

Report Progetto Sistemi Embedded e Real-Time:

Gestore portiera auto con lettore carta NFC e rilevamento urti

Lista componenti:

- Elegoo Mega 2560 R3 Board ATmega2560 ATMEGA16U2
- ICQUANZX Display LCD seriale IIC I2C TWI 1602
- RFID RC522 13,56MHz con Reader
- Elegoo Kit Elettronico con Breadboard (utilizzati 3 led, 2 bottoni, 2 resistenze, cavetti vari)
- ARCELI DC 3.3-5V Buzzer Alarm per Arduino
- SW420 Vibration Shaker

Il progetto consiste nello sviluppare la “serratura” di un’automobile utilizzando come chiave una carta NFC. È stato utilizzato un ARDUINO MEGA 2560 collegato ad un lettore carta RFID RC522. È stato poi utilizzato un display LCD per la visualizzazione dell’output. Sono inoltre presenti tre led: uno rosso indicante lo stato di chiusura dell’automobile, uno verde indicante lo stato di apertura del veicolo e uno blu indicante se la chiave (o carta) è in prossimità dell’automobile. Inoltre, sono presenti due bottoni, uno per l’apertura e uno per la chiusura dell’auto. Infine, è stato utilizzato un sensore di vibrazioni per rilevare gli urti e un buzzer per segnalare l’evento (come un antifurto).

Di default l’ARDUINO parte con il LED rosso acceso in quanto la macchina all’inizio è chiusa. In memoria è salvata la chiave valida, l’unica autorizzata per l’accesso del veicolo. Ogni volta che una carta viene letta viene confrontata con questa chiave e se uguale, tramite la pressione dei due bottoni è possibile aprire o chiudere l’automobile (viene inoltre visualizzato sul display LCD il TAG UID della chiave valida). La chiave è nelle vicinanze dell’auto, e viene quindi acceso il led blu indicante la prossimità. Fino a quando si è in questo stato il veicolo può essere aperto e chiuso a piacere (questo per simulare che la carta si trovi ad una distanza dall’auto che sia all’interno di un range prestabilito). Quando la chiave viene passata nuovamente per il lettore NFC allora essa viene considerata fuori portata, e per riavere accesso alla macchina sarà necessaria una nuova lettura. Il sensore di vibrazione viene utilizzato per simulare la rilevazione di un urto da parte dell’automobile, e se viene quindi rilevata una forte vibrazione si attiva il buzzer per evidenziare l’allarme (viene inoltre mostrato su display LCD un output relativo all’impatto avvenuto).

Per lo sviluppo del software sono stati utilizzati cinque task, uno per la lettura NFC, uno per la rilevazione di vibrazioni, uno per la visualizzazione sul display e uno per l’apertura e la chiusura dell’auto. Ognuno di questi ha accesso a variabili globali e il loro codice è protetto da un mutex che permette la mutua esclusione di questi task.

- Il primo task si occupa di leggere le chiavi utilizzate e le confronta con la chiave valida, in caso positivo setta a *true* la variabile *cardAuth* e aggiorna di conseguenza lo stato del led di prossimità controllando se il numero di volte in cui è stata passata è un numero pari o dispari.
- Il task di rilevazione di vibrazioni, in caso di urto (simulata da una forte vibrazione), fa suonare il buzzer, che si spegnerà in automatico dopo un determinato numero di tick, ma che può essere spento manualmente convalidando la carta nel veicolo, in quanto questo task è attivo solo quando il proprietario non si trova nelle vicinanze del veicolo.

- Il task del display si occupa di scrivere sul relativo schermo LCD il TAG della carta che è stata validata e che quindi permette l'accesso al veicolo. Si occupa inoltre di scrivere sul display in caso di urto rilevato "Impact Alarm".
- Il task di apertura se la carta è stata autorizzata, e se è stato premuto il tasto di apertura apre l'automobile, aggiornando anche il led verde.
- Il task di chiusura se la carta è stata autorizzata, e se è stato premuto il tasto di chiusura chiude l'automobile, aggiornando anche il led rosso.

Lo scheduler utilizzato per la gestione dei task è quello PRIORITY BASED, abbiamo infatti assegnato al task di lettura RFID il valore di priorità 2, al task di rilevazione di vibrazioni priorità 2, mentre al task del display, a quello di apertura e di chiusura è stato assegnato il valore 1.

Il progetto rispetta inoltre la conformità MISRA C, seguendo le sue linee guida di sviluppo software.

Pedinatura:

