Iniziare a sviluppare usando STM32-comStick di HITEX

By E.M.

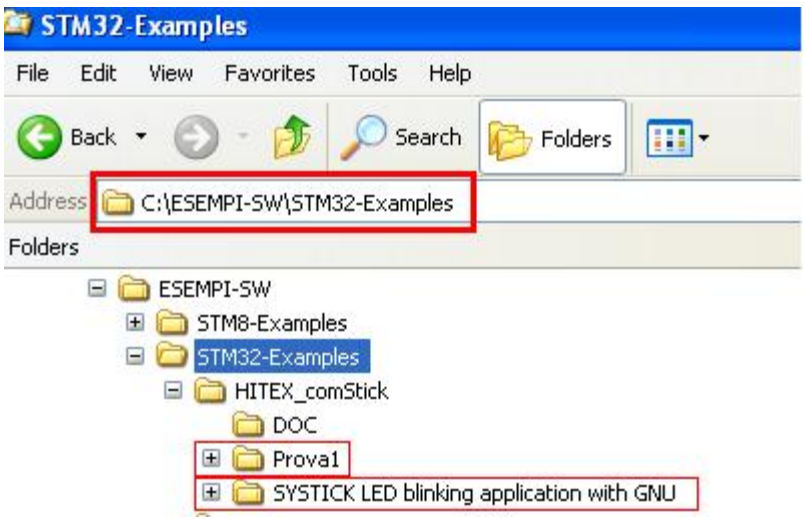
Per sviluppare un nuovo progetto usando il **STM32-comStick** di **HITEX**, la cosa più semplice da fare è partire da uno degli esempi forniti da Hitex per poi modificarlo per adeguarlo alle nostre necessità.
Per capire come fare, di seguito, c'è un esempio sviluppato passo per passo.

SUGGERIMENTI:

Vi consigliamo di guardare la video guida:
HiTOP *Universal user interface for all Hitex test and analysis tools*
http://www.hitex.com/index.php?id=551
Che spiega in modo sintetico ma chiaro le principali funzionalità dell'IDE di HITEX.

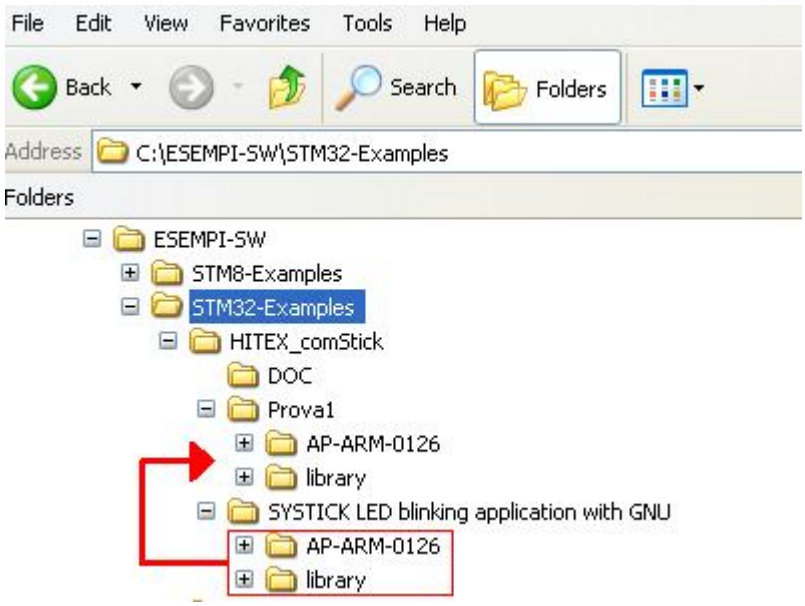
Partiamo con i seguenti presupposti:

1) Svilupperemo usando il **STM32-comStick** di **HITEX**
2) Useremo le **librerie di STM ver.3.1.0** che potete prendere all'indirizzo qui sotto:
   http://www.st.com/mcu/familiesdocs-110.html#Firmware
3) Useremo l'esempio di partenza di HITEX chiamato:
   SYSTICK LED blinking application with GNU
   che si puo prendere all'indirizzo qui sotto riportato:
   http://www.hitex.com/index.php?id=1676
4) La directory di lavoro sarà:
   Prova1
5) **Creiamo una struttura di directory come sotto** evidenziato in rosso



