



IIS PRIMO LEVI

FCA

FIAT CHRYSLER AUTOMOBILES

ALTERNANZA SCUOLA LAVORO
Sistema semaforico per
traffico AGV

1

Analisi funzionale

INTRODUZIONE

Nel corso del triennio abbiamo realizzato un sistema semaforico per il traffico dei carrelli AGV. I carrelli servono per trasportare i componenti delle automobili dal magazzino alla linea di produzione.



INTRODUZIONE

Il progetto commissionatoci da FCA consiste nella creazione di un sistema semaforico che gestisca l'incrocio tra il percorso di un carrello guidato da un AGV e la strada normalmente percorribile nella fabbrica.

L'AGV si muove automaticamente seguendo un itinerario sempre uguale: prima va alla zona di carico dove vengono posti sul carrello i vari pezzi che serviranno in zona di montaggio. Dopo attraverserà la strada per arrivare dall'operaio incaricato che scaricherà il necessario e farà ripartire l'AGV, il quale tornando alla zona iniziale riinizierà il suo ciclo.

Il nostro obiettivo era creare un sistema che gestisse due semafori, i quali si devono attivare al passaggio dell'AGV in modo da avvisare in tempo chi dovesse attraversare quel passaggio. Per realizzare il nostro scopo abbiamo utilizzato: due semafori controllati da una scheda Arduino, due sensori bluetooth e due sensori PIR collegati ad altre due schede Arduino.

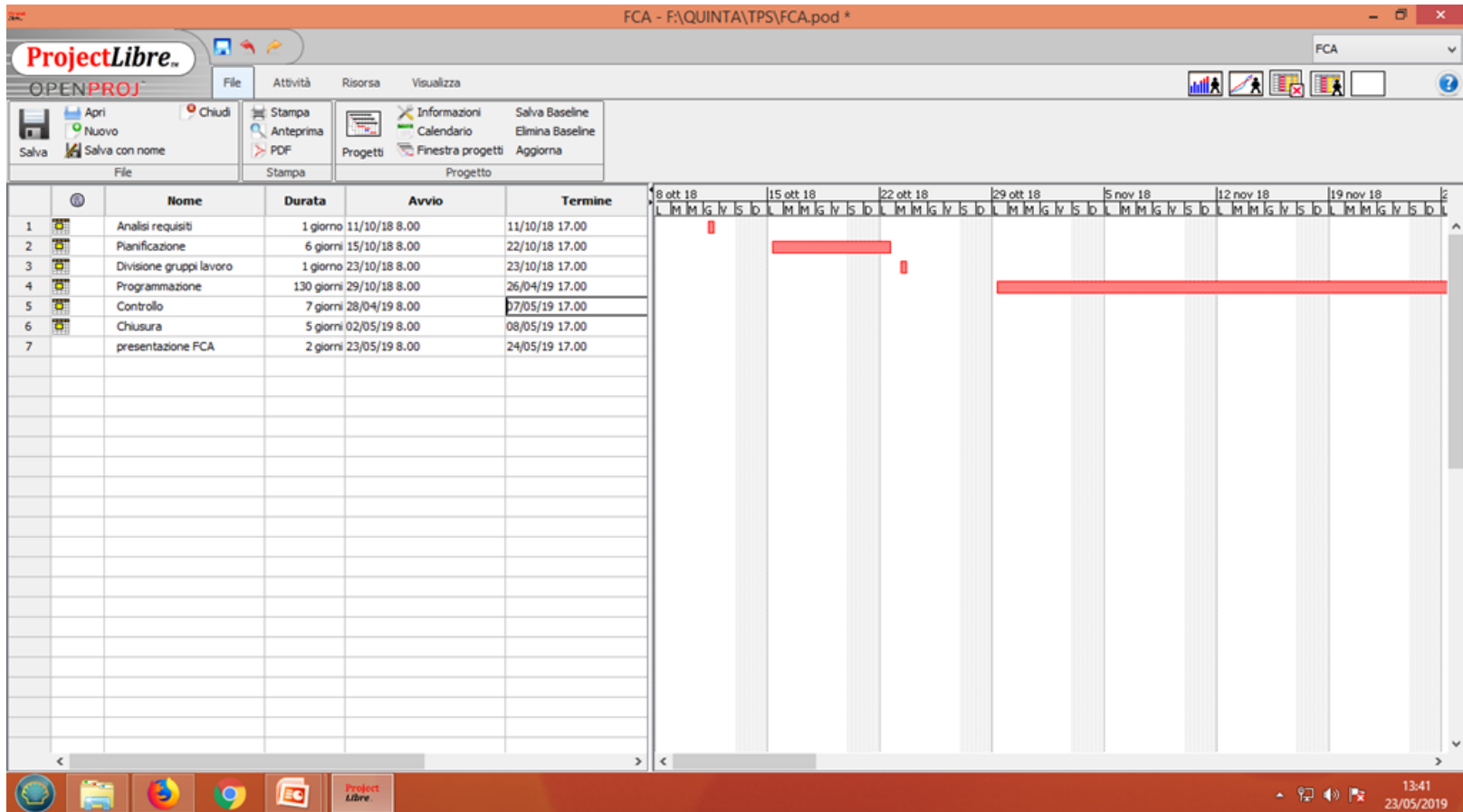
I due semafori saranno normalmente verdi, ma quando l'AGV passerà davanti al sensore PIR (posizionato poco prima dell'attraversamento) diventeranno rossi fino a che il carrello non sarà passato dall'altro lato della strada, liberando il passaggio. Collegate ai sensori PIR ci saranno due schede Arduino, che, una volta ricevuto dai sensori PIR il segnale che il carrello è passato, invieranno tramite il sensore bluetooth alla scheda dei semafori il segnale di farli diventare rossi. Questa riceverà il segnale sul sensore bluetooth posto su di essa, e tramite un programma codificato in c++ cambierà colore ai semafori.

COME ABBIAMO AFFRONTATO IL PROGETTO

In questi tre anni abbiamo lavorato in gruppi, divisi in base alle nostre professionalità ed interessi specifici, per svolgere il lavoro in modo più efficace e veloce.

Abbiamo riscontrato delle difficoltà nella realizzazione per via di specifiche di FCA restrittive, ma abbiamo cercato soluzioni alternative ai problemi incontrati rispettando i limiti di esigenze di magazzino, costi di progetto e di tempo.

FASI DEL PROGETTO: DIAGRAMMA DI GANTT



TEMPISTICHE PIANIFICATE DAL TEAM LEADER

Per gestire le tempistiche dello sviluppo abbiamo utilizzato il programma Project Libre che permette di creare un diagramma di Gantt per monitorare lo stato di avanzamento lavori di ogni fase del progetto.

SINCRONIZZAZIONE ATTIVITA' SULLE DUE SCHEDE ARDUINO

PRIMO GRUPPO DI LAVORO:

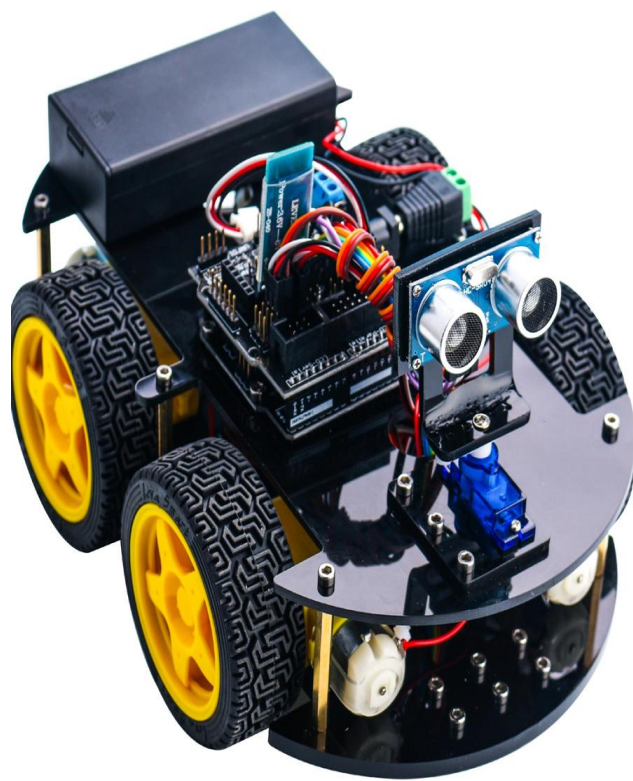
codice C++ per gestione sensore PIR e modulo bluetooth master di invio dati alla seconda scheda.

