

Proyecto 1

Objetivos

Objetivo General

- Que el estudiante interactúe con el microcontrolador Arduino.

Objetivos Específicos

- Adquirir y aplicar conocimiento sobre dispositivos electromecánicos (motores).
- Aprender a utilizar los protocolos de comunicación UART.
- Adquirir y aplicar conocimiento de pantallas LCD en Arduino.
- Adquirir y aplicar conocimiento de memoria EEPROM en Arduino.
- Aprender y aplicar el funcionamiento del lenguaje, funciones disponibles y librerías para programar el microcontrolador.

Descripción

Una bodega de clasificación de paquetes solicita una solución para la distribución automática, hacia sus puertas de embarque, de sus paquetes dependiendo del tamaño de los mismos por medio de bandas transportadoras.

Se requiere presentar un prototipo simulado para el manejo propuesto y aprobación de la implementación requerida.

El prototipo debe considerar los siguientes factores (que serán explicados detenidamente más adelante):

- Usuario Administrador.
- CRUD de usuarios.
- Pantalla LCD para mostrar mensajes localmente.
- Teclado matricial 4x4
- Manejo de token para sincronización con aplicación.
- Estadísticas por usuario.
- Aplicación Bluetooth para el manejo del sistema.

Especificaciones

Pantalla LCD

Todas las acciones que se ejecuten deben mostrarse como estado en la pantalla LCD, sin excepción alguna, así sea que se entregó un paquete a una banda, llegó a la puerta de embarque o se sincronizó con la aplicación Bluetooth, el login fue correcto, el token fue aceptado, etc; estos mensajes deben permanecer por al menos 3 segundos antes de cambiar a una pantalla limpia, **todos los mensajes a mostrar responden, como mínimo, a este tiempo.**

Todo el tiempo que el sistema esté listo para enviar un nuevo paquete dejará fijo un mensaje indicando que está listo; aunque no hay un requerimiento específico para este mensaje no debe ser ambiguo. El sistema está listo para distribuir las cajas hasta que se haya sincronizado con el dispositivo Bluetooth.

Cada vez que el sistema se inicie debe mostrar el estado de este.

El mensaje de inicio debe mostrar:

- Línea superior: "➡Bienvenido◀"
- Línea inferior: "Grupo##-SeccionX"

Luego de iniciado debe mostrar un mensaje con el tiempo transcurrido durante dicho proceso, para finalmente mostrar que se completó el inicio y que se encuentra lista para el siguiente paso.

CRUD De Usuarios

El crud de usuarios será híbrido.

La creación de nuevos usuarios será exclusivamente desde el teclado matricial 4x4, se debe apoyar en la pantalla LCD para poder crear un menú completamente entendible y fácil para el usuario.

Únicamente el Administrador puede realizar este crud, para ello se establecen las siguientes credenciales:

Sección	Nombre	Usuario	Password
A	Admin-##A	A1973C	AD##DA
B	Admin-##B	B2684D	BC##CB

Dónde ## responde al número de grupo, si fuese menor a 10 el primer dígito será 0.

Las credenciales de usuario deberán ser almacenados en la memoria EEPROM, sus características son:

- Código de usuario netamente numérico
- Password alfanumérico considerando únicamente las letras del teclado.

Estructura para usuario

	Min	Max
Código de usuario	6	6
Password	1	8
Nombre	1	16

Desde la aplicación bluetooth, logueado como Administrador se colocarán los nombres de los usuarios; esta será la única función de CRUD hecha desde la aplicación.

El borrado de un usuario debe realizarse únicamente desde el teclado 4x4.

Todos los datos de los usuarios deben ser almacenados en EEPROM.

Login

El login se requiere que sea validado por medio de un token que será generado por la aplicación y luego ingresado manualmente en el teclado matricial.

Los token generados deberán de ser de longitud 6, la mitad serán letras y la mitad números, estos deben de estar mezclados, cualquier token que tenga todos los números y letras contiguas deberá ser descartado.

Flujo de login:

- Se realiza el login en el teclado matricial.
- Se conecta con la aplicación Bluetooth.
- En la aplicación se muestra una pantalla de login, donde se deberá volver a solicitar las credenciales del usuario logueado en el teclado.
- Si las credenciales coinciden en ambas partes la aplicación generará un token que debe ser ingresado en el teclado
- Introducir el token en el teclado matricial, si el token coincide permitirá ir hacia la aplicación de manipulación del dispositivo.

Todo el tiempo que un usuario esté logueado debe mostrar, en la aplicación Bluetooth, el nombre del usuario, en caso de no haberlo configurado no debe permitir el login desde la aplicación.

El usar el usuario administrador para loguearse no debe permitir utilizar el equipo, en sí, solo permitirá ver reportes, tanto en la LCD como en la aplicación Bluetooth. Administrador desde la aplicación también podrá colocar nombres a cada usuario.

Flujo de la simulación

Los tamaños de los paquetes a identificar son 3 (pequeño, mediano y grande), por lo que se necesita simular 4 bandas transportadoras.

Para visualizar las bandas transportadoras se deben de simular con, por lo menos, 10 leds; un led encendido indica la posición instantanea del paquete, lo que significa que por medio de ir encendiendo los leds se simulará el movimiento visualmente.

Se requiere que los paquetes sean identificados todo el trayecto, por tamaño, por lo que se solicita se utilicen leds de dos colores usando un color para un tamaño, el segundo color para otro tamaño e intercambiando los colores (al menos dos veces antes que avance a la siguiente posición) para el tercer tamaño.