STEP n.1

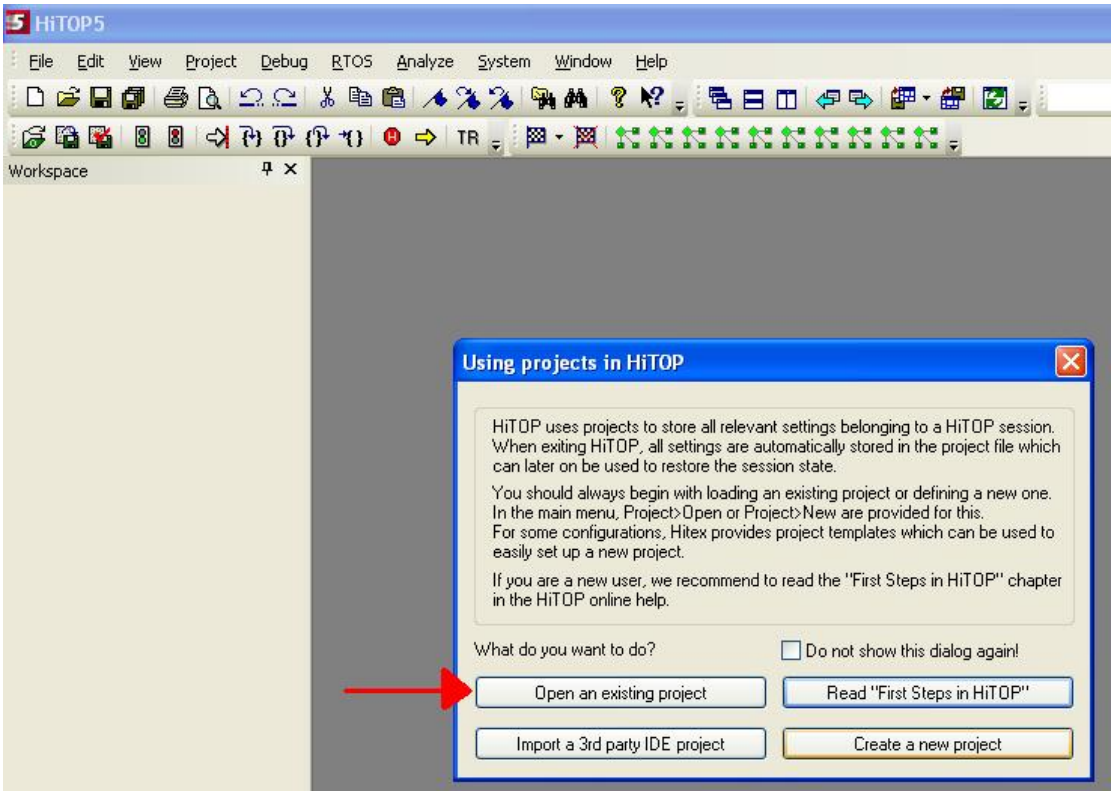
Copiamo le directory che si trovano sotto la cartella:
C:\ESEMPI-SW\STM32-Examples\HITEX_comStick\SYSTICK LED blinking application with GNU
Nella nostra cartella di lavoro:
Prova1
Vedere la figura sotto che è esplicativa di quanto dobbiamo fare.



