

Gruppo 4

MANUALE PROGRAMMATORE

1. Introduzione

Il programma sviluppato, per il microprocessore STM32F413RGT6, permette l'interazione tramite comunicazione seriale USART1. Lo sviluppo del programma è volto alla modifica della frequenza di lavoro del microprocessore e dalla possibilità di comunicare attraverso la seriale CAN-bus.

2. Struttura del progetto

Il progetto è composto da tre file principali:

- Sniffer_CanBus_TxRx_80_16mhz.c: contenente il main del programma, le intestazioni e le varie funzioni.
- STM32F413XX.H: contenente i define per poter permettere l'inizializzazione

3. Sniffer_CanBus_TxRx_80_16mhz.c

In uno stato iniziale si dichiarano tutti i prototipi che ci sono utili, per poi andare nel main() in cui prima viene chiamata la funzione "init", dove vengono configurate GPIOA e GPIOB oltre che attivato il clock su queste e USART1, per poi stampate a schermo, attraverso un ciclo while dove all'interno è presente uno "switch case", le possibili scelte da poter fare tra la variazione di frequenza "gestisci_menu_frequenza()" e la gestione della comunicazione CAN "Gestione_Sniffer_CAN".

Il void gestisci_menu_frequenza(void) ha al suo interno un ciclo while che ci consente di scegliere le due frequenze di lavoro scelte, attraverso le funzioni set_clock_16MHz() e set_clock_80MHz(), Il baud rate non verrà cambiato in base alla frequenza selezionata, verrà solo variato il fattore BRR.

Il void Gestione_Sniffer_CAN(long cpu_freq) ha il ruolo di:

- Configurare le GPIOB
- Configurare il Timing con target 500 kbps, usando il range ideale di time quanta
- Abilitare GPIOB e CAN1
- Effettuare il reset del CAN
- Analizza la frequenza di lavoro impostata precedentemente
- Configura e avvia il CAN1
- Avviare la comunicazione in RX o la trasmissione in TX
- Lettura a schermo tramite seriale dei messaggi ricevuti

Gruppo 4

MANUALE PROGRAMMATTORE

4. Analisi della Comunicazione

Per verificare che il dispositivo funzioni qual'ora non si avesse a disposizione nient'altro che la scheda per la comunicazione, si può usufruire di un oscilloscopio effettuando un test come è possibile notare in *Figura 1* e *Figura 2*.



Figura 1: inizio del messaggio. 2 microsecondi è il nominal bit time. La frequenza di trasmissione è 500kbps (tipico per un'implementazione automotive)



Figura 2: Inizio e fine del messaggio CAN, si può notare CANH in giallo e CANL in blu.

Messaggio CAN:

DHex: 0x123 (Un ID per test)

DLC: 8 (Invieremo il massimo dei byte consentiti, ovvero 8)

Data L (Low): 0x01020304

Data H (High): 0x05060708