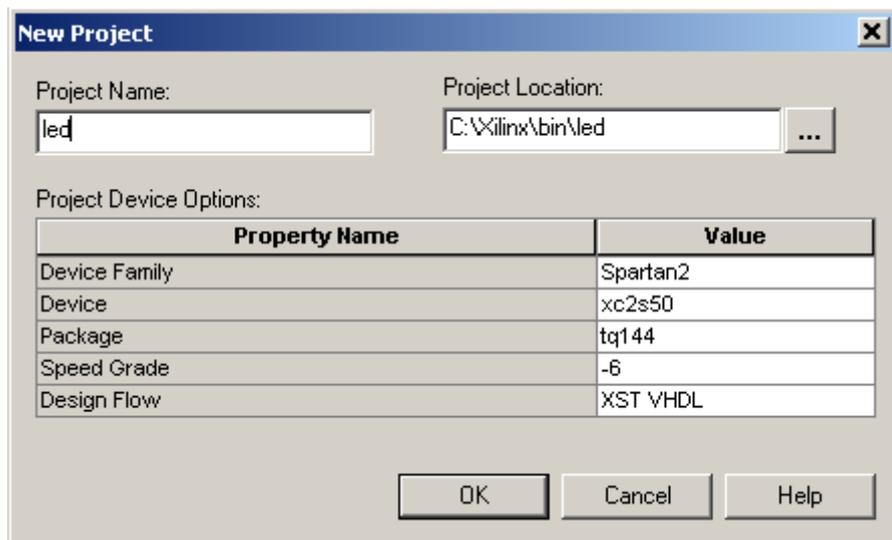


Esercitazione n° 1

Questa esercitazione guidata ha lo scopo di mostrare come creare un semplice progetto per pilotare l'accensione e lo spegnimento di un led attraverso uno degli switch della XSA board. Per la realizzazione del progetto si prevede l'utilizzo dell'editor schematico ECS.

- Avviare il Project Navigator attraverso l'icona  .
- È ora necessario creare un nuovo progetto: File → New Project.
Comparirà una finestra che dovrà riportare i seguenti parametri nelle Project Device Options:



Property Name	Value
Device Family	Spartan2
Device	xc2s50
Package	tq144
Speed Grade	-6
Design Flow	XST VHDL

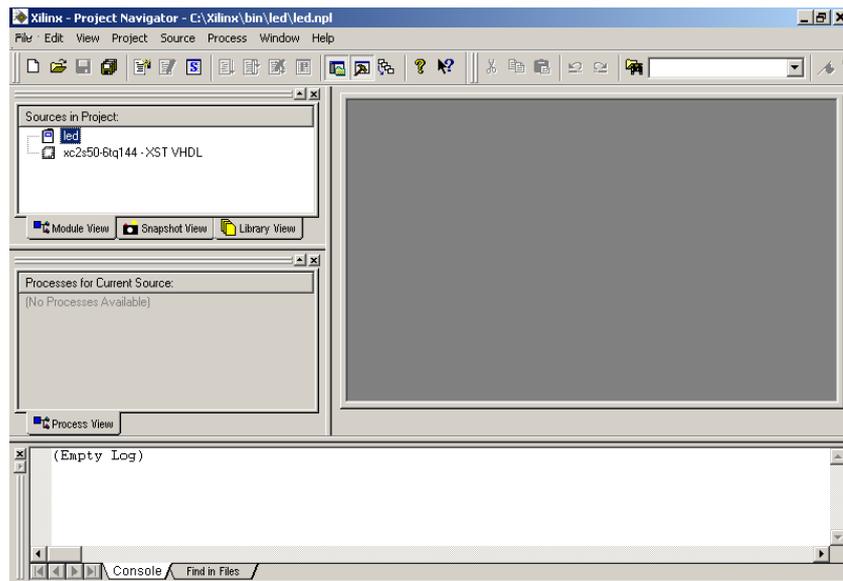
Nelle 'Project Device Options' sono contenute le caratteristiche del dispositivo che si andrà a programmare. Nel nostro caso il dispositivo appartiene alla famiglia Spartan 2, in particolare si tratta del dispositivo xc2s50 che si trova all'interno di un package che ha codice tq144ams0201 (entrambe le informazioni sono leggibili sul package dell'FPGA). Lo Speed Grade è un dato relativo alla velocità del dispositivo, mentre in corrispondenza del Design Flow va specificato il linguaggio dei tool di sviluppo che verranno utilizzati nella fase di editing e di sintesi.