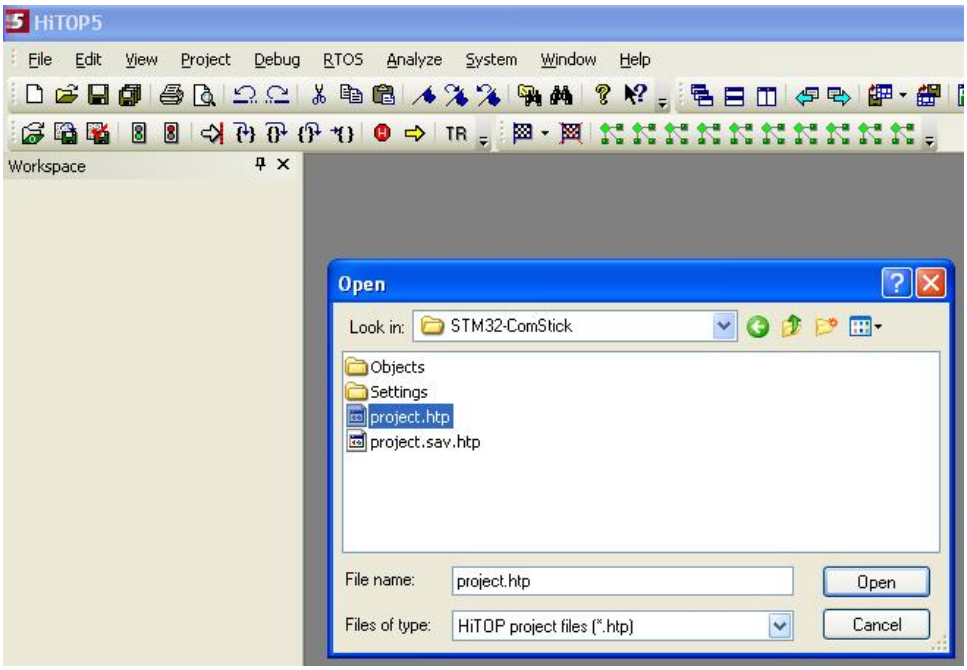
STEP n.2

Mandiamo in esecuzione **HiTOP53-STM32-comStick** la cui icona è sotto riportata.

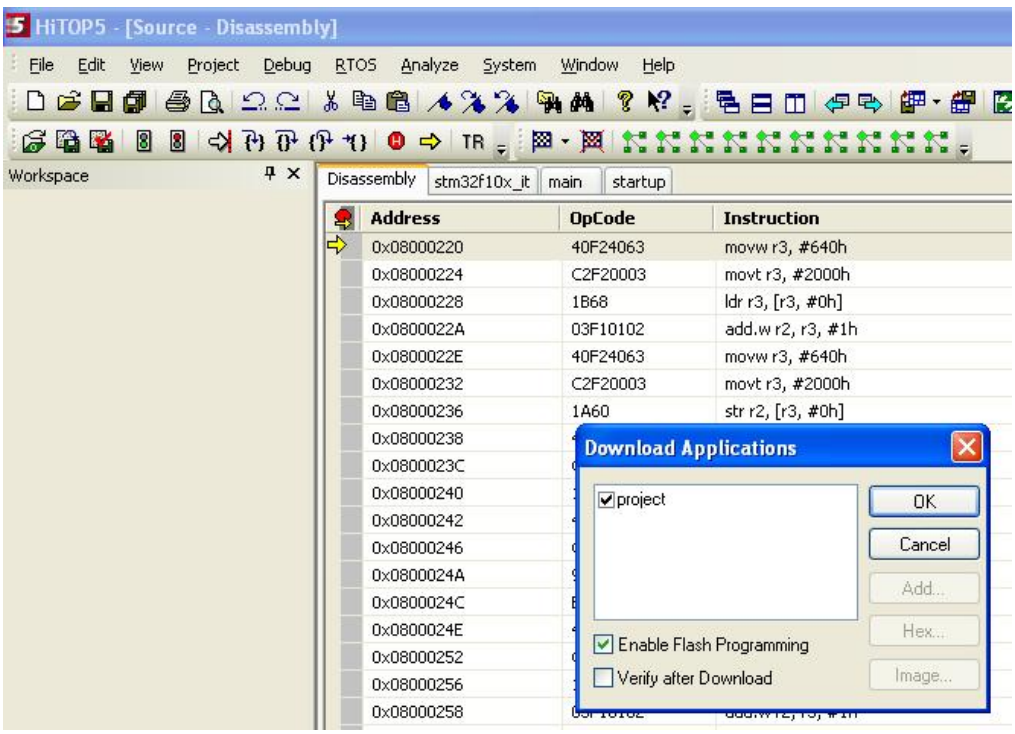
La pagina che vi deve comparire è sotto riportata.
Da questa pagina premete: **Open an existing project**



