

# Programación en la EDU-CIAA-NXP

Guillermo Steiner

Centro de Investigación en Informática para la Ingeniería  
Universidad Tecnológica Nacional, F.R.C.

<http://ciiii.frc.utn.edu.ar>

Córdoba, Argentina



2016

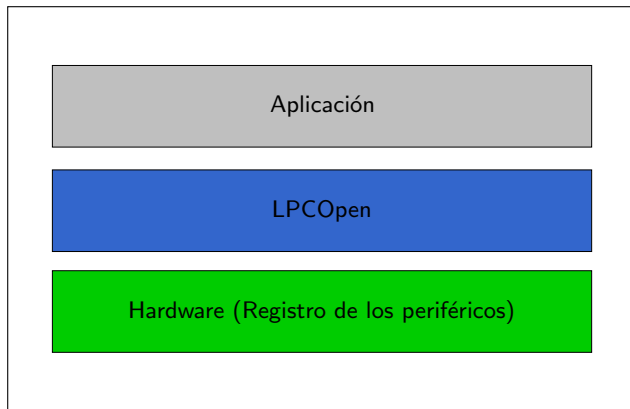
## Hardware Abstraction Layer

Es la parte de software que se relaciona directamente con el hardware. Su función es proveer una interfaz entre los recursos del hardware y la aplicación o el sistema operativo

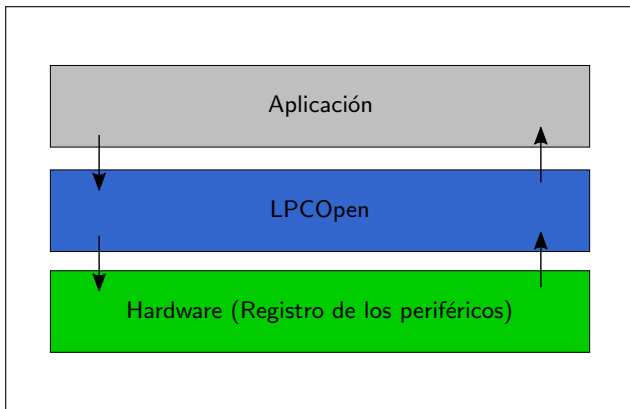
## LPCOpen

Librería de NXP la cual provee una API común para el acceso al hardware del microcontrolador.

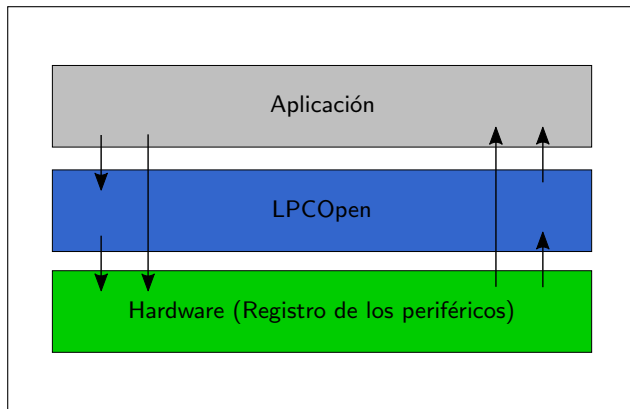
# Capas de Abstracción



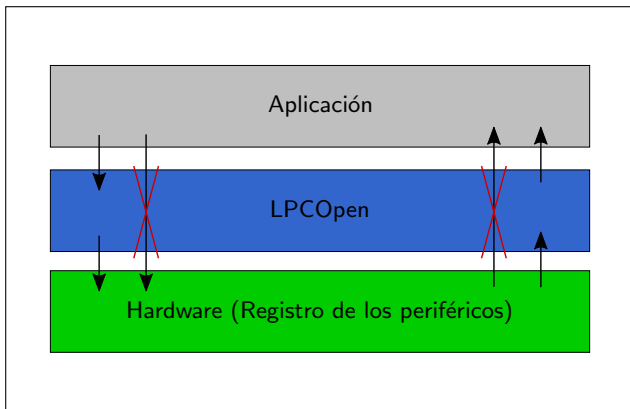
# Capas de Abstracción



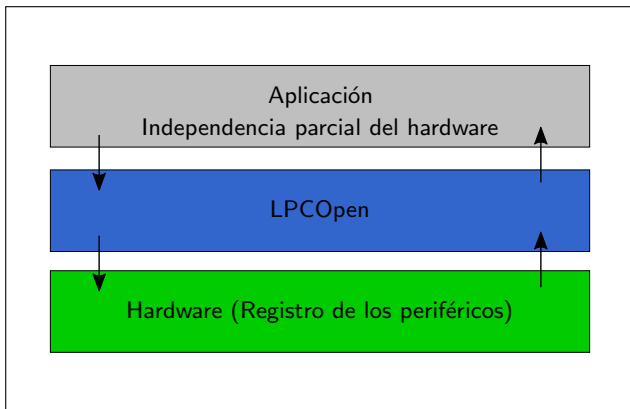
# Capas de Abstracción



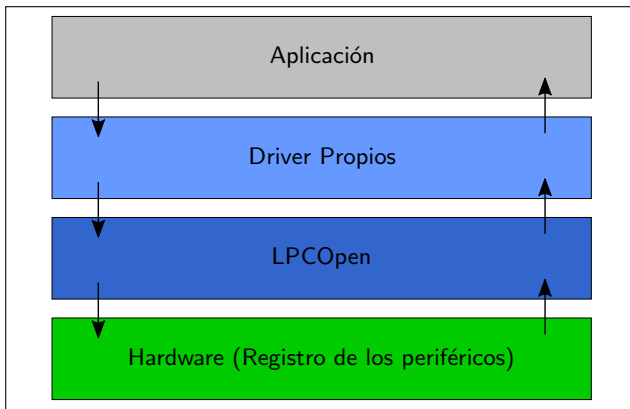
# Capas de Abstracción



# Capas de Abstracción

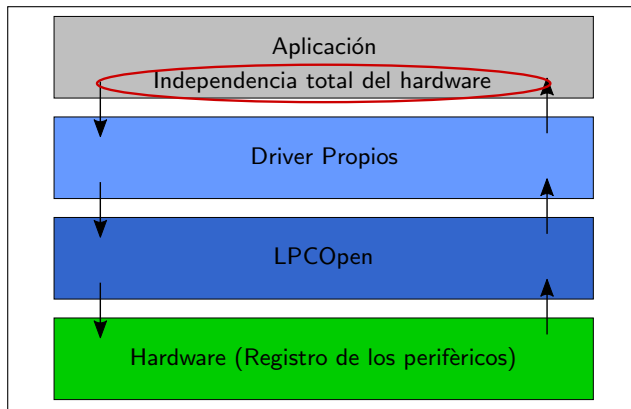


# Capas de Abstracción





# Capas de Abstracción



# Puertos de Entrada Salida (GPIO) con LPCOpen

- Para el uso de las librerías LPCOpen, se deberá incluir en el .c el archivo chip.h
- La implementación de las librerías para la GPIO se encuentran en gpio\_18xx\_43xx.c

# Pasos para utilizar la GPIO

## primer paso

Definir uso del pin configurando la System Control Unit (SCU)

```
Chip_SCU_PinMux(2,0,MD_PUP,FUNC4); /* mapea P2_0 en GPIO5[0] y habilita el pull-up*/
```

## segundo paso

Inicializar la GPIO

```
Chip_GPIO_Init(LPC_GPIO_T* pGPIO);
```

## tercer paso

Definimos Dirección

```
Chip_GPIO_SetDir(LPC_GPIO_T* pGPIO, uint8_t portNum,uint32_t portValue,uint8_t out);
```

# Funciones disponibles para el manejo de GPIO

- `Chip_GPIO_ClearValue();`
- `Chip_GPIO_SetValue();`
- `Chip_GPIO_SetPinOutLow();`
- `Chip_GPIO_SetPortOutHigh();`
- `Chip_GPIO_SetPinToggle();`
- `Chip_GPIO_SetPortToggle();`

# Uso de la EDU-CIAA

# ¿Qué es el CIAA Firmware?

El firmware es el programa que se ejecuta en la CPU del microcontrolador.

- RTOS basado en el estándar OSEK
- HAL basado en el estándar POSIX
- Protocolos de comunicación MODBUS
- Tests Unitarios para cada módulo.
- Biblioteca de Terceros

# ¿Cómo se graba la CIAA?

El proceso de programación de la memoria de la CIAA y depuración del software se lleva a cabo a través de un debug server, cuyos componentes son:

- JTAG debugger Opendous
- OpenOCD
- arm-none-eabi-gdb

# Estructura Firmware

- doc, Documentacion, la estructura no está definida
- examples, Ejemplos
- externals, Herramientas y Fuentes externos.
- modules, Modulos de CIAA Firware.
- out Archivos generados.
- Makefile archivo makefile prncipal
- Makefile.config Demo de configuración personal de Makefile
- Makefile.mine Configuración personal del Makefile.
- Readme