La directory che compare in Project Location sarà quella in cui verrà creata la sottodirectory con il nome del progetto inserito nella casella Project Name. Nel nostro esempio scegliamo il project name 'led'

Una volta inseriti tutti i parametri necessari premere

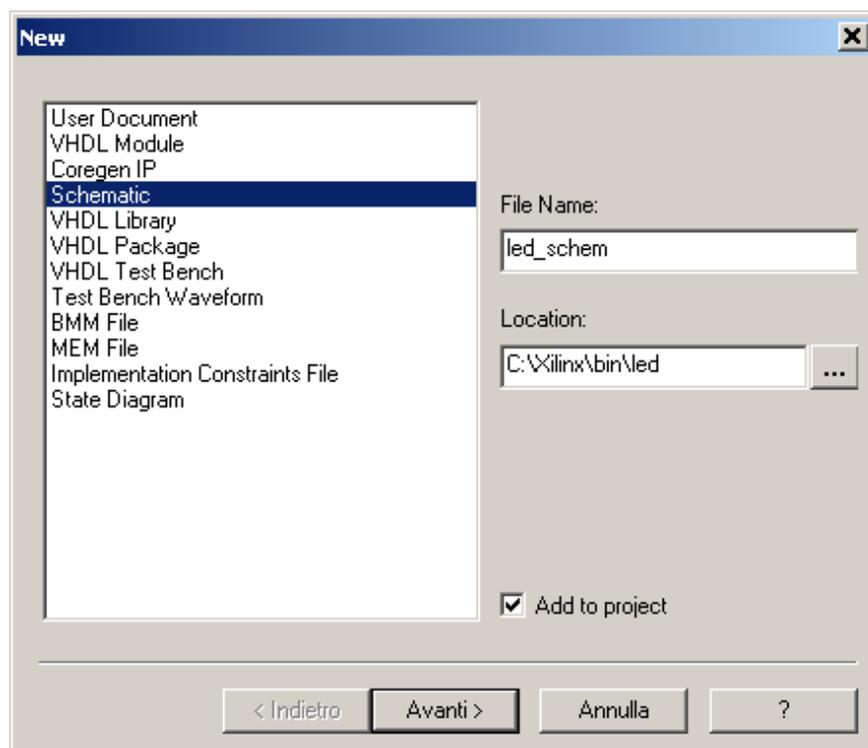


Ci troviamo nella seguente condizione:



Si tratta quindi di creare i file sorgente del progetto, in particolare, come precedentemente detto, lo schema logico del circuito e il file contenente i vincoli (constraints) di progetto.