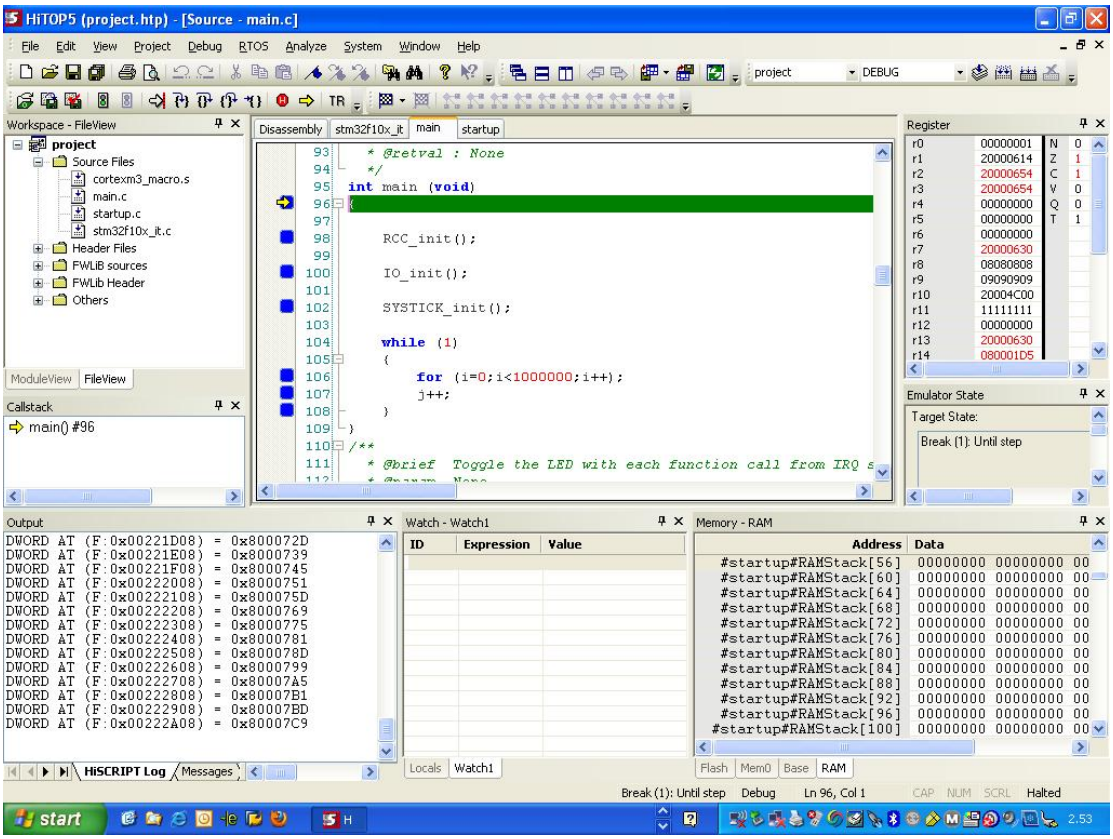
Nel menù che compare spostiamo nella directory:
C:\ESEMPI-SW\STM32-Examples\HITEX_comStick\Prova1\AP-ARM-0126\HiTOP\STM32-ComStick
e apriamo il file:
project.htp
e poi premete **Open** (guardate la figura sotto).



A questo punto vi deve comparire la pagina sotto riportata dove, dovete premere **OK**.
ATTENZIONE: dovete aver collegato al PC, via porta USB, il STM32-comStick.



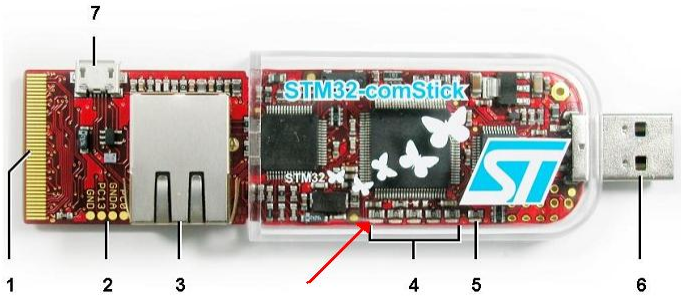
Se tutto è andato bene dovrete avere una pagina simile a quella sotto riportata.



STEP n.3

Per verificare che tutto sia OK mandate in esecuzione il programma premendo sull'icona  .

