|  |  |  |  |
| --- | --- | --- | --- |
| **INFORMATICA II**  **Final – primer tema** | **Total Hojas** | **Duración** | **02/oct/2014** |
|  | **45 minutos** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre y Apellido** | **Nº Legajo** | **Calificación** | **Docente Evaluador / Firma** |
|  |  |  |  |

**Tema inicial (8051 o LPC)**

**Nota Inicial:**

*La implementación puede realizarse bajo el concepto de mono-usuario/mono-dedo.*

*Tener presente que el barrido del teclado deberá estar sincronizado en forma lógica con BoardHW, lo cual impactará en la implementación de esta última función.*

|  |  |
| --- | --- |
| Se cuenta con la siguiente función clásica de barrido de teclado matricial, la cual está adaptada para un teclado de 3x4 y se es ejecutada desde una rutina periódica cada 6ms.  El teclado que se quiere controlar con esta rutina, si bien cumple con la condición de ser de 3x4, posee la característica de tenerse que mantener activas las líneas de barrido KEY\_ROW correspondientes por al menos 1ms  Se solicita:  **1.-** Modificar la función para satisfacer las necesidades de HW.  **Nota:** *tener en cuenta que la aplicación no se puede quedar detenida esperando el ms, es decir, no realizar una modificación que sea bloqueante.*  **2.-** Para que todas las teclas se continúen barriendo cada 6ms, mencione que modificaciones se deberían plantear sobre la rutina periódica que la invoca (solo describir las modificaciones).  **3.-** Finalmente, implementar las funciones de barrido de teclado y la primitiva de lectura de teclado. Los prototipos de estas funciones son:  void **Barrido\_Teclado** (void);  unsigned char **get\_key** (void);  La función de barrido será invocada desde una interrupción de temporización (p/ej: systick para el LPC; Timer 0 para el 8051), bajo el supuesto de barrido tomado en el punto anterior  La función get\_key, se podrá utilizar en cualquier parte de la aplicación según necesidad. | char BoardHW(void)  {  // Barro fila 1  **KEY\_ROW\_1 = OFF;**  KEY\_ROW\_2 = ON;  KEY\_ROW\_3 = ON;  if(!KEY\_COL\_1) return KEY\_1;  if(!KEY\_COL\_2) return KEY\_4;  if(!KEY\_COL\_3) return KEY\_7;  if(!KEY\_COL\_4) return KEY\_A;  // Barro fila 2  KEY\_ROW\_1 = ON;  **KEY\_ROW\_2 = OFF;**  KEY\_ROW\_3 = ON;  if(!KEY\_COL\_1) return KEY\_2;  if(!KEY\_COL\_2) return KEY\_5;  if(!KEY\_COL\_3) return KEY\_8;  if(!KEY\_COL\_4) return KEY\_0;  // Barro fila 3  KEY\_ROW\_1 = ON;  KEY\_ROW\_2 = ON;  **KEY\_ROW\_3 = OFF;**  if(!KEY\_COL\_1) return KEY\_3;  if(!KEY\_COL\_2) return KEY\_6;  if(!KEY\_COL\_3) return KEY\_9;  if(!KEY\_COL\_4) return KEY\_B;  // y si no  return NO\_KEY;  } |

**Nota:** *No se solicita la configuración del timer.*

*No se requiere modificar interrupción de temporización, sin embargo, si le simplifica el análisis Se puede considerar que la frecuencia del timer es de 1kHz (una interrupción cada 1 ms).*

|  |  |  |  |
| --- | --- | --- | --- |
| **INFORMATICA II**  **Final – segundo tema** | **Total Hojas** | **Duración** | **02/oct/2014** |
|  | **60 minutos** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre y Apellido** | **Nº Legajo** | **Calificación** | **Docente Evaluador / Firma** |
|  |  |  |  |

**Segundo Tema (8051 o LPC)**

Se posee un simple terminal de control de acceso, el cual está compuesto por un teclado de 3x4; un (1) indicador lumínico; un (1) relé de salida para el control de una puerta automática y una comunicación serie para verificación del número ingresado y gestión remota de la puerta.