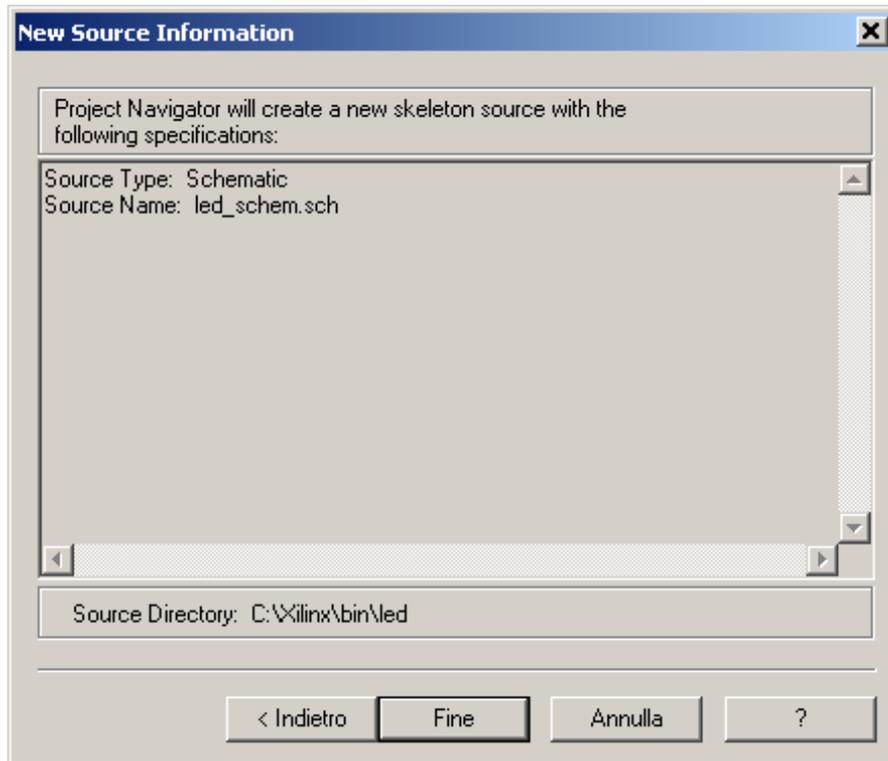
- Per aggiungere il sorgente schematico: Project → New Source



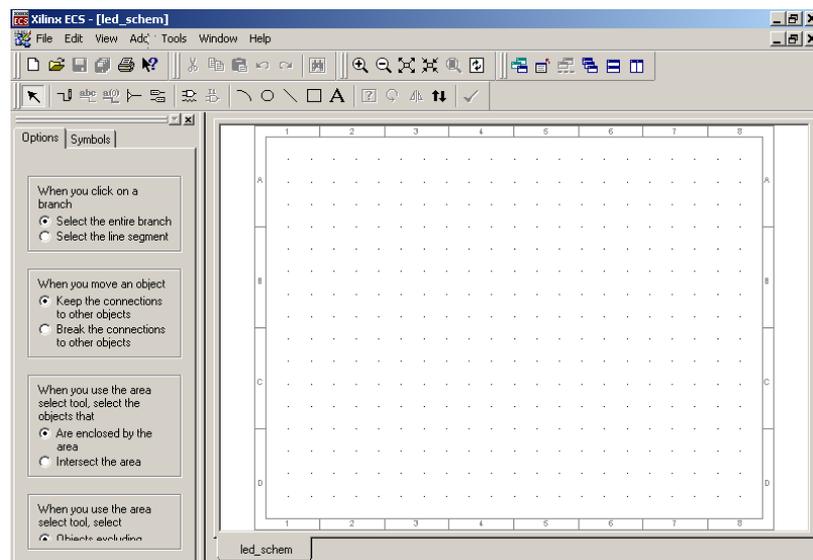
Clickare “Schematic” come nella figura e scegliere un nome per il file schematico. Poi



Comparirà la seguente schermata

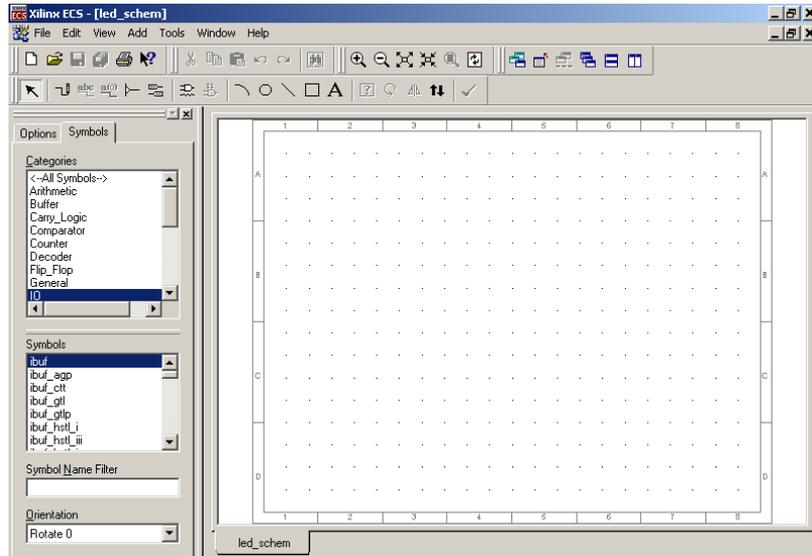


Clickare .
Si aprirà la finestra dell'ECS:

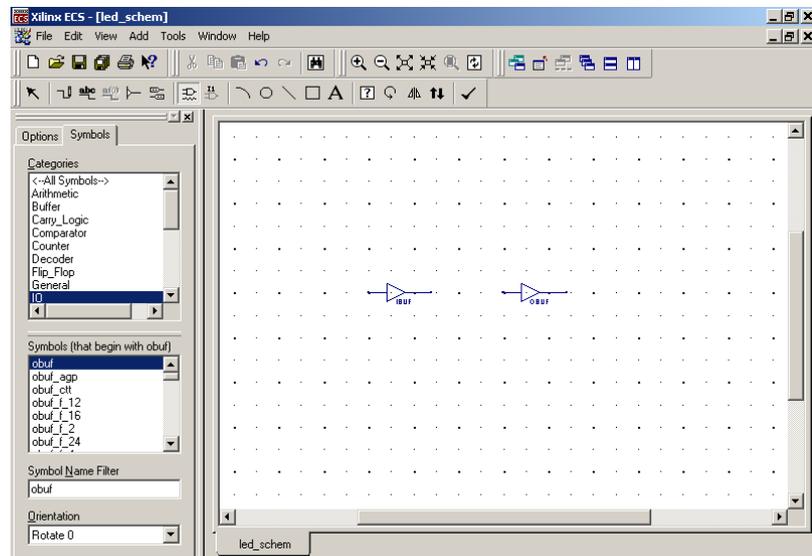


A questo punto bisogna creare lo schema.

- Saranno necessari due buffer, uno in entrata (ibuf) e uno in uscita (obuf), in corrispondenza dei quali si troveranno un I/O Marker di entrata e uno di uscita rispettivamente. Nella successiva figura è mostrato come selezionare l'elemento ibuf :

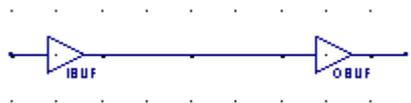


Analogamente per il buffer in uscita (obuf):



Ora bisogna creare la logica tra l'ingresso e l'uscita. Nel caso del led si tratta semplicemente di collegare con un filo lo switch (input) con il led (output).

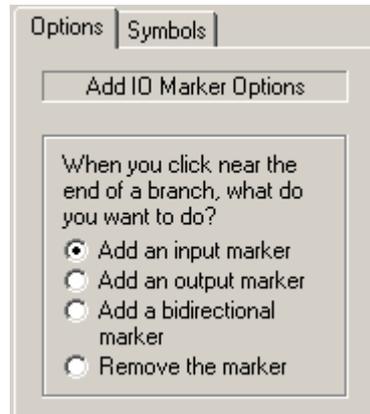
Per fare ciò, clickare su  e collegare le due estremità di ibuf e obuf:



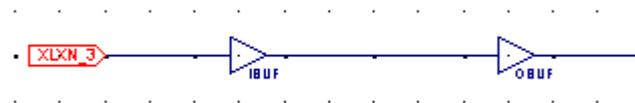
Ora è necessario aggiungere gli I/O Marker, in ingresso e in uscita. Gli I/O Marker non possono essere collegati direttamente a un buffer: bisogna che ci sia del filo tra i buffer e gli I/O Marker. Aggiungiamo pertanto un trattino di filo a ciascuna estremità dei buffer:



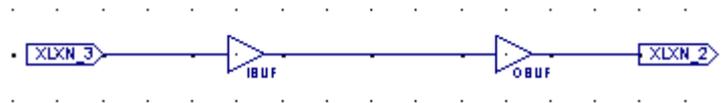
Ora è possibile aggiungere gli I/O Marker selezionando . Quando si aggiunge l'I/O Marker in input, fare attenzione che nelle I/O Marker Options sia puntata la casella "Add an input Marker":



