



INSTITUTO DE
INGENIERÍA
ELÉCTRICA



FACULTAD DE
INGENIERÍA
UDELAR



UNIVERSIDAD
DE LA REPÚBLICA
URUGUAY

Sistemas embebidos para tiempo real

Este material didáctico fue elaborado por docentes del Departamento de Electrónica de la Universidad de la República a lo largo a varios años. Se pone a disposición de la comunidad bajo la licencia “Creative Commons Attribution 4.0 International License”.

Ver detalles de la licencia aquí: <https://creativecommons.org/licenses/by/4.0/>



Co-funded by the
Erasmus+ Programme
of the European Union



INSTITUTO DE
INGENIERÍA
ELÉCTRICA



FACULTAD DE
INGENIERÍA
UDELAR



UNIVERSIDAD
DE LA REPÚBLICA
URUGUAY

Sistemas embebidos para tiempo real

Presentación del curso

Julián Oreggioni
juliano@fing.edu.uy

Índice

- Contenido
- Objetivos
- Metodología
- Reglas del curso
- Aprobación
- Programa
- Material

Contenido

- ¿De qué se trata?
- Sistemas embebidos
 - Hardware
 - Software
- Programación de sistemas embebidos
 - Interfaz hardware-software
 - Arquitectura de software

Ejemplos de sistemas embebidos



- Miele dishwashers
- Microprocesador:
8-bit Motorola 68HC05

- Otros en la cocina:
 - Lavarropas
 - Horno-microondas
 - Heladera?
 - Cafetera?
 - Licuadora?

Fuente: Daniel W. Lewis, “Fundamentals of Embedded Software”

Ejemplos de sistemas embebidos

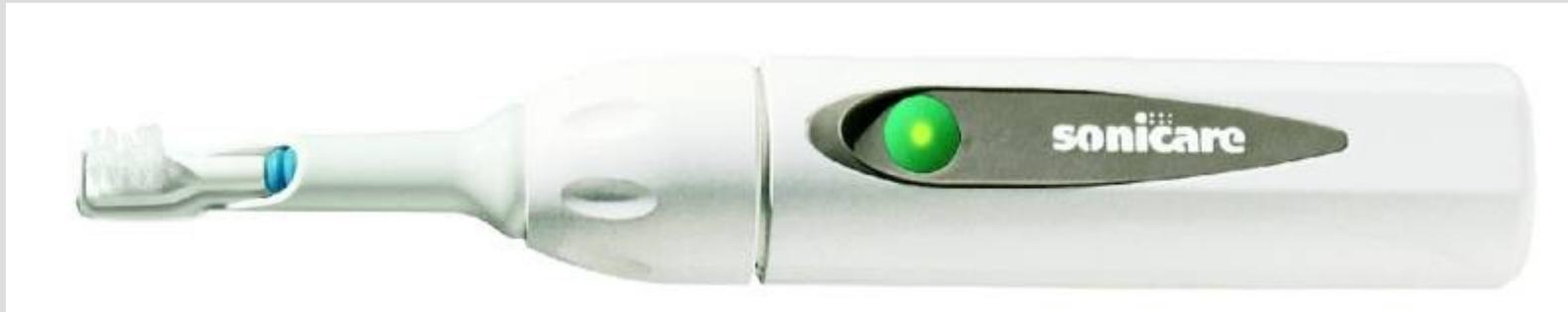


- Hunter Programmable Digital Thermostat.
- Microprocesador: 4-bit.
- Otros en casa:
 - Aire acondicionado
 - Alarma
 - TV?
 - Equipo de música?
 - Ventilador?
 - Computadora? Celular?

Fuente: Daniel W. Lewis, “Fundamentals of Embedded Software”

Ejemplos de sistemas embebidos

- Producto:
Sonicare Plus toothbrush.
- Microprocesador:
8-bit Zilog Z8



Fuente: Daniel W. Lewis, "Fundamentals of Embedded Software"

Ejemplos de sistemas embebidos

- Tecnologías de comunicación inalámbrica (Bluetooth low energy, wifi, etc.)



www.fitbit.com



www.sticknfind.com



www.meterplug.com



Ejemplos de sistemas embebidos



- Producto:
Vendo V-MAX 720 vending machine.
- Microprocesador:
8-bit Motorola 68HC11.

