



Sensor de temperatura I2C: TC74

IES Juan de la Cierva

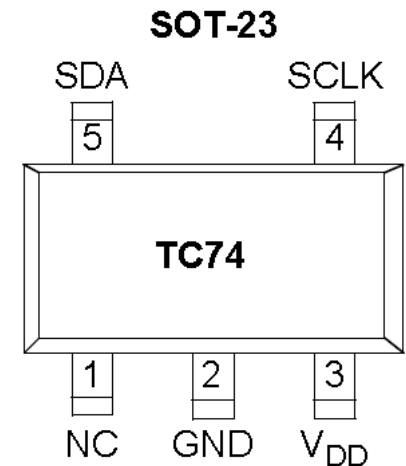
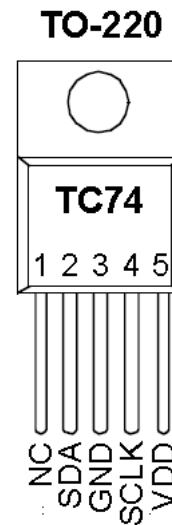


Plataforma Universal Microcontrolada

Sensor de Temperatura TC74



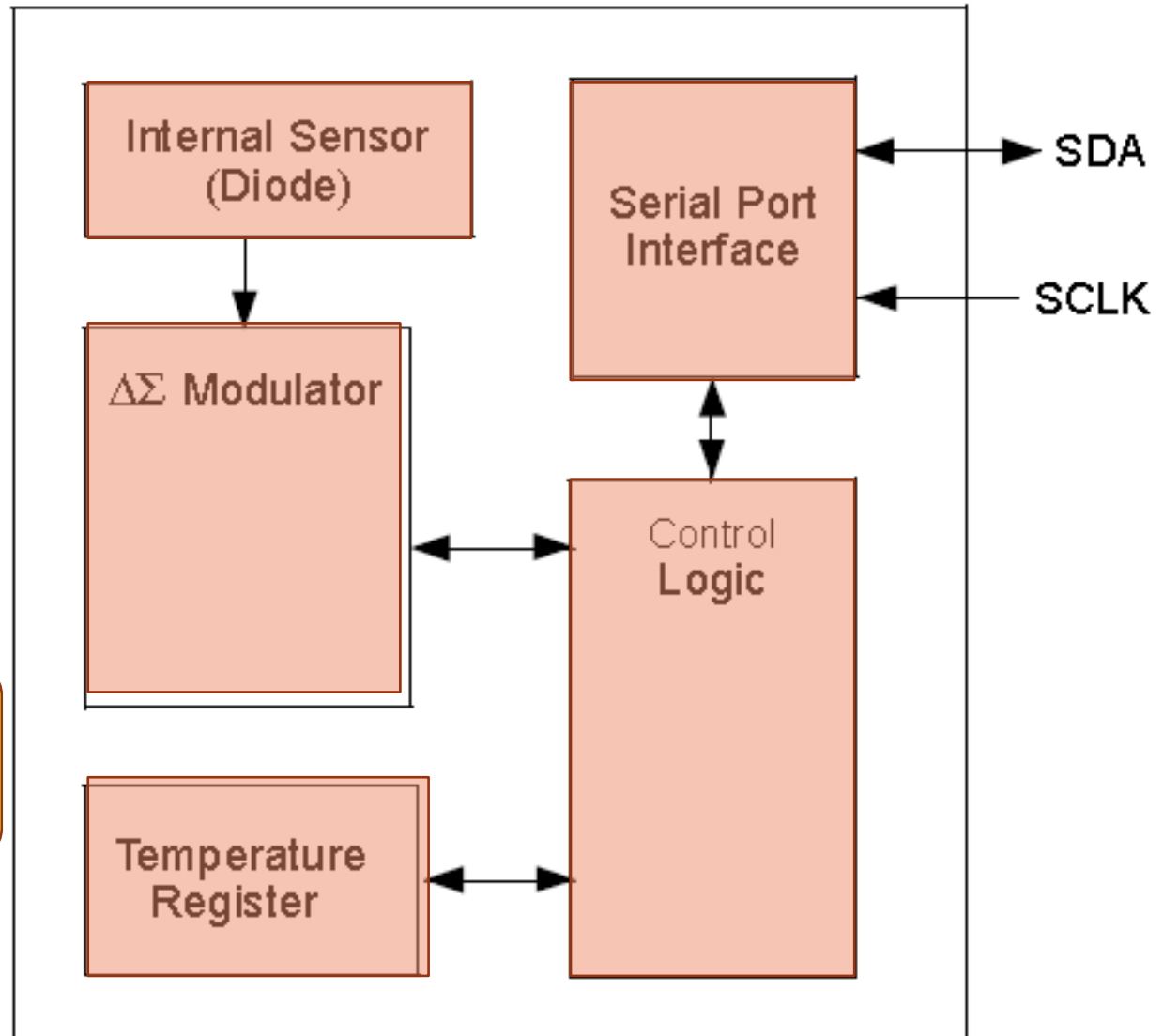
- Combinación de sensor de Temperatura, conversor A/D de 8 bits e interface I2C en un solo C.I.
- Siempre será dispositivo esclavo y estará recibiendo o emitiendo datos cuando se lo solicite el Maestro del bus
- Dirección asignada en I2C como esclavo: **1-0-0-1-1-0-1** (hay otras 7 direcciones distintas que se podrían asignar bajo pedido a Microchip)