SECONDO GRUPPO DI LAVORO:

codice C++ per gestione modulo bluetooth slave di ricezione dati dalla prima scheda e per gestione del semaforo

Le attività sono state svolte in parallelo e solo al termine sono stati effettuati i test di comunicazione

IL PROTOTIPO UTILIZZATO PER I TEST IN SOSTITUZIONE DELL' AGV



Il progetto con FCA si occupa di gestire un impianto semaforico utilizzando le schede arduino come sistema di gestione



I SENSORI DI TRACCIAMENTO LINEE UTILIZZATI DAL PROTOTIPO PER SEGUIRE IL PERCORSO



Il sensore si presenta come una piccola scheda elettronica i cui componenti principali sono costituiti da un led emettitore ad infrarossi ed un ricevitore.



IL CODICE NECESSARIO A FAR SEGUIRE UN PERCORSO AL PROTOTI

```
#define LT1 digitalWrite(10)
#define LT2 digitalWrite(4)
#define LT3 digitalWrite(2)

#define ENA 5
#define ENB 11
#define IN1 6
#define IN2 7
#define IN3 8
#define IN4 9

#define ABS 115

void forward() {
  analogWrite(ENA, ABS);
  analogWrite(ENB, ABS);
  digitalWrite(IN1, HIGH);
  digitalWrite(IN2, LOW);
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, HIGH);
  Serial.println("go forward!");
}

void back() {
  analogWrite(ENA, ABS);
  analogWrite(ENB, ABS);
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, HIGH);
  digitalWrite(IN3, HIGH);
  digitalWrite(IN4, LOW);
  Serial.println("go back!");
}

void left() {
  analogWrite(ENA, ABS);
  analogWrite(ENB, ABS);
  digitalWrite(IN1, HIGH);
  digitalWrite(IN2, LOW);
  digitalWrite(IN3, HIGH);
  digitalWrite(IN4, LOW);
  Serial.println("go left!");
}
```



IL CODICE NECESSARIO A FAR SEGUIRE UN PERCORSO AL PROTOT

```
void right(){
  analogWrite(ENA, ABS);
  analogWrite(ENB, ABS);
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, HIGH);
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, HIGH);
  Serial.println("go right!");
}

void stop(){
  digitalWrite(ENA, LOW);
  digitalWrite(ENB, LOW);
  Serial.println("Stop!");
}

void setup(){
  Serial.begin(9600);
}

void loop() {

  //LT1=digitalRead(10);
  //LT2=digitalRead(4);
  //LT3=digitalRead(2);
  if(LT2){
    forward();
  }
  else if(LT1) {
    left();
    while(LT1);
  }
  else if(LT3) {
    right();
    while(LT3);
  }
}
```



SISTEMA SEMAFORICO AGV PER FCA

Il nostro obiettivo era creare un sistema che gestisse due semafori, i quali si devono attivare al passaggio dell'AGV in modo da avvisare in tempo chi dovesse attraversare quel passaggio.

Per realizzare il nostro scopo abbiamo utilizzato:

- due semafori controllati ad una scheda Arduino
- due sensori PIR collegati ad altre due schede Arduino.
- dei sensori bluetooth per la comunicazione fra le schede

L'ELEMENTO CHIAVE: LA SCHEDA ARDUINO

Una piattaforma Open Source

Scheda fisica programmabile in linguaggio
C/C++



I SENSORI BLUETOOTH:

Il modulo HC-06 Bluetooth si presenta con quattro pin, due per l'alimentazione (VCC, GND) e due per la comunicazione seriale con Arduino (RX, TX).



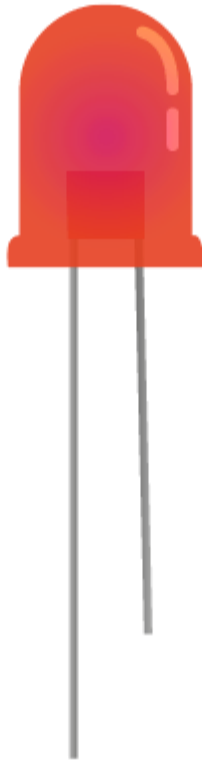
I SENSORI PIR

Questo modulo è un sensore a infrarossi in grado di fornire un segnale alto a 3V quando rileva un corpo in movimento.



I MODULI SEMAFORICI:

Per simulare le luci del semaforo abbiamo utilizzato dei semplici led: uno rosso, uno giallo ed uno verde.



I MODULI UTILIZZATI NEL PROTOTIPO ROBOT CAR:

Sensori di tracciamento: componenti sistemati sotto alla robot car per leggere la linea di nastro adesivo nera



DISPOSITIVI FONDAMENTALI PER IL FUNZIONAMENTO GENERALE:

Breadbord

Cavi di collegamento