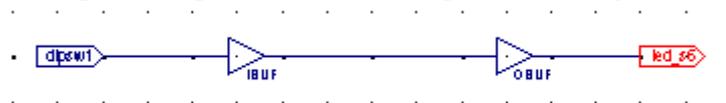
successivamente clickare sul puntino rosso all'estremità "input" dello schema; si avrà:



Analogamente selezionando dalle IO Marker Options "Add an output marker" e clickando sul puntino rosso all'estremità output dello schema. Si ottiene quanto segue:



È quindi possibile modificare i nomi di ciascuno degli elementi dello schema. Ad esempio facendo doppio-click su gli I/O Marker cambiamo il nome per facilitare la lettura. In particolare utilizziamo, per esempio, il nome dei dispositivi collegati con i marker:



dove con dipsw1 chiamo appunto il DIP Switch 1 della XSA Board e con led_s6 il led che voglio far accendere sul display 7 segmenti (vedi XSA Board User Manual a pag. 24)

Salviamo il file: File → Save.

Ora è necessario avere chiaro lo schema circuitale della XSA board.

- Con l'aiuto del manuale utente, a pag. 25 constatiamo quanto segue:

Seven-Segment LED

The XSA Board has a 7-segment LED digit for use by the FPGA or the CPLD. The segments of this LED are active-high meaning that a segment will glow when a logic-high is applied to it.

The LED shares the same pins as the eight bits of the Flash RAM data bus.

Four-Position DIP Switch

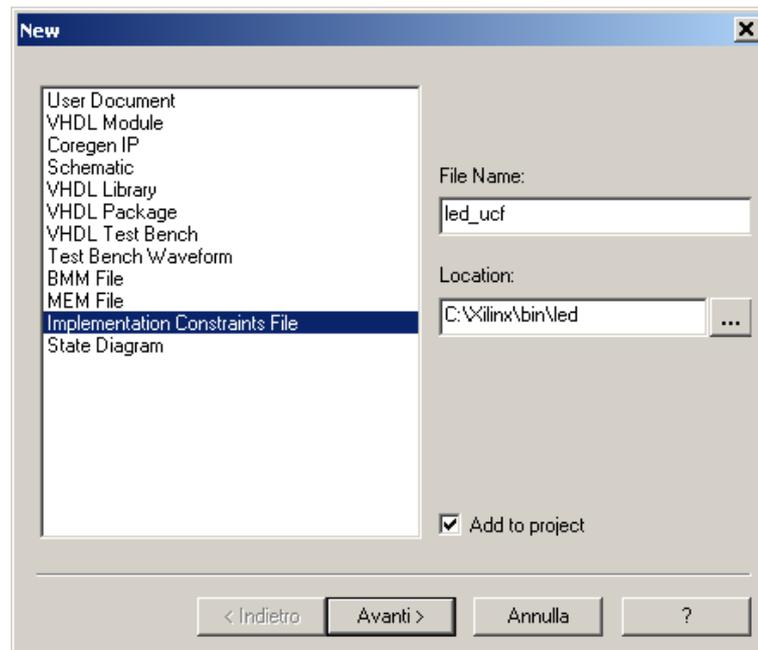
The XSV Board has a bank of four DIP switches accessible from the CPLD and FPGA. When closed or ON, each switch pulls the connected pin of the FPGA and CPLD to ground. Otherwise, the pin is pulled high through a resistor when the switch is open or OFF.

When not being used, the DIP switches should be left in the open or OFF configuration so the pins of the FPGA and CPLD are not tied to ground and can freely move between logic low and high levels.