Adicionalmente se requiere que se active dos motores stepper por banda simulada, estos motores se activará cada vez que en la banda se este transportando un producto y se desactivará al ser llevado al final. (por banda un motor con driver y otro sin driver)

En la aplicación Bluetooth se presionará un botón que indique el tamaño de paquete y eso accionará la primer banda transportadora.

Desde el teclado matricial no se puede hacer el envío de paquetes.

Al final de la primer banda transportadora se necesita que el motor espera a la activación de un servo motor para entregar los paquetes de la banda 1 hacia las siguientes bandas; las bandas deben estar separadas por angulos de 45 grados entre ellas, ocupando 135 grados por las 3; el servomotor se debe posicionar en el angulo correcto antes de proceder a entregar el paquete de la banda uno hacia la banda correspondiente según su tamaño; una vez el servomotor se encuentre en posición el paquete puede continuar su camino.

Cada banda transportadora puede llevar un paquete a la vez, una vez se encuentra libre la banda transportadora puede permitir accionar el envío de otro paquete, esto quiere decir que máximo dos paquetes pueden encontrarse en movimiento en un instante dado y la opción de enviar paquete debe estar deshabilitada si hay dos en tránsito.

Al final de cada banda se debe acumular la cantidad de paquetes disponibles en un momento dado y dar la opción de despachar envío para cada puerta, lo que reiniciará a 0 la cantidad de paquetes para una puerta de embarque específica. Cada vez que se envíe un despacho deberá sonar un buzzer, con distinto tono para cada puerta de envío.

Logout

En la aplicación Bluetooth debe existir un botón de Logout, esto hará un logout de ambos lugares.

Reportes

Se requiere reportes (tanto en LCD como en aplicación Bluetooth) que indiquen, por usuario, cantidad de paquetes procesados por tamaño y cantidad de envíos despachados por tamaño.

Se debe tener en memoria, por lo menos, a 5 usuarios con todos sus datos almacenados.

En la pantalla LCD, a la hora de mostrar reportes por usuario deben ser elegibles por nombre, no por código de usuario.

Todos los reportes deben incluir el código de empleado y nombre, incluyendo los mostrados en la pantalla LCD.

Otras consideraciones

Si fuese necesario el uso de más de un arduino, la comunicación entre ellos deberá ser por medio de comunicación Serial. Se recomienda usar un dispositivo como maestro.

El menú en pantalla LCD debe ser amigable, si se elabora un menú ambiguo no se otorgarán puntos por él.

La señal del último led de cada banda simulada hara las veces de la señal de cercanía de un sensor ultrasónico.

Materiales

- Leds de dos colores (al menos 40)
- Motores stepper (8)
- Servomotor (1)
- Teclado alfanumérico matricial de 4x4
- Módulo Bluetooth
- Pantalla LCD.
- Buzzer.

Entregables

- Manual Técnico (elaborado en MD).
- Manual de usuario (elaborado en MD).
- Código funcional de la simulación.
- Archivo de simulación en Proteus.
- Código funcional de la aplicación. Si la aplicación fuese generada en algún medio que no provea código se deben incluir todas las capturas de su desarrollo.
- Link al repositorio de gitlab.

Entrega

- La entrega, en UEDI, será el link al repositorio de gitlab
- El nombre del repositorio responde a la siguiente estructura:
 - ACE1-
 - Año
 - Período
 - Código curso
 - Sección
 - "G" + Número de Grupo.
 - PROY1

ej. ACE1-222S0778BG13PROY1

- Debe existir más de un commit por cada integrante del grupo.
- Realizar el último commit y hacer su entrega en UEDI antes de 23:59 horas del 9 de septiembre.

- Se ejecutará un checkout hacia el último commit hecho antes de la fecha de entrega.
- Se debe agregar al auxiliar de su sección como miembro del repositorio
 - Sección A: @Orbp1403
 - Sección B: @pinedaMario

Observaciones y Restricciones

- Debido a la naturaleza de la práctica, no se califican entregas que no estén integradas en una sola solución.
- Se debe tener en memoria, por lo menos, a 5 usuarios con todos sus datos almacenados.
- Se deben respetar los requerimientos sobre el uso de driver en motores stepper.
- Se debe entregar los manuales técnico y de usuario, de lo contrario se asumirá que el estudiante copió.
- Es requerido el uso de EEPROM.
- La realización de la práctica es de forma grupal
- Se debe presentar la práctica simulada en Proteus, arduino también será simulado con alguna librería para Proteus.
- Librerías permitidas:
 - No se permiten librerías fuera de las acá listadas**
 - Stepper.h <https://www.arduino.cc/en/reference/stepper>
 - LiquidCrystal_I2C.h <https://www.arduino.cc/reference/en/libraries/liquidcrystal-i2c/>
 - EEPROM <https://www.arduino.cc/en/Reference/EEPROM>
 - Keypad.h <https://www.arduino.cc/reference/en/libraries/keypad/>
 - Servo.h <https://www.arduino.cc/reference/en/libraries/servo/>
- Se calificará encapsulamiento.
- Si una acción no tiene mensaje en pantalla LCD no tendrá puntos la acción.
- Si no se cumplen los requerimientos de los reportes no se obtendrá nota por ellos.
- El día de la calificación se harán preguntas, modificación de código sobre aspectos utilizados en la elaboración del proyecto, las cuales se considerarán en la nota final.
- **Copias parciales o totales tendrán una nota de 0 puntos y los involucrados serán reportados a la Escuela de Ciencias y Sistemas**