Fuente: Daniel W. Lewis, “Fundamentals of Embedded Software”

Ejemplos de sistemas embebidos



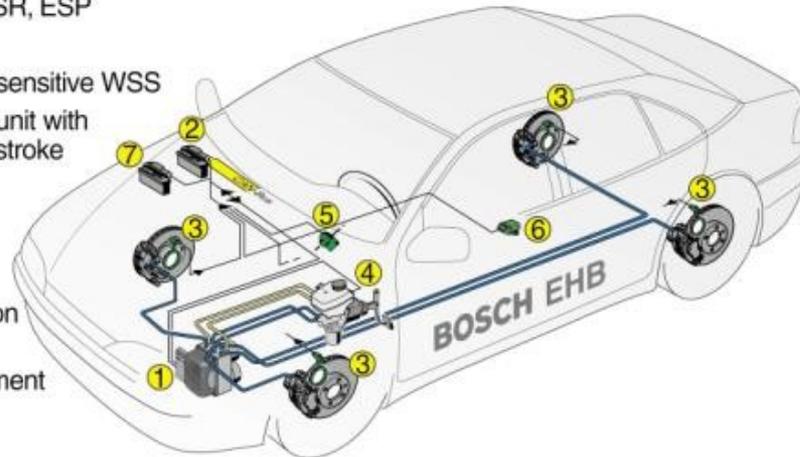
- Producto:
CoinCo USQ-712 coin changer.
- Microprocesador:
8-bit Motorola 68HC912.

Fuente: Daniel W. Lewis, "Fundamentals of Embedded Software"

Ejemplos de sistemas embebidos

Bosch Electrohydraulic Brake EHB

- 1 Electrohydraulic actuator for EHB, ABS, ASR, ESP
- 2 EHB - ECU
- 3 Active, direction-sensitive WSS
- 4 Brake operation unit with integrated pedal stroke sensor
- 5 Steering wheel angle sensor
- 6 Yaw rate and lateral acceleration sensor
- 7 Engine management ECU



BOSCH



Reproduction free of charge with notation "Photo: Bosch".

Photo photo No. 1-K1-10583

Fuente: Daniel W. Lewis, "Fundamentals of Embedded Software"

Ejemplos de sistemas embebidos

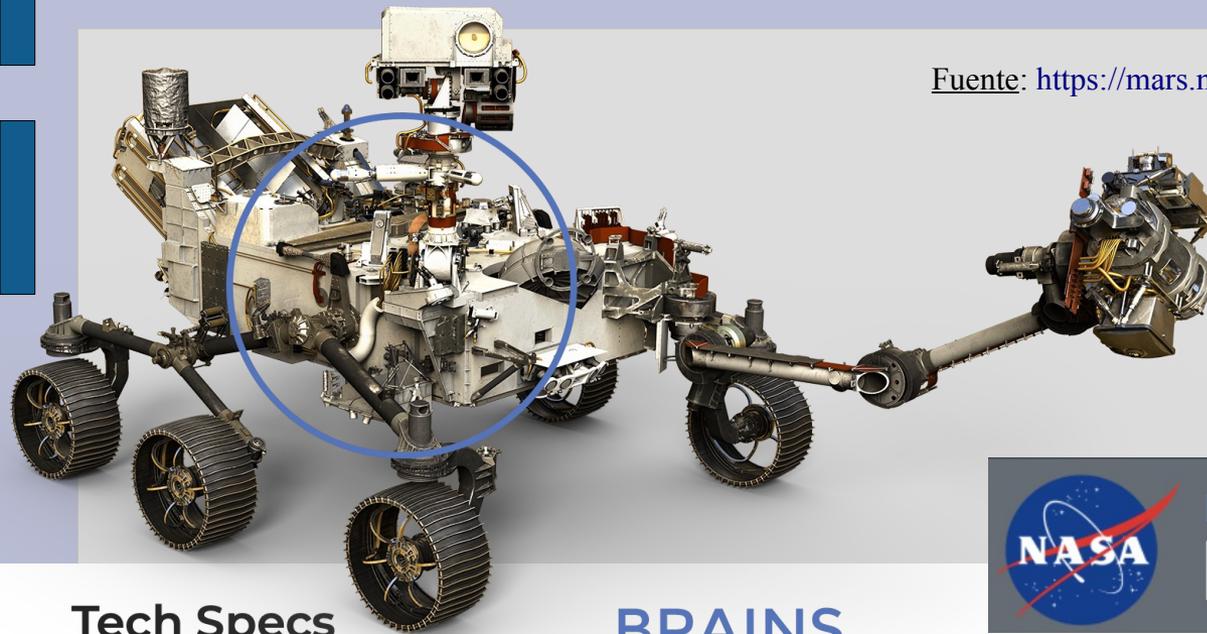


- Producto: NASA's Mars Sojourner Rover.
“First wheeled vehicle to rove another planet”
- Microprocesador:
8-bit Intel 80C85

Fuente: Daniel W. Lewis, “Fundamentals of Embedded Software”

Ejemplos de sistemas embebidos

Fuente: <https://mars.nasa.gov/mars2020/spacecraft/rover/brains/> , 1/3/2021



Microprocesador
de 32 bits (2001)

Tech Specs

BRAINS



Processor

- Radiation-hardened central processor with PowerPC 750 Architecture: a BAE RAD 750
- Operates at up to 200 megahertz speed, 10 times the speed in Mars rovers Spirit and Opportunity's computers

Memory

- 2 gigabytes of flash memory (~8 times as much as Spirit or Opportunity)
- 256 megabytes of dynamic random access memory
- 256 kilobytes of electrically erasable programmable read-only memory

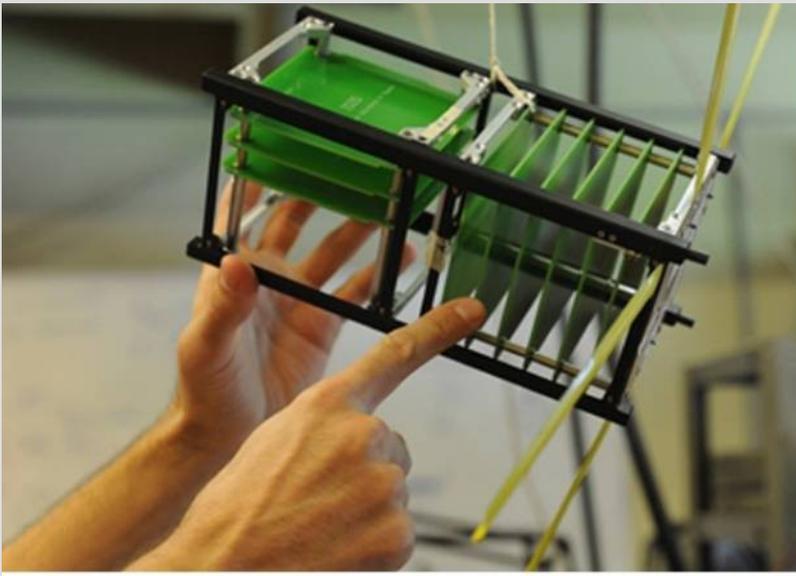
Ejemplos de sistemas embebidos



- Producto:
GloboSat01 IIE (2008)
- Microprocesadores:
 - ATmega2560 (aplicación)
 - PIC16F877A (telemetría)

Ejemplos de sistemas embebidos

- AntelSat (IIE + ANTEL)
- Microprocesador: MSP430



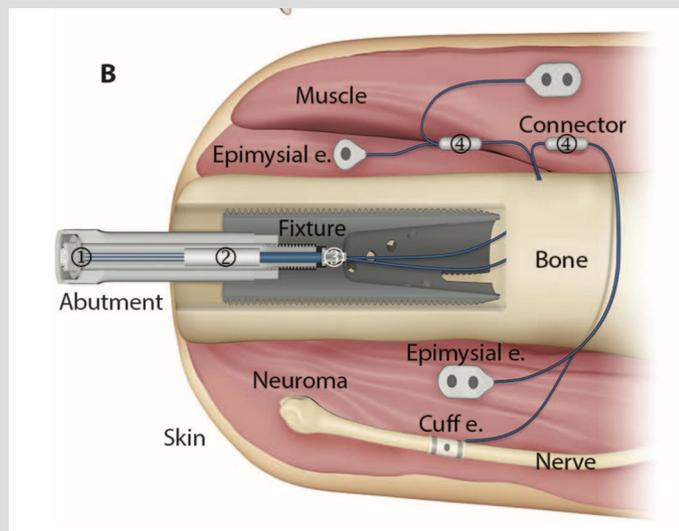
Ejemplos de sistemas embebidos

EEG inalámbrico



Causa et al, LASCAS 2018

Ortiz-Catalan et al, ST MED 2014



Control de prótesis neurales



Comportamiento animal

Cabrera et al, CAFE 2023



Algunos proyectos Sisem de años anteriores

- **Juegos:** UARTetris: Tetris controlado por UART(2021) y Viborita (CCuP, 2019)
- **Control e Industrial:** Monitoreo y control de una estufa de leña de alto rendimiento (2019), Calentador de agua inteligente (SmartSUN, 2019), Control de Cortina Metálica de Garaje (2021) y otros
- **Campo y Animales:** Determinación de la dinámica espacial de una oveja (2020), Sistema de Invernadero Automatizado (SINVA, 2020), Sistema de confinamiento virtual para mascotas (2023) y otros
- **Ing. Biomédica:** Adquisidor de señales de signos vitales (2020), Mouse controlado por gestos (GesturesMouse, 2019), Medida de la impedancia transtorácica en pacientes pediátricos (2019) y otros
- Ver detalles en página web del curso.

Objetivos del curso

- Enfrentar proyectos de desarrollo de software para sistemas embebidos
- Describir diferentes arquitecturas de software para sistemas embebidos.
- Seleccionar la arquitectura adecuada para cada aplicación particular.
- Aplicar adecuadamente las técnicas correspondientes para implementar el sistema.

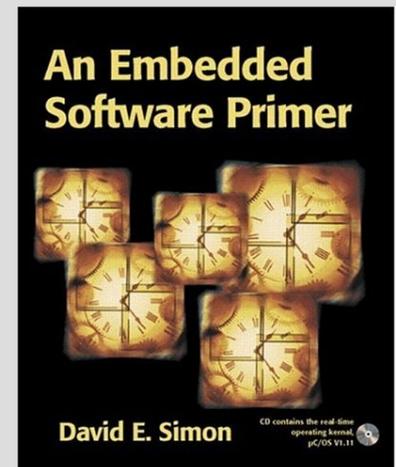
Metodología

- Clases teóricas
- Laboratorios
- Proyecto final grupal

Clases teóricas

- 2 clases por semana de 1,5 hs. hasta primeros parciales
 - Presentación y debate
 - Actividades en clase
 - Trabajo domiciliario

- Libro de texto: Simon



- Clases de 2021 disponibles en OpenFing

Laboratorios

- Carga: 4 sesiones
- Complejidad creciente / acumulativos
- Trabajo en equipo: 2 o 3 estudiantes
- Plataforma de hardware (préstamo a cada estudiante):
 - MSP430 LaunchPad de Texas Instruments (MSP430G2553)
- **Basado en trabajo previo domiciliario**
- **Defensa presencial con cuestionario**
- Probar/desarrollar con los mismos equipos que van hacer las defensas

Laboratorios

1: Introducción al entorno de desarrollo

- Control de versiones (GIT)
- Lenguaje C

2: Interrupciones

- Periféricos = BCM+, Timer, ADC
- Documentación automática de código (Doxygen)
- Problema de datos compartidos

3: Diseño basado en interrupciones (parte 1)

- Round-Robin con interrupciones
- UART (comunicación serial)
- Debug con Osciloscopio y Analizador Lógico

4: Diseño basado en interrupciones (parte 2)

- Planificación por encolado de funciones
- Bajo consumo (LPM, Energy-Trace, Ultra Low Power Advisor)
- Test (ASSERT)

Proyecto final

- En equipo (grado)
 - 3 estudiantes (para grado, posgrado puede variar)
 - 80 horas/persona (alta carga)
- Plataformas:
 - Plataformas de hardware (préstamo a cada estudiante):
 - MSP430 LaunchPad de Texas Instruments (MSP430G2553)
 - Boosterpack Sensor de Texas Instruments y otros sensores o módulos
 - Analog Discovery 2 from Digilent (1 placa por grupo)
- Tema
 - Depende de cantidad de estudiantes
 - Problema especificado por los docentes
 - Propuesto por estudiantes (si tienen un idea plantearlo cuanto antes)

Reglas del curso

- Reglamento del curso:
 - Disponible en EVA
 - Recorrida rápida conjunta del documento
- Cronograma
 - Disponible en EVA
 - Recorrida rápida conjunta del documento

Próximos pasos

- 14/3 son las 1eras defensa de laboratorios.
- Definir grupos rápido:
 - Mandar mail a juliano@fing.edu.uy y mgonzalez@fing.edu.uy con propuesta antes del miércoles 6/3 a las 20 horas.

Próximos pasos

- 14/3 son las 1eras defensa de laboratorios.
- Entrega de placas
 - lunes 4/3 de 11:30 a 12:00 hs
 - Miércoles 6/3 de 18 a 19 hs
 - Jueves 7/3 de 9 a 9:50 hs (a las 10 hay clase)
 - Laboratorio de medidas del IIE
 - Si no pueden concurrir mandar mail a vcabrera@fing.edu.uy, rodrigog@fing.edu.uy y jlema@fing.edu.uy

Reglas del curso

- Clases teóricas:
 - en caso de inasistencia hay ejercicios de teórico
- Asistencia obligatoria:
 - 4 Laboratorios
 - Proyecto: 2 presentaciones (hito y final)
- Honestidad académica:
 - Trabajo independiente (individual o grupal)
 - “copy & paste” vs. paráfrasis & referencias
 - Intercambio de ideas vs. “copiar”
 - ChatGPT

Aprobación del curso

- En Laboratorio (40 puntos) se evaluará mediante:
 - Cuestionario individual vía EVA
 - Realización de tareas previas
 - Desempeño y actitud durante la defensa
- El Proyecto (60 puntos) se evaluará mediante:
 - Elaboración de la especificación
 - Realización del proyecto
 - Presentaciones (hito intermedio y defensa)
 - Entrega final: Código, Documentación, Video, etc.

Temario del curso

1. Introducción y conceptos básicos.
2. Microcontroladores e Interrupciones.
3. Arquitecturas de software embebido
4. Introducción a los SO para tiempo real (RTOS)
5. Sistemas embebidos de bajo consumo
6. Criterios de diseño de un sistema embebido
7. Otros temas de Sistemas Embebidos

Material del curso

- Página web de la asignatura:
 - <https://eva.fing.edu.uy/course/view.php?name=sisem>
- Transparencias del curso.
- Material para laboratorio y proyectos.
- Bibliografía básica:
 - “An Embedded Software Primer”, David E. Simon, Addison-Wesley Professional , 1999.
ISBN: 020161569X
 - "Real-Time Embedded Systems. Design principles and engineering practices", Xiaocong Fan, 2015
ISBN 978-0-12-801507-0