**Hanoi University of Science and Technology**



**ISTITUTO DI ELETTRONICA - TELECOMUNICAZIONI**

**ESRC LAB**

**Topic:**

**GIOCO DEL SERPENTE su Kit DE1**

**Gruppi di studenti:**

**Gruppo 1 - K53:**

**Vu Quang Trong**

**Do Son Tung**

Hanoi 8/2011

1. introdurre
   1. filo

Dopo il completamento della pratica di laboratorio su Kit Altera DE1, continuiamo a sviluppare capacità progettuali e applicate nella pratica, che sta sviluppando un sistema completo su Kit DE1 con il tema:

**"Utilizzando il kit Altera DE1 per creare un gioco Snake per i giocatori con un'interfaccia grafica, comunicare con i giocatori tramite tastiera PS2 e VGA".**

* 1. Membri e posti di lavoro assegnare

|  |  |
| --- | --- |
| Immagine andate qui | Immagine andate qui |
| Vu Quang Trong | Do Son Tung |
| (Leader) |  |
| 0973.750.337 | 0168.9.929.537 |
| [vuquangtrong@gmail.com](mailto:vuquangtrong@gmail.com) | [tungmontaint@gmail.com](mailto:tungmontaint@gmail.com) |
| Mappatura del tema generale.  Logico stato del sistema di blocco.  blocco di controllo solido.  blocchi di display grafico.  blocchi di testo visualizzato.  E blocchi altri accessori necessari. | Imparare la connessione PS2.  Imparare e driver VGA.  Imparare IC AudioCodec e controllare il volume del suono. |

* 1. Requisiti di argomento
     1. Requisiti funzionali
* hardware
  + Il gioco gira interamente su Kit DE1
  + Prendi il controllo della tastiera PS2
  + Mostra sulla risoluzione dello schermo VGA di 640 x 480
  + Suono attraverso un altoparlante 2.0 con WM8731 codec audio IC con manioca sul Kit DE1
* software
  + Quartus II 9.1 SP2 utilizzando
  + Pure linguaggio VHDL
  + Il gioco ha tutti gli elementi per diventare un gioco completo:
    - Il giocatore controlla serpente con 4 pulsanti direzionali della tastiera.
    - La tasti funzionano come PAUSE, diga giocatori menu di selezione con le azioni di gioco.
    - Lungo serpente durante l'alimentazione, o hanno altre interazioni seconda esche gusto.
    - Comandamenti moriranno se il serpente ha colpito le pareti o la coda, e si avrà un nuovo thread di riproduzione solidi.
    - Condizioni per giocare di nuovo se e solo se il tuo punteggio riproduce maggiore di 0, l'originale, i giochi saranno 3, può aumentare se le esche alimentari LIVE\_UP, e diminuisce come è morto il serpente. Se il numero di riproduzioni la schiena è 0, il gioco finirà.
    - I diversi livelli di difficoltà, che è la velocità del serpente crescerà più di un certo numero di volte Predator, seguita dalla schermata di gioco con diversi ostacoli.
    - Il punteggio per il gioco, il punteggio viene calcolato al livello attuale e il tipo di esca che i serpenti mangiano.
    - Salva nome di giocatori quando i punti giocatore per guadagnare qualche punto nella top 5
    1. requisiti non funzionali
* Tempo di risposta meno sequenze di tasti per aiutare i giocatori, quando la navigazione di controllo facilità, premere velocità di circa 4 volte / sec => 250 ms di tempo di risposta
* Visualizzato sullo schermo 640x480 VGA con 8 colori base, utilizzando 3 bit per colore.
* Frequenza Lo schermo è abbastanza grande per garantire la visualizzazione regolare, prendiamo è la frequenza di scansione 60Hz.
* leggi di controllo vengono applicate nel modo seguente:
  + I serpenti non possono tornare indietro nella direzione del viaggio, vale a dire se si è in movimento all'indietro premendo i pulsanti non funzionano, quando i denti sono simili a sinistra, a destra, o verso il basso.
* Telaio di giocatori:

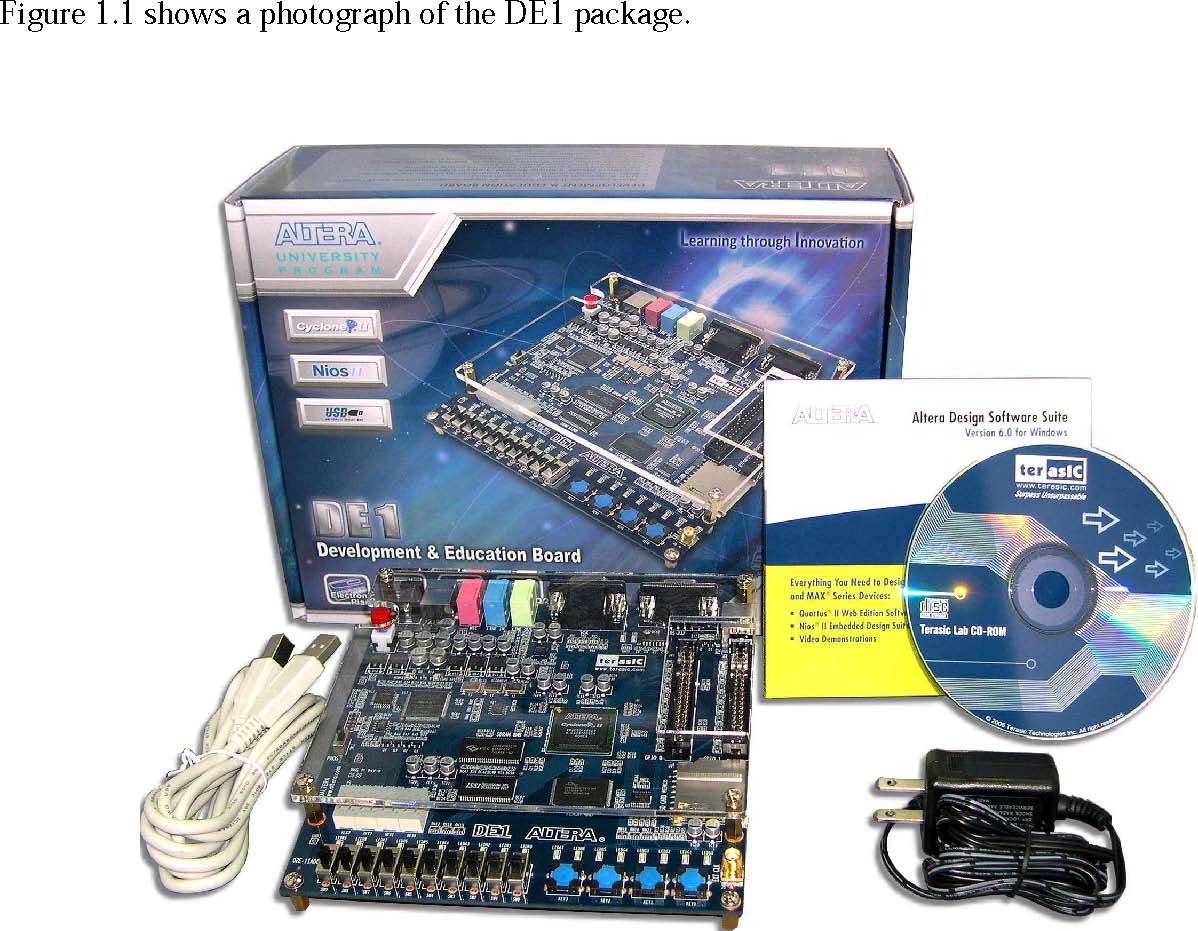


schermata di gioco è suddivisa in due riquadri principali:

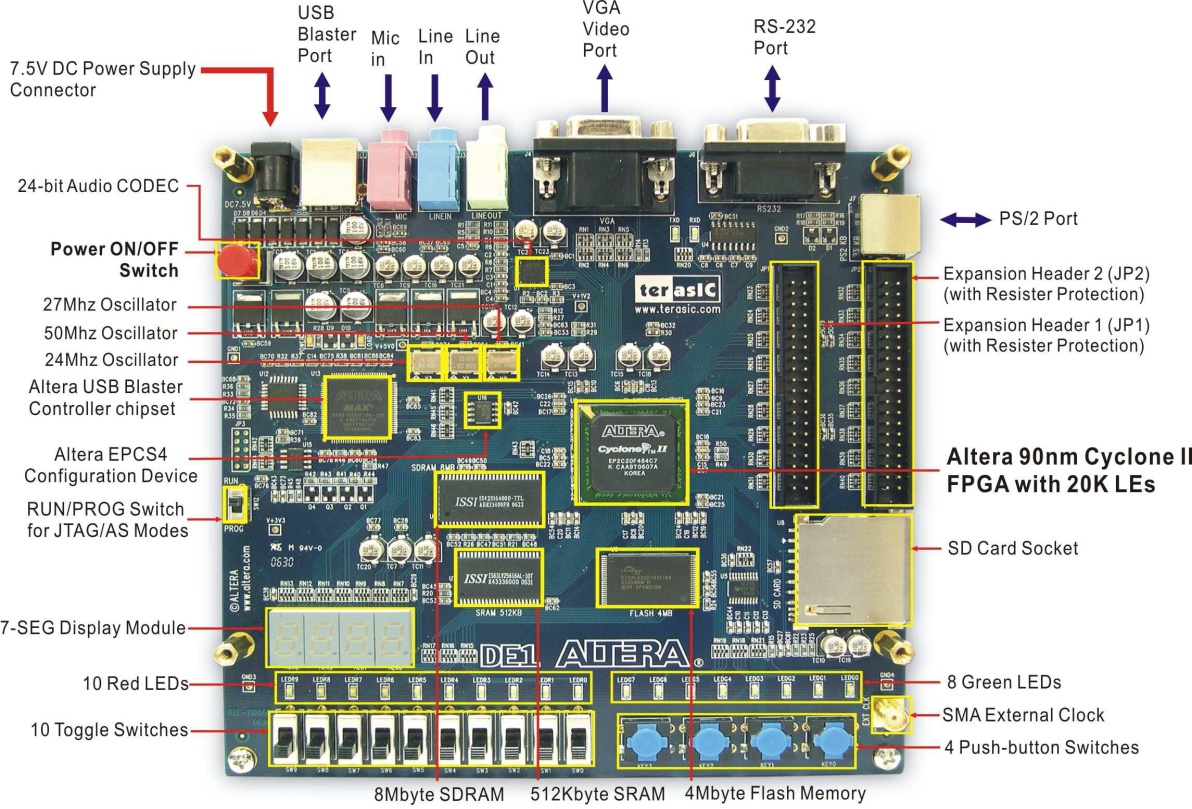
* Informazioni sul display cornice del giocatore:
  + Tra cui punteggio, livello, stage e numero di giocatori rimasti solidi.
  + 160x480 dimensioni.
  + Il testo viene visualizzato con dimensioni 32x16.
* parte di struttura dell'esposizione si gioca:
  + dimensioni 480x480
  + Serpenti, prede, e le pareti sono costruite composito 16x16 blocco quadrato.
  + Questa sezione può essere per visualizzare un messaggio quando è necessario.
  1. hardware consigliato
     1. kit DE1
        1. L'introduzione di KIT DE1

KIT DE1 un'Altera.Muc dei suoi sviluppatori di prodotto allo scopo di creare DE1 KIT è quello di fornire uno strumento per servire l'ideale di design avanzato in alcuni settori come multimedia, stoccaggio , la rete ...

KIT DE1 da usare, abbiamo bisogno di connettersi a un computer che esegue il software Microsoft Windows.



* + - 1. componenti Kit su DE1



• Dispositivo di Altera Cyclone II FPGA 2C20

• Dispositivo di configurazione seriale Altera - EPCS4

• Blaster USB (a bordo) per la programmazione e API controllo utente; Sia il JTAG e Active Serial

(AS) modalità di programmazione sono supportati

• 512 Kbyte SRAM

• 8 Mbyte SDRAM

Memoria • 4 Mbyte Flash

• Vano scheda SD

• 4 interruttori a pulsante

• 10 interruttori basculanti

• 10 LED utente rossi

• 8 LED utente reen

• oscillatore 50 MHz, 27 MHz e 24 MHz oscillatore oscillatore per sorgenti di clock

• CODEC audio di qualità CD a 24 bit con il line-in, line-out, microfono jack-in e

