

Maestría en Sistemas Embebidos
Universidad de Buenos Aires

Sistema de control para estación autónoma marítima de monitoreo de ruido ambiente

Esp. Ing. Patricio Bos

Director:

Dr. Ing Ariel Lutenberg

Jurados:

Dr. Ing. Pablo Gómez
Ing. Juan Manuel Cruz
Mg. Lic Igor Prario



Agenda

- Motivación
- Planificación
- Metodología
- Implementación
- Testing
- Demo
- Conclusiones

Agenda



Motivación

Planificación

Metodología

Implementación

Testing

Demo

Conclusiones

Motivación



- ▶ ¿Por qué acústica submarina?

Motivación



- ▶ ¿Por qué acústica submarina?
- ▶ ¿Qué es el nivel de ruido?



Motivación

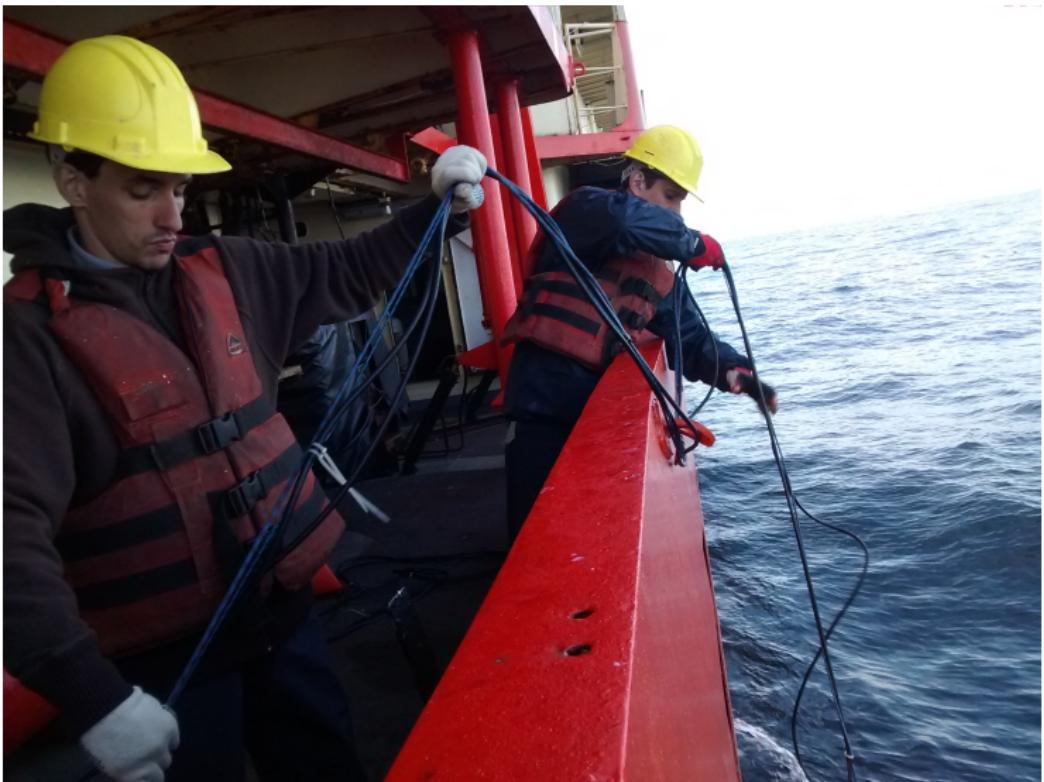
- ▶ ¿Por qué acústica submarina?
- ▶ ¿Qué es el nivel de ruido?
- ▶ ¿Por qué interesa medirlo?

Motivación



- ▶ ¿Por qué acústica submarina?
- ▶ ¿Qué es el nivel de ruido?
- ▶ ¿Por qué interesa medirlo?
- ▶ ¿Qué disciplinas lo necesitan?

Antecedentes



Objetivo

General



- ▶ Prototipo de estación autónoma.

Objetivo

General



5

- ▶ Prototipo de estación autónoma.
- ▶ Medición de señales acústicas.

Objetivo

General



- ▶ Prototipo de estación autónoma.
- ▶ Medición de señales acústicas.
- ▶ Medición de parámetros ambientales.

Objetivo

General



- ▶ Prototipo de estación autónoma.
- ▶ Medición de señales acústicas.
- ▶ Medición de parámetros ambientales.
- ▶ Almacenamiento de la información.

Objetivo

General



- ▶ Prototipo de estación autónoma.
- ▶ Medición de señales acústicas.
- ▶ Medición de parámetros ambientales.
- ▶ Almacenamiento de la información.
- ▶ Transmisión en tiempo real.

Objetivo

Particular



- ▶ Desarrollar un firmware de control para la CIAA-NXP.

Objetivo

Particular



6

- ▶ Desarrollar un firmware de control para la CIAA-NXP.
- ▶ Arquitectura multicore modular y flexible.

Objetivo

Particular



6

- ▶ Desarrollar un firmware de control para la CIAA-NXP.
- ▶ Arquitectura multicore modular y flexible.
- ▶ Mecanismos de comunicación, sincronización y concurrencia.

Objetivo

Particular



6

- ▶ Desarrollar un firmware de control para la CIAA-NXP.
- ▶ Arquitectura multicore modular y flexible.
- ▶ Mecanismos de comunicación, sincronización y concurrencia.
- ▶ Interfaz de usuario



Agenda

Motivación

Planificación

Metodología

Implementación

Testing

Demo

Conclusiones



Diagrama en bloques

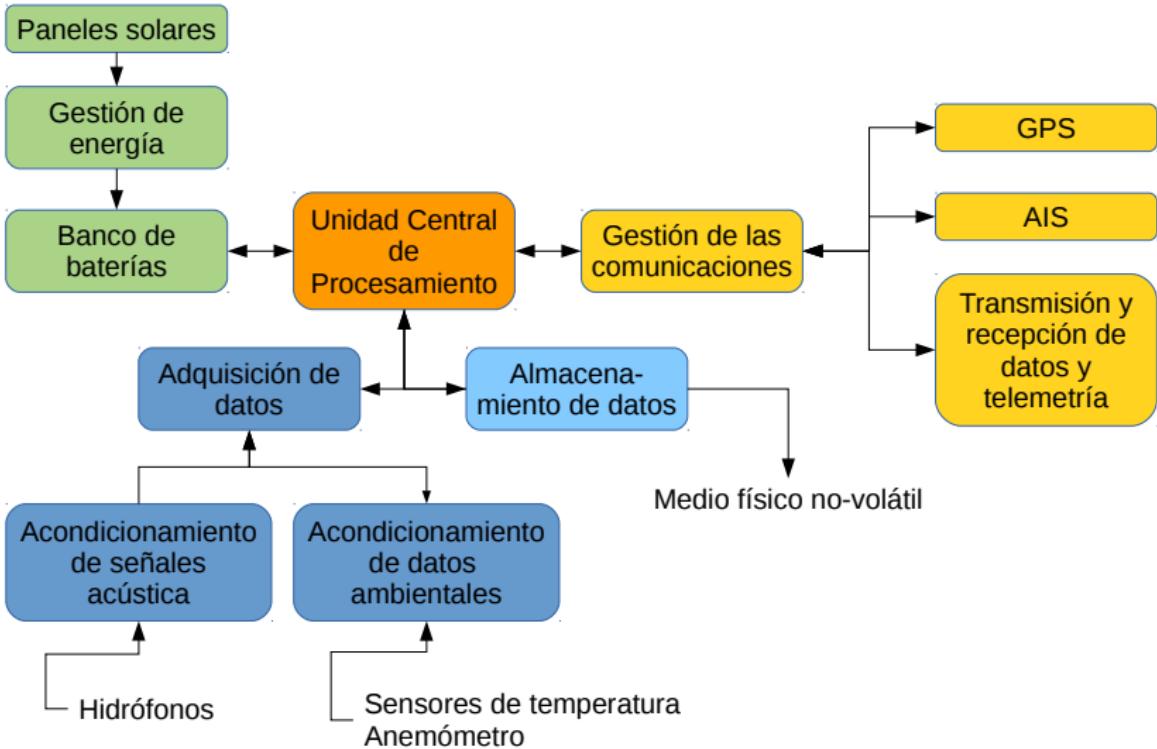
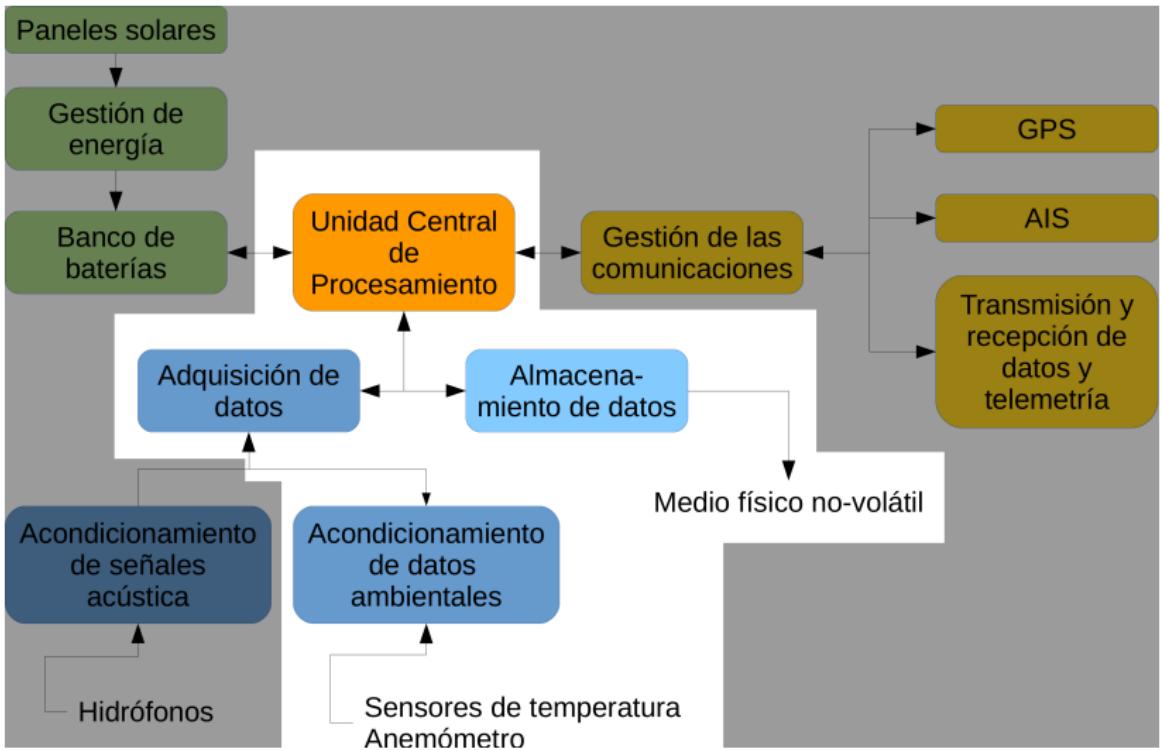


