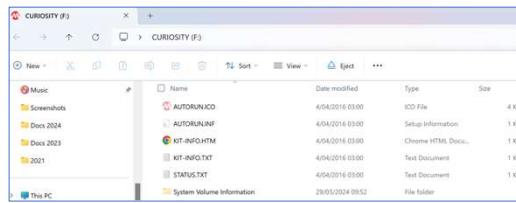
Si es que no se usa el Curiosity Nano

- Necesitamos un microcontrolador en formato DIP para ser usado en el protoboard (ejemplo PIC18F45Q43 ó PIC18F45Q10)
- Necesitamos un programador (ejemplo PICKIT5)



Funciones del debugger/programmer del Curiosity Nano PIC18F57Q43 / PIC18F47Q10

- Proveer energía al módulo (dos líneas)
 - VBUS (conexión directa de la energía del Puerto USB conectado)
 - VTG (conexión de salida de voltaje del sistema ajustable/controlado integrado en el módulo)
- Programar el microcontrolador onboard (el PIC18F57Q43 / PIC18F47Q10)
- Proveer un puerto serial en la PC para comunicación UART
- Al conectarlo a la PC, aparecerá una unidad de disco llamado "Curiosity"

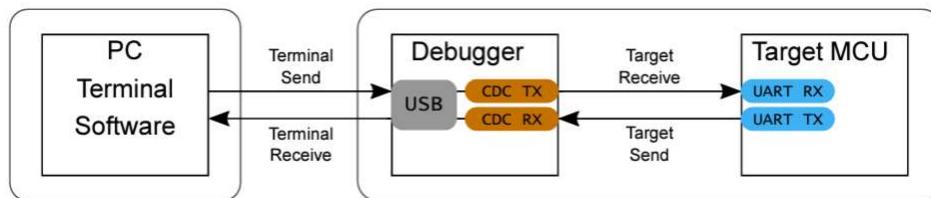


Aclaración sobre la unidad de disco Curiosity

- Al conectar el Curiosity Nano PIC18F57Q43 / PIC18F47Q10 a la PC aparecerá una unidad de disco llamado "Curiosity".
- Esta unidad de disco **NO DEBE DE USARSE COMO UNIDAD DE ALMACENAMIENTO!, NO ES UNA MEMORIA USB!**
- Esta unidad de disco sirve para hacer drop-off de un firmware

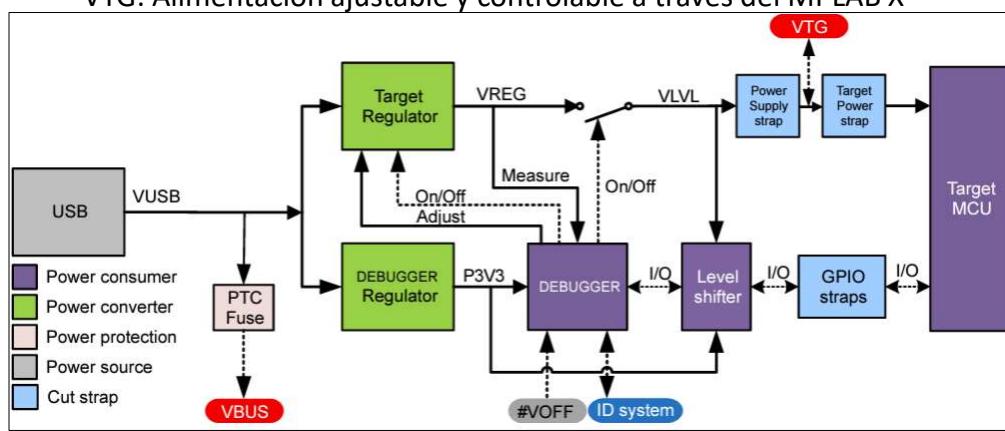
El módulo Curiosity Nano PIC18F57Q43 / PIC18F47Q10

- Al conectar el Curiosity Nano a la PC con el cable de datos USB-MicroUSB se habilitará un puerto serial para hacer labores de depuración/comunicación
- Dirigirse al Administrador de Dispositivos del Windows (dentro del Panel de Control) para que puedan identificar el puerto serial que se ha habilitado (COMx)
- Software terminal serial puede ser el PuTTY, HyperTerminal, Arduino IDE



El módulo Curiosity Nano PIC18F57Q43 / PIC18F47Q10

- Posee dos terminales de voltaje de alimentación disponibles para el usuario:
 - VBUS: Conexión directa de la alimentación del puerto USB (5V DC)
 - VTG: Alimentación ajustable y controlable a través del MPLAB X



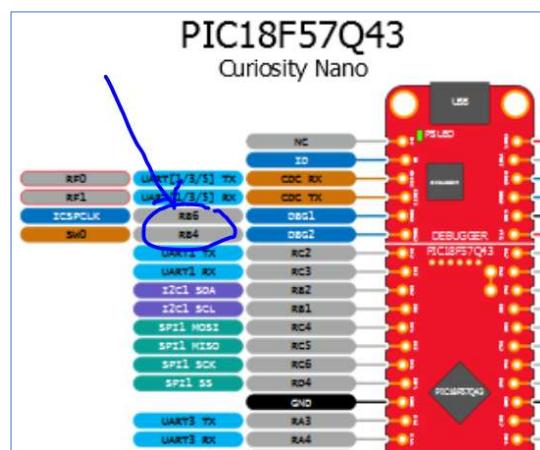
Recomendaciones al usar el Puerto VTG

•NO CONECTAR LOS PINES “VBUS” Y “VTG” A LA VEZ!!!