• DAC VGA (rete resistiva 4-bit) con connettore VGA-out

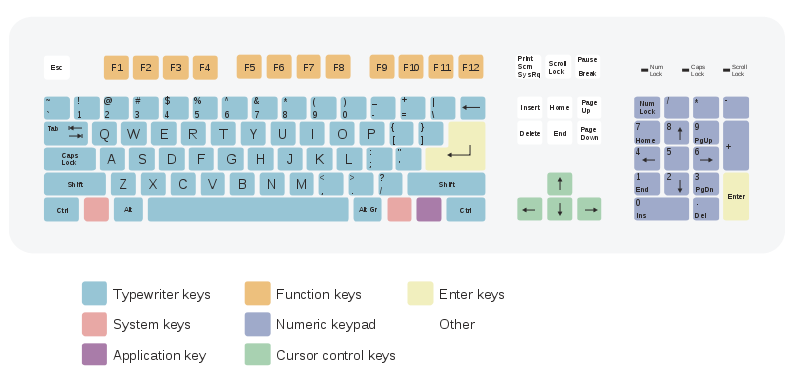
• RS 232 ricetrasmettitore e connettore 9 poli

• Connettore mouse PS / 2 / Tastiera

• Due intestazioni di espansione 40 pin con resistenza Protec

• Alimentato da un adattatore di HOAC DC 7.5V o un cavo USB

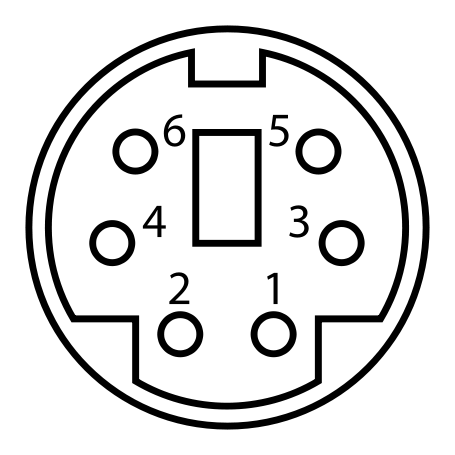
* + 1. PS2 Keyboard



Con la nostra gamma progettato questo progetto, interessati solo nella direzione del movimento tasti cane: su, giù, sinistra, destra. Due funzioni chiave per mettere in pausa e scegliere: Esc, Enter.Co in grado di sviluppare altri pulsanti richiesti da ciascun soggetto.

I dettagli su come ricevere e trasmettere dati chiave dalla tastiera saranno elencati nei dettagli del sistema.

**pinout PS2**

Utilizzare connettore PS2 standard per collegare una tastiera con KIT DE1

Pin 1 + DATI I dati

Pin 2 Non collegato Non collegato \*

pin GND 3 Gr

Batteria Vcc +5 V DC 4 a 275 mA

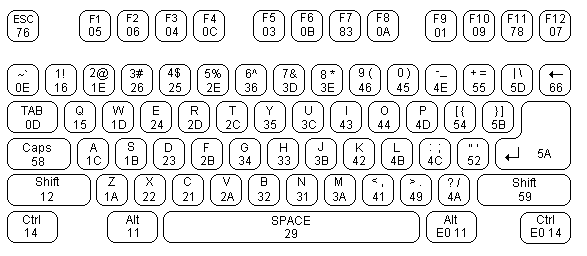
Pin 5 + CLK

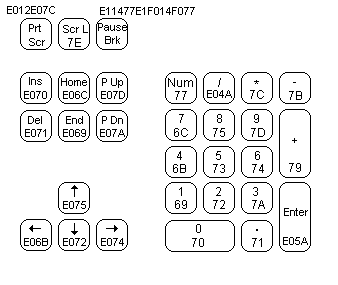
Pin 6 Non collegato Non collegato \*\*

Non standard PS2 ha qualche altro standard di connessione comune è USB e wireless (wireless). Ambito di temi progettuali legati solo con la PS2, abbiamo semplicemente introdurre e conoscere gli altri due standard.

connettore di interfaccia PS2 è un 6-pin MINI DIN.

**2.1.2.3. codice di scansione**

Una tastiera comprende una matrice chiave e un processore incorporato per controllare il funzionamento dei tasti e inviare scansione codice appropriato.