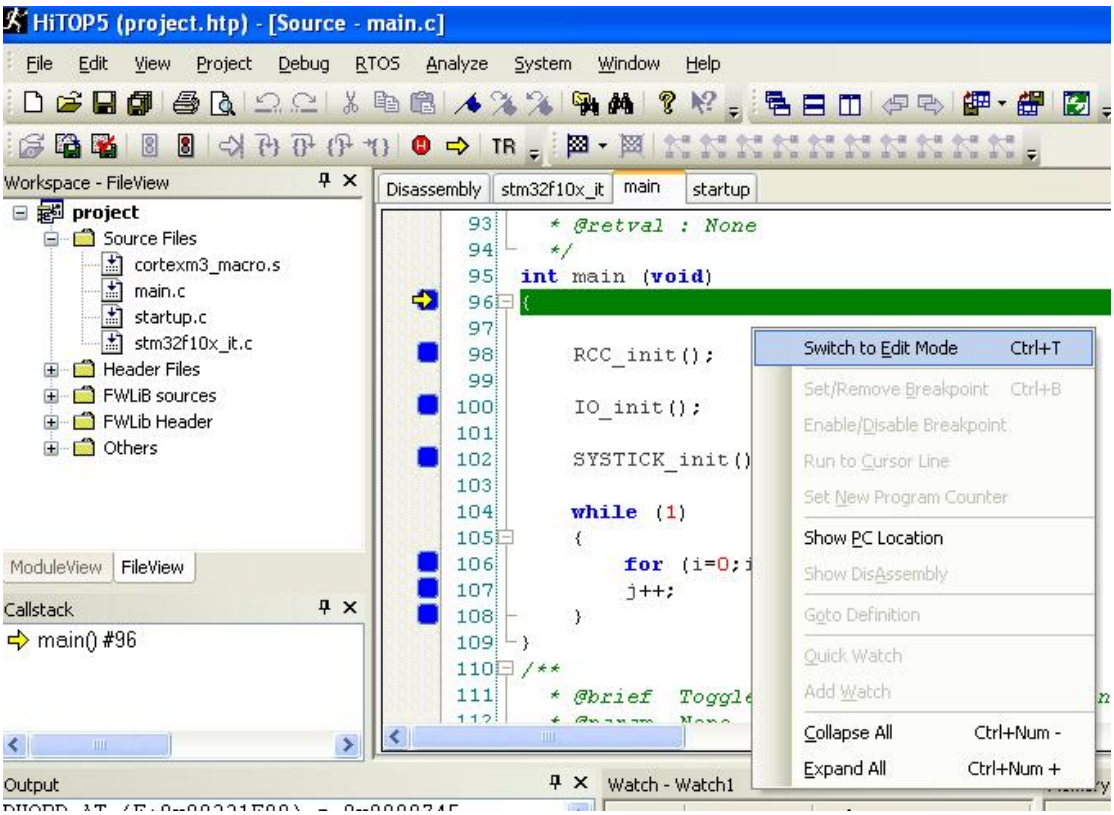
Se è tutto OK, con il programma in esecuzione, dovrete vedere lampeggiare il primo LED arancione sulla sinistra (vedere figura sotto, freccia rossa).



Bene, il LED lampeggia e allora andiamo a modificare il programma per adattarlo alle nostre esigenze che sono:

- **Usare esclusivamente le librerie di STM**
- **Far lampeggiare tutti i LED**

Per apportare modifiche o visualizzare variabili si deve premere sull'icona  che fermerà l'esecuzione del programma.
Per entrare in **editor** dovrete premere il **tasto destro del mouse**, quando vi troviamo all'interno della pagina che visualizza il codice in C e **selezionare Switch to Edit Mode** (vedere figura sotto riportata).



STEP n.4

Il programma scritto da HITEX fondamentalmente svolge le seguenti funzioni:

- Configura la MCU
- Configura gli I/O (GPIO)
- Configura i CLOCK (RCC)
- Configura il SYSTICK
- Configura gli INTERRUPT (NVIC)

Il led lampeggia allo scadere del SYSTICK ed è ovviamente gestito in Interrupt.

I files che realizzano ciò sono:

main.c	Setup routines and main loops
tm32f10x_it.c	Interrupt vectors
stm32f10x_lib.c	Library definition module
stm32f10x_gpio.c	Library for GPIO module
stm32f10x_rcc.c	Library for RCC module
stm32f10x_systick.c	Library for SYSTICK module
stm32f10x_nvic.c	Library for NVIC module

Ci sono altri files importanti al fine della configurazione corretta della MCU che sono:

main.h	General inclusions
stm32f10x_conf.h	Library configuration file
stm32f10x_lib.h	Library inclusions file
stm32f10x_type.h	Definitions and types
stm32f10x_it.h	Interrupt vector pre-declarations

In particolare vi evidenziamo il file **stm32f10x_conf.h** che serve per abilitare o disabilitare le periferiche e per impostare il clock macchina che verrà usato dal SW per calcolare per esempio i valori da porre nei registri delle USART in modo da ottenere i BaudRate richiesti.