- El ajuste de voltaje de este puerto se realiza en el MPLABX
- Antes de grabar el dispositivo en el MPLABX, entrar a propiedades del proyecto y activar la energía (opción Power)

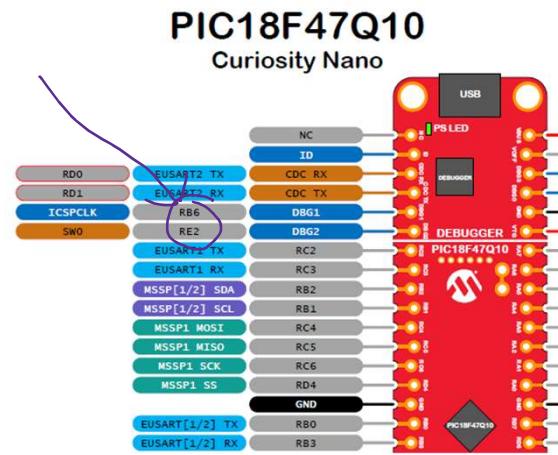
El módulo Curiosity Nano PIC18F57Q43

- No encuentro los pines RB4 y RB6 en el Curiosity Nano...
- Los puedes encontrar no rotulados y siguiendo el diagrama de pines en la documentación.



El módulo Curiosity Nano PIC18F57Q10

- No encuentro los pines RB6 y RE2 en el Curiosity Nano...
- Los puedes encontrar no rotulados y siguiendo el diagrama de pines en la documentación.

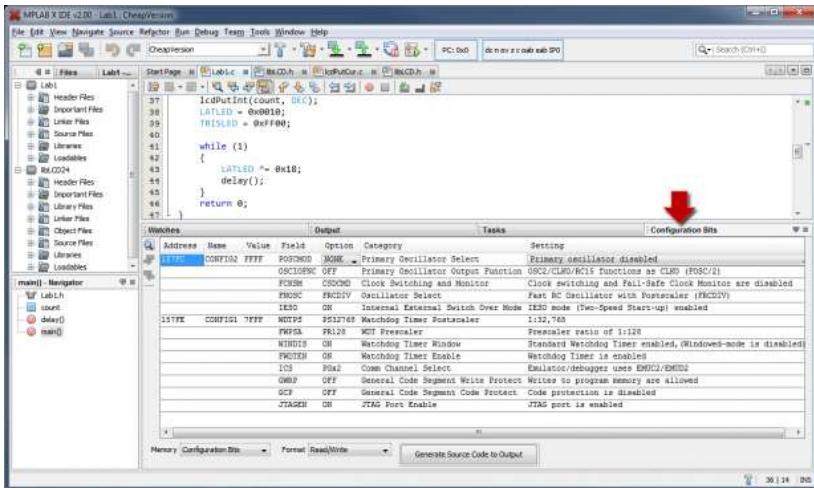


Comentario:



- Pero dichos algoritmos. ¿En dónde son ejecutados? En hardware!
- Se tiene que construir el hardware primero!

El MPLAB X IDE



- Descargable desde el siguiente link:

<https://www.microchip.com/mplab/mplab-x-ide>

<https://www.microchip.com/development-tools/pic-and-dspic-downloads-archive>

El MPLAB X IDE



- Entorno de desarrollo integrado (Integrated Design Environment - IDE) para los microcontroladores PIC y AVR de Microchip.
- Disponible para Windows, Linux y macOS
- Desarrollo basado en proyecto
- Emplea el compilador XC (XC8, XC16 y XC32), descargable de manera separada.
- Soporte a herramientas de desarrollo propias de Microchip (módulos Curiosity Nano, programadores PICKIT, depuradores ICD, etc)
- Última versión: v6.25 (lanzado el 4 de marzo del 2025)

MPASM vs. PICASM (XC8 Assembler)

- MPASM fué el lenguaje de programación hasta la v5.35 del MPLABX, actualmente en obsolescencia.
- XC8 PIC Assembler es el nuevo formato de lenguaje y soportado por la nueva versión ~~6.20~~ 6.25
- Las instrucciones en los microcontroladores no han variado, solo la sintaxis de programación (el core del microcontrolador es el mismo).
- Evolución: MPLAB -> MPLAB X (32bits) -> MPLAB X (64bits)
- Documentación: /Program Files/Microchip/xc8/v3.10/docs/MPLAB_XC8_PIC_Assembler_User_Guide.pdf
- Última versión del XC8: v3.10 (lanzado el 29 de agosto del 2025)

Importancia de tener las hojas técnicas de los IC's a usar:

- Las hojas técnicas (datasheet) son proporcionadas por el fabricante del IC's y se detallan todas las funcionalidades, capacidades, configuraciones, limitaciones, etc de dicho dispositivo, es la información mas fiel.
- En nuestro caso tendremos siempre presente la hoja técnica del microcontrolador PIC18F57Q43 rev. H (actualizado al 2024)

		PIC18F27/47/57Q43																												
		28/40/44/48-Pin, Low-Power, High-Performance Microcontroller with XLP Technology																												
Introduction																														
This PIC18-Q43 microcontroller family is available in 28/40/44/48-pin devices for real-time control applications. This family features a 12-bit Analog-to-Digital Converter (ADC), a 12-bit Current Comparison Counter (CCC) with dual independent capture, an advanced multi-touch sensor averaging, filtering, oversampling and threshold comparison. This family showcases a new 16-bit Pulse-Width Modulator (PWM) module which provides dual independent outputs on the same time base. Additional features include vectorized interrupt controller with fixed latency for handling interrupt system bus arbiter, Direct Memory Access (DMA) capability, Universal Asynchronous Receiver/Transmitter (UART), Support for IEEE 1451, Digital-to-Analog (D/A), Digital Addressable Lighting Interface (DALI) and Local Interconnect Network (LIN) protocols, Serial Peripheral Interface (SPI), I ² C, memory features like Memory Access Partition (MAP) to support users in data protection and bootloader applications, and Device Information Area (DIA), which stores factory calibration values to help improve temperature sensor accuracy.																														
PIC18-Q43 Family Types																														
Table 1. Devices Included in This Data Sheet																														
Device		Programmable Flash	Programmable RAM	Data EEPROM	EEPROM	EE240	EE400	EE480	EE512	EE1K	EE2K	EE4K	EE8K	EE16K	EE32K	EE64K														
PIC18F27Q43		128K	8192	1024	YY	25Y	3/4	3/5	3	1	3	8	32	1	2/1	2/1														
PIC18F47Q43		128K	8192	1024	YY	36Y	3/4	3/5	3	1	3	8	32	1	2/1	2/1														
PIC18F57Q43		128K	8192	1024	YY	44Y	3/4	3/5	3	1	3	8	43	1	2/1	2/1														