**Il funzionamento della tastiera** :

- Quando si preme un tasto, poi il 1 fanno trasmissioni codici dei tasti.

- Quando una chiave viene mantenuta costante, lo stato è noto come il trucco typematic il codice viene trasmesso continuamente con una certa velocità. In modalità predefinita, una tastiera PS2 rendere -code trasmesso a circa 100 ms dopo il 1 chiave è stata mantenuta nel range di 0,5 s.

- Quando il tasto viene rilasciato, la prima rompere il codice (0xF0) trasmissioni quindi rendere il codice chiave per identificare quale tasto è stato rilasciato.

**il trasferimento dei dati Way**

Dispositivo PS2 (tastiera) collegato a KIT scambiare dati DE1 e attraverso il percorso di dati e di clock 2



Dati Strada comprende 11 bit



Orologio della via è stato portato in un segnale di clock separata.

Dati vengono trasmessi quando il cambiamento di clock, e sono attivo basso (fronte di discesa).

* + 1. Monitor VGA

**introdurre**

VGA (Video Graphics Array: Video Graphics Array) è stato introdotto per i PC IBM sono supportate dall'hardware grafico e monitorare hinh.Chung PC di progettazione si interfaccerà con 8 colori di base con una risoluzione di 640x480 per lo schermo CRT.

**meccanismo di base di azione di un CRT**

Schema a blocchi:



Figura 1. L'intensità del fascio di elettroni e luminosità del punto è determinata dal livello di tensione del segnale video di ingresso, il segnale è mono mono.Tin livelli di tensione prestazioni analogiche tra 0 e 0,7 modifiche.

- La bobina di deflessione verticale ed una orizzontale controllo bobina di deflessione del flusso di elettroni e viaggio in cui gli elettroni passano decisioni sui monitor hinh.Voi oggi, gli elettroni capo sono guidati da sinistra a destra, dall'alto in basso.

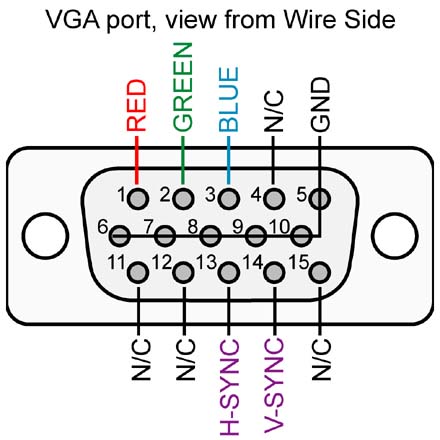
**VGA scansione**



Quando la tensione viene applicata alla bobina di deflessione honrizontal e aumentato costantemente, gli elettroni capo si sposteranno dalla sinistra alla destra. Dopo aver raggiunto l'angolo destro, i raggi saranno rapidamente coprire angolo posteriore sinistro quando la tensione di 0V (HSYNC). Fino elettroni capo alla stazione a valle, il monitor di tensione sarà messo nella bobina di deflessione verticale, verranno prese raggi incappucciati torna dello schermo (VSYNC) e continuare il processo come mostrato in figura

Segnale hsync usato per scandire lo schermo in righe e segnali VSYNC utilizzati per analizzare l'intero schermo con una frequenza di 25MHz tasso di pixel (25 milioni di pixel avviene in 1s) per creare VGA schermo 640x480 risoluzione .

**pin VGA**

****

segnale VGA tra cui 5 attività: due hsync e segnale vsync, tre segnali video sono di colore rosso, blu, verde è collegato a 15 piedi

|  |  |  |  |
| --- | --- | --- | --- |
| rosso | verde | blu | colore qUA |
| 0 | 0 | 0 | nero |
| 0 | 0 | 1 | blu |
| 0 | 1 | 0 | verde |
| 0 | 1 | 1 | Ciano |
| 1 | 0 | 0 | rosso |
| 1 | 0 | 1 | magenta |
| 1 | 1 | 0 | giallo |
| 1 | 1 | 1 | bianco |

**il trasferimento dei dati Way**

Per essere in grado di ricevere i dati trasmessi e visualizzati sullo schermo, abbiamo progettato un circuito che comprende contatore vga\_sync e il segnale di clock con 2 bo.Mach hsync e segnale vsync è collegata direttamente allo schermo, abbiamo cane usato per monitorare i segnali di scansione orizzontale e verticale hinh.Hai decodificati da un circuito temporizzatore ed è disponibile in uscita 2 segnali sono pixel\_x, segnale di uscita che indica pixel\_y.Hai relazioni tra posizione di scansione e la posizione corrente di un anh.Mach video\_on punto di segnale al cane spento o sul display.



