

Actividad 5: ALARMA

Materia Programación de Microcontroladores.
Santiago Germino, 20/4/2018.

Capturas de pantalla



```
royconejo@proconejo: ~  
File Edit View Search Terminal Help  
  
Bienvenido a...  
ACS: ALARMA  
Comandos:  
  'i' Este mensaje.  
  'p' Ingresar contraseña.  
  'x' Cancelar operación.  
  'a' Sensores.  
  's' Buffers de I/O.  
  'm' Máquina de estado.  
  'c' Borrar pantalla.  
  
Alarma DESARMADA, presione 'p' para ARMAR.
```

```
royconejo@proconejo: ~
File Edit View Search Terminal Help
: 123
Validando dato ingresado...
Comparando contraseña...
Contraseña incorrecta.
Alarma DESARMADA, presione 'p' para ARMAR.
Ingrese contraseña y presione [ENTER] para continuar.
: 1234
Validando dato ingresado...
Comparando contraseña...
Contraseña OK.
En 6 segundo(s) la alarma cambiará a estado ARMADA.
Armando alarma...
Alarma ARMADA, presione 'p' para DESARMAR.
Ingrese contraseña y presione [ENTER] para continuar.
: xx
Validando dato ingresado...
Comparando contraseña...
Contraseña incorrecta.
Comando no reconocido.
Alarma ARMADA, presione 'p' para DESARMAR.
SENSOR DE PUERTA ACTIVADO.
Modo INTRUSO se activará en 6 segundo(s). Presione 'p' para CANCELAR.
ALERTA DE INTRUSO! Presione 'p' para CANCELAR.
```

Características

1. Comandos de diagnóstico del sistema concurrentes e independientes a la máquina de estado del sistema de alarma.
2. LED verde prende y apaga para indicar transición de DESARMADO hacia ARMADO.
3. Luego de ingresar correctamente el password, la transición de ARMANDO a ARMADO se puede CANCELAR (presionar tecla 'x').
4. Lógica preparada para hasta 31 sensores sin cambios.
5. Sensores TOGGLEAN al presionar el botón correspondiente (para facilitar demostración). Presionar 's' para ver estado de sensores, en particular el de la puerta que no se reproduce en un led.
6. Tres reintentos desde ARMADO a INTRUSO iniciado por el sensor PUERTA.
7. Este programa **se diseñó para interactuar con una terminal** y presenta todas las ventajas inherentes (ver CuteCom vs Picocom en Decisiones de diseño).

Decisiones de diseño

UART

Segun datasheet, el integrado LPC4337 dispone de "16 B Receive and Transmit FIFOs.". A una velocidad de 115200 bps configurado como 8N1 (9 bits), es necesario que la tarea que levanta los datos de la FIFO de entrada se llame a una frecuencia mínima de $115200 / 9 = 12800$ bytes por segundo. Esto dividido por el tamaño de la FIFO (16) da como resultado la cantidad de FIFOs que se llenarían en un segundo: 800.

De este resultado se desprende que, para que no ocurriese overflow, la tarea que levanta todos los datos de la FIFO (uartRecvTask) deberia ejecutarse a 1/800: 1.25 milisegundo. Como no existen valores intermedios, la tarea se debe ejecutar en cada milisegundo.

Bajando la velocidad a 9600 bps, la frecuencia de la tarea puede ser 12 veces más lenta: 15 milisegundos. Se eligió 14 para tener un margen de seguridad.

En cuanto al buffer de entrada, uno de 160 bytes debe consumirse (tarea uartProcessTask) cada 12.5 o 150 ms según sea 115200 o 9600 bps. Se elige 140 ms nuevamente para tener un margen de seguridad.

En la salida, para no producir un overflow en la FIFO (sobrecargando de datos que no llegará a enviar) ni tampoco generar tiempos de demora esperando que se libere espacio, hay que enviar hasta 16 bytes de datos por cada entrada a la tarea en no menos de 2 a 15 ms según sea 115000 o 9600. Por este motivo, se elige llamar a la tarea uartSendTask cada 16 ms.

En el caso de la salida el buffer es mucho más amplio, dado que la frecuencia de envío de datos es muy baja pero el tamaño de los mismos muy grande en relación a la FIFO (a veces de hasta varios kilobytes).

Sistema operativo

Se modificó para soportar "contextos de memoria" para cada tarea y pasaje de ticks de ejecución como parámetros.

Picocom vs CuteCom

CuteCom no es un emulador de terminal ni trabaja con la terminal del sistema ya que solo envía y recibe líneas de texto y las copia sin más en un cuadro de texto. En consecuencia, la interacción con el usuario se dificulta enormemente. En particular, con CuteCom no es posible generar un prompt para que el usuario pueda ingresar y editar un valor. Otra desventaja es que solo representa ASCII básico (no acentos ni ñ).

Por estos motivos se decidió no usar CuteCom como referencia para desarrollar la interfaz de usuario.

Como iniciar picocom:

```
picocom /dev/ttyUSB1 --baud=9600  
salir con C-a C-q
```

Libreria

Se aprovechó la oportunidad para desarrollar el core de la librería del proyecto RETRO-CIAA. La misma cuenta con funciones optimizadas para microcontrolador para manejar FEMs, VARIANTS, ARRAYS, UART en modo productor/consumidor, ingreso y validación de datos, etc.