Revisión de documentos

- Hoja técnica del PIC18F57Q43
 - <https://ww1.microchip.com/downloads/aemDocuments/documents/MCU08/ProductDocuments/DataSheets/PIC18F27-47-57Q43-Data-Sheet-40002147F.pdf> (rev. F)
 - <https://ww1.microchip.com/downloads/aemDocuments/documents/MCU08/ProductDocuments/DataSheets/PIC18F27-47-57Q43-Microcontroller-Data-Sheet-XLP-DS40002147.pdf> (rev. H 2024)
- Hoja técnica del Curiosity Nano PIC18F57Q43
 - <http://ww1.microchip.com/downloads/en/DeviceDoc/PIC18F57Q43-Curiosity-Nano-HW-UserGuide-DS40002186B.pdf>
- MPLAB XC8 PIC Assembler User Guide
 - //Program Files/Microchip/xc8/v3.10/docs/MPLAB_XC8_PIC_Assembler_User_Guide.pdf

Revisión de documentos

- Hoja técnica del PIC18F57Q10
 - <https://ww1.microchip.com/downloads/aemDocuments/documents/MCU08/ProductDocuments/DataSheets/PIC18F27-47Q10-Micorcontroller-Data-Sheet-DS40002043.pdf>
- Hoja técnica del Curiosity Nano PIC18F57Q10
 - <https://ww1.microchip.com/downloads/aemDocuments/documents/MCU08/ProductDocuments/UserGuides/PIC18F47Q10-Curiosity-Nano-Hardware-User-Guide-40002103.pdf>
- MPLAB XC8 PIC Assembler User Guide
 - //Program Files/Microchip/xc8/v3.10/docs/MPLAB_XC8_PIC_Assembler_User_Guide.pdf

Importancia de los comentarios en un código fuente

- Cuando uno desarrolla un programa, en cualquier lenguaje de programación, es fundamental colocar comentarios.
- Los comentarios no añaden espacio de memoria luego de la compilación.
- Los comentarios sirven para recordar ideas, configuraciones, procesos, algoritmos, etc que le permitan al programador en un tiempo después ver lo que hizo en dicho momento.
- En MPASM ó XC8 PICASM los comentarios van antecedidos por un punto y coma (;)

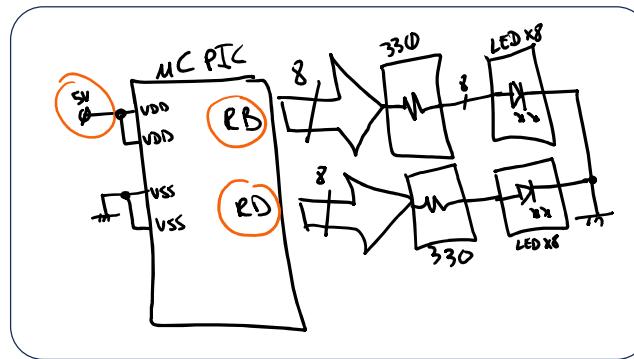
Consideraciones importantes al usar simuladores

- El uso de simuladores ha permitido acelerar los procesos de validación de circuitos eléctricos y electrónicos, **pero** no es un determinante a la hora de validar en forma física.
- En la mayoría de casos en ingeniería electrónica el producto final es algo físico por lo que no solamente podemos fiarnos de una simulación y dar por sentado que la propuesta funcione correctamente.
- En Proteus hay elementos que no se muestran en el momento de hacer simulaciones.
- El software Proteus a la fecha no posee módulo de simulación del microcontrolador PIC18F57Q43 pero si para el PIC18F47Q10

Sobre el MPLAB X IDE

1. Crear un proyecto (seleccionar Standalone Project)
2. Seleccionar el dispositivo microcontrolador (PIC18F57Q43 / PIC18F47Q10)
3. Seleccionar la herramienta “pic-as” (XC8 PIC Assembler)
4. Crear el archivo header (*.inc) e incluirle los bits de configuración (Window / Target Memory Views / Configuration Bits), dicho archivo debe de estar en la carpeta “Header Files”
5. Crear el archivo fuente (*.s) e incluir el archivo header dentro del cuerpo, dicho archivo debe de estar en la carpeta “Source Files”
6. Para compilar: 
7. El archivo generado de la compilación tiene extensión *.hex ó *.elf
8. Para grabar en el Curiosity Nano: 

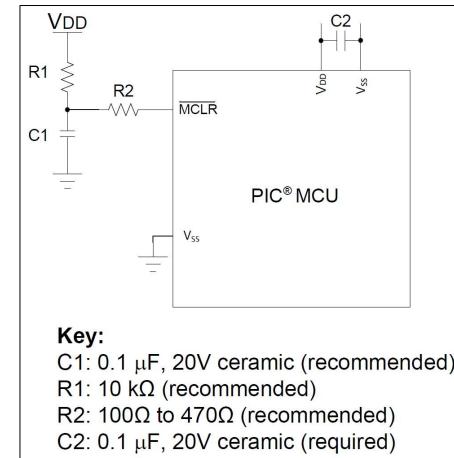
Hay dos pines de Vdd y dos pines de Vss, es necesario conectar todos?



Se deben de conectar todos los pines de alimentación para que el microcontrolador pueda obtener mayor capacidad de corriente en caso lo requiera la aplicación.