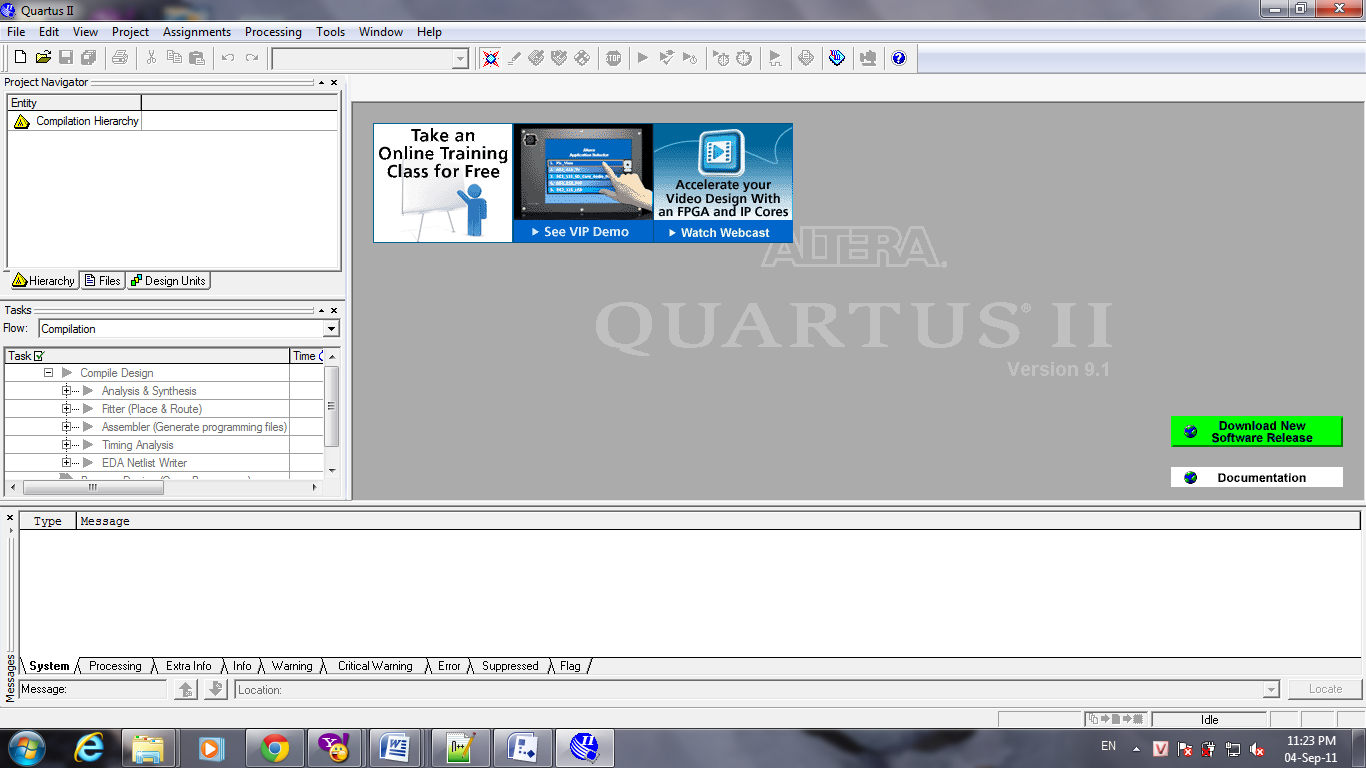
**blocco controllore VGA**

Un circuito per generare il segnale video 3 viene indicato come RGB (rosso verde blu) di ingresso è pixel\_x e pixel\_y, video\_on. Il valore di un colore viene visualizzato sullo schermo dipende dalla posizione corrente di pixel (pixel\_x e pixel\_y) e segnali di dati e cane all'esterno