Sensor de Temperatura TC74

- **Funcionalidad: ¿Qué puede hacer y para qué puede servir?**
 - Medida de temperatura: de -65°C a 127°C
 - Precisión de $\pm 2^{\circ}\text{C}$ de $+25^{\circ}\text{C}$ a 85°C y $\pm 3^{\circ}\text{C}$ de 0°C a 125°C
 - Resolución de 8 bits: 1°C
 - Proporciona valor en binario y valores negativos en complemento a 2
 - Tensión de alimentación: 2,7V a 5,5V
 - Consumo “en operación” de $200\ \mu\text{A}$ y “en standby” de $5\ \mu\text{A}$
- **Interface: ¿Cómo nos comunicamos con él desde un microcontrolador?**
 - Es esclavo de un bus I2C
 - Tiene una dirección asignada en el bus
 - Tramas de lectura y escritura en I2C
 - Comandos admitidos y formatos conocidos especificados a continuación

TC74: Diagrama de bloques funcional



No necesita de componentes
Externos para funcionar

Funcionamiento: Medida de la temperatura

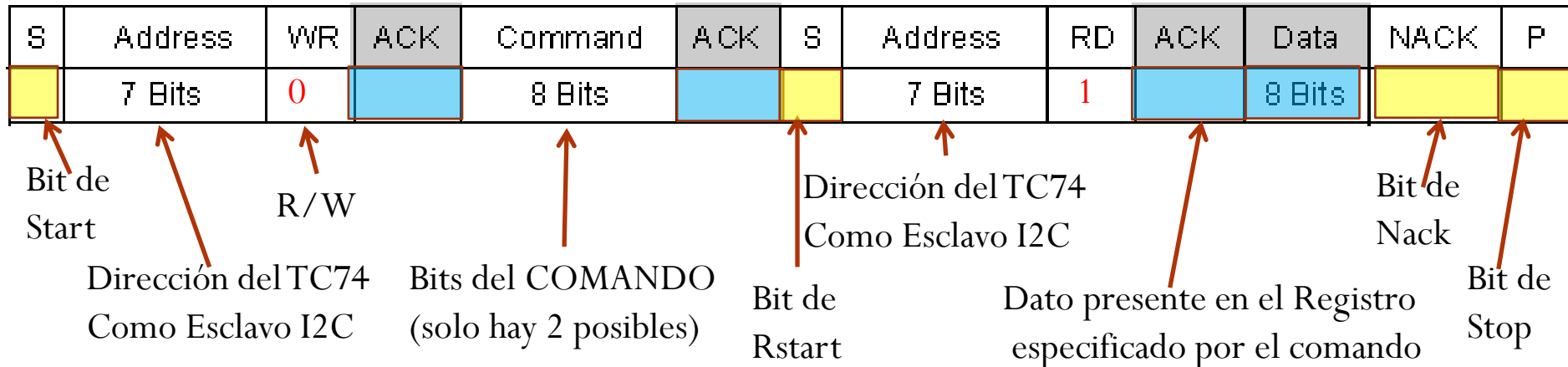
- El integrado realiza la adquisición y conversión de temperaturas mediante un sensor de estado sólido con una resolución de $\pm 1^{\circ}\text{C}$
- En modo de operación normal, se toman y convierten 8 muestras por segundo y el resultado de la conversión se almacena en un registro interno al que se puede acceder en cualquier momento vía serie mediante el bus I2C
- El C.I. se puede situar en modo de bajo consumo (Stand-by) en el que se suspende la adquisición de temperatura y queda retenido en el registro interno el resultado de la última conversión.
- Internamente existe un registro de configuración (**CONFIG**) donde hay un bit (**SHDN**)
- que establece el modo de trabajo: **normal** (SHDN = 0) o de **bajo consumo** (SHDN=1)
- El registro de configuración interno se puede escribir y se puede leer, mientras que el registro de temperatura sólo puede ser leído
- Las lecturas y escrituras siguen las tramas típicas en un bus I2C