Detalles técnicos iniciales

- Voltaje de alimentación del PIC18F57Q43 / PIC18F47Q10
 - Voltajes menores a -0.5V son perjudiciales
 - Voltaje de operación máximo 5.5V
 - Voltaje de operación mínimo 1.8V



Fin de la sesión

Agenda 23/01/2026 15:00

- Detalle de configuración del módulo de oscilador
- Detalle de configuración de puertos de E/S
- Segundo ejemplo práctico: Titilar un LED

Plantilla para programas en XC8 PIC Assembler

- Tener en cuenta el modelo de microcontrolador a usar, éste debe de coincidir desde la creación del proyecto hasta el detalle de la directiva PROCESSOR

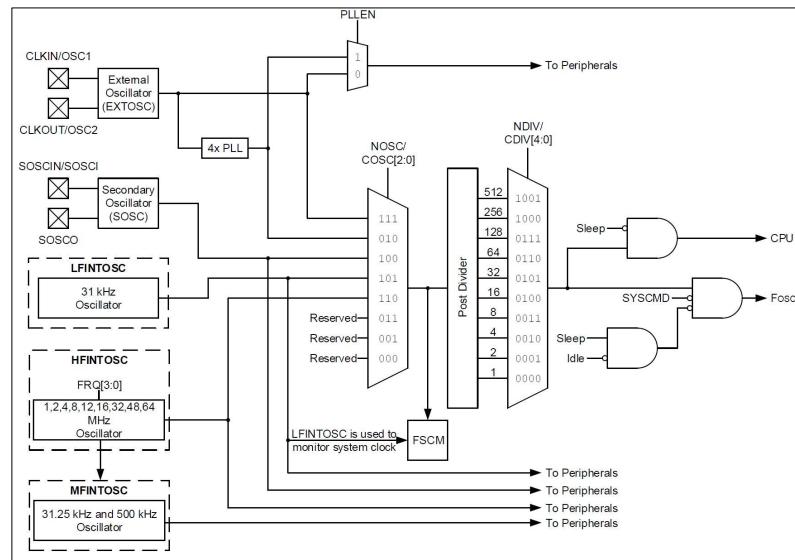
```

1      PROCESSOR 18F57Q43
2      #include "cabecera.inc"
3
4      PSECT principal, class=CODE, reloc=2, abs
5      principal:
6          ORG 000000H          ;Vector de RESET
7          bra configuro      ;Salto hacia etiqueta configuro
8
9          ORG 000020H          ;Zona de programa de usuario
10     configuro:
11         ;Configuraciones de la aplicación (puertos, periféricos, etc)
12
13     inicio:
14         ;Detalle de lo que va a hacer el microcontrolador
15         bra inicio          ;Salto hacia etiqueta inicio
16
17     end principal
18

```

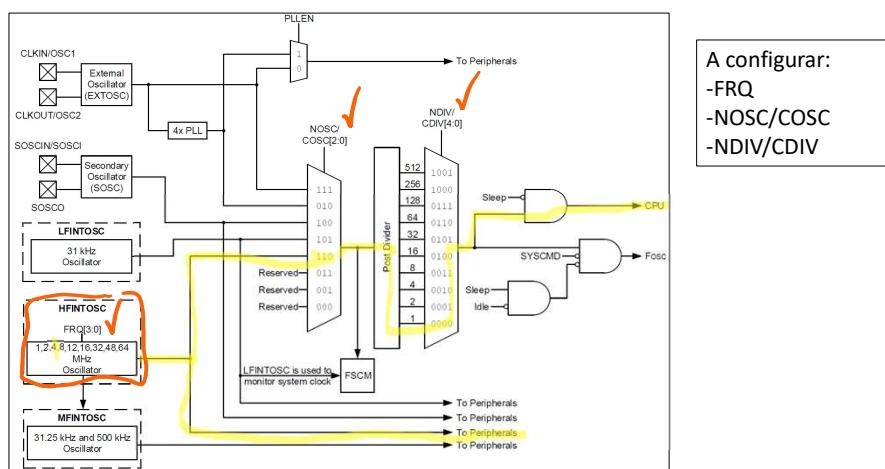
Detalles técnicos del módulo de oscilador (fuente de reloj)

- El modulo de oscilador del PIC18F47Q10 es similar al del PIC18F57Q43
- Oscilador externo (oscilador primario) sirve para que se obtenga la Fuente de reloj de manera externa, ya sea por un cristal o una Fuente de onda cuadrada
- Oscilador secundario sirve para colocar un cristal de 32.768KHz y hacer aplicaciones de reloj en tiempo real con el Timer1.
- El PLL de este microcontrolador sirve para incrementar la frecuencia de trabajo, hasta 64MHz.
- Usaremos el oscilador interno HFINTOSC para el CPU en todas nuestras aplicaciones.
- MFINTOSC no puede derivarse hacia el CPU, solo hacia periféricos.



Configuración de fuente de reloj con HFINTOSC

Para obtener 4MHz al CPU y periféricos a partir del oscilador interno:



Configuración de fuente de reloj con HFINTOSC

Registros implicados en la configuración del reloj
(PIC18F57Q43):

A configurar:
 -FRQ
 -NOSC/COSC
 -NDIV/CDIV

12.6 Register Summary - Oscillator Module

Address	Name	Bit Pos.	7	6	5	4	3	2	1	0
0xAC	ACTCON	7:0	ACTEN	ACTUD			ACTLOCK		ACTORS	
0xAD	OSCCON1	7:0			NOSC[2:0]				NDIV[3:0]	
0xAE	OSCCON2	7:0			COSC[2:0]				CDIV[3:0]	
0xAF	OSCCON3	7:0	CSWHOLD	SOSCPWR		ORDY	NOSCR			
0xB0	OSCTUNE	7:0					TUN[5:0]			
0xB1	OSCFRQ	7:0							FRQ[3:0]	
0xB2	OSCSTAT	7:0	EXTOR	HFOR	MFOR	LFOR	SOR	ADOR		PLLR
0xB3	OSCEN	7:0	EXTOEN	HFOEN	MFOEN	LFOEN	SOSCEN	ADOEN		PLLEN

Configuración de fuente de reloj con HFINTOSC

Registros implicados en la configuración del reloj
(PIC18F57Q10):

A configurar:
 -FRQ
 -NOSC/COSC
 -NDIV/CDIV

Address	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x00										
...	Reserved									
0xED2										
0xED3	OSCCON1	7:0			NOSC[2:0]				NDIV[3:0]	
0xED4	OSCCON2	7:0			COSC[2:0]				CDIV[3:0]	
0xED5	OSCCON3	7:0	CSWHOLD	SOSCPWR		ORDY	NOSCR			
0xED6	OSCSTAT	7:0	EXTOR	HFOR	MFOR	LFOR	SOR	ADOR		PLLR
0xED7	OSCEN	7:0	EXTOEN	HFOEN	MFOEN	LFOEN	SOSCEN	ADOEN		
0xED8	OSCTUNE	7:0					HFTUN[5:0]			
0xED9	OSCFRQ	7:0						HFFRQ[3:0]		