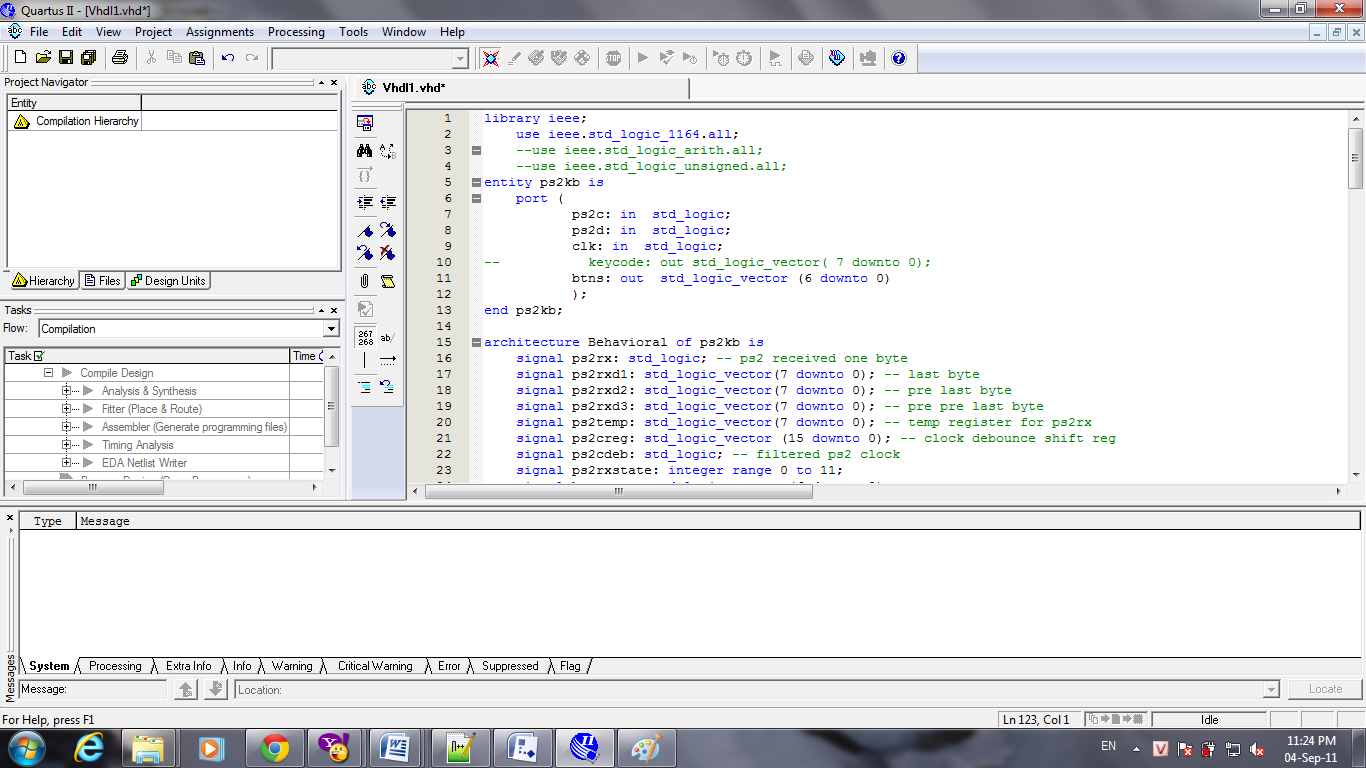
* 1. L'introduzione di software
     1. Quartus II

**interfaccia principale**

Quartus Web Edition 9.1 per la programmazione e caricati su KIT DE1

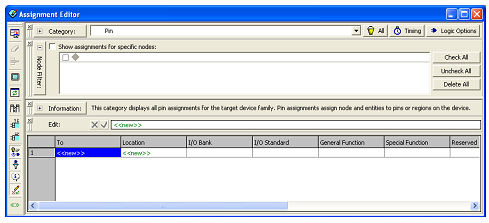


**2.2.1.2 Editor**



**assegnare gamba**

Fase 1: Fare clic su Assegnazioni> Assegnazione Editor. Nella Categoria selezionare Pin. Fare doppio clic sul nuovo << >>. Clicca su segnali dal menu a discesa, come viene assegnato il piede. Quindi, fare doppio clic sulla casella a casella di destra per il segnale da assegnare (colonna Location). Scegli dalla lista o il piede PIN\_XYZ può digitare direttamente nella casella Posizione.



