

Relazione Laboratorio Hardware N°2

Scopo del laboratorio: realizzare uno smart home controller che vada a gestire una serie di sensori e combinarne le letture per controllare un sistema di riscaldamento e condizionamento.

Questo esercizio è stato realizzato utilizzando tutti i sensori e attuatori forniti nel seguente modo:

Setup: sensore di temperatura per rilevare la temperatura dell'ambiente

- 1) Ipotizzati due diversi set point di base per l'attuatore FAN si è proceduto con la creazione di uno sketch in grado di modulare la velocità della ventola in base alla temperatura fissata a due valori di riferimento: il primo più basso aziona la ventola con una velocità ridotta che andrà poi ad aumentare fino ad arrivare al 100% della velocità in prossimità di un valore di temperatura di soglia massima.
- 2) Realizzazione dello stesso tipo di algoritmo ma con logica inversa in quanto si parla ora di un LED simulante un riscaldamento. In questo caso la potenza del LED è inversamente proporzionale al valore di temperatura (più si abbassa la temperatura, più sale la potenza del riscaldamento). Per risolvere questi due punti si sono utilizzati due pin di GPIO in modalità output PWM tramite una funzione a onda quadra dove il duty cycle sale in riferimento all'algoritmo scritto sopra.
- 3) Realizzazione di un algoritmo per monitorare la presenza di una persona nell'ambiente considerato tramite un sensore PIR che è stato collegato ad un GPIO con interrupt per poter

avviare la procedura nel momento in cui si rileva un minimo cambiamento di stato da parte del sensore. È stato poi messo un tempo di timeout per verificare invece l'assenza di persone.

- 4) Realizzazione di un algoritmo simile al precedente ma relativo ad un sensore di rumore. Si è creato un buffer circolare per perfezionare la verifica della presenza di persone durante il timeout prefissato. In questo punto si è sostituito l'interrupt dal sensore PIR a quello di rumore, in quanto il sensore PIR avendo diversi potenziometri al suo interno può essere settato anche il tempo in cui il suo stato rimane attivo da quando rileva un movimento. Questo ne ha facilitato la lettura tramite il polling.
- 5) È stato modificato l'algoritmo di questi due ultimi sensori utilizzando la tecnica del sensor fusion poiché entrambi i sensori sono relativi al controllo della presenza nell'ambiente e uno dei due potrebbe essere ingannato da falsi positivi (ad esempio il sensore di rumore può notificare una presenza semplicemente rilevando il passaggio di un'ambulanza).
- 6) Aggiornamento dei dieci set point della temperatura a cui gli attuatori iniziano a lavorare; qui di seguito è riportato un esempio con il setpoint della temperatura a 25°C in entrambi i casi ma con due differenti intervalli, uno in presenza e uno in assenza, rispettivamente di 10°C e 5°C:
 - a) float tempFanMinNoPeople = 25;
 - b) float tempFanMaxNoPeople = 30;
 - c) float tempLedMinNoPeople = 20;
 - d) float tempLedMaxNoPeople = 25;
 - e) float tempFanMinWithPeople = 25;
 - f) float tempFanMaxWithPeople = 35;
 - g) float tempLedMinWithPeople = 15;
 - h) float tempLedMaxWithPeople = 25;
- 7) Inserimento del pannello LCD, settato in modo da mostrare due pagine diverse che esauriscono tutti i dati rilevati e gestiti da Arduino. Le pagine scorrono con un intervallo di 5 secondi.
- 8) Implementazione di un algoritmo in grado di decodificare una stringa inserita tramite interfaccia seriale per poter modificare i setpoint di accensione e spegnimento dei due attuatori FAN e LED (riscaldamento). La stringa inserita è formattata come segue:

```
//esempio stringa 25.1/26.0/20.0/21.0/23.0/28.0/15.0/22.0
```

Ogni elemento della stringa corrisponde al set di valori impostati nell'ordine descritto al punto 6.