Interface I2C del TC74

- Esclavo de un bus I2C con una dirección preestablecida: 1001101, no podrían convivir más de un TC74 en el mismo bus salvo que le encarguemos a Microchip sensores con otras direcciones bajo pedido: 1001xxx posibles direcciones
- Interface a dos hilos: SCL entrada del reloj generado por el Maestro del bus y SDA entrada o salida de datos en función del sentido establecido por el maestro tras un byte de control inicial
- La **máxima frecuencia** de reloj de entrada admitida por el dispositivo es de **100kHz**
- Las tramas de comunicación siempre se inician de manera idéntica:
 - Bit de START
 - Byte de Control con la dirección del TC74 seguido de bit 0 (escritura) ó 1 (lectura)
 - Bit de ACK por parte del TC74
 -lo que sigue ya depende del comando en particular

LECTURA DE UN BYTE EN EL REGISTRO

Read Byte Format

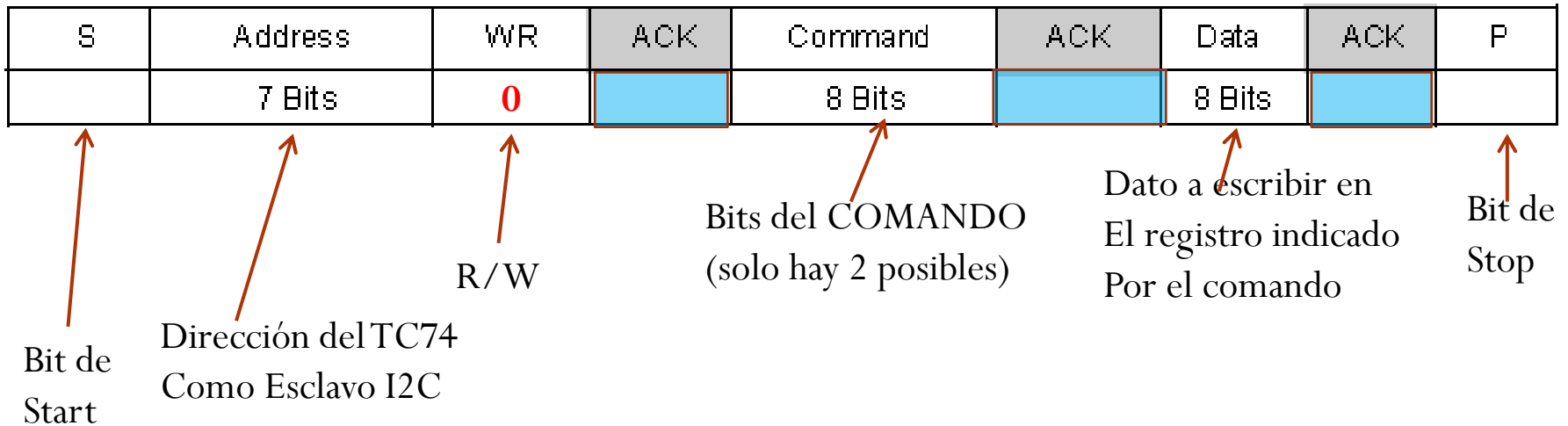


- En “amarillo” lo que tiene que poner el Maestro: el microcontrolador
- En azul lo que pondrá el Esclavo: TC7

Con la primera parte de la trama se indica a qué registro se desea acceder (dependerá del comando : RTR o RWCR) y con la segunda parte de la trama se realiza la lectura del byte

ESCRITURA DE UN BYTE EN EL REGISTRO

Write Byte Format



COMANDOS POSIBLES:

- RTR=0x00 Leer Registro Temperatura
- RWCR=0x01 Leer/Escribir Configuración

En este caso solo tendría sentido **ESCRIBIR CONFIGURACIÓN**
(Comando RWCR=0x01)

Registros internos: Registros de configuración

b7	b6	b5	b4	b3	b2	b1	b0
Bit de Standby	Data Ready	Reservados (se leen como 0)					
Lect./Escrit.	Sólo lectura	N/A					

- **Bit de Standby:** si está a 1 sitúa al C.I. en bajo consumo, sin realizar conversiones, si a 0 en modo normal
- **Data Ready:** valor de temperatura disponible en el registro si está a 1 el bit. Estará a cero al alimentar el C.I. antes de que se haga la primera conversión y también cuando está en modo standby

Tras un “Power On Reset”, el registro arranca con el valor **0x00**, por tanto en **modo normal**