**Figura finestra Editor 6. Assegnazione utilizzato per assegnare perni**

Fase 2: Come sopra, perform assegnare alle altre piedini di ingresso

Fase 3: Dopo aver completato i pin assegnati, fare clic su File> Salva. Chiudi la finestra Assegnazione Editor, fare clic su Sì e ricompilare il circuito.

**Attenzione** : Scegliere il nome del segnale coincide con la DE1\_pin\_assigment.csv gamba del tavolo quando abbiamo solo bisogno di assegnare assigment> importazione assigment nel nostro percorso di quel file e premere OK DE1\_pin\_assigment.csv e seguire il punto 3 è la perdita xong.Khong tempo per assegnare manualmente sulla mappa.

**compilare**

Una volta terminato di scrivere codice per un programma che è necessario compilare per creare i file utilizzati per caricare KIT DE1

Fase 1: Fare clic su Seleziona produzione> Avvia complicazione. Compilare successo (o insuccesso) saranno informati sul dialogo rilassarsi dopo le estremità di compilazione. Confermare facendo clic sul pulsante OK.

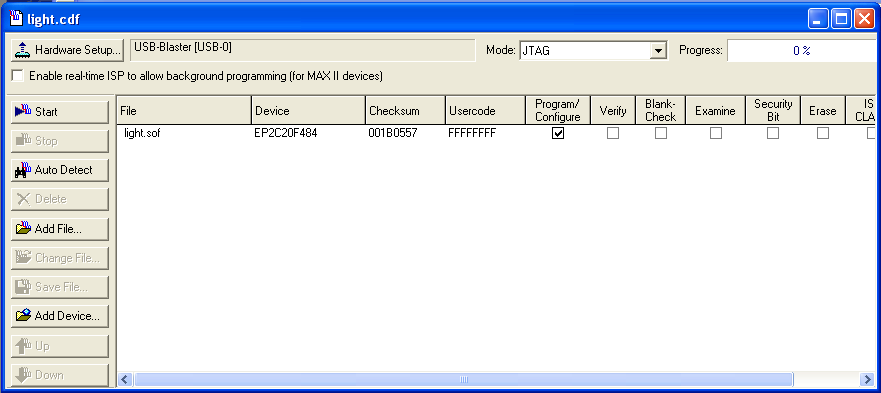
Passo 2: Quando la traduzione è completa, viene dato un rapporto compilatore. Questa finestra può anche essere aperto in qualsiasi momento facendo clic scegliere produzione> Complication Report. In questo rapporto include una serie di categorie sulla finestra di sinistra, fare clic sulle categorie per vedere i dettagli di questo annuncio si apre sulla finestra a destra.

Fase 3: correggere i bug

Selezionare Analisi e sintesi> Messaggi per visualizzare i messaggi di errore. Fare doppio clic sul primo messaggio di errore, gli errori della riga di comando saranno segnati sulla editor di testo, correggerlo e quindi ricompilare il progetto.

**Caricare KIT**

Fase 1: spazzolatura commutazione interruttore RUN / PROG a RUN. Fare clic su Strumenti> Programmer alla finestra come mostrato in Figura 11. Controllare l'opzione Programma / configue per consentire xxxxxx.sof file di configurazione caricato.



Fase 2: Fare clic sul pulsante Start sulla finestra di sinistra per caricare il file nella configurazione FPGA. Dopo caricato correttamente nella FPGA, controllare questo circuito implementazione su FPGA per eseguire razze esattamente come ufficiali del desiderio o meno.

* 1. schema a blocchi del sistema
     1. sistema ampia



* + 1. Blocchi e funzioni.

|  |  |
| --- | --- |
| unità | funzioni |
| PS2 | Ricezione di segnali dalla tastiera, l'invio di segnali di controllo. |
| VGA\_SYNC | Controllare la visualizzazione sul VGA |
| ARTICOLI | Creazione di prede per i serpenti. |
| SNAKE | blocco di controllo del comportamento solido. |
| MAPPE | La creazione di diversi ostacoli nella schermata di gioco. |
| testi