

Continental Learning

Laboratory 3.A SW

Laboratorul 4:

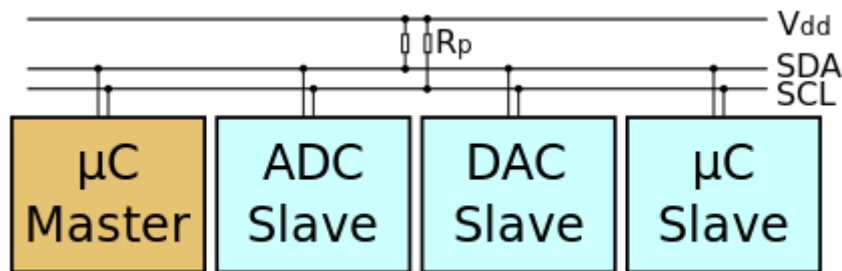
- Prezentarea protocolului de comunicație I2C
- Prezentarea driver-ului pentru touch screen STMPE811
- Configurarea I2C.
- Aplicații demonstrative.

1. Prezentarea protocolului de comunicare I2C.

I2C reprezintă prescurtarea denumirii protocolului de comunicație **I**nter-**I**ntegrated **C**ircuit.

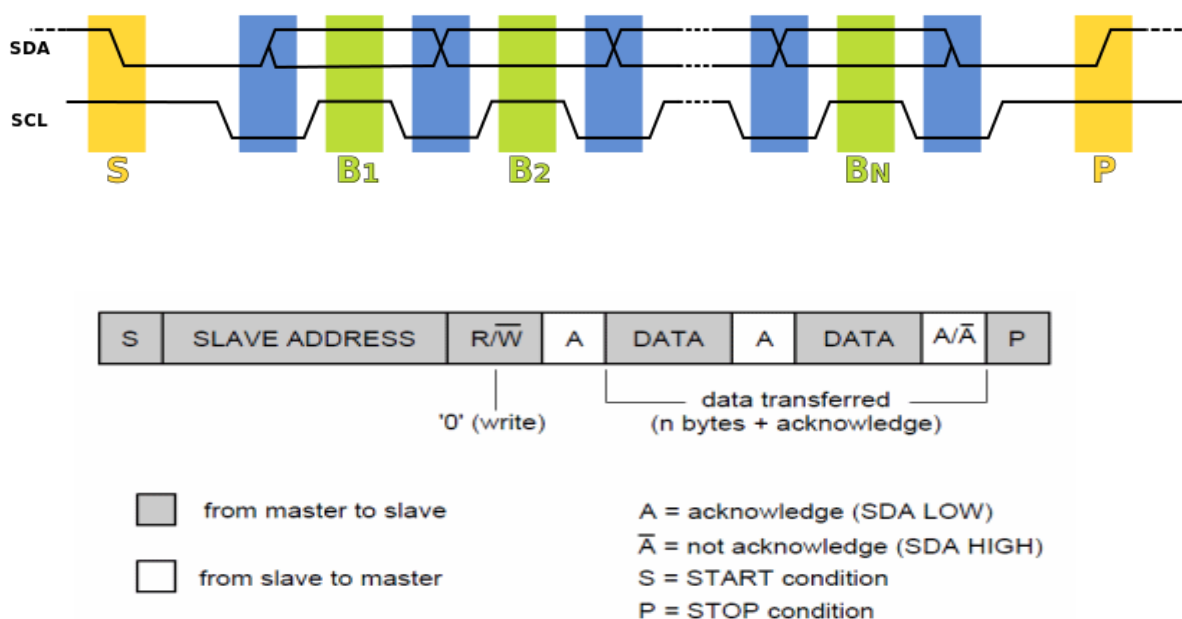
I2C este un protocol de comunicație multi-master sau multi-slave patentat de Philips Semiconductors (NXP Semiconductors) și este destinat comunicației între două sau mai multe microprocesoare, microcontrollere sau a unor dispozitive sau drivere, pe distanțe scurte.

Acest protocol de comunicație este unul sincron, de viteză mică, putând avea o viteză maximă de 400KHz (în cazul utilizării pe microcontroller).



Spre deosebire de protocolul SPI acest protocol este caracterizat doar de două linii digitale SCL(serial clock) și SDA(serial data) ce sunt de tip open-drain, necesitând astfel montarea a câte unui rezistor de pull-up (fixare la VDD) pentru a face posibilă comunicația.

În ceea ce privește modul în care sunt codificate la nivel HW informațiile transmise sau recepționate, figurile de mai jos ilustrează un cadru complet al acestui tip comunicație.



2. Prezentarea driver-ului pentru touch screen STMPE811.

STMPE811 este un controller produs de ST Microelectronics și este destinat pentru ecranele tactile resistive cu 4 linii analogice. Acesta detine două magistrale de comunicație SPI și I2C, un ADC configurabil de 10 – 12 biți, un sensor de temperatură și un buffer de 128 de cuvinte pe 32 de biți.

În cadrul laboratorului, interfațarea cu acest controller se va face prin intermediul magistralei de comunicație I2C, folosind driver-ul software atasat laboratorului, importând fișierele STMPE811.c și STMPE811.h.

Descrierea sumară a interfețelor furnizate de driver-ul software:

❖ **void STMPE811_init (void)**

- funcția transmite totii parametrii necesari inițializării driver-ului STMPE811. Această funcție trebuie apelată după ce s-a făcut initializarea I2C.

❖ **void STMPE811_write_byte(unsigned char addr, unsigned char value)**

- Funcția transmite adresa registrului **addr** și valoare **value** ce se va scrie la acea adresă.