Configuración de fuente de reloj con HFINTOSC

Registro OSCCON1:

12.5.2 OSCCON1							
Name: OSCCON1 Address: 0x0AD Oscillator Control Register 1							
Bit	NOSC[2:0]		NDIV[3:0]				
Access	R/W	f	R/W	f	R/W	q	R/W
Reset							
6	0	0	0	1	0	0	0
5	0	0	0	1	0	0	0
4	0	0	0	1	0	0	0
3	0	0	0	1	0	0	0
2	0	0	0	1	0	0	0
1	0	0	0	1	0	0	0
0	0	0	0	1	0	0	0

Bits 6:4 – NOSC[2:0] New Oscillator Source Request^(1,2,3)
Requests a new oscillator source per the [NOSC/COSC Clock Source Selection Table](#).

Bits 3:0 – NDIV[3:0] New Divider Selection Request
Requests the new postscaler division ratio per the [NDIV/CDIV Clock Divider Selection Table](#).

Notes:

- The default value is determined by the RSTOSC Configuration bits. See the Reset Oscillator (RSTOSC) selection table for the RSTOSC selections.
- If NOSC is written with a reserved value, the operation is ignored and neither NOSC nor NDIV is written.
- When CSWEN = 0, these bits are read-only and cannot be changed from the RSTOSC.

Table 12-2. NOSC/COSC Clock Source Selection Table

NOSC / COSC	Clock Source
111	EXTOSC ⁽¹⁾
110	HFINTOSC ⁽²⁾
101	LFINTOSC
100	SOSC
011	Reserved
010	EXTOSC + 4xPLL ⁽³⁾
001	Reserved
000	Reserved

Table 12-3. NDIV/CDIV Clock Divider Selection Table

NDIV / CDIV	Clock Divider
1111-1010	Reserved
1001	512
1000	256
0111	128
0110	64
0101	32
0100	16
0011	8
0010	4
0001	2
0000	1

Configuración de fuente de reloj con HFINTOSC

Registro OSCFRQ:

12.5.6 OSCFRQ							
Name: OSCFRQ Address: 0x0E7							
Bit	FRQ[3:0]						
Access	R/W	0	R/W	0	R/W	0	R/W
Reset							
6	0	0	0	1	0	0	0
5	0	0	0	1	0	0	0
4	0	0	0	1	0	0	0
3	0	0	0	1	0	0	0
2	0	0	0	1	0	0	0
1	0	0	0	1	0	0	0
0	0	0	0	1	0	0	0

Bits 3:0 – FRQ[3:0] HFINTOSC Frequency Selection

FRQ	Nominal Freq (MHz)
1111-1001	Reserved
1000	64
0111	48
0110	32
0101	16
0100	12
0011	8
0010	4
0001	2
0000	1

Configuración de fuente de reloj con HFINTOSC

Registro OSCEN:

12.5.8 OSCEN

Name: OSCEN
Address: 0x0000

Oscillator Enable Register

Bit	EXTOPEN	HFOEN	MFOEN	LFOEN	SOSCEN	ADOPEN	PLEN
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0
Bit 7 – EXTOPEN External Oscillator Enable							
Value	Description						
1	EXTOSC is explicitly enabled, operating as specified by FEXTOSC						
0	EXTOSC can be enabled by a peripheral request						
Bit 6 – HFOEN HFINTOSC Enable							
Value	Description						
1	HFINTOSC is explicitly enabled, operating as specified by OSCFRQ						
0	HFINTOSC can be enabled by a peripheral request						
Bit 5 – MFOEN MFINTOSC Enable							
Value	Description						
1	MFINTOSC is explicitly enabled						
0	MFINTOSC can be enabled by a peripheral request						
Bit 4 – LFOEN LFINTOSC Enable							
Value	Description						
1	LFINTOSC is explicitly enabled						
0	LFINTOSC can be enabled by a peripheral request						
Bit 3 – SOSCEN Secondary Oscillator Enable							
Value	Description						
1	SOSC is explicitly enabled, operating as specified by SOSCPWR						
0	SOSC can be enabled by a peripheral request						
Bit 2 – ADOPEN ADCRC Oscillator Enable							
Value	Description						
1	ADCRC is explicitly enabled						
0	ADCRC may be enabled by a peripheral request						
Bit 0 – PLEN PLL Enable⁽¹⁾							
Value	Description						
1	EXTOSC multiplied by the 4x system PLL is used by a peripheral request						
0	EXTOSC is used by a peripheral request						

Note:
1. This bit only controls external clock source supplied to the peripherals and has no effect on the system clock.

Configuración de fuente de reloj con HFINTOSC

- Configuraremos el PIC18F57Q43 / PIC18F47Q10 para que funcione a 4MHz con el HFINTOSC

```

1  PROCESSOR 18F57Q43
2  #include "cabecera.inc"
3
4  PSECT principal, class=CODE, reloc=2, abs
5  principal:
6  ORG 000000H ;Vector de RESET
7  bra configuro ;Salto hacia etiqueta configuro
8
9  ORG 000020H ;Zona de programa de usuario
10 configuro:
11 ;Configuraciones de la aplicación (puertos, periféricos, etc)
12 ;Configuración de la fuente de reloj para 4MHz con el HFINTOSC
13 movlb 0H ;Bank 0 donde estan los registros de conf del osc
14 ;OSCFRQ = 02H
15 movlw 02H ;muevo literal 02H a wreg
16 movwf OSCFRQ, 1 ;muevo contenido de wreg hacia OSCFRQ
17 ;OSCCON1 = 60H
18 movlw 60H ;muevo literal 60H a wreg
19 movwf OSCCON1, 1 ;muevo contenido de wreg hacia OSCCON1
20 ;OSCEN = 40H
21 movlw 40H ;muevo literal 40H a wreg (0100 0000)
22 movwf OSCEN, 1 ;muevo contenido de wreg hacia OSCEN
23
24 inicio:
25 ;Detalle de lo que va a hacer el microcontrolador
26 bra inicio ;Salto hacia etiqueta inicio
27
28 end principal

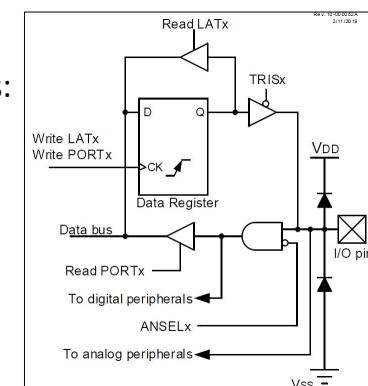
```