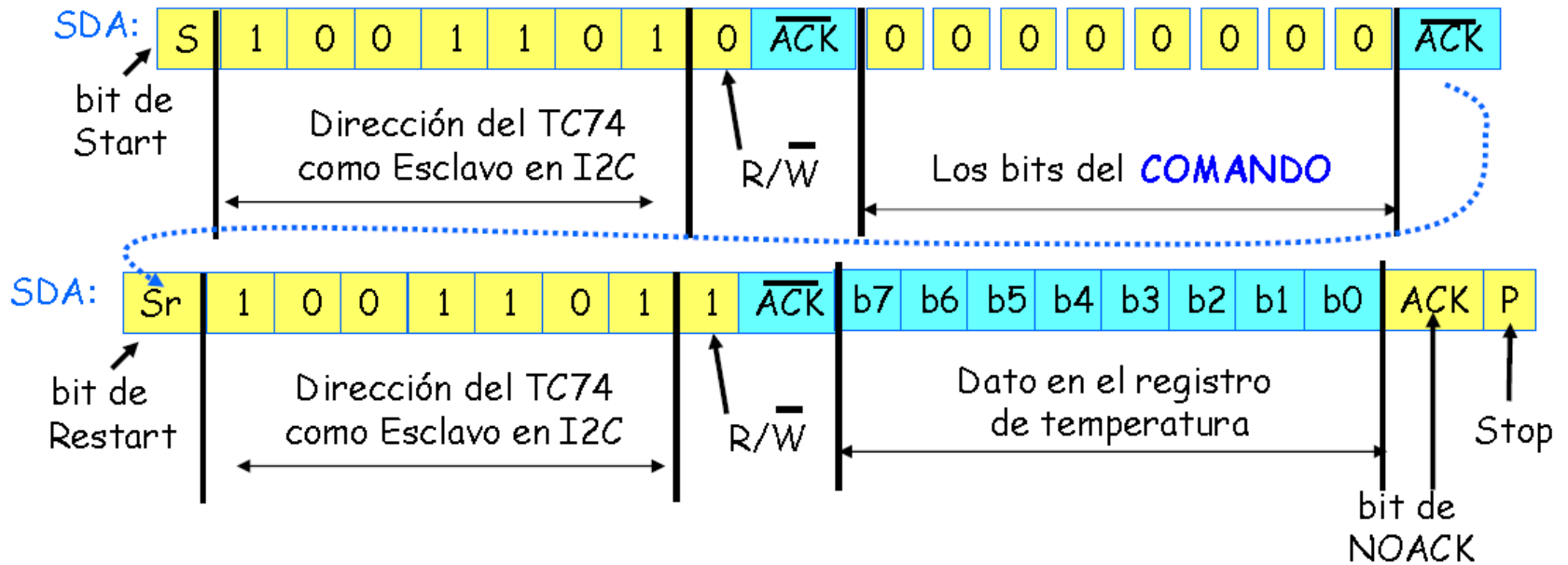
Registros internos: Registro de temperatura

D[7]	D[6]	D[5]	D[4]	D[3]	D[2]	D[1]	D[0]
MSB	X	X	X	X	X	X	LSB

- Nos proporciona el valor en binario resultante de la última conversión A/D, cada **unidad** representa **un grado centígrado**
- Tras un **POR**, se carga con 0x00 hasta que finalice la primera conversión, se sabe porque el bit **DataReady** del registro **CONFIG** pasa a 1
- Los datos se representan en complemento a 2.

Actual Temperature	Registered Temperature	Binary Hex
+130.00°C	+127°C	0111 1111
+127.00°C	+127°C	0111 1111
+126.50°C	+126°C	0111 1110
+25.25°C	+25°C	0001 1001
+0.50°C	0°C	0000 0000
+0.25°C	0°C	0000 0000
0.00°C	0°C	0000 0000
-0.25°C	-1°C	1111 1111
-0.50°C	-1°C	1111 1111
-0.75°C	-1°C	1111 1111
-1.00°C	-1°C	1111 1111
-25.00°C	-25°C	1110 0111
-25.25°C	-26°C	1110 0110
-54.75°C	-55°C	1100 1001
-55.00°C	-55°C	1100 1001
-65.00°C	-65°C	1011 1111

