

# **Aplicaciones de uso de software – Uso de Diagramas de Estado en Sistemas Embebidos**



## Información relevante

### 5to Congreso Virtual de Microcontroladores y sus Aplicaciones

Disponemos con una buena variedad de Áreas Temáticas y esperamos contar con vuestra asistencia y una muestra de sus trabajos a efectos de compartir con toda nuestra comunidad. Además de los trabajos contaremos con Seminarios Web en el transcurso del Congreso que versarán sobre los temas que nos interesan

### Taller de Sistemas Embebidos

Asignatura correspondiente a la **actualización 2023** del Plan de Estudios 2020 y resoluciones modificatorias, de Ingeniería Electrónica de FIUBA

### Más información . . .

. . . sobre el **5to Congreso** . . . <https://www.cvm.utn.edu.ar/>

. . . sobre el **Taller de Sistemas Embebidos** . . . <https://campusgrado.fi.uba.ar/course/view.php?id=1217>

*Por Ing. Juan Manuel Cruz, partiendo de la platilla Salerio de Slides Carnival*

*Este documento es de uso gratuito bajo Creative Commons Attribution license (<https://creativecommons.org/licenses/by-sa/4.0/>)*

*You can keep the Credits slide or mention SlidesCarnival (<http://www.slidescarnival.com>), Startup Stock Photos (<https://startupstockphotos.com/>), Ing. Juan Manuel Cruz and other resources used in a slide footer*



# ¡Hola!

Soy Juan Manuel Cruz  
Taller de Sistemas Embebidos  
Consultas a: [jcruz@fi.uba.ar](mailto:jcruz@fi.uba.ar)

# 1

## Contexto

Actualización 2023 del Plan de Estudios 2020 y resoluciones . . .



*Taller de Sistemas Embebidos*  
*Segundo Ciclo*  
*Asignatura obligatoria*  
*3<sup>er</sup> año (6° cuatrimestre)*  
*6 horas semanales (2 encuentros)*



## Objetivos & Contenidos

### ■ (TA134) Taller de Sistemas Embebidos

<https://campusgrado.fi.uba.ar/course/view.php?id=1217>

#### ▷ Objetivos:

- ▷ Elaboración de un **Proyecto** de Sistemas Embebidos (**Intermedio**), junto con el diseño y cálculo, ...

#### ▷ Contenidos:

- ▷ Microcontroladores utilizados en Sistemas Embebidos. Introducción a la arquitectura de Microcontroladores de 32 bit o superior y sus componentes básicos, ...



## Objetivos & Contenidos

- ▷ Modelo del programador, modos de operación, mapa de memoria, registros, Stack, FPU, core peripherals, ...
- ▷ Estrategias de control de periféricos, ...
- ▷ Conversión A/D y D/A, ...
- ▷ Metodologías de diseño, buenas prácticas, uso de repositorios, ...
- ▷ Programación en lenguaje C, estilo de codificación, ...
- ▷ Programación Modular, Bare Metal (sin Sistema Operativo), Gobernada por Eventos, Máquinas/Diagramas de Estado, ...
- ▷ Técnicas de verificación y validación, On chip debugging, ...



# Proyecto Intermedio



Un enfoque centrado en la **práctica propia de la carrera** más que en el desarrollo teórico disciplinar, con eje en la **participación** de las y los **estudiantes**





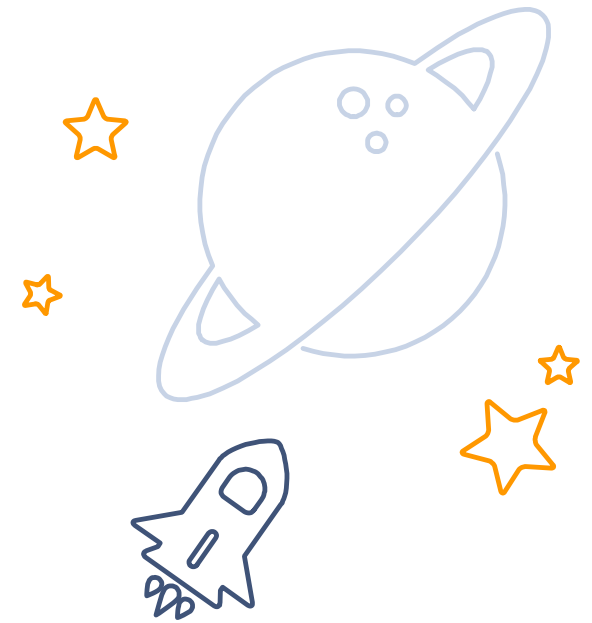
# ¡Hola!