Configuración de fuente de reloj con HFINTOSC

- ¿Por qué siempre se trabaja con 4MHz y no mas rápido? ¿Qué pasaría si trabajamos siempre a 32MHz?
 - Porque trabajar a mayor frecuencia el microcontrolador consumirá mas energía.
 - El microcontrolador trabajará mas rápido
 - Se trabaja a 4MHz debido a que cada instrucción en ensamblador se ejecuta a 1us.

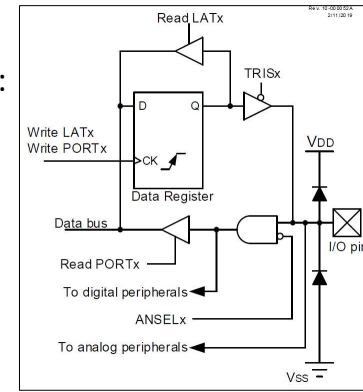
Manejo de puertos de E/S en el PIC18F57Q43

- Revisar 19.0 en la hoja técnica
- **Por defecto los puertos están como entradas analógicas.**
- Se tienen los siguientes registros para manipular los puertos:
 - TRISx Para configurar el sentido del puerto (entrada ó salida), cero para salida y uno para entrada.
 - PORTx Para leer el puerto
 - LATx Para escribir el puerto
 - ANSELx Para configurar el puerto en analógico o digital, uno para analógico y cero para digital.
 - WPUx Para habilitar las resistencias de pullup: 1 activado, 0 desactivado
 - SLRCON Para configurar la velocidad de respuesta en el puerto configurado como salida.

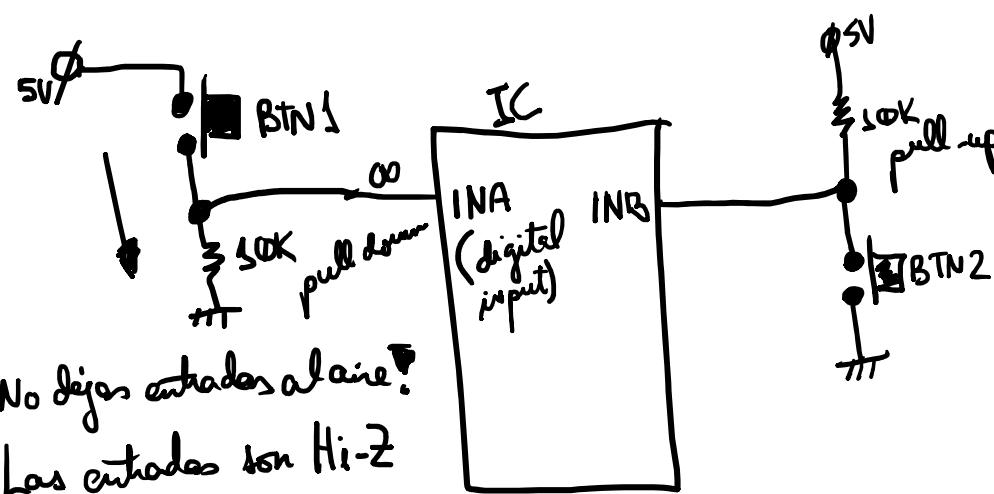


Manejo de puertos de E/S en el PIC18F57Q10

- Revisar 16.0 en la hoja técnica
- **Por defecto los puertos están como entradas analógicas.**
- Se tienen los siguientes registros para manipular los puertos:
 - TRISx Para configurar el sentido del puerto (entrada ó salida), cero para salida y uno para entrada.
 - PORTx Para leer el puerto
 - LATx Para escribir el puerto
 - ANSELx Para configurar el puerto en analógico o digital, uno para analógico y cero para digital.
 - WPUx Para habilitar las resistencias de pullup: 1 activado, 0 desactivado
 - INLVlx Para determinar el umbral de voltaje en modo entrada, si es umbral TTL o ST CMOS
 - SLRCONx Para configurar la velocidad de respuesta en el puerto configurado como salida.