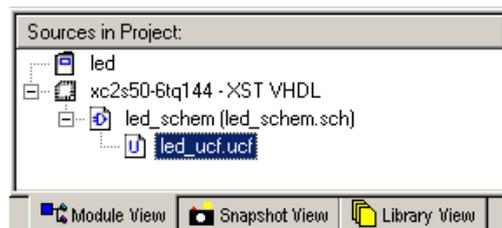
The DIP switches also share the same pins as the uppermost four bits of the Flash RAM address bus. If the Flash RAM is programmed with several FPGA bitstreams, then the DIP switches can be used to select a particular bitstreams which will be loaded into the FPGA by the CPLD on power-up.

Ciò significa che se si volesse rispettare l' ON/OFF del DIP Switch 1 scelto per attivare il led, si dovrebbe introdurre nello schema un invertitore fra i due buffer ibuf e obuf. Tuttavia, per esigenze di chiarezza, non si effettuerà la modifica poiché dal punto di vista qualitativo il funzionamento del progetto è in ogni caso garantito.

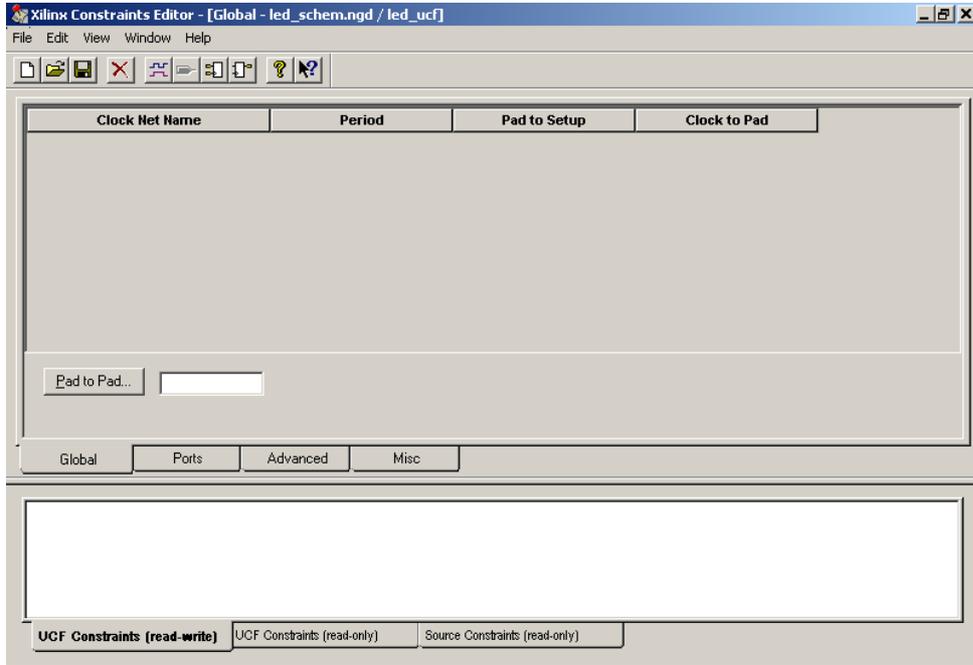
- Passiamo ora alla definizione dei vincoli, che per ora sarà molto semplice. Si tratta solo di assegnare a ciascuno degli I/O Marker il pin corrispondente sulla XSA board. Per fare ciò si ritorni alla schermata del Project Navigator, poi Project → New Source; assegnare un nome al nuovo file sorgente:



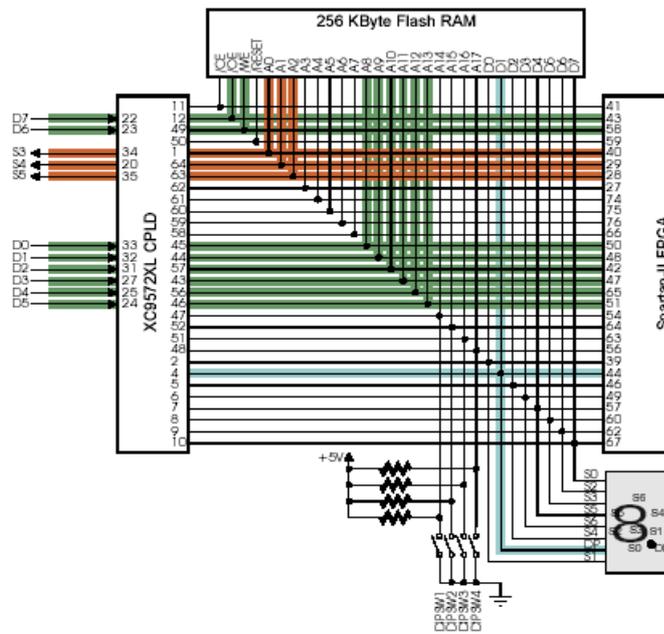
poi ,  e . Fare quindi doppio-click su led_ucf.ucf nel menu a tendina Module View posto in alto a sinistra:



comparirà la seguente schermata:

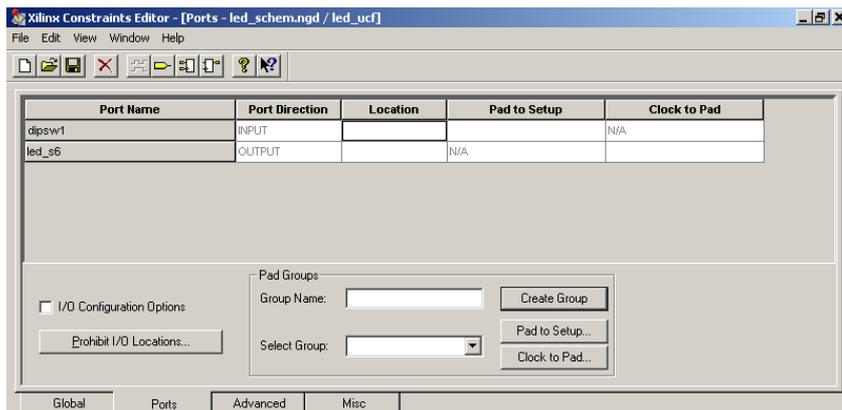


Dal manuale utente della XSA board, a pag. 28 ricaviamo:

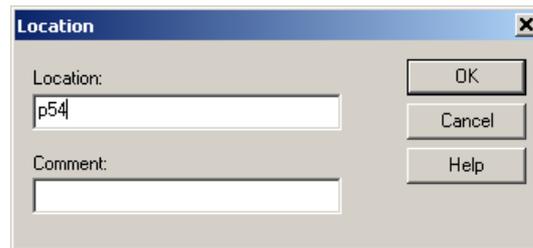


da cui si vede che il led s6 del display 7 segmenti è collegato alla posizione 49 sull’FPGA, e il dipsw1 alla posizione 54.

Selezionare il menu a tendina “Ports” e fare doppio-click sulla casella bianca in coordinate “dipsw1, Location”:



Nell'inserire la posizione nella location, il numero va preceduto da una 'p', cioè per il dipsw1:



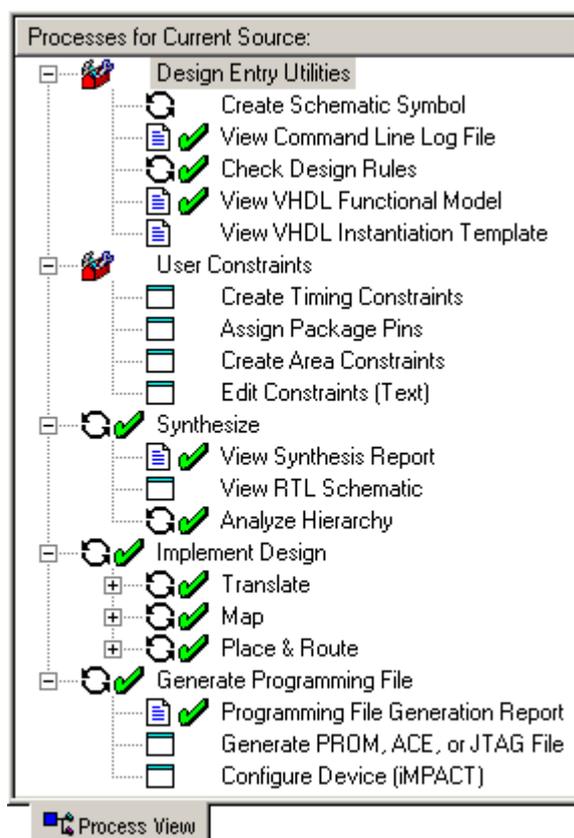
Analogamente per il led_s6 facendo doppio-click sulla casella in coordinate "led_s6, **Location**", si inserisca 'p49'.