Maggiori dettagli sui files sopra menzionati li potete trovare nella Application Example AE-CORTEX-0102.pdf
http://www.hitex-download.de/examples/st/stm32-comstick/AE-CORTEX-0101.pdf)
che avete scaricato dal sito Hitex quando avete preso l'esempio che stiamo usando.

Per poter modificare il programma di HITEX occore sapere dove sono collegati i LED e per questo motivo aprite il data sheet del STM32-comStick che si trova nella cartella qui sotto riportata:
C:\Program Files\Hitex\HiTOP53-STM32-comStick\STM32-ComStickView\Doc
E che si chiama:

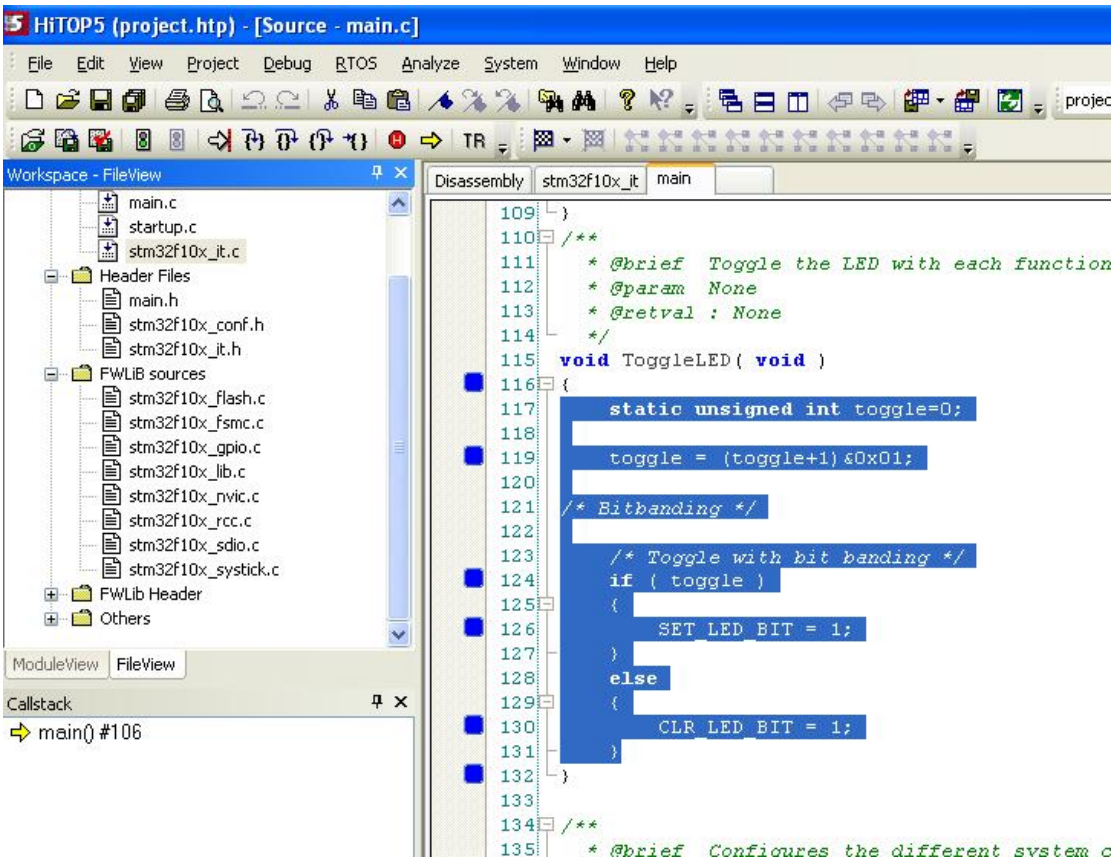
stm32-io-board-ds.pdf

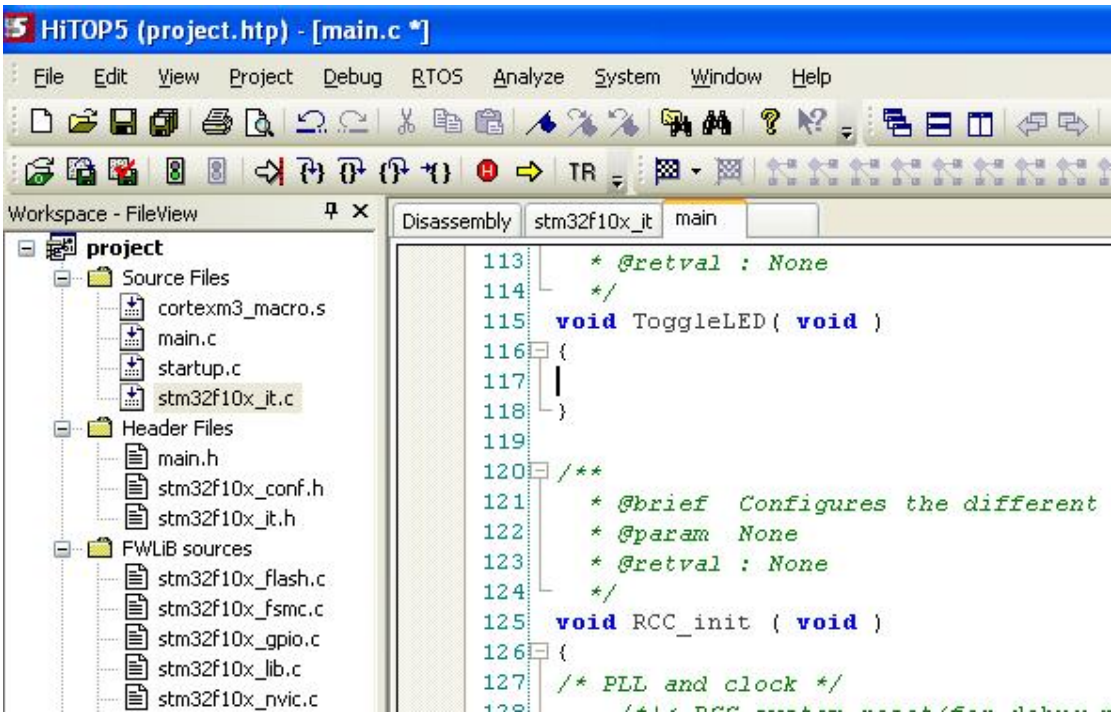
Dal manuale del STM32-comStick scopriamo che i LED sono collegati come qui sotto riportato:

LEDs					