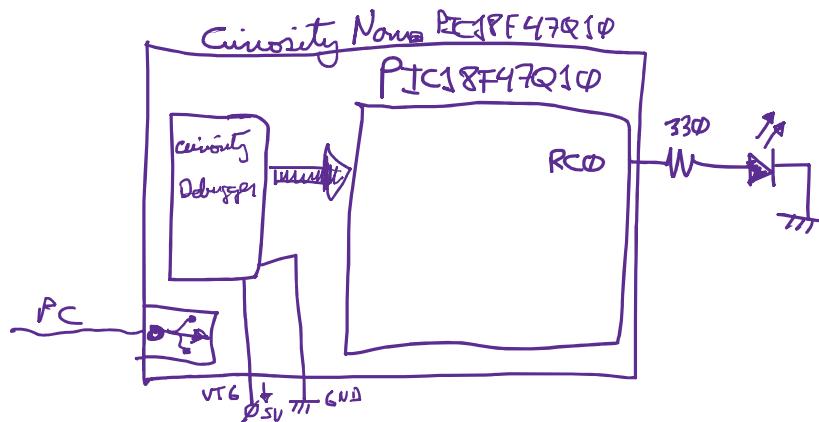


Diferencia entre pulsadores activos en alto y activos en bajo



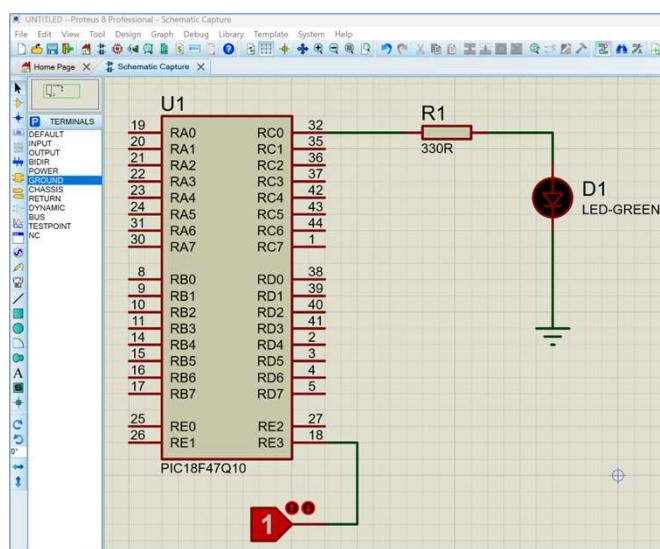
Ejemplo: Titilar un LED con periodo de 500ms

- Empleando el Curiosity Nano PIC18F47Q10, usaremos el RC0 para conectar un LED



Ejemplo: Titilar un LED con periodo de 500ms

- Circuito en Proteus:



Ejemplo: Titilar un LED con periodo de 500ms

- Análisis y diagrama de flujo:

Periodo de 500ms :

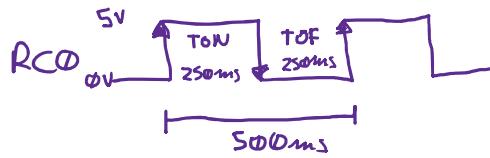
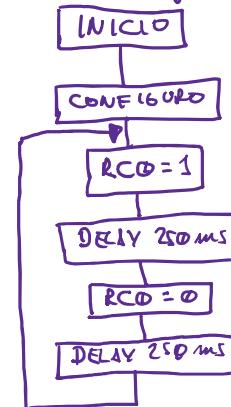
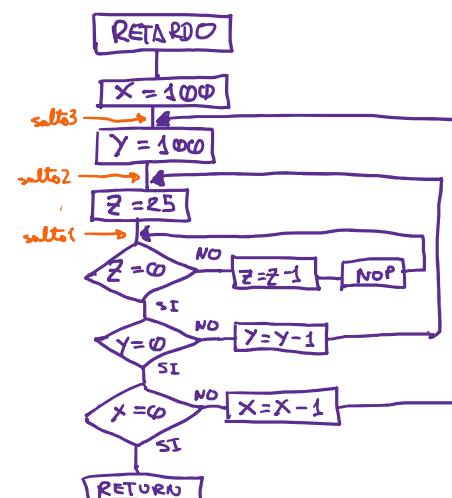
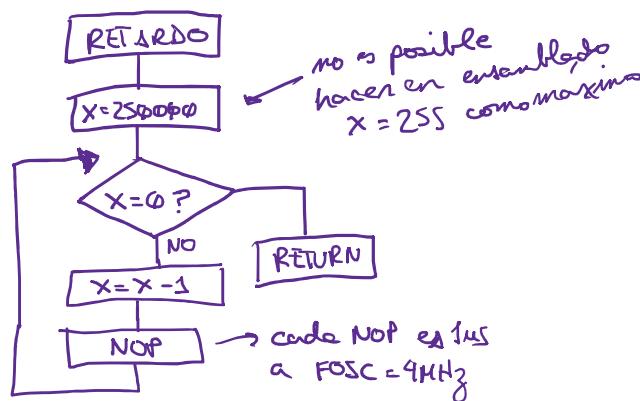


Diagrama de flujo:



Ejemplo: Titilar un LED con periodo de 500ms

- Obtención de un retardo de 250ms



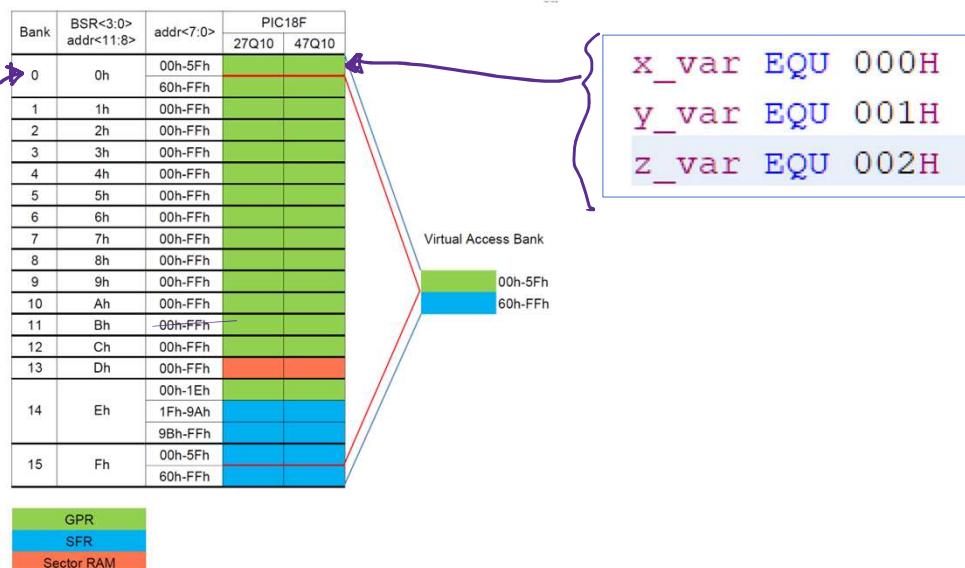
Ejemplo: Titilar un LED con periodo de 500ms

- Configuration bits:

	4	FEXTOSC OFF	External Oscillator mode Selection Bits	Oscillator not enabled
	6	RSOSC HFINTOSC_...	Power-up default value for COSC bits	HFINTOSC with HFREQ = 4 MHz and CDIV = 4:1
300001	CONFIG1H FF	- -	-	-
	1	CLKOUTEN OFF	Clock Out Enable bit	CLKOUT function is disabled
	1	CSHEN ON	Clock Switch Enable bit	Writing to NOSC and NDIV is allowed
	1	FCMEN ON	Fail-Safe Clock Monitor Enable bit	Fail-Safe Clock Monitor enabled
300002	CONFIG2L 3D	- -	-	-
	1	MCLR EXTMCLR	Master Clear Enable bit	MCLR pin (RE3) is MCLR
	0	PWRTE ON	Power-up Timer Enable bit	Power up timer enabled
	1	LPOREN OFF	Low-power BOR enable bit	Low power BOR is disabled
	0	BOREN OFF	Brown-out Reset Enable bits	Brown-out Reset disabled
300003	CONFIG2H FF	-	-	-
	3	BORV VBOR_190	Brown Out Reset Voltage selection bits	Brown-out Reset Voltage (VBOR) set to 1.90V
	1	ZCD OFF	ZCD Disable bit	ZCD disabled. ZCD can be enabled by setting the ZCDSEN bit of ZCDCON
	1	PPSWAY ON	PPSLOCK bit One-Way Set Enable bit	PPSLOCK bit can be cleared and set only once; PPS registers remain locked after one clear.
	1	STVREN ON	Stack Full/Underflow Reset Enable bit	Stack full/underflow will cause Reset
	1	XINST OFF	Extended Instruction Set Enable bit	Extended Instruction Set and Indexed Addressing Mode disabled
300004	CONFIG3L 9F	- -	-	-
	1F	WDTCPSP WDTCPSP_31	WDT Period Select bits	Divider ratio 1:65536; software control of WDTPS
	0	WDIE OFF	WDT operating mode	WDT Disabled
300005	CONFIG3H FF	- -	-	-
	7	WDTCWS WDTCWS_7	WDT Window Select bits	window always open (100%); software control; keyed access not required
	7	WDTCSC SC	WDT input clock selector	Software Control
300006	CONFIG4L FF	-	-	-
	1	WR10 OFF	Write Protection Block 0	Block 0 (000800-003FFFFh) not write-protected
	1	WR11 OFF	Write Protection Block 1	Block 1 (004000-007FFFFh) not write-protected
	1	WR12 OFF	Write Protection Block 2	Block 2 (008000-00BFFFFh) not write-protected
	1	WR13 OFF	Write Protection Block 3	Block 3 (00C000-00FFFFh) not write-protected
	1	WR14 OFF	Write Protection Block 4	Block 4 (010000-013FFFFh) not write-protected
	1	WR15 OFF	Write Protection Block 5	Block 5 (014000-017FFFFh) not write-protected
	1	WR16 OFF	Write Protection Block 6	Block 6 (018000-01BFFFFh) not write-protected
	1	WR17 OFF	Write Protection Block 7	Block 7 (01C000-01FFFFh) not write-protected
300007	CONFIG4H DF	- -	-	-
	1	WRIC OFF	Configuration Register Write Protection bit	Configuration registers (300000-300008h) not write-protected
	1	WRIB OFF	Boot Block Write Protection bit	Boot Block (000000-0007Fh) not write-protected
	1	WRID OFF	Data EEPROM Write Protection bit	Data EEPROM not write-protected
	1	SCANE ON	Scanner Enable bit	Scanner module is available for use, SCANNRD bit can control the module
	0	LVP OFF	Low Voltage Programming Enable bit	HV on MCLR/VPP must be used for programming

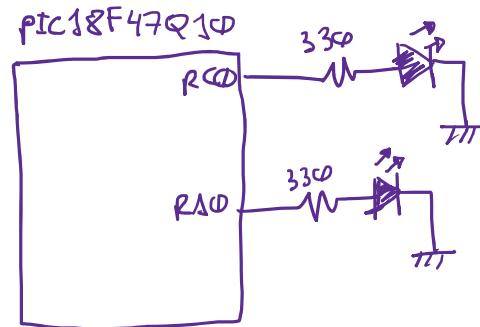
Ejemplo: Titilar un LED con periodo de 500ms

- Usando posiciones en la memoria RAM (GPR)



Ejercicio:

- Modificar el ejemplo anterior para que tenga dos LEDs y parpadeen en completo (si uno esta encendido el otro esta apagado)
- Revisión y discusión a las 6PM



Ejercicio: Modificar el ejemplo anterior para que tenga dos LEDs y parpadeen en completo (si uno esta encendido el otro esta apagado)

- Solo se tiene que agregar la configuración del puerto del segundo LED, como salida digital.
- En el código agregar el estado de funcionamiento del segundo LED: mientras el primero esta encendido se apaga el segundo y viceversa

```

23      ;configuramos las E/S
24      movlb OFH    ;Al Bank15, para que Q43 es Bank4 (4H)
25      bcf TRISC, 0, 1    ;RC0 como salida
26      bcf ANSELC, 0, 1    ;RC0 como digital
27      bcf TRISA, 0, 1    ;RA0 como salida
28      bcf ANSELA, 0, 1    ;RA0 como digital
29
30      inicio:
31      bsf LATC, 0, 1    ;RC0 = 1 prendemos el primer LED
32      bcf LATA, 0, 1    ;RA0 = 0 apagamos el segundo LED
33      call retardo    ;subrutina de retardo
34      bcf LATC, 0, 1    ;RC0 = 0 apagamos el primer LED
35      bsf LATA, 0, 1    ;RA0 = 1 encendemos el segundo LED
36      call retardo    ;subrutina de retardo
37      bra inicio       ;salta a inicio

```

Ejercicios propuestos:

- Modificar el circuito y diagrama de flujo para que con un pulsador se cambie el periodo de titileo entre 500ms y 250ms.
- Basado en el punto anterior y usando el mismo pulsador intercambiar el periodo de titileo entre 500ms, 250ms, 100ms y 50ms.
- Efecto del auto fantástico (corrimiento de iluminación de ocho LEDs)

Fin de la sesión!