❖ **unsigned char STMPE811_read_byte(unsigned char addr)**

- Funcția returnează valoarea octetului citit de la adresa **addr**.

❖ **void STMPE811_write_word(unsigned char addr, uint16_t value)**

- Funcția scrie valoarea **value** pe 16 biți la adresa **addr**.

❖ **uint16_t STMPE811_read_word(unsigned char addr)**

- Funcția returnează o valoare pe 16 biți citită de la adresa **addr**.

❖ **uint16_t STMPE811_read_x()**

- Funcția returnează o valoare pe 12biți a liniei analogice X citită de ADC-ul controller-ului.

❖ **uint16_t STMPE811_read_y()**

- Funcția returnează o valoare pe 12biți a liniei analogice Y citită de ADC-ul controller-ului.

❖ **void STMPE811_get_touch(uint16_t *x_pos, uint16_t *y_pos, unsigned char orientation)**

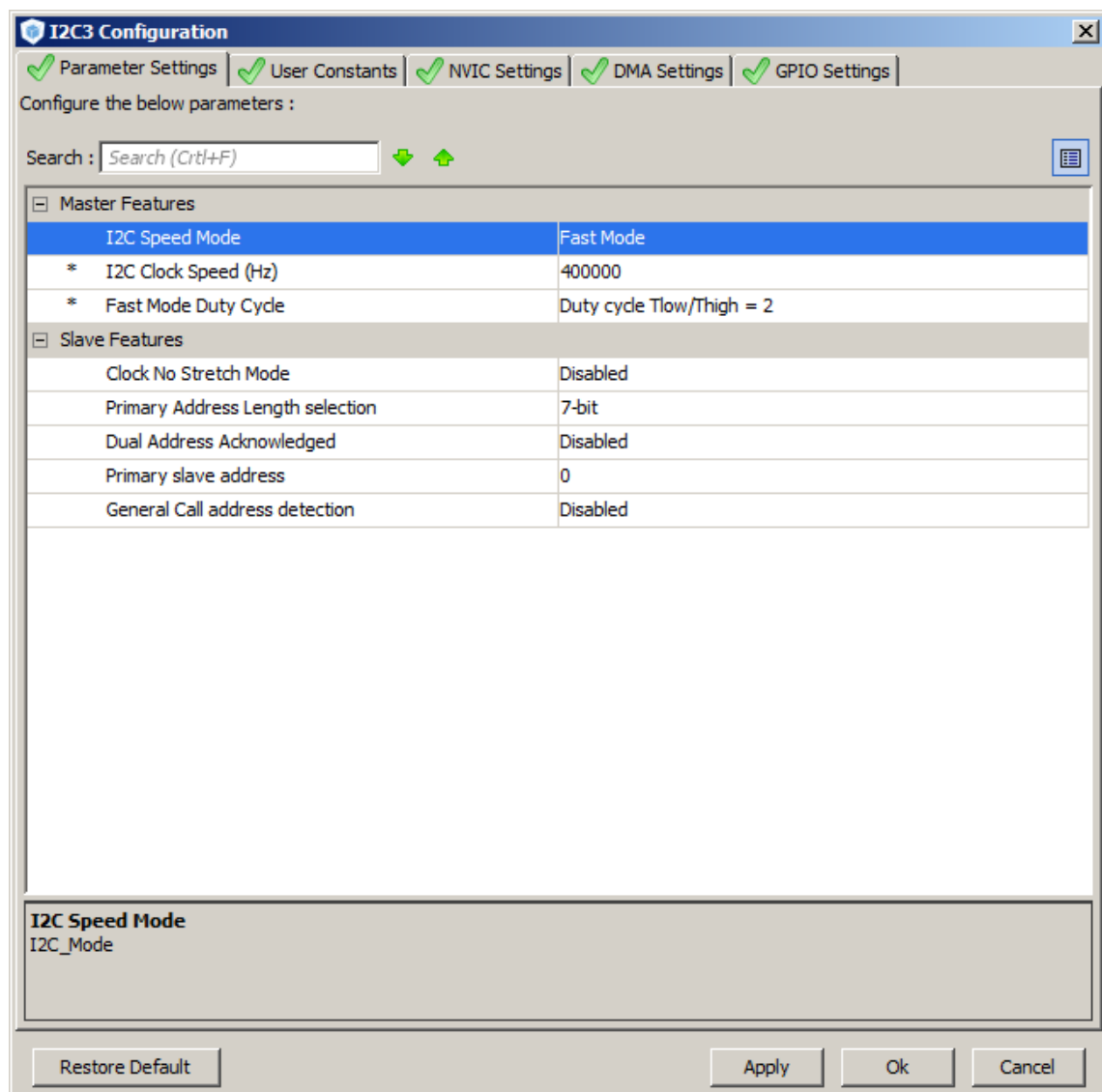
- Funcția furnizează valorile coordonatelor x și y în funcție de orientarea precizată: Landscape sau Portrait.

3. Configurarea I2C pentru STM32F429 folosind STM32CubeMx.

Un prim pas reprezintă selecția perifericului I2C3 la care este conectat controller-ul STMPE811.



Configurarea se va face respectând întocmai parametrii din figura de mai jos:



4. Aplicații:

1. Utilizând interfețele furnizate de driver-ele software (ILI9341 și STMPE811) , afișați în timp real pe LCD, coordonatele x și y citite de la ecranul tactil.
2. Sa se creeze o aplicație ce simulează apăsarea unui buton, folosind funcțiile ecranului tactil.