Trama para Lectura de la Temperatura

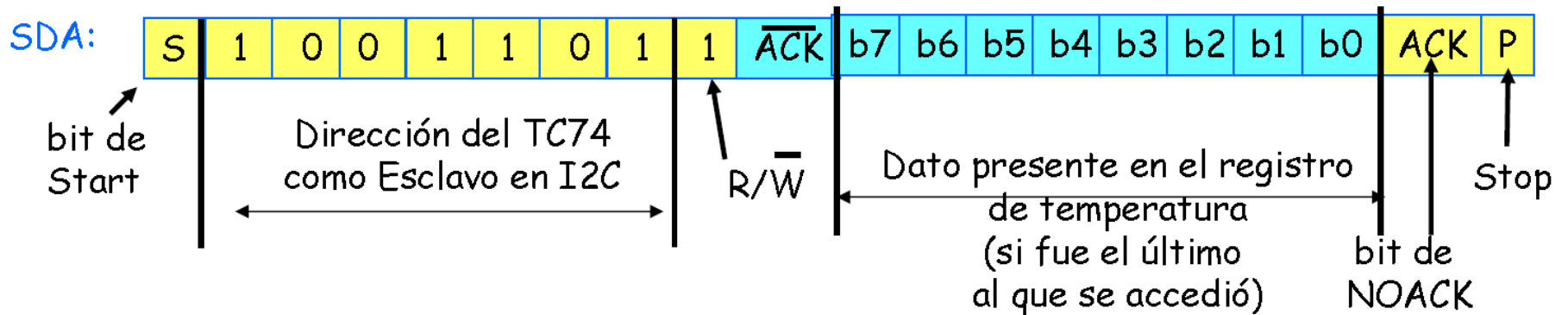


En “amarillo” lo que tiene que poner el Maestro: el microcontrolador PIC
En azul lo que pondrá el Esclavo: TC74

Lectura de Temperatura con secuencia de subprogramas a llamar para I2C

- Generación de bit de start **B_START**
- Envío de dirección de esclavo con bit R/W=0 **TX**
- Envío del comando RTR = 0x00 **TX**
- Envío de bit de restart **B_RESTART**
- Envío de dirección de esclavo y bit R/W=1 **TX**
- Recoger byte del bus **RX**
- Generar bit de NOACK **B_NOACK**
- Generar bit de STOP **B_STOP**

Otra posible lectura de temperatura



Si se desean recoger sucesivos valores de temperatura, sería suficiente con haber “escrito” al principio un comando de lectura de temperatura ($RTR = 0x00$) y luego se podrían leer los valores de temperatura enviando sucesivas tramas como la mostrada arriba

Otra posible Lectura de un registro

- Generación de bit de start **B_START**
- Envío de dirección de esclavo y bit R/W=1 **TX**
- Recoger byte del bus **RX**
- Generar bit de NOACK **B_NOACK**
- Generar bit de STOP **B_STOP**



MINISTERIO
DE EDUCACIÓN



Unión Europea

Fondo social Europeo

El FSE invierte en tu futuro

- Realizado dentro del “Proyecto PLUMA (Plataforma Universal Microcontrolada) Desarrollo de Aplicaciones Didácticas e Industriales”, encuadrado en la convocatoria del Ministerio de Educación según Resolución de 5 de abril de 2011, de la Secretaria de Estado de Educación y Formación Profesional, por la que se convocan ayudas destinadas a la realización de proyectos de innovación aplicada y transferencia del conocimiento en la formación profesional del sistema educativo.
- Según la Resolución de 21 de octubre de 2011, de la Secretaria de Estado de Educación y Formación Profesional, por la que se conceden ayudas destinadas a la realización de proyectos de innovación aplicada y transferencia del conocimiento en la formación profesional del sistema educativo a este proyecto se le conceden 537.627,94 € de ayuda.
- El proyecto cuenta con la cofinanciación del Fondo Social Europeo.

Se hace constar de forma expresa que el contenido difundido y publicado por este medio compromete exclusivamente al autor o autores del mismo.

Licencia

Este documento ha sido liberado bajo Licencia GFDL 1.3 (GNU Free Documentacion License). Se incluyen los términos de la licencia en español (Castellano) al final del mismo.

Copyright (C) 2012 PROYECTO PLUMA.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with with the Invariant Sections being Texts and graphics of Front-Cover, this license and the obligaciones de los beneficiarios de las ayudas. A copy of the license is included in the section entitled "GNU Free Documentation License".

Copyright (c) 2012 PROYECTO PLUMA.

Se otorga permiso para copiar, distribuir y/o modificar este documento bajo los términos de la Licencia de Documentación Libre de GNU, Versión 1.3 o cualquier otra versión posterior publicada por la Free Software Foundation; siendo las Secciones Invariantes los Textos y gráficos de Cubierta Delantera, la licencia y las obligaciones de los beneficiarios de las ayudas. Una copia de la licencia está incluida en la sección titulada Licencia de Documentación Libre de GNU.