

Progetto SISTEMI OPERATIVI 2022-2023

Simone Cappabianca - Mat: 5423306
simone.cappabianca@edu.unifi.it

Dicembre 31, 2023

Contents

1	Istruzioni per la compilazione e esecuzione	3
2	Sistema obiettivo	3
3	Elementi facoltativi	4
4	Progettazione e implementazione	5
5	Esecuzione	6

1 Istruzioni per la compilazione e esecuzione

Tutti i comandi sottostanti vanno eseguiti dalla directory del progetto.
Per l'installazione del progetto è necessario eseguire i seguenti comandi:

1. `make all`
2. `make install`

Per disinstallare il progetto è necessario eseguire il comando:

1. `make uninstall`

Per eseguire il progetto è necessario aprire una prima shell ed eseguire il comando con l'opzione di lancio che desideriamo (**NORMALE/ARTIFICIALE**):

1. `./bin/ecu.out "OPZIONE-DI-LANCIO"`

In una seconda shell è necessario eseguire il comando:

1. `./bin/hmi_output.out`

2 Sistema obiettivo

Il progetto è stato sviluppato sulla distribuzione linux **Ubuntu 22.04 LTS**.

3 Elementi facoltativi

#	Elemento Facoltativo	Realizzato (SI/NO)	Metodo o file principale
1	Ad ogni accelerazione, c'è una probabilità di 10^{-5} che l'acceleratore fallisca. In tal caso, il componente throttle control invia un segnale alla Central ECU per evidenziare tale evento, e la Central ECU avvia la procedura di AR-RESTO	NO	
2	Componente "forward facing radar"	NO	
3	Quando si attiva l'interazione con park assist, la Central ECU sospende (o rimuove) tutti i sensori e attuatori, tranne park assist e surround view cameras.	NO	
4	Il componente Park assist non è generato all'avvio del Sistema, ma creato dalla Central ECU al bisogno.	SI	
5	Se il componente surround view cameras è implementato, park assist trasmette a Central ECU anche i byte ricevuti da surround view cameras.	NO	

6	Componente "surround view cameras"	NO	
7	Il comando di PARCHEGGIO potrebbe arrivare mentre i vari attuatori stanno eseguendo ulteriori comandi (accelerare o sterzare). I vari attuatori interrompono le loro azioni, per avviare le procedure di parcheggio.	NO	
8	Se la Central ECU riceve il segnale di fallimento accelerazione da "throttle control", imposta la velocità a 0 e invia all'output della HMI un messaggio di totale terminazione dell'esecuzione.	NO	

4 Progettazione e implementazione

Le scelte implementative per la realizzazione del progetto sono le seguenti:

- il componente **Central ECU** si occupa di generare i processi dei componenti necessari per esecuzione ad esclusione dell'output della Human-Machine Interface;
- la **Human-Machine Interface** è stata divisa in due processi distinti uno relativo all'input e uno relativo all'output.

Nello specifico il **Central ECU** genera i processi dei seguenti componenti:

1. **front windshield camera**;
2. **steer-by-wire**;

3. **throttle control**;
4. **brake-by-wire**;
5. **Human-Machine interface** (input);
6. **park assist**.

Mentre i processi dei primi 5 componenti vengono generati al momento dell'avvio della **Central ECU**, il componente **park assist** viene creato quando **Central ECU** riceve il comando *PARCHEGGIO*.

Per quanto riguarda la comunicazione tra i processi è stato utilizzato un socket di tipo FIFO nei seguenti casi:

- **front windshield camera** → **Central ECU**;
- **Central ECU** → **steer-by-wire**;
- **Central ECU** → **brake-by-wire**;
- **Central ECU** → **throttle control**;
- **Central ECU** → **Human-Machine Interface output**;
- **park assist** → **Central ECU**.

Sono stati usati i segnali invece nei seguenti casi:

- **Human-Machine Interface input** → **Central ECU** per la gestione tutti i comandi in input (*INIZIO*, *PARCHEGGIO*, *ARRESTO*);
- **Central ECU** → **brake-by-wire** per la gestione del comando *ARRESTO*
- **Central ECU** → **brake-by-wire** per la gestione del comando *PERICOLO*

5 Esecuzione