Si osservi che nel compiere queste operazioni, nella sezione **UCF Constraints (read-write)** del menu a tendina a fondo pagina, compare un testo che è esattamente quello del file di testo led_ucf.ucf.

A questo punto salviamo (file → Salva) e chiudiamo l'editor.

Il progetto è ora completo.

- È possibile verificare che siano state rispettate le regole per il disegno del progetto (doppio-click su Check Design Rules), operare la sintesi (doppio-click su Synthesize), implementare il disegno (doppio-click su Implement Design) e generare il file che andrà poi caricato sulla XSA board (doppio-click su Generate Programming File):

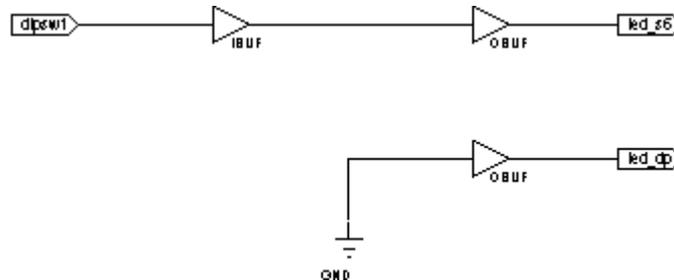


Verrà generato un file dal nome 'led_schem.bit'; esso dovrà essere caricato nella Board secondo la procedura standard.

- **Attenzione:** una volta caricato il programma, oltre al led s6 (pilotato con il dipsw1) si accenderà anche il led dp del display a 7 segmenti. Quest'ultimo rimarrà sempre acceso (anche nei successivi progetti) e non costituirà un problema ai fini dell'esercitazione.

ATTENZIONE

Tuttavia se si desiderasse tenerlo spento sarà sufficiente modificare lo schematico e il file 'led_ucf.ucf'. Lo schematico andrà modificato, a partire dalla finestra dell'ECS, aggiungendo un collegamento a terra (Symbols → General → gnd), cui si collegherà – come già fatto precedentemente – prima un Output Buffer e poi un Output Marker (che chiameremo 'led_dp'):



Si Salvi quindi il file.

Si ritorni alla finestra del Project Navigator e si proceda con un doppio click sul file 'led_ucf.ucf' nella finestra 'Module View'. Si aprirà il 'Constraints Editor' che, nel menu a tendina 'Ports' andrà compilato, come segue:

Port Name	Port Direction	Location	Pad to Setup	Clock to Pad
dipsw1	INPUT	p54		N/A
led_dp	OUTPUT	p44	N/A	
led_s6	OUTPUT	p49	N/A	

Si salvi il file e si ritorni al Project Navigator.

A questo punto sarà necessario riavviare l'implementazione del disegno e la generazione del file di programmazione facendo doppio click rispettivamente su 'Implement Design' e 'Generate Programming File' dal menu 'Process View'.

**Il procedimento appena descritto crea un conflitto tra FPGA e CPLD che può portare al danneggiamento dei dispositivi !:
(vedi XSA Board V1.2, User Manual – pp.28)**

Il punto decimale sul display a sette segmenti viene utilizzato dalla CPLD per indicare che l'FPGA e' stata correttamente programmata. Pertanto per "spegnere" il LED non si deve FORZARE a massa il piedino 44, ma piuttosto fare si che la CPLD "rilasci" (metta in alta impedenza) il piedino 4 (connesso al LED).

Per fare questo bisogna piuttosto abbassare il segnale condiviso dalla linea \overline{CE} della Flash Memory (piedino 41 dell'FPGA – piedino 11 della CPLD)