Diagrama en bloques

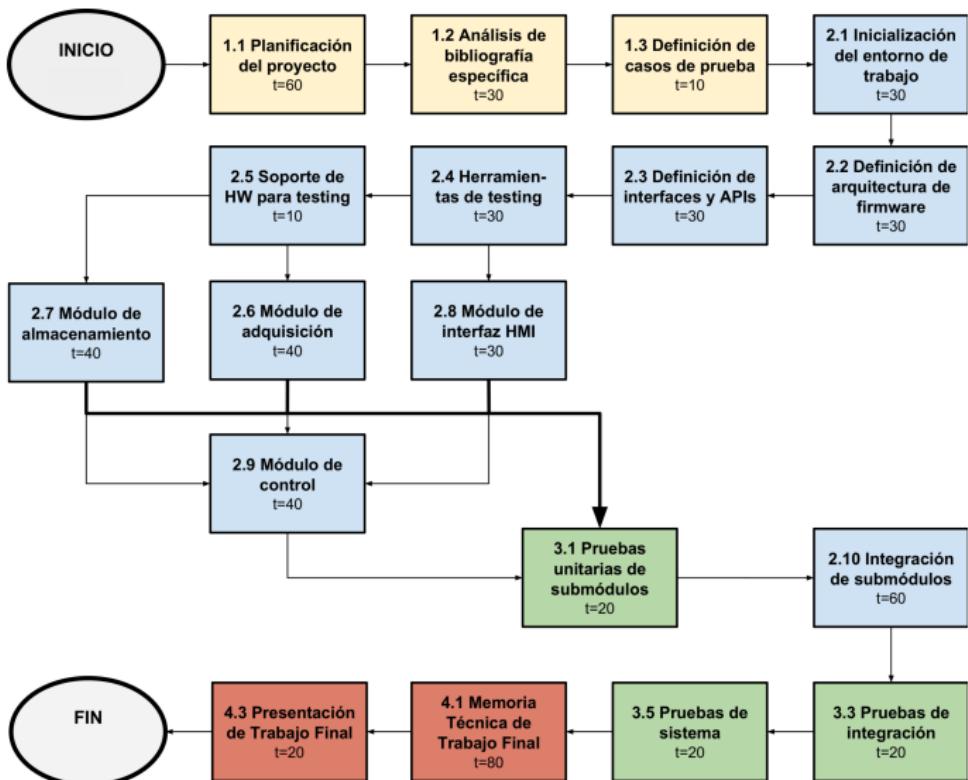


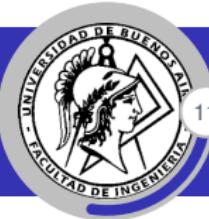
Planificación en etapas



Etapa	Horas	Hitos
Documentación y análisis preliminar	100	Plan de trabajo
		Presentación de plan de trabajo
Diseño e implementación	340	Documentación de submódulos
Verificación y validación	60	Reporte de pruebas unitarias
		Reporte de pruebas de integración
		Reporte de resultados de casos de prueba
Proceso de cierre	100	Memoria Técnica
		Presentación de Trabajo Final

Desglose de tareas en AoN





Agenda

Motivación

Planificación

Metodología

Implementación

Testing

Demo

Conclusiones

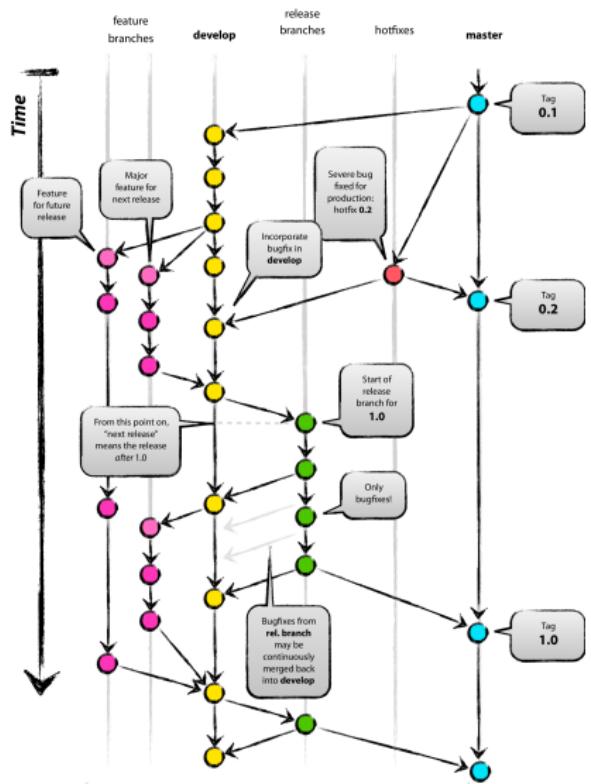
Modelo de ramas

A successfull git branching model



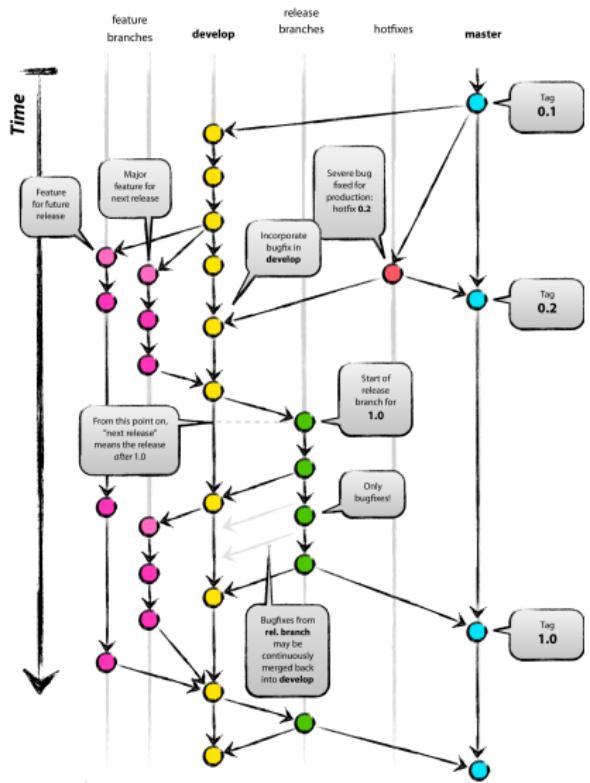
Modelo de ramas

A successfull git branching model



Modelo de ramas

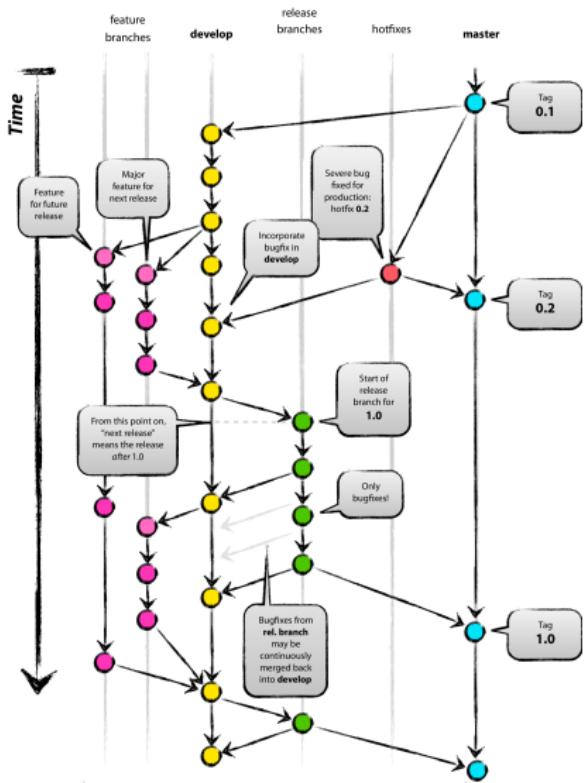
A successfull git branching model



Ramas creadas:

Modelo de ramas

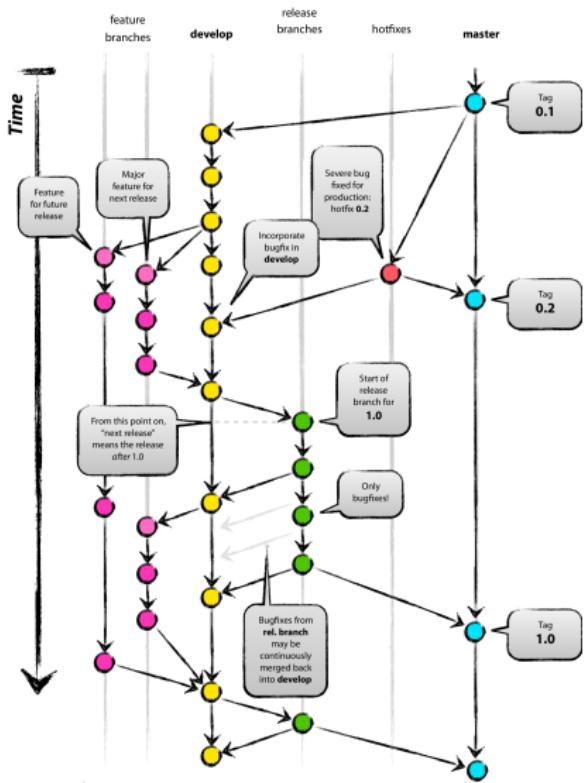
A successfull git branching model



Ramas creadas:
▶ Master

Modelo de ramas

A successfull git branching model

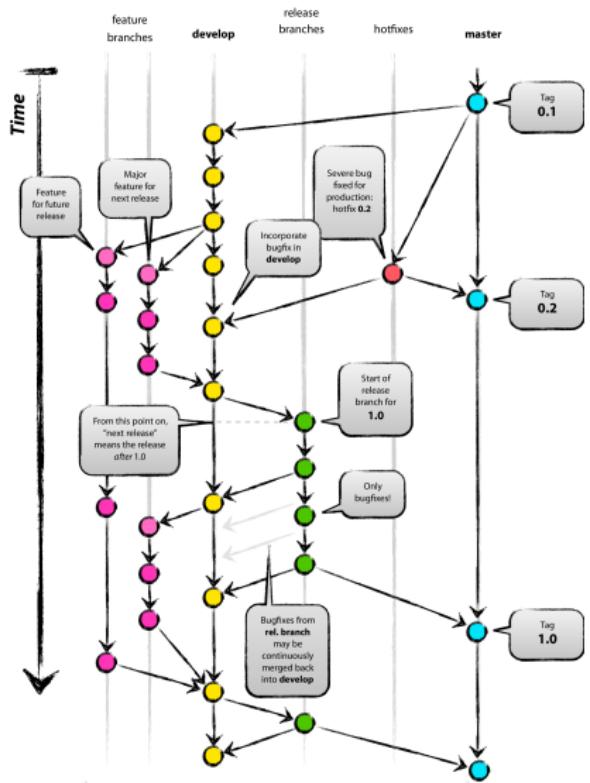


Ramas creadas:

- ▶ Master
- ▶ Develop

Modelo de ramas

A successfull git branching model

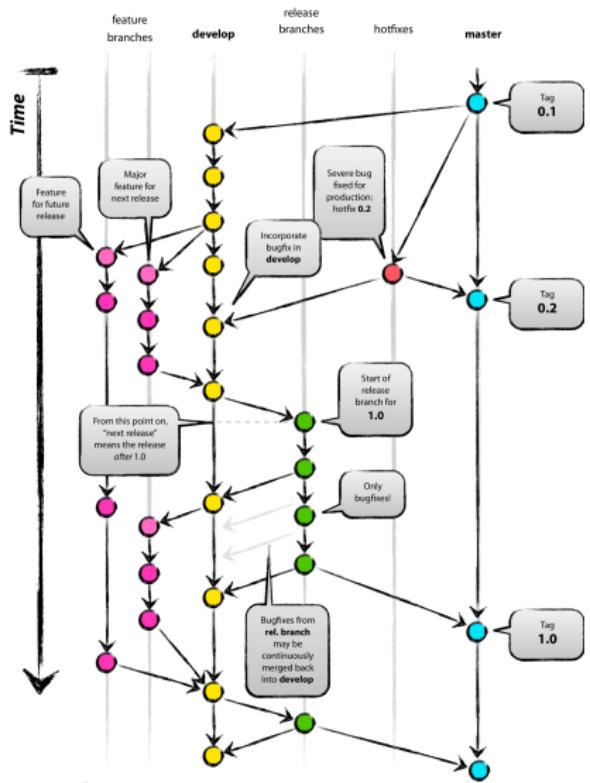


Ramas creadas:

- ▶ Master
- ▶ Develop
- ▶ Adquisición

Modelo de ramas

A successfull git branching model

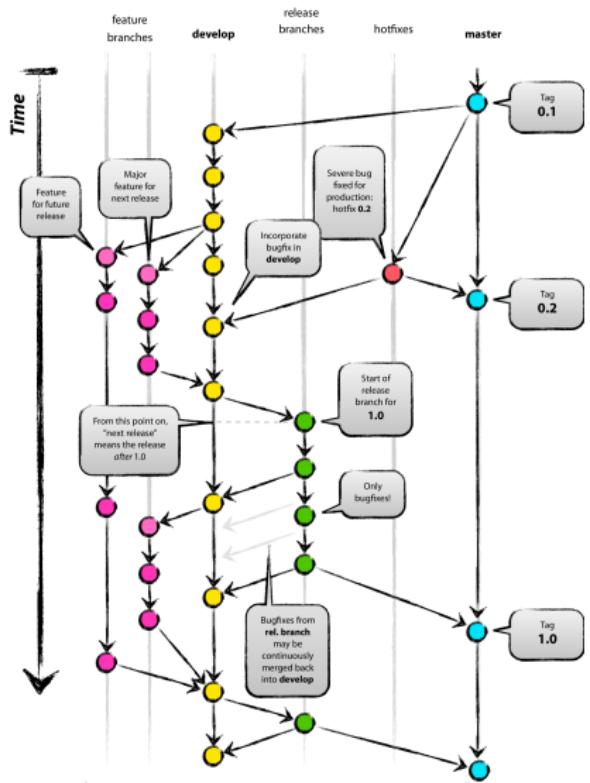


Ramas creadas:

- ▶ Master
- ▶ Develop
- ▶ Adquisición
- ▶ Almacenamiento

Modelo de ramas

A successfull git branching model

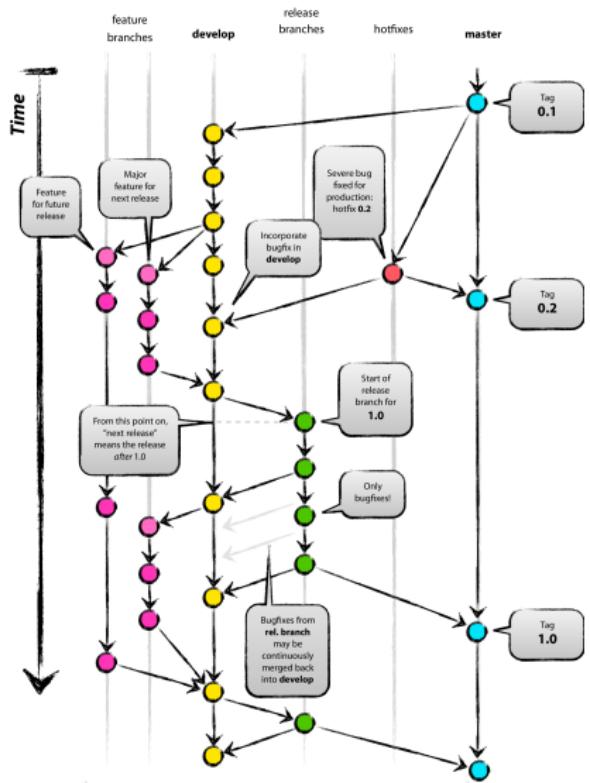


Ramas creadas:

- ▶ Master
- ▶ Develop
- ▶ Adquisición
- ▶ Almacenamiento
- ▶ Interfaz de usuario

Modelo de ramas

A successfull git branching model

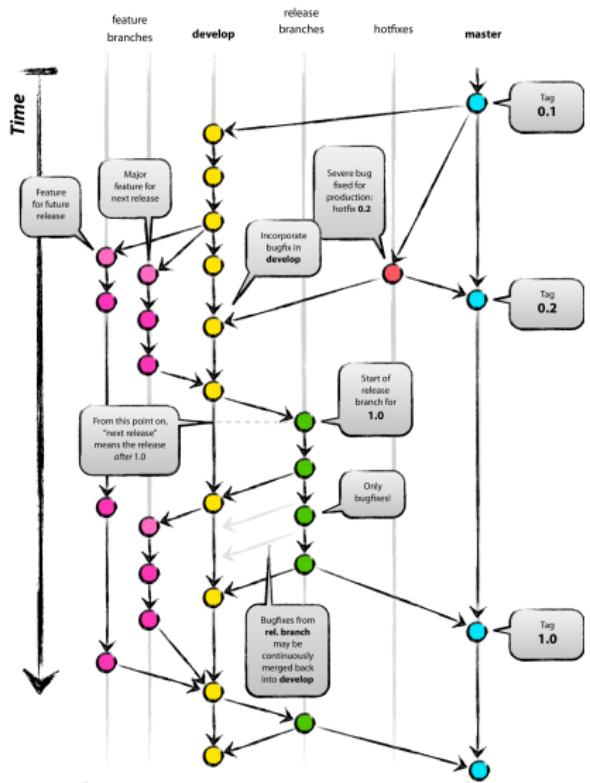


Ramas creadas:

- ▶ Master
- ▶ Develop
- ▶ Adquisición
- ▶ Almacenamiento
- ▶ Interfaz de usuario
- ▶ Control

Modelo de ramas

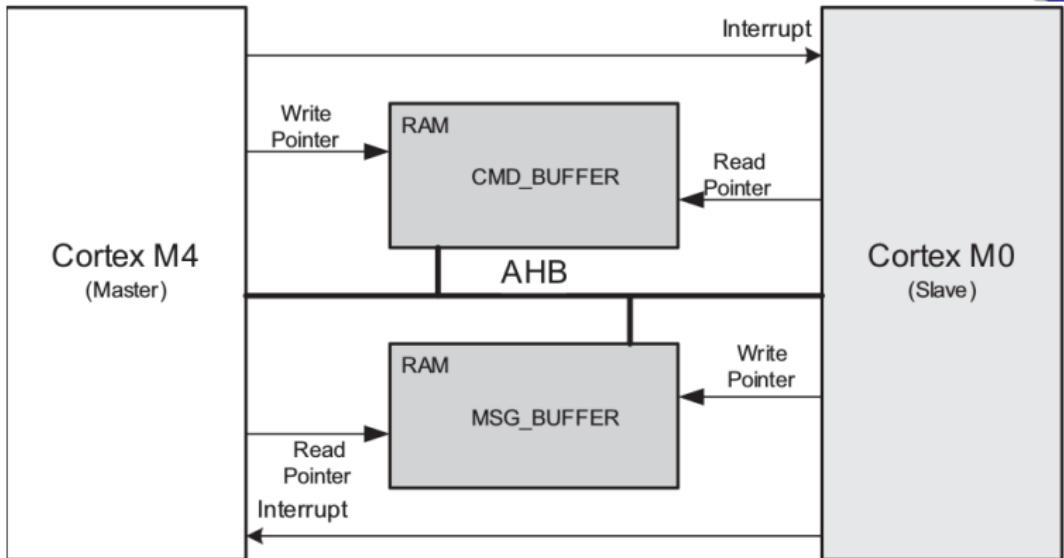
A successfull git branching model



Ramas creadas:

- ▶ Master
- ▶ Develop
- ▶ Adquisición
- ▶ Almacenamiento
- ▶ Interfaz de usuario
- ▶ Control
- ▶ Testing

Inter Process Communications



= M0 subsystem



= M4 subsystem



= shared

Protothreads

Adam Dunkel



14

- ▶ Multitasking cooperativo.

Protothreads

Adam Dunkel



14

- ▶ Multitasking cooperativo.
- ▶ Conjunto de macros o co-rutinas.

Protothreads

Adam Dunkel



14

- ▶ Multitasking cooperativo.
- ▶ Conjunto de macros o co-rutinas.
- ▶ Bloqueo sin cambio de contexto.

Protothreads

Adam Dunkel



14

- ▶ Multitasking cooperativo.
- ▶ Conjunto de macros o co-rutinas.
- ▶ Bloqueo sin cambio de contexto.
- ▶ Control de flujo más lineal

Protothreads

Adam Dunkel



14

- ▶ Multitasking cooperativo.
- ▶ Conjunto de macros o co-rutinas.
- ▶ Bloqueo sin cambio de contexto.
- ▶ Control de flujo más lineal
- ▶ 2 bytes de overhead por thread

Agenda



Motivación

Planificación

Metodología

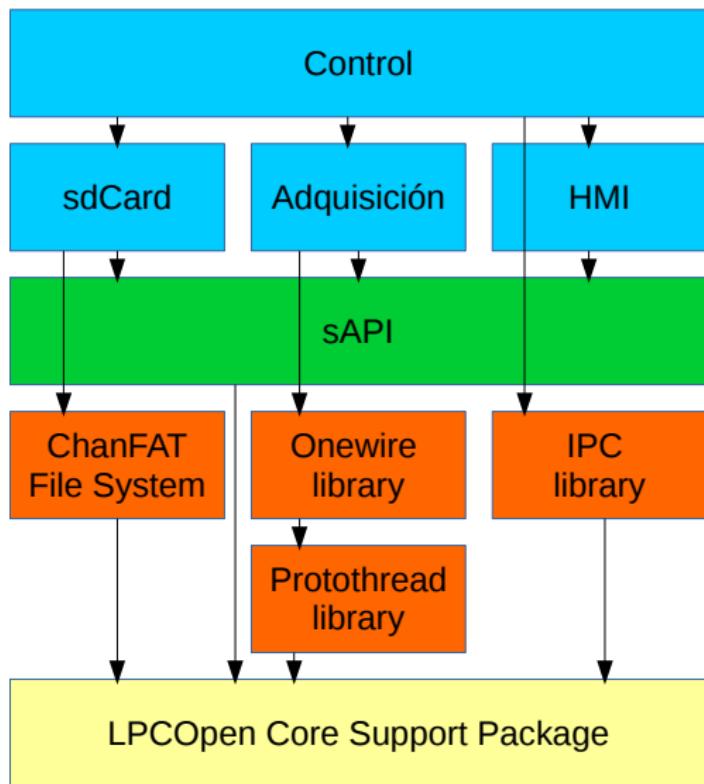
Implementación

Testing

Demo

Conclusiones

Modelo de capas



Arquitectura de módulos



```
typedef struct {  
  
    cpuid_t cpuid;  
  
    moduleID_t moduleID;  
  
    funcPtr_t eventHandler;  
  
    tick_t period;  
  
    moduleStatus_t status;  
} module_t;
```

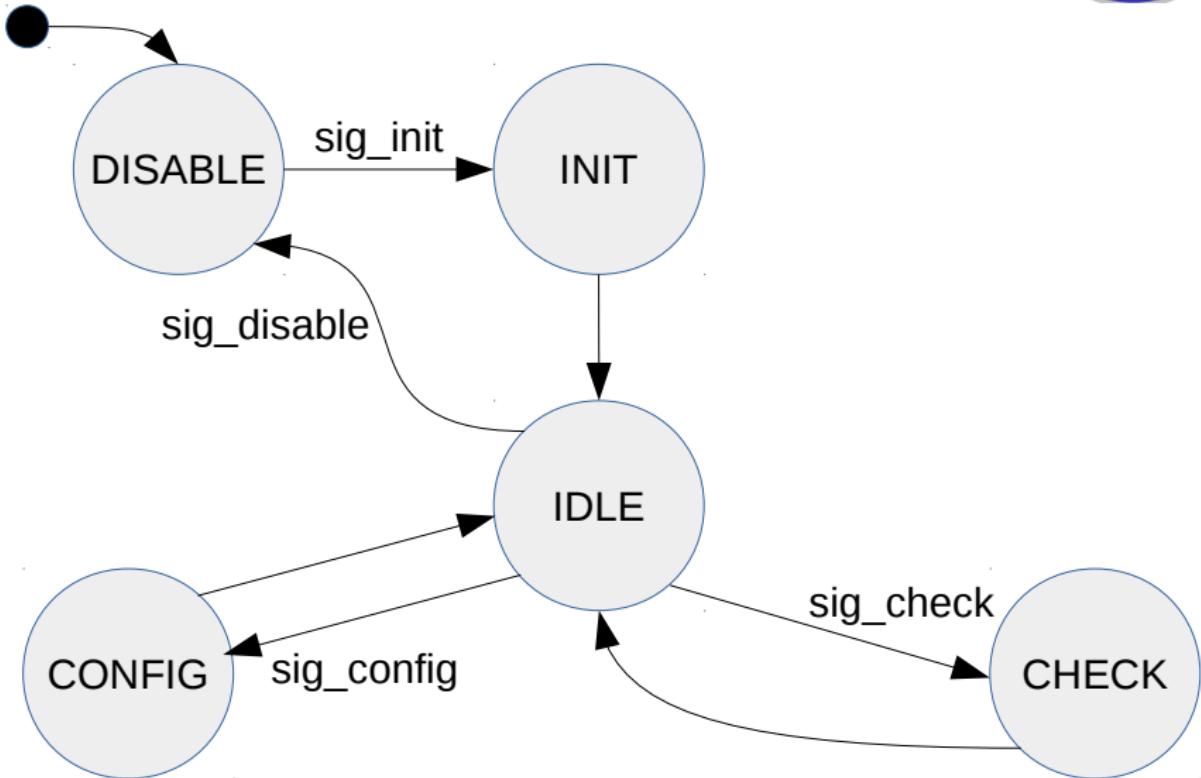
```
typedef struct {  
  
    struct {  
        cpuid_t cpuid;  
        moduleID_t moduleID;  
    } id;  
  
    signal_t signal;  
    (void *) data;  
} ipcex_msg_t;
```

Máquina de Estados Finitos

Módulo genérico



18



Arquitectura de Firmware

Disparada por eventos



```
int main(void)
{
    bool_t goToSleep = FALSE;

    prvSetupHardware();
    FR_register_all_modules();
    FR_broadcast_signal(sig_init);

    while(TRUE) { // the main loop

        goToSleep = FR_dispatch_tasks();

        if (goToSleep == TRUE)
            __WFI();
    }
    return 0;
}
```