	V507	V506	V505	V504	V503
Port	PB1	PB0	PB9	PE15	PB5

Set the dsired port to 'high' in order to light up the corresponding LED.

Aprite il **main.c**, individuate la funzione **void ToggleLED(void)** e cancellate il contenuto come sotto evidenziato.





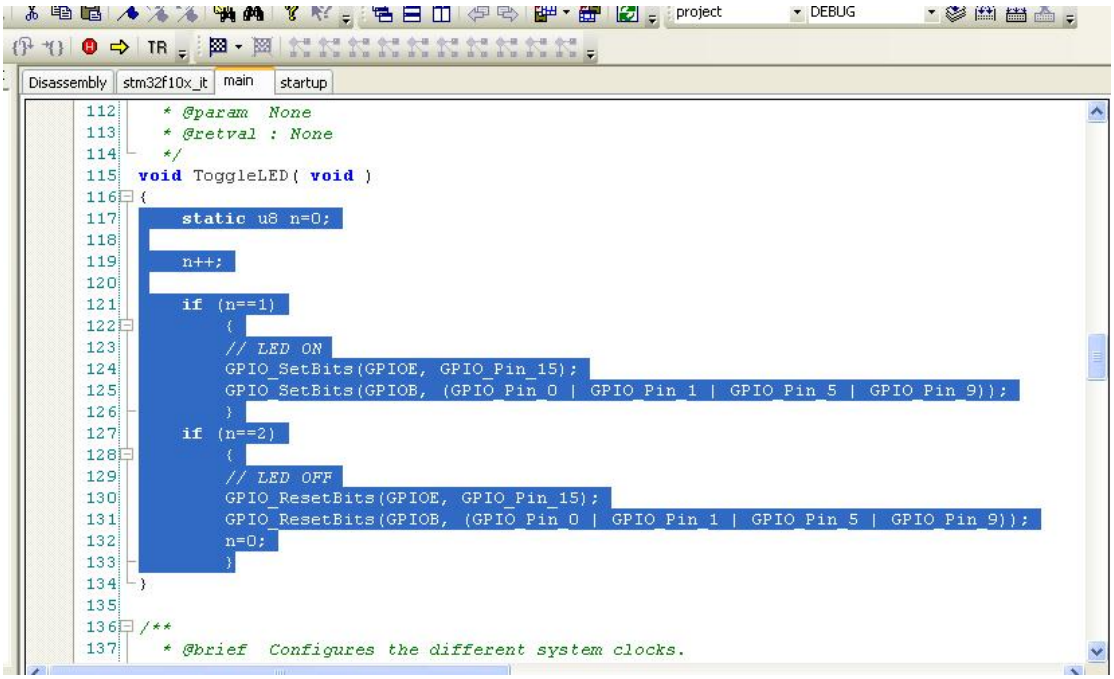
Il contenuto da scrivere all'interno di **void ToggleLED(void)** sarà:

```
static u8 n=0;

n++;

if (n==1)
        {
        // LED ON
        GPIO_SetBits(GPIOE, GPIO_Pin_15);
        GPIO_SetBits(GPIOB, (GPIO_Pin_0 | GPIO_Pin_1 | GPIO_Pin_5 | GPIO_Pin_9));
        }
if (n==2)
        {
        // LED OFF
        GPIO_ResetBits(GPIOE, GPIO_Pin_15);
        GPIO_ResetBits(GPIOB, (GPIO_Pin_0 | GPIO_Pin_1 | GPIO_Pin_5 | GPIO_Pin_9));
        n=0;
        }
```

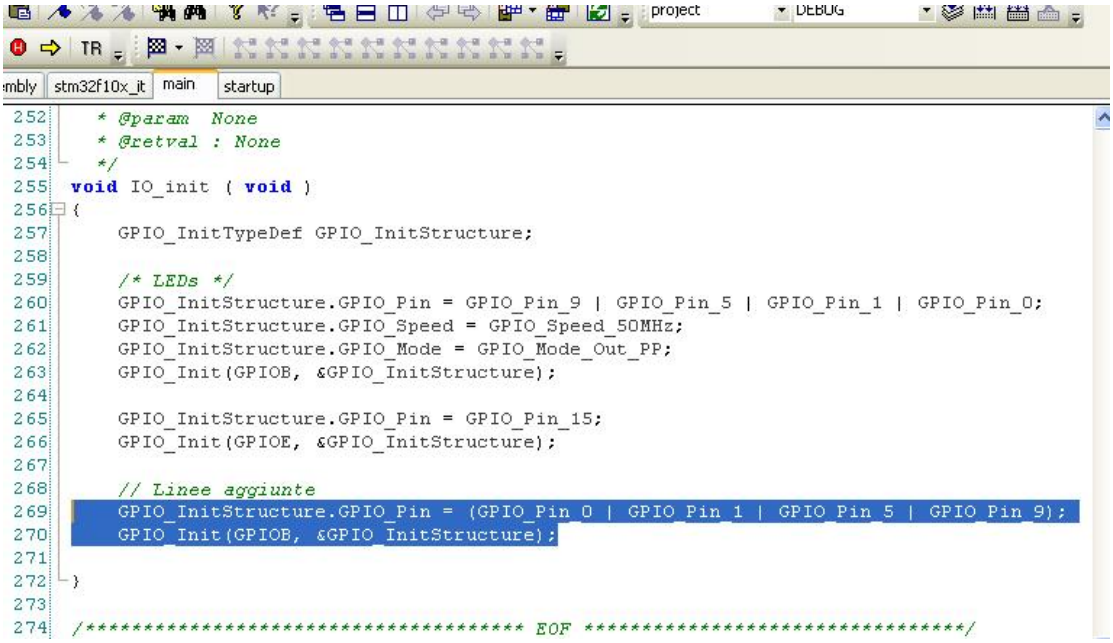
Come sotto evidenziato.



Adesso individuate la funzione **void IO_init (void)** e aggiungete in fondo le linee sotto:

```
GPIO_InitStructure.GPIO_Pin = (GPIO_Pin_0 | GPIO_Pin_1 | GPIO_Pin_5 | GPIO_Pin_9);
GPIO_Init(GPIOB, &GPIO_InitStructure);
```

guardate la figura sotto.



Adesso non ci resta che compilare il programma premendo sull'icona  e se abbiamo fatto tutto bene comparirà la videata qui sotto riportata.
Premete **OK** e mandate in esecuzione il programma, tutti i LED dovranno lampeggiare.