

Seminario de Electrónica: Sistemas Embebidos - Trabajo Práctico N° 1

LPC43xx Entradas y Salidas (Digitales) de Propósito General (GPIO) – Diagrama de Estado

Objetivo:

- **Uso del IDE** (edición, compilación y depuración de programas)
- **Uso de GPIO & Diagrama de Estado** (manejo de Salidas y de Entradas Digitales en Aplicaciones)
- **Documentar lo que se solicita en c/ítems**

Referencias (descargar del Campus Virtual del curso a fin de usarlas durante la realización del TP):

- **Diagrama de Estado:** <http://campus.fi.uba.ar/mod/resource/view.php?id=51884>
- **DE-GNU MCU Eclipse & Yakindu SCT:** <http://campus.fi.uba.ar/mod/resource/view.php?id=79378>
- **LPC435X_3X_2X_1X Product Data Sheet:** <http://campus.fi.uba.ar/mod/resource/view.php?id=28519>
- **LPC43XX User Manual (Chapter 1, 18 & 19):** <http://campus.fi.uba.ar/mod/resource/view.php?id=77765>
- **EDU-CIAA-NXP (web site):** <http://proyecto-ciaa.com.ar/devwiki/doku.php?id=desarrollo:edu-ciaa:edu-ciaa-nxp>
- **EDU-CIAA-NXP (esquemático):** http://www.proyecto-ciaa.com.ar/devwiki/lib/exe/fetch.php?media=desarrollo:edu-ciaa:edu-ciaa-nxp:edu-ciaa-nxp_color.pdf
- **EDU-CIAA-NXP (pinout):** http://proyecto-ciaa.com.ar/devwiki/lib/exe/fetch.php?media=desarrollo:edu-ciaa:edu-ciaa-nxp_pinout_a4_v4r2_es.pdf

1. Uso del IDE (Integrated Development Environment) GNU MCU Eclipse (p/Linux o p/Windows)

- Previo a éste TP ya se Descargó, Instaló, Ejecutó, Configuró y Licenció todo lo necesario para desarrollar éste TP
 - Instaló **CIAA-LAUCHER**, ejecutó **GNU MCU Eclipse**, agregó **firmware_v3**, configuró Debug para EDU-CIAA-NXP, agregó el plug-in de **eGIT**, agregó **Yakindu StateChart Tools (STC)**, gestionó la licencia respectiva y al recibirla la cargó
 - Antes de ejecutar asegúrese tener conectada la placa **EDU-CIAA-NXP** a su PC (recuerde conectarla **siempre al mismo puerto USB**) a través de la interfaz **Debug** (para **editar & simular** statecharts **NO** necesita **conectar** ninguna **placa**)
 - Seleccionar el Workspace: **C:\CIAA\CIAA_Software_1.1-Win\workspaces\eclipse-ws**
 - En el archivo **program.mk** podrá configurar el **programa** en el que se trabajara:
PROGRAM_PATH = examples/c/sapi/statecharts
PROGRAM_NAME = 1_toggle
 - En el archivo **board.mk** podrá configurar la **placa** a utilizar:
BOARD = edu_ciaa_nxp
 - Verifique tener en la carpeta **examples/c/sapi/statecharts/1_toggle/** los archivos:
 - toggle.sct** Yakindu SCT Statechart Model file
 - toggle.sgen** Yakindu SCT Code Generator Model file
 - Para **Editar** el **modelo**: Doble clic sobre **toggle.sct**
 - Para **Simular** el **modelo**: Clic derecho sobre **toggle.sct** -> Run Us -> 1 Satechart Simulation
 - Para Editar la generación de código del modelo: Doble clic sobre **toggle.sgen**
 - Actividad **a desarrollar** cuando cuente con una placa **EDU-CIAA-NXP**
 - Para Generar el código del modelo: Clic derecho sobre **toggle.sgen** -> Generate Code Artifacts (Artifacts => **gen/Toggle.c**, **gen/Toggle.h**, **gen/ToggleRequired.h** y **inc/sc_types.h**)
 - Actividad **a desarrollar** cuando cuente con una placa **EDU-CIAA-NXP**
 - Compilar **firmware_v3** => Configurar **Debug** => Probar **Debug**
 - Actividad **a desarrollar** cuando cuente con una placa **EDU-CIAA-NXP**
 - Documentar** mediante tablas c/texto e imágenes la estructura de **archivos**, su tipo/contenido (especialmente **readme.txt**) de c/proyecto importado
 - Actividad **a desarrollar** cuando cuente con una placa **EDU-CIAA-NXP**
 - Documentar** mediante tablas c/texto e imágenes la secuencia de **funciones** invocadas durante la ejecución del ejemplo de aplicación, en qué archivo se encuentran, su descripción detallada, qué efecto tiene la aplicación sobre el hardware (identificar circuitos, puertos, pines, niveles, etc.) así como la interacción entre las mismas
 - Actividad **a desarrollar** cuando cuente con una placa **EDU-CIAA-NXP**
 - Idem b** pero con **datos** (definiciones, constantes, variables, estructuras, etc.)
 - Actividad **a desarrollar** cuando cuente con una placa **EDU-CIAA-NXP**

- d. En **PROGRAM_PATH** = **examples/c/sapi/statecharts** encontrará varios ejemplos más con los que podrá hacer pruebas:
- | | |
|--|--|
| PROGRAM_NAME = 2_blink | PROGRAM_NAME = 4_buttons |
| PROGRAM_NAME = 3_idleBlink | PROGRAM_NAME = 5_application |

2. Uso del **IDE** (Integrated Development Environment) **GNU MCU Eclipse** (p/Linux o p/Windows)

- a. Para **editar & simular** un **modelo** que cumpla las consignas del TP debe:
- En el path **examples/c/sapi/statecharts** crear la carpeta **0_Test**
 - Copiar el modelo **toggle.sct** y pegar en el path **examples/c/sapi/statecharts/0_Test**
 - Renombrar el modelo **toggle.sct** y su contenido acorde a las consignas de éste **TP** (o hacerlo con aquel ejemplo que mejor se ajuste a sus necesidades)
 - Para facilitar la corrección de su trabajo se le ha pedido previamente:
 - Ingresar** a <https://github.com/>
 - Crear** una **cuenta** si no dispone de una
 - Informar** su nombre de usuario al docente (para ser agregado al *team* correspondiente a su grupo, o lo que hayan recomendado los ayudantes)
 - Crear** un **repositorio** denominado **TP1**. (La URL del mismo será parecida a <https://user...name/TP1>, o lo que hayan recomendado los ayudantes)
 - Realizar** un **commit/push** inicial del **modelo** actual.
 - De ahora en adelante, **actualizar su repositorio** mediante **commit/push**
- b. Para **generar el código** del **modelo** debe:
- Copiar la carpeta del modelo que mejor se ajuste a sus necesidades y pegar en el path **examples/c/sapi/statecharts** renombrado acorde a las consignas de éste **TP**
 - Además de renombrar el modelo debe buscar el nombre del modelo copiado y reemplazar por el nombre del nuevo modelo en los archivos: **.sct**, **.sgen** y **main.c**
 - Para facilitar la corrección de su trabajo se le ha pedido previamente:
 - Ingresar** a <https://github.com/>
 - Realizar** un **commit/push** inicial del **código** actual.
 - De ahora en adelante, **actualizar su repositorio** mediante **commit/push**
 - Actividad **a desarrollar** cuando cuente con una placa **EDU-CIAA-NXP**

3. **Implementar** (editar, simular y generar el código) el **modelo** de control de **Panel de Control** de un generador de señales (tensión de 0 a 10V, frecuencia de 20 a 20.000Hz y 3 formas de señal)
- Editar y simular:** Actividad desarrollada durante la clase del **30/04**
 - Generación de código:** Actividad **a desarrollar** cuando cuente con una placa **EDU-CIAA-NXP**
4. **Implementar** (editar, simular y generar el código) el **modelo** de control de **Puerta Corrediza** automatizada (motor con movimiento en dos sentidos, sensor de presencia y fines de carrera)
- Editar y simular:** Actividad desarrollada durante la clase del **30/04**
 - Generación de código:** Actividad **a desarrollar** cuando cuente con una placa **EDU-CIAA-NXP**
5. **Implementar** (editar, simular y generar el código) el **modelo** de control de **Portón Levadizo** automatizado (motor con movimiento en dos sentidos, control remoto de apertura/cierre, fines de carrera y señalización luminosa)
- Editar y simular:** Actividad desarrollada durante la clase del **30/04**
 - Generación de código:** Actividad **a desarrollar** cuando cuente con una placa **EDU-CIAA-NXP**
6. **Implementar** (editar, simular y generar el código) el **modelo** de control de **Escalera Mecánica** unidireccional automatizada (motor c/movimiento en un sentido y dos velocidades, sensores de ingreso, egreso y señalización luminosa)
- Editar y simular:** Actividad que desarrollará: **07/05** (editar, simular), **14/05** (editar, simular) y hasta el **21/05** (generar el código)
 - Generación de código:** Actividad **a desarrollar** cuando cuente con una placa **EDU-CIAA-NXP**
7. **Implementar** (editar, simular y generar el código) el **modelo** de control de **Horno Microondas** (3 modos de cocción seleccionable por botón de modo, botón de comenzar/terminar y sensor de apertura de puerta)
- Editar y simular:** Actividad que desarrollará: **07/05** (editar, simular), **14/05** (editar, simular) y hasta el **21/05** (generar el código)
 - Generación de código:** Actividad **a desarrollar** cuando cuente con una placa **EDU-CIAA-NXP**
8. **Implementar** (editar, simular) el **modelo** de control de **xxxx xxxx** (dicho modelo debe ser una parte de la implementación de su TPF)
- Editar y simular:** Actividad que desarrollará: **14/05** (editar, simular) y **21/05** (editar, simular)
 - Generación de código:** Actividad **a desarrollar** cuando cuente con una placa **EDU-CIAA-NXP**