Interfaz de usuario

Pantalla principal



20

The diagram illustrates the main screen of the application. At the top, there is a decorative header with the text 'EAMMIRA' and 'r1.0'. Below this is a menu bar labeled 'Menu principal:' containing options '(C) -> Modo Configuracion.' and '(D) -> Modo Debug.'. To the right of the menu bar is a box labeled 'Encabezado'. The main area contains a 'Menú de opciones' box. At the bottom left, there is a 'Barra de estado' box showing status information like '<cmdline>:' and '<status>:' followed by a small green square icon. To the right of the status bar is a 'Línea de comandos' box.

```
EAMMIRA r1.0
=====
Menu principal:
(C) -> Modo Configuracion.
(D) -> Modo Debug.

<cmdline>: ■
<status>:

Barra de estado
Línea de comandos
Menú de opciones
Encabezado
```

Interfaz de usuario

Modo configuración



21

```
MAIN MENU - SYSTEM
=====
Modulo 1-WIRE:
=====
(D) -> Habilitar / Deshabilitar.
(I) -> Info del modulo.
(C) -> Configurar.
(A) -> Autocomprobacion.

(X) -> Para volver al menu anterior.

<cmdline>:D
<status>:[1-WIRE] status: READY
Esta seguro que desea cambiar el estado del modulo (s/n)■
```

Agenda



22

- Motivación
- Planificación
- Metodología
- Implementación
- Testing**
- Demo
- Conclusiones

Testing

Niveles de abstracción



23

- ▶ Pruebas unitarias.

Testing

Niveles de abstracción



23

- ▶ Pruebas unitarias.
- ▶ Pruebas funcionales.

Testing

Niveles de abstracción



23

- ▶ Pruebas unitarias.
- ▶ Pruebas funcionales.
- ▶ Pruebas de sistema.

Testing

Pruebas unitarias



Test ID	Tipo	Nombre del Test		Descripción	
ALM_001	Positivo	Inicializar el FileSystem		Verificar la correcta inicialización del sistema de archivos	
ALM_002	Test ID	Tipo	Nombre del Test		Descripción
ALM_003	CON_001	Positivo	Obtener el estado de un módulo		Obtener un valor válido de estado cuando pide el estado de un módulo válido
ALM_004	CON_002	Test ID	Tipo	Nombre del Test	
ALM_005	CON_003	HMI_001	Positivo	Obtener la posición del cursor en la pantalla	Obtener la posición X,Y de dónde se encuentra el cursor en la pantalla de la terminal serie
ALM_006	CON_004	HMI_002	Positivo	Nombre del Test	
ALM_007	CON_005	HMI_003	Positivo	ADQ_001	Identificar cada ds18b20 conectado al bus 1-wire
ALM_008	CON_006	HMI_004	Positivo	ADQ_002	Detectar presencia de cada ds18b20 identificado
ALM_009	CON_007	HMI_005	Positivo	ADQ_003	Adquirir valores de temperatura
ALM_010	CON_008	HMI_006	Positivo	ADQ_004	Adquirir periódicamente temperatura
ALM_011	CON_009	HMI_007	Positivo	ADQ_005	Leer configuración de ds18b20
ALM_012	CON_010	HMI_008	Positivo	ADQ_006	Escribir configuración de ds18b20
ALM_013	CON_011	HMI_009	Positivo	ADQ_007	Convertir temperatura a celcius
ALM_014	CON_012	HMI_010	Positivo	ADQ_008	Leer el estado del controlador de bus 1-wire
ALM_015	CON_013	HMI_011	Negativo	ADQ_009	Desactivar el controlador del Bus 1-wire
ALM_016	CON_014	HMI_012	Negativo	ADQ_010	Activar el controlador del Bus 1-wire
ALM_017	CON_015	HMI_013	Negativo	ADQ_011	No identificar los ds18b20 conectados al bus 1-wire
ALM_018	CON_016	HMI_014	Negativo	ADQ_012	No detectar presencia de ds18b20 identificados
ALM_019	CON_017	HMI_015	Negativo	ADQ_013	No adquirir valores de temperatura
ALM_020	CON_018	HMI_016	Negativo	ADQ_014	No adquirir periódicamente temperatura
	CON_019	HMI_017	Negativo	ADQ_015	No leer configuración de ds18b20

Testing

Pruebas de sistema: plantilla



25

PROYECTO: EAMMRA

ID CASO DE PRUEBA:

Test diseñado por:

NOMBRE DEL CASO:

Fecha de diseño:

Versión de firmware:

Test ejecutado por:

Fecha de ejecución:

Pre condiciones:

Post condiciones:

ÉXITO:

FALLA:

Resumen del Test:

Step #	Descripción	Resultado Esperado	Resultado Obtenido
1			
2			
3			

Testing

Pruebas de sistema: casos de uso



Testing

Pruebas de sistema: casos de uso



PROYECTO: EAMMRA

ID CASO DE PRUEBA:	UC01	Test diseñado por:	Patricio Bos
NOMBRE DEL CASO:	Medición de temperatura autónoma	Fecha de diseño:	01/10/18
Versión de firmware:	1.0	Test ejecutado por:	Patricio Bos

PROYECTO: EAMMRA

ID CASO DE PRUEBA:	UC02	Test diseñado por:	Patricio Bos
NOMBRE DEL CASO:	Cambio de periodo de adquisición de temp.	Fecha de diseño:	01/10/18
Versión de firmware:	1.0	Test ejecutado por:	Patricio Bos

PROYECTO: EAMMRA

Pre condiciones:	ID CASO DE PRUEBA: UC03	Test diseñado por:	Patricio Bos
	NOMBRE DEL CASO: Cambio de perfil de consumo eléctrico	Fecha de diseño:	03/10/18
	Versión de firmware: 1.0	Test ejecutado por:	Patricio Bos

Resumen del Test:

Step #	Descripción	Resumen del Test:
1	Energizar el sistema.	Post condiciones: EXITO: Nueva configuración de consumo aplicada. FALLA: Nueva configuración de consumo no aplicada.
2	(sistema) En la Lluvia	Step # Descripción Resumen del Test: Test de cambio de perfil de consumo. Después de energizar el sistema, se inicializan los módulos y la estación comienza a funcionar con el perfil de consumo optimizado para performance. Después de 1 minuto, el usuario ingresa a la interfaz en modo configuración y cambia el perfil de consumo en las opciones del módulo de control. El sistema debe enviar un mensaje de confirmación a través de la interfaz.
3	(sistema) Quiero cambiar el consumo	1 Energizar el sistema Resumen del Test: Test de cambio de perfil de consumo. Después de energizar el sistema, se inicializan los módulos y la estación comienza a funcionar con el perfil de consumo optimizado para performance. Después de 1 minuto, el usuario ingresa a la interfaz en modo configuración y cambia el perfil de consumo en las opciones del módulo de control. El sistema debe enviar un mensaje de confirmación a través de la interfaz.
4	(sistema) sig. config	2 (sistema) Esperar que se cumpla 1 minuto desde el paso #1
5	(sistema) sig. config al tiempo	3 (sistema) En la Lluvia
6	(sistema) El módulo de control recibe la señal sig. config	4 (sistema) En la Lluvia
7	(sistema) El módulo de control recibe la señal sig. config	5 (sistema) En la Lluvia
8	En la Lluvia	6 (sistema) En la Lluvia
9	En la Lluvia	7 (sistema) En la Lluvia

Step # Descripción

Step #	Descripción	Resultado Esperado	Resultado Obtenido
1	Energizar el sistema	Mensaje de startup en la consola	ok
2	(sistema) Inicialización de todos los módulos	Mensajes de inicialización de cada módulo en la consola	ok

Resumen del Test:

1	En la Lluvia	Test de cambio de perfil de consumo. Después de energizar el sistema, se inicializan los módulos y la estación comienza a funcionar con el perfil de consumo optimizado para performance. Después de 1 minuto, el usuario ingresa a la interfaz en modo configuración y cambia el perfil de consumo en las opciones del módulo de control. El sistema debe enviar un mensaje de confirmación a través de la interfaz.	
2	(sistema) Esperar que se cumpla 1 minuto desde el paso #1	-	-
3	(sistema) En la Lluvia	3 Esperar que se cumpla 1 minuto desde el paso #1	-
4	(sistema) En la Lluvia	4 En la consola, ingresar al modo configuración. Luego Ingresar al módulo de control	Visualizar en la consola el menú contextual del configuración del módulo de control
5	(sistema) En la Lluvia	5 En la consola, ingresar a la opción cambiar configuración de perfil de consumo. Luego confirmar el cambio ingresando 's'.	Visualizar perfil de consumo actual y mensaje de confirmación antes de aplicar el cambio
6	(sistema) En la Lluvia	6 (sistema) Se encola un mensaje para el módulo de control con la señal sig_config y el nuevo valor para el perfil de consumo.	-
7	(sistema) En la Lluvia	7 (sistema) El módulo de control recibe la señal sig_config y un nuevo valor de perfil de consumo.	Visualizar en la barra de estado de la consola un mensaje de módulo de control con el nuevo perfil de consumo aplicado.
8	(sistema) En la Lluvia	8 (sistema) El módulo de control encola un mensaje para el módulo HMI con la señal sig_disable.	La consola queda deshabilitada.
9	(sistema) En la Lluvia	9 (sistema) El módulo HMI recibe el mensaje y deshabilita la UART	ok