|  |  |
| --- | --- |
| El funcionamiento del sistema es el siguiente:  La persona que se acerca a la puerta debe ingresar su clave presionando cuatro (4) dígitos numéricos y la tecla “Enter”.  En caso de querer reiniciar el ingreso, se puede cancelar la operación presionando la tecla “C”.  Estando los cuatro dígitos ingresados, el ingreso de un nuevo dígito será omitido.  En caso de presionar la tecla “Enter” antes de haber ingresado los cuatro (4) dígitos numéricos, esta será omitida.  Al ingresarse el 4to digito, el indicador lumínico se pondrá en verde. |  |

Una vez ingresado los 4 dígitos, al presionarse la tecla “Enter”, se enviará vía PS (ver trama más abajo) la información ingresada por teclado a un sistema remoto, quedando a la espera de su respuesta por un tiempo máximo de 2 segundos (este tiempo puede ser considerado desde el inicio del envío de la trama o finalizado el envío – explicitarlo). En caso de no recibir la respuesta o la trama recibida sea inválida la operación se debe cancelar (similar a la presión de la tecla “C”).

En caso de respuesta afirmativa, se debe accionar el relé de la puerta automática durante seis (6) segundo, mientras que el indicador lumínico se pondrá en amarillo durante el mismo período, luego se apagará. En caso de respuesta negativa (o falta de la misma), se colocará el indicador lumínico en rojo durante 2 segundos y luego se apagará. Durante este proceso el ingreso de cualquier tecla es omitido.

Funciones disponibles (*no implementar, solo utilizar*)

**void set\_led (char color);** // en donde **color**= 0 – apagado / 1 – verde / 2 – amarillo / 3 - rojo

**void set\_rele (char st);** // en donde **st**= OFF – relé en reposo / ON – relé activado

**void set\_time (int ms);** /\* la función **set\_time** dispara un temporizador el cual durará **ms** milisegundos. Al vencer el temporizador, éste se detendrá y colocará la variable global **XYZ** en cero (0), mientras que al invocar a **set\_time**, esta variable se pondrá en uno (1).

Si **set\_time** se usa con **ms**<=0 la variable XYZ se pondrá en uno (1) y el temporizador de detendrá \*/

Tramas de envío/recepción

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Trama de envío: | 0xAA | D1 | D2 | D3 | D4 | Chk | 0xFF |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Trama de recepción: | 0xAA | D1 | D2 | D3 | D4 | ANS | 0xFF |

En donde:

D1 a D4 son los dígitos ingresados,

Chk es el checksum producto de realizar el exor entre los dígitos ingresados

ANS es la respuesta (ANS: 0xA5 acceso permitido / ANS: 0xEE acceso denegado)

0xAA y 0xFF los bytes de inicio y fin de la trama respectivamente

**Se pide**:

1**.-** Realice la máquina de estados para el manejo de los datos ingresados por teclado, tanto gráficamente como su implementación en C, contemplando:

- La lógica antes detallada.

- el manejo del indicador lumínico y relé (mediante las funciones descriptas)

- consulta al sistema remoto y acción dependiendo de la respuesta.

**2.-** realice las funciones de envío y recepción de datos vía PS (por interrupciones) incluyendo el armado y análisis de las tramas a enviar y análisis de las tramas recibidas.

**Notas**:

**a.-** Teclado: utilizar las funciones realizadas en el ejercicio anterior (no volver a implementarlas)

**b.-** No se pide ningún tipo de inicialización.

**c.-** Ni bien inicia el sistema, el indicador lumínico se encontrará apagado y el relé en reposo.

**d.-** El manejo del relé e indicador lumínico se hará mediante las funciones mencionadas. No las implemente.

**e.-** No se requiere implementar el main.

**f.-** Para el manejo de las temporizaciones utilizar set\_time.

**g.**- Queda a criterio del implementador, si los dígitos se manejan en código ASCII o no. Explicitarlo.