Soy Juan Manuel Cruz  
Taller de Sistemas Embebidos  
Consultas a: [jcruz@fi.uba.ar](mailto:jcruz@fi.uba.ar)



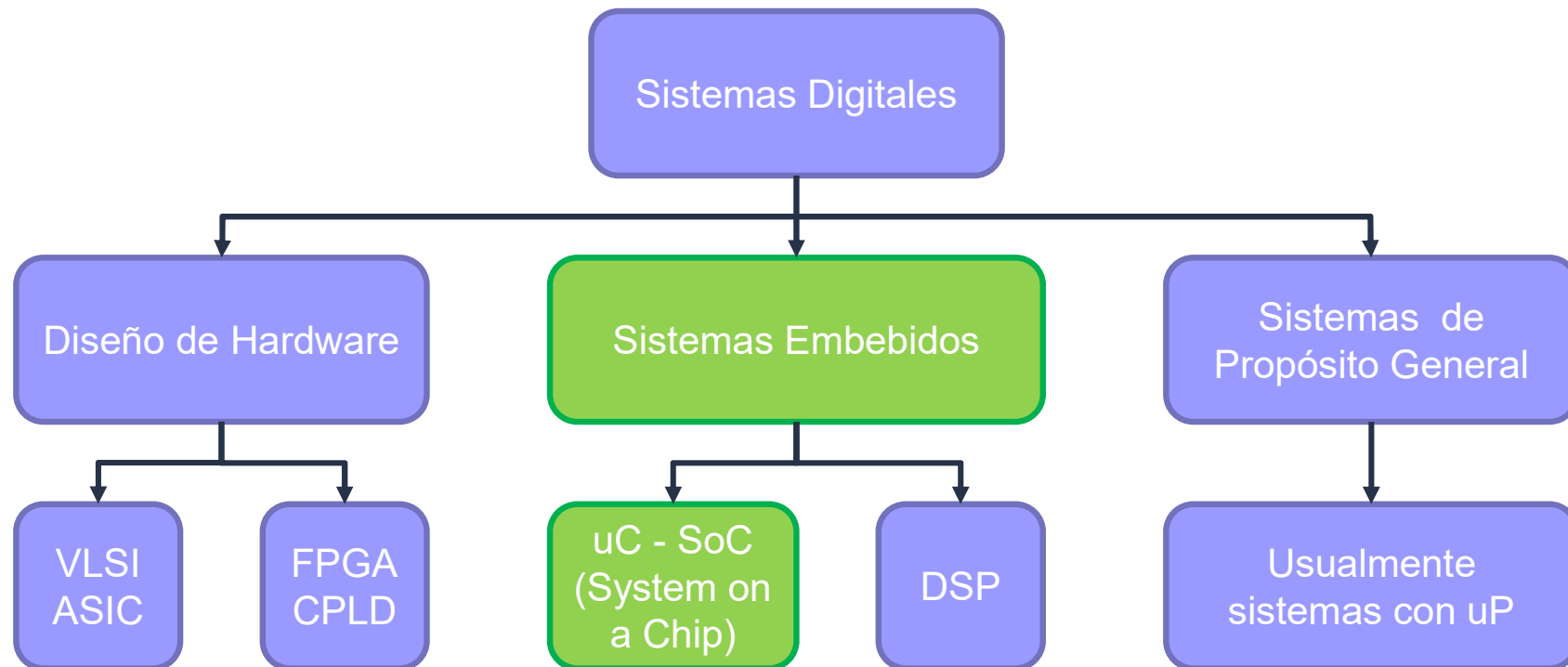
# Formar Ingenieros

El ingeniero aplica su intelecto y las habilidades técnicas y profesionales adquiridas, para planificar, diseñar y crear cualquier cosa que solucione problemas de la vida real





## ¿En qué nos concentraremos?





## Estado del Arte

- Contamos con **plataformas** (uC - SoC/DSP/FPGA/ASIC/ etc.) de **rendimiento** y **recursos** en **crecimiento** que permitan atender el **incremento** del **procesamiento**, necesario para soportar periféricos avanzados con capacidad de atender nuevas **conectividades** e **interfaces de usuario** requeridas por el **mercado** (usuarios)
- Variada oferta de **plataformas** competitivas en **costo**, **disponibilidad**, **soporte**, **herramientas** (de HW & SW); en especial en el campo de los **microcontroladores** de nueva generación (ARM: 32 bits))
- Esto permite recurrir a las mejores prácticas de **Ingeniería de Software**, al uso de **modelos**, **lenguajes de alto nivel**, con y sin un **sistema operativo de tiempo real** (RTOS o Bare Metal), empleando técnicas de programación específicas para lograr **eficiencia**, **confiabilidad** y **re-usabilidad**



# Solución Adecuada

... lo más **simple** posible, previa determinación del objetivo de **excelencia** a cumplir, obviamente contando con la **documentación debida** y recurriendo a la **metodología de trabajo adecuada**



# 2

## Solución Adecuada

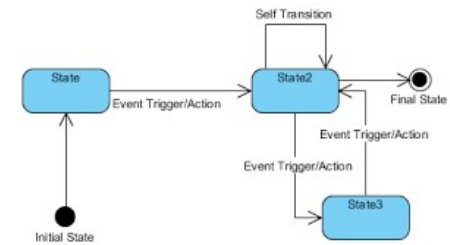
1er Cuatrimestre de 2024, dictado por primera vez . . .



*Diseño/Desarrollo/Depuración de  
Hardware y Software: Circuito eléctrico  
(electrónica analógica/digital/  
alimentación) – Circuito Impreso –  
Producto – Manufactura – Programas  
(Firmware/Middleware/Software)*



# Diseño/Desarrollo/Depuración de Software



El **Unified Modeling Language (UML)** es un **lenguaje** de **modelado** de propósito general

- Su **objetivo** es definir una **forma estándar** de **visualizar** la forma en que se ha diseñado un **sistema** (similar a los planos utilizados en otros campos de la ingeniería)
- **No** es un **lenguaje** de **programación**, es más bien un **lenguaje visual**

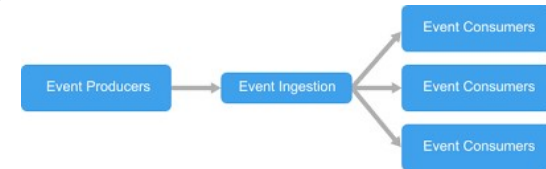
**State Machine Diagrams (Máquinas de Estado) | Unified Modeling Language (UML)**, se usan para **representar** la **condición** del **sistema** (o parte del mismo) en **instantes** de **tiempo** finito

- Diagrama que representa el **comportamiento** mediante **transiciones** de **estados** finitos
- Se conoce como **State-Chart Diagrams (Diagrama de Estados)**
- De manera simple, se utiliza un **diagrama** de **estado** para **modelar** el **comportamiento dinámico** de una clase en respuesta al **tiempo** y a **estímulos** externos cambiantes.  
Podemos decir que todas y cada una de las clases tienen un estado, pero no modelamos todas las clases utilizando diagramas de estado





## Diseño/Desarrollo/Depuración de Software



- Como solución a **Sistemas Embebidos** orientados a **Control**, tenemos los **Sistemas Disparados/Activados** por **Tiempo** o por **Evento** (**Time- and Event-Triggered Systems**)
  - ▶ Un **disparador/activador** (**trigger**) es un **suceso** (**evento/sincronismo**) que provoca el **inicio** de alguna **acción** en el **Sistema de Control**
  - ▶ La acción puede ser la ejecución de una tarea leyendo una variable y calculando un nuevo valor de una variable de corrección, o el envío de un mensaje informando valores actuales de variables como presión o temperatura
- En el **Control activado/disparado** por **Evento**, una acción se inicia sólo si ocurre un **evento** significativo (**condición de programa/encuesta/interrupción**)
  - ▶ Por ejemplo, un sensor enviaría un mensaje solo si la temperatura ha cambiado más de 3 °C desde que se envió el último mensaje



## Diseño/Desarrollo/Depuración de Software

- Codificaremos en C
  - ▷ "Nothing better than C", Linus Torvalds
- Aplicando el estándar de codificación
  - ▷ Embebidos Embedded C Coding Standard by Michael Barr
- Codificaremos soluciones del tipo:
  - ▷ Estructurada, Modular
    - ▷ Escutar, Procesar y Actuar
  - ▷ Bare Metal (sin Sistema Operativo)
    - ▷ Super-Loop (Polling & Interrupts), con modelado de tareas (Diagramas de Estado)
    - ▷ Event-Triggered Systems

# 3

## ¿Vamos bien ?

1er Cuatrimestre de 2024, dictado por primera vez . . .

“ *Hablando de aplicaciones de uso de software, la pregunta del millón es:  
¿ Uso de Diagramas de Estado en  
Sistemas Embebidos ?*



## Necesitamos una lista de acciones

■ El material de esta charla esta disponible en:

- ▶ El sitio web del proveedor de la **herramienta** (crudo en: <https://www.itemis.com/en/>)
- ▶ O bien, referido en del Campus Virtula FIUBA Grado del curso (semielaborado en: <https://campusgrado.fi.uba.ar/course/view.php?id=1217&section=0#tabs-tree-start>), a saber:
  - ▶ Referencias -> **Conferences**
  - ▶ Teoría -> **Semanas 1 a 3**
  - ▶ Práctica -> **Semanas 1 a 2**
  - ▶ Práctica -> **Semanas 3 a 4**



## Necesitamos una lista de acciones

■ Vamos por partes: Práctica -> [Semanas 1 a 2](https://campusgrado.fi.uba.ar/course/view.php?id=1217&section=24#tabs-tree-start)  
(<https://campusgrado.fi.uba.ar/course/view.php?id=1217&section=24#tabs-tree-start>):

- ▷ TP0 – Herramientas – folder
  - ▷ Descargar: [TdSE\\_Guia\\_TP0-Herramientas.pdf](#)
  - ▷ Buscar: Itemis
  - ▷ Realizar:
    - ▷ Paso TP0-00-05: Registrarse ... (**Tarea para el hogar**)
    - ▷ Paso TP0-00-06: Descargar, instalar, licenciar ... (**Tarea para el hogar**)



## Necesitamos una lista de acciones

■ Vamos por partes, Práctica -> [Semanas 3 a 4](https://campusgrado.fi.uba.ar/course/view.php?id=1217&section=25#tabs-tree-start)  
(<https://campusgrado.fi.uba.ar/course/view.php?id=1217&section=25#tabs-tree-start>):

- ▷ TP1 – Diagramas de Estado – Modelado – folder
  - ▷ Descargar: [TdSE\\_Guia\\_TP1-Diagramas\\_de\\_Estado-Modelado.pdf](#)
  - ▷ Buscar: Tour
  - ▷ Realizar:
    - ▷ Paso TP1-01-02: Ingresar ... (**Tarea para el hogar**)
    - ▷ Paso TP1-01-03: Simular ... (**Tarea para el hogar**)



## Necesitamos una lista de acciones

- Vamos por partes, Itemis -> Products -> [Itemis Create](https://www.itemis.com/en/products/itemis-create/)  
(<https://www.itemis.com/en/products/itemis-create/>):
  - ▷ Documentation (<https://www.itemis.com/en/products/itemis-create/documentation/>)
    - ▷ User Guide ([https://www.itemis.com/en/products/itemis-create/documentation/user-guide/overview\\_what\\_are\\_state\\_machines?hsLang=en](https://www.itemis.com/en/products/itemis-create/documentation/user-guide/overview_what_are_state_machines?hsLang=en))
      - ▷ What is a state machine? ...
      - ▷ What is itemis CREATE? ...
      - ▷ Quick reference ...





## Necesitamos una lista de acciones

- Vamos por partes, Itemis -> Products -> [Itemis Create](https://www.itemis.com/en/products/itemis-create/)  
(<https://www.itemis.com/en/products/itemis-create/>):
  - ▷ Documentation (<https://www.itemis.com/en/products/itemis-create/documentation/>)
    - ▷ User Guide ([https://www.itemis.com/en/products/itemis-create/documentation/user-guide/overview\\_what\\_are\\_state\\_machines?hsLang=en](https://www.itemis.com/en/products/itemis-create/documentation/user-guide/overview_what_are_state_machines?hsLang=en))
      - ▷ [itemis CREATE for Eclipse ...](#) (Tarea para el hogar)
      - ▷ [Statechart language reference ...](#) (Tarea para el hogar)
      - ▷ ...



## Necesitamos una lista de acciones

- Vamos por partes, Itemis -> Products -> **Itemis Create**  
(<https://www.itemis.com/en/products/itemis-create/>):
  - ▷ Documentation (<https://www.itemis.com/en/products/itemis-create/documentation/>)
    - ▷ **Tutorials & Videos** (**Tarea para el hogar**)  
(<https://www.itemis.com/en/products/itemis-create/documentation/tutorials>)
    - ▷ **Examples** (**Tarea para el hogar**) (<https://www.itemis.com/en/products/itemis-create/documentation/examples/>)
    - ▷ ...



## Necesitamos una lista de acciones

- Vamos por partes, Referencias -> **Conferences**  
(<https://campusgrado.fi.uba.ar/course/view.php?id=1217&section=14#tabs-tree-start>):
  - ▷ Itemis-Create.zip & STM32CubeIDE.zip – folder
    - ▷ Descargar: **Itemis-Create.zip**
    - ▷ Importar a Itemis Create Desktop: el proyecto **c\_project-application**, contenido en **Itemis-Create.zip**
    - ▷ Editar ... c\_project-application/model/**statechart.ysc**
    - ▷ Identificar y analizar los **modelos** del proyecto y sus **elementos** constitutivos (declaraciones, regiones, pseudo estado inicial, estados simples, estados compuestos, transiciones, eventos, guardas, efectos, variables, constantes, ...)
    - ▷ ...



## Necesitamos una lista de acciones

- ▷ Editar ... c\_project-application/model/[statechart.sgen](#)
- ▷ Identificar y analizar sus **elementos** constitutivos
- ▷ Identificar los **bibliotecas** ([sc\\_timer\\_service](#))
  - ▷ ... c\_project-application/model/inc/[sc\\_timer\\_service.h](#)
  - ▷ ... c\_project-application/model/src/[sc\\_timer\\_service.c](#)
- ▷ Generar **código C** e identificar los **archivos** generados
  - ▷ ... c\_project-application/model/inc/[sc\\_prefix.h](#), [sc\\_prefix\\_required.h](#) & [sc\\_types.h](#)
  - ▷ ... c\_project-application/model/src/[sc\\_prefix.c](#)
  - ▷ ...



## Necesitamos una lista de acciones

■ Vamos por partes, Referencias -> [Conferences](#)

(<https://campusgrado.fi.uba.ar/course/view.php?id=1217&section=14#tabs-tree-start>):

- ▷ Itemis-Create.zip & STM32CubeIDE.zip – folder
  - ▷ Descargar: [STM32CubeIDE.zip](#)
  - ▷ Importar a STM32CubeIDE: el proyecto [tdse-itemis\\_create-application-nucleo\\_f103rb](#), contenido en [STM32CubeIDE.zip](#)
  - ▷ Compilar y depurar el proyecto [tdse-itemis\\_create-application-nucleo\\_f103rb](#) ...
    - ▷ Ejecutar [Resume (F8)]
    - ▷ Observar el estado del [led verde](#) (LD2) de la placa NUCLEO-F103RB
    - ▷ Observar la [evolución de variables globales](#) en Live Expressions view



## Necesitamos una lista de acciones

- ▷ Mantener oprimido el **botón azul** (BT1) de su placa NUCLEO-F103RB
  - ▷ Observar el estado del **led verde** (LD2) de la placa NUCLEO-F103RB
  - ▷ Observar la evolución de **variables globales** en Live Expressions view
- ▷ Dejar de oprimir el **botón azul** (BT1) de su placa NUCLEO-F103RB
  - ▷ Observar el estado del **led verde** (LD2) de la placa NUCLEO-F103RB
  - ▷ Observar la evolución de **variables globales** en Live Expressions view
- ▷ Contrastar el comportamiento del **led** y la evolución de **variables globales** observados con el modelo del proyecto
- ▷ Contrastar el **tiempo de ejecución** de la tarea `task_sc_prefix_update()`
- ▷ ...



## Necesitamos una lista de acciones

- ▷ ...
- ▷ Detener [Suspend]
- ▷ Resetear [Reset the chip and restart debug session]
- ▷ Ejecutar por pasos [Step Into (F5)] o [Step Over (F6)]
  - ▷ Seguir la ejecución hasta llegar al código generado por [Itemis Create](#)
- ▷ ...



## Referencias

- Todas mis presentaciones terminan con la lista de referencias y sus respectivos link, en este caso, todas ellas están contenidas en páginas previas de esta presentación

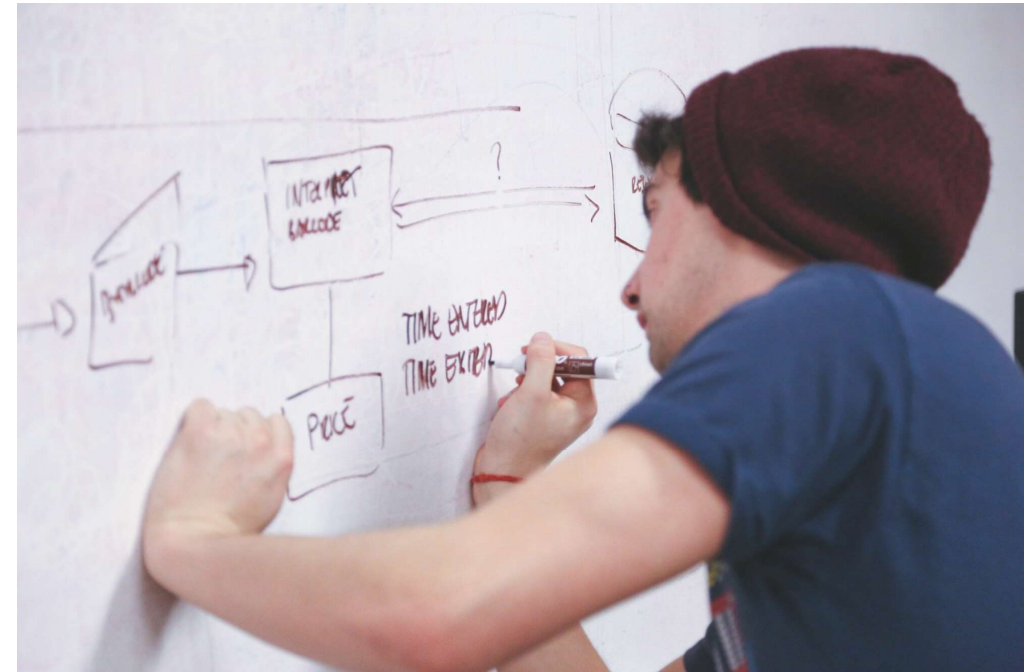




## Manos a la obra con el . . .

. . . Proyecto Intermedio

. . . un enfoque centrado en la práctica propia de la carrera más que en el desarrollo teórico disciplinar, con eje en la participación de las y los estudiantes



A person with short dark hair, seen from the back, is looking at a wall covered in various design sketches, photos, and notes. The wall is a collage of creative work, including wireframes, hand-drawn diagrams, and photographs of people and objects. The person is wearing a grey and black striped sweater. A dark blue arrow points from the left towards the person's head.

Las y los estudiantes preguntarán:  
**¿en qué lío nos metimos?**



# ¡Muchas gracias!

¿Preguntas?

...

Consultas a: [jcruz@fi.uba.ar](mailto:jcruz@fi.uba.ar)