Testing

Pruebas de sistema: casos de uso



PROYECTO:	EAMMRA
ID CASO DE PRUEBA:	UC01
NOMBRE DEL CASO:	Medición de temperatura autónoma
Versión de firmauario:	1.0
Test diseñado por:	Patricio Bos
Fecha de diseño:	01/10/18
Test ejecutado por:	Patricio Bos

PROYECTO:	EAMMRA
Pre condiciones:	<p>ID CASO DE PRUEBA: UCO2 NOMBRE DEL CASO: Cambio de periodo de adquisición de temp. Versión de firmware: 1.0 Test diseñado por: Patricio Bos Fecha de diseño: 01/10/18 Test ejecutado por: Patricio Bos</p>

	PROYECTO:	EAMMRA	
Post condiciones:	Pre condiciones:		
	El CASO DE PRUEBA: NOMBRE DEL CASO: Versión de firmware:	UIC03 Cambio de perfil de consumo eléctrico 1.0	Test diseñado por: Patricio Bos Fecha de diseño: 03/10/18 Test ejecutado por: Patricio Bos Fecha de ejecución: 01/11/18
Resumen del Test	Post condiciones:		
	Pre condiciones:	2 sensores DS18B20 conectados al puerto I2C-FAT12 conectado al puerto SPI de la CIAA-NXP. 1 Tarjeta módem con firmware FAT12 conectada al puerto SPI de la CIAA-NXP. Terminal serie conectada en configuración BNU 115200 a la USART 1.	

Step #	Desc.	Resumen del Test:		
Step #	Desc.	Post condiciones:	EXITO: Nueva configuración de consumo aplicada. FALLA: Nueva configuración de consumo no aplicada.	
1	Energ			
2	(siste	Step #	Descripción	Test de cambio de perfil de consumo. Después de energizar el sistema, se inicializan los módulos y la estación comienza a funcionar con el perfil de consumo optimizado para performance. Después de 1 minuto, el usuario ingresa a la interfaz en modo configuración y cambia el perfil de consumo en las opciones del módulo de control. El sistema debe enviar un mensaje de comunicación a través de la interfaz.
3	(siste quiero para e	1	Energ	Resumen del Test:
4	(siste sig. te	2	(siste	
5	(siste sig. al tempo	3	Esper	
6	(siste sig. as El mó tempo El mó tarjet de tie	4	En la Luego	1 Energizar el sistema 2 (sistema) Inicialización de todos los módulos
7	Esper	5	En la Luego	3 Esperar que se cumpla 1 minuto desde el paso #1
8	En la	6	(siste ción rido	4 En la consola, ingresar al modo configuración. Luego ingresar al módulo de control
9	En la	7	(siste sig. co Cuan módul	5 En la consola, ingresar a la opción cambiar configuración de perfil de consumo. Luego confirmar el cambio ingresando 's'.
		8	En la	6 (sistema) Se encolla un mensaje para el módulo de control con la señal sig_config y el nuevo valor para el perfil de consumo.
		9	En la	7 (sistema) El módulo de control recibe la señal sig_config y el nuevo valor de perfil de consumo. Se cambia el periodo del sysTickTimer de 1 ms a 10 ms. Se cambia el valor periodo en la estructura de control del módulo de almacenamiento a 10 a. Se envían mensajes confirmando el cambio de configuración a través de la interfaz.
		B		8 (sistema) El módulo de control encolla un mensaje para el módulo HMI con la señal sig_disable. El módulo HMI recibe el mensaje y deshabilita la UART
				La consola queda desabilitada.

Casos de uso:

Testing

Pruebas de sistema: casos de uso



PROYECTO: EAMMRA	
ID CASO DE PRUEBA:	UC01
NOMBRE DEL CASO:	Medición de temperatura autónoma
Versión de firmamape:	1.0
Test diseñado por:	Patricio Bos
Fecha de diseño:	01/10/18
Test ejecutado por:	Patricio Bos

PROYECTO:	EAMMRA
Pre condiciones:	ID CASO DE PRUEBA: UC02 NOMBRE DEL CASO: Cambio de periodo de adquisición de temp. Añadir los siguientes: 1.0

Post condiciones:	Pre condiciones:	PROYECTO: EAMMRA	TESTADO POR: JESÚS GONZÁLEZ GARCÍA
	El CASO DE PRUEBA: NOMBRE DEL CASO: Versión de firmware:	UIC03 Cambio de perfil de consumo eléctrico 1.0	Test diseñado por: Patricio Bos Fecha de diseño: 03/11/2018 Test ejecutado por: Patricio Bos Fecha de ejecución: 01/12/2018
Resumen del Test:	Post condiciones:	2 sensores DS18B20 conectados al puerto FAT1 y FAT2 de la CIAA-NXP. 1 Tarjeta microSD con firmware FAT32 conectada en la SPI del puerto P1 de la CIAA-NXP.	
	Pre condiciones:	Terminal serie conectada en configuración BM1_115200 a la USART_USB.	

Step #	Desc.	Resumen del Test:		
Step #	Desc.	Post condiciones:	EXITO: Nueva configuración de consumo aplicada. FALLA: Nueva configuración de consumo no aplicada.	
1	Energ			
2	(siste	Step #	Descripción	Test de cambio de perfil de consumo. Después de energizar el sistema, se inicializan los módulos y la estación comienza a funcionar con el perfil de consumo optimizado para performance. Después de 1 minuto, el usuario ingresa a la interfaz en modo configuración y cambia el perfil de consumo en las opciones del módulo de control. El sistema debe enviar un mensaje de comunicación a través de la interfaz.
3	(siste quiero para e	1	Energ	Resumen del Test:
4	(siste sig. te	2	(siste	
5	(siste sig. al tempo	3	Esper	
6	(siste sig. as El mó tempo El mó tarjet de tie	4	En la Luego	1 Energizar el sistema 2 (sistema) Inicialización de todos los módulos
7	Esper	5	En la Luego	3 Esperar que se cumpla 1 minuto desde el paso #1
8	En la	6	(siste ción rido	4 En la consola, ingresar al modo configuración. Luego ingresar al módulo de control
9	En la	7	(siste sig. co Cuan módul	5 En la consola, ingresar a la opción cambiar configuración de perfil de consumo. Luego confirmar el cambio ingresando 's'.
		8	En la	6 (sistema) Se encolla un mensaje para el módulo de control con la señal sig_config y el nuevo valor para el perfil de consumo.
		9	En la	(sistema) El módulo de control recibe la señal sig_config y el nuevo valor de perfil de consumo. Se cambia el periodo del sysTickTimer de 1 ms a 10 ms. Se cambia el valor periodo en la estructura de control del módulo de almacenamiento a 10 a. Se envían mensajes confirmando el cambio de configuración a través de la interfaz.
		B		8 (sistema) El módulo de control encolla un mensaje para el módulo HMI con la señal sig_disable. El módulo HMI recibe el mensaje y deshabilita la UART
				La consola queda desabilitada.

Casos de uso:

► UC01: Adquisición autónoma

Testing

Pruebas de sistema: casos de uso



PROYECTO:	EAMMRA
ID CASO DE PRUEBA:	UC01
NOMBRE DEL CASO:	Medición de temperatura autónoma
Versión de firmauario:	1.0

PROYECTO:	EAMMRA
Pre condiciones:	<p>ID CASO DE PRUEBA: UCO2 NOMBRE DEL CASO: Cambio de periodo de adquisición de temp. Versión de firmware: 1.0 Test diseñado por: Patricio Bos Fecha de diseño: 01/10/18 Test ejecutado por: Patricio Bos</p>

		PROYECTO:	EAMMRA
Post condiciones:	Pre condiciones:	UIC03 Cambio de perfil de consumo eléctrico Versión de firmware: 1.0	Test diseñado por: Patricio Bos Fecha de diseño: 03/10/2018 Test ejecutado por: Patricio Bos Fecha de ejecución: 01/11/2018
Resumen del Test	Post condiciones:	2 sensores DS18B20 conectados al puerto 1-WIRE (GPO3)(0) de la CIAA-NXP. 1 Tarjeta microSD con formato FAT32 conectada al puerto SPI de la CIAA-NXP. Terminal serie conectada en configuración BNU 115200 a la USART-USB	
Step #	Desc	Resumen del Test:	
1	Energ	Post condiciones:	EXITO: Nueva configuración de consumo aplicada. FALLA: Nueva configuración de consumo no aplicada.
2	(sistema)	Step #	Desc
		Test de cambio de perfil de consumo. Despues de energizar el sistema, se inicializan los motores y la tarjetón comienza a funcionar con el perfil de consumo optimizado.	

Casos de uso:

- ▶ UC01: Adquisición autónoma
 - ▶ UC02: Cambio de período

			Resumen del Test:
3	(sistema que se quiso para el control)	1 Energizar el sistema.	para performance. Después de 1 minuto, el usuario ingresa a la interfaz en modo configuración y cambia el perfil de consumo en las opciones del módulo de control. El sistema debe enviar un mensaje de confirmación a través de la interfaz.
4	(sistema sig. de tiempo)	2 (sistema que se quiso para el control)	
5	(sistema sig. de tiempo)	3 Esperar	Step # Descripción
			1 Energizar el sistema
6	(sistema sig. de tiempo) El módulo de control tarjeta de tiempo	4 En la consola Luego	2 (sistema) Inicialización de todos los módulos
			3 Esperar que se cumpla 1 minuto desde el paso #1
7	Esperar	5 En la consola Luego	4 En la consola, ingresar al modo configuración. Luego ingresar al módulo de control
8	En la	6 (sistema) Cargar el perfil de consumo	5 En la consola, ingresar a la opción cambiar configuración de perfil de consumo. Luego confirmar el cambio ingresando 's'.
9	En la	7 Cargar el nuevo módulo	6 (sistema) Se encola un mensaje para el módulo de control con la señal sig_config y el nuevo valor para el perfil de consumo.
			7 (sistema) El módulo de control recibe la señal sig_config y un nuevo valor de perfil de consumo. Se actualiza la señal de medida del TickAtTime de 1 ms a 10 ms. Se cambia el valor period en la estructura de control del módulo de almacenamiento 10 a 1. Se envían mensajes confirmando el cambio de configuración a través de la interfaz.
			8 (sistema) El módulo de control encola un mensaje para el módulo HMI con la señal sig_desable. El módulo HMI recibe el mensaje y desabilita la UART
			La consola queda desabilitada.

Testing

Pruebas de sistema: casos de uso



PROYECTO: EAMMRA

ID CASO DE PRUEBA: UC01
 NOMBRE DEL CASO: Medición de temperatura autónoma
 Versión de firmware: 1.0

Test diseñado por: Patricio Bos
 Fecha de diseño: 01/10/18
 Test ejecutado por: Patricio Bos

PROYECTO: EAMMRA

Pre condiciones: ID CASO DE PRUEBA: UC02
 NOMBRE DEL CASO: Cambio de periodo de adquisición de temp.
 Versión de firmware: 1.0

Test diseñado por: Patricio Bos
 Fecha de diseño: 01/10/18
 Test ejecutado por: Patricio Bos

PROYECTO: EAMMRA

Post condiciones: ID CASO DE PRUEBA: UC03
 NOMBRE DEL CASO: Cambio de perfil de consumo eléctrico
 Versión de firmware: 1.0

Test diseñado por: Patricio Bos
 Fecha de diseño: 01/10/18
 Test ejecutado por: Patricio Bos
 Fecha de ejecución: 01/11/18

Resumen del Test:

Step # Desc
 1 Energizar el sistema.

Step # Desc
 2 (sistema) Iniciar el sistema.

Step # Desc
 3 (sistema) Esperar que se cumpla 1 minuto.

Step # Desc
 4 (sistema) En la consola, ingresar al modo configuración.

Step # Desc
 5 (sistema) En la consola, ingresar al módulo de control.

Step # Desc
 6 (sistema) En la consola, ingresar a la opción cambiar configuración de perfil de consumo.

Step # Desc
 7 (sistema) Luego confirmar el cambio ingresando 's'.

Step # Desc
 8 (sistema) Se encola un mensaje para el módulo de control con la señal sig_config y el nuevo valor para el perfil de consumo.

Step # Desc
 9 (sistema) El módulo de control recibe la señal sig_config y un nuevo valor de perfil de consumo.

Se cambia el periodo del systickTimer de 1 ms a 10 ms. Se cambia la estructura de memoria y la estructura de control del módulo de almacenamiento 10 a 1. Se envían mensajes confirmando el cambio de configuración a través de la interfaz.

(sistema) El módulo de control encola un mensaje para el módulo HMI con la señal sig_desabilita. El módulo HMI recibe el mensaje y deshabilita la UART.

Test diseñado por: Patricio Bos
 Fecha de diseño: 01/10/18
 Test ejecutado por: Patricio Bos
 Fecha de ejecución: 01/11/18

Casos de uso:

- ▶ UC01: Adquisición autónoma
- ▶ UC02: Cambio de período
- ▶ UC03: Cambio de perfil

PROYECTO:	EAMMRA
ID CASO DE PRUEBA:	UC01
NOMBRE DEL CASO:	Medición de temperatura autónoma
Versión de firmware:	1.0
Pre condiciones:	
Post condiciones:	
Resumen del Test:	
Step #	Descripción
1	Energizar el sistema.
2	(sistema) Iniciar el sistema.
3	(sistema) Esperar que se cumpla 1 minuto.
4	(sistema) En la consola, ingresar al modo configuración.
5	(sistema) En la consola, ingresar al módulo de control.
6	(sistema) En la consola, ingresar a la opción cambiar configuración de perfil de consumo.
7	(sistema) Luego confirmar el cambio ingresando 's'.
8	(sistema) Se encola un mensaje para el módulo de control con la señal sig_config y el nuevo valor para el perfil de consumo.
9	(sistema) El módulo de control recibe la señal sig_desabilita. El módulo HMI recibe el mensaje y deshabilita la UART.

Testing

Caso de uso UC01



Step #	Descripción	Resultado Esperado	Resultado Obtenido
1	Energizar el sistema	Mensaje de startup en la consola	ok
2	(sistema) Inicialización de todos los módulos	Mensajes de inicialización de cada módulo en la consola	ok
3	(sistema) Expira el timeout para la tarea periódica de adquisición de temperatura. Se encola una señal sig_timeout para el módulo de adquisición	-	-
4	(sistema) El módulo de adquisición recibe la señal sig_timeout y realiza una adquisición de temperatura.	-	-
5	(sistema) El módulo de adquisición envía una señal sig_write al módulo de almacenamiento con el dato de temperatura medido	-	-
6	(sistema) El módulo de almacenamiento recibe la señal sig_write y un valor de temperatura para guardar. El módulo de almacenamiento adquiere una marca de tiempo del RTC del sistema. El módulo de almacenamiento escribe una entrada en la tarjeta microSD con el valor de temperatura y una marca de tiempo	-	-
7	Esperar que se cumpla 1 minuto desde el paso #1	-	-
8	En la consola, ingresar al modo debug	Visualizar en la consola el menú contextual del modo debug	ok
9	En la consola, ingresar a la opción ver archivo de registro	Visualizar entradas con valores de temperatura con saltos regulares en la marca de tiempo	ok

Testing

Caso de uso UC02



Step #	Descripción	Resultado Esperado	Resultado Obtenido
1	Energizar el sistema	Mensaje de startup en la consola	ok
2	(sistema) Inicialización de todos los módulos	Mensajes de inicialización de cada módulo en la consola	ok
3	Esperar a que se cumpla 1 minuto desde el paso #1	-	-
4	En la consola, ingresar al modo configuración. Luego ingresar al módulo de adquisición 1-WIRE	Visualizar en la consola el menú contextual del configuración del módulo de adquisición	ok
5	En la consola, ingresar a la opción cambiar configuración del archivo de registro. Luego ingresar 2000 ms	Visualizar entradas con valores de temperatura con saltos regulares en la marca de tiempo	ok
6	(sistema) Se encola un mensaje para el módulo de adquisición con la señal sig_config y el nuevo valor para el período de adquisición.	-	-
7	(sistema) El módulo de adquisición recibe la señal sig_config y un nuevo valor de período. Cuando expira el nuevo timeout, el módulo de adquisición envía una señal sig_write con un valor de temperatura al módulo de almacenamiento.	-	-
8	En la consola, ingresar al modo debug	Visualizar en la consola el menú contextual del modo debug	ok
9	En la consola, ingresar a la opción ver archivo de registro	Visualizar entradas con valores de temperatura con saltos regulares en la marca de tiempo	ok

Testing

Caso de uso UC03



Step #	Descripción	Resultado Esperado	Resultado Obtenido
1	Energizar el sistema	Mensaje de startup en la consola	ok
2	(sistema) Inicialización de todos los módulos	Mensajes de inicialización de cada módulo en la consola	ok
3	Esperar que se cumpla 1 minuto desde el paso #1	-	-
4	En la consola, ingresar al modo configuración. Luego ingresar al módulo de control	Visualizar en la consola el menú contextual del configuración del módulo de control	ok
5	En la consola, ingresar a la opción cambiar configuración de perfil de consumo. Luego confirmar el cambio ingresando 's'.	Visualizar perfil de consumo actual y mensaje de confirmación antes de aplicar el cambio	ok
6	(sistema) Se encola un mensaje para el módulo de control con la señal sig_config y el nuevo valor para el perfil de consumo.	-	-
7	(sistema) El módulo de control recibe la señal sig_config y un nuevo valor de perfil de consumo. Se cambia el periodo del sysTickTimer de 1 ms a 10 ms. Se cambia el valor period en la estructura de control del módulo de almacenamiento 10 a 1. Se envían mensajes confirmando el cambio de configuración a través de la interfaz	Visualizar en la barra de estado de la consola un mensaje del módulo de control con el nuevo perfil de consumo aplicado.	ok
8	(sistema) El módulo de control encola un mensaje para el módulo HMI con la señal sig_disable. El módulo HMI recibe el mensaje y deshabilita la UART	La consola queda deshabilitada.	ok

Agenda



Motivación
Planificación
Metodología
Implementación
Testing
Demo
Conclusiones



Agenda

Motivación

Planificación

Metodología

Implementación

Testing

Demo

Conclusiones



Resultados obtenidos

- ▶ Metodología de trabajo.



Resultados obtenidos

- ▶ Metodología de trabajo.
- ▶ Arquitectura modular multicore.



Resultados obtenidos

- ▶ Metodología de trabajo.
- ▶ Arquitectura modular multicore.
- ▶ Mecanismos de control y despacho de tareas y eventos.



Resultados obtenidos

- ▶ Metodología de trabajo.
- ▶ Arquitectura modular multicore.
- ▶ Mecanismos de control y despacho de tareas y eventos.
- ▶ 4 módulos funcionales.



Resultados obtenidos

- ▶ Metodología de trabajo.
- ▶ Arquitectura modular multicore.
- ▶ Mecanismos de control y despacho de tareas y eventos.
- ▶ 4 módulos funcionales.
- ▶ Documentación completa.

Cumplimiento

Requerimientos funcionales del sistema



- 2.1 El sistema debe adquirir datos de un array de sensores de temperatura a intervalos regulares con un período de adquisición seleccionable. ✓
- 2.2 El sistema debe adquirir datos de un anemómetro a intervalos regulares con un período de adquisición seleccionable. ✗
- 2.3 El sistema debe almacenar los datos de temperatura y velocidad de viento adquiridas junto con una marca de tiempo identificatoria en un medio físico no volátil. ✓
- 2.4 El sistema debe poder operar con dos perfiles de consumo de energía máximo desempeño y mínimo consumo de energía. ✓
- 2.5 El sistema debe contar con una interfaz serie cableada que permita realizar operaciones de configuración y mantenimiento. ✓

Trabajo futuro



34

- ▶ Completar requerimientos y funcionalidades.



Trabajo futuro

- ▶ Completar requerimientos y funcionalidades.
- ▶ Analizar distintas configuraciones.



Trabajo futuro

- ▶ Completar requerimientos y funcionalidades.
- ▶ Analizar distintas configuraciones.
- ▶ Servidor de integración continua.



Trabajo futuro

- ▶ Completar requerimientos y funcionalidades.
- ▶ Analizar distintas configuraciones.
- ▶ Servidor de integración continua.
- ▶ Diseñar hardware *ad-hoc*.



Trabajo futuro

- ▶ Completar requerimientos y funcionalidades.
- ▶ Analizar distintas configuraciones.
- ▶ Servidor de integración continua.
- ▶ Diseñar hardware *ad-hoc*.
- ▶ Completar objetivos generales.

¿Preguntas?

Protothreads

Multitasking cooperativo



36

```
struct pt { unsigned short lc; };

#define PT_THREAD(name_args)    char name_args

#define PT_BEGIN(pt)           switch(pt->lc) { case 0:

#define PT_WAIT_UNTIL(pt, c)   pt->lc = __LINE__; \
                           case __LINE__: \
                           if(!(c)) return 0

#define PT_END(pt)             } pt->lc = 0; return 2

#define PT_INIT(pt)            pt->lc = 0
```

Protothreads

Multitasking cooperativo



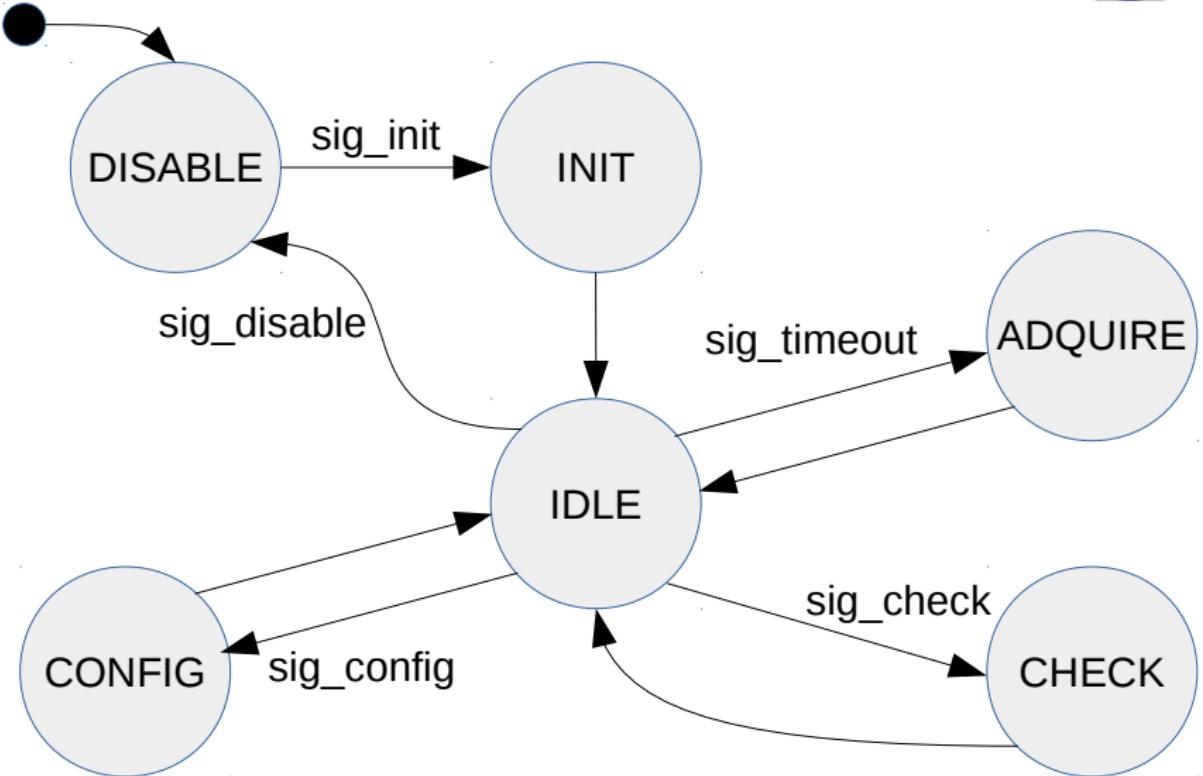
37

```
static  
PT_THREAD(example( struct pt  
    *pt ))  
{  
    PT_BEGIN(pt);  
  
    while(1) {  
        PT_WAIT_UNTIL(pt,  
            counter == 1000);  
        printf("Threshold  
reached\n");  
        counter = 0;  
    }  
    PT_END(pt);  
}
```

```
static  
char example( struct pt *pt)  
{  
  
    switch(pt->lc) { case 0:  
  
        while(1) {  
            pt->lc = 12; case 12:  
            if (!(counter == 1000))  
                return 0;  
            printf("Threshold  
reached\n");  
            counter = 0;  
        }  
        pt->lc = 0; return 2;  
    }  
}
```

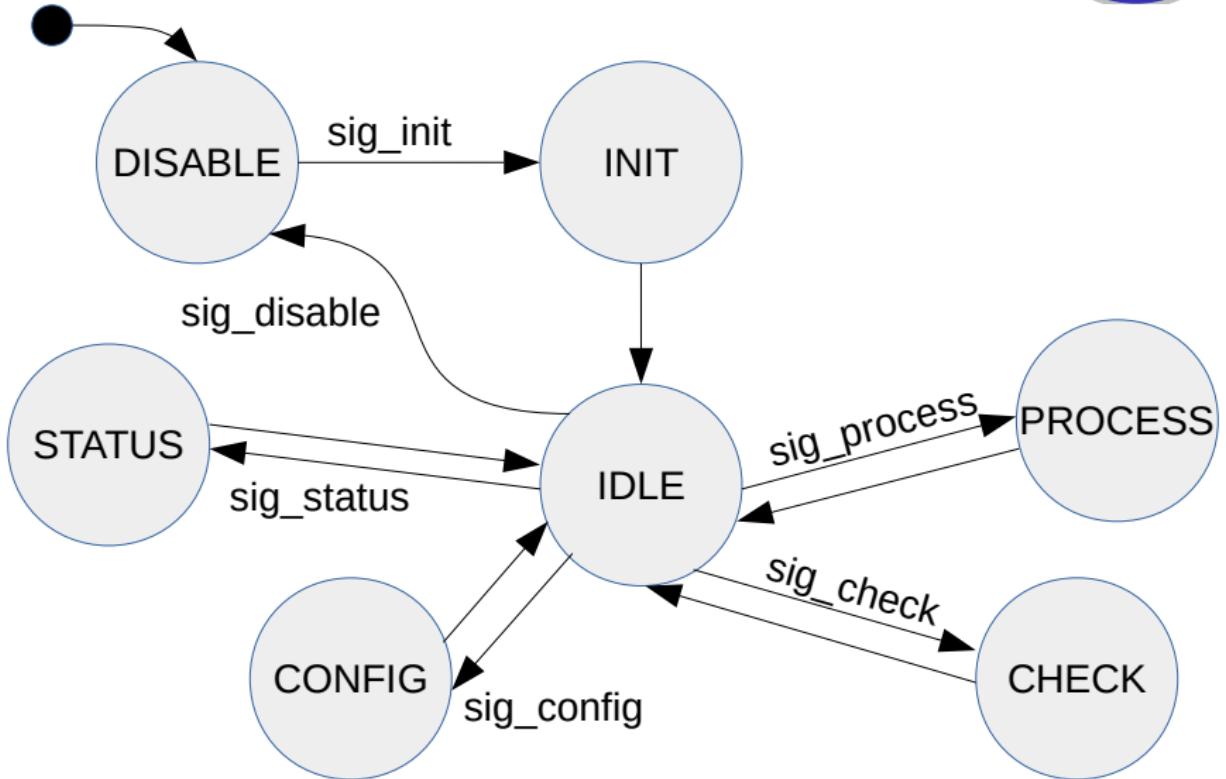
Máquina de Estados Finitos

Módulo adquisición



Máquina de Estados Finitos

Módulo HMI



Máquina de Estados Finitos

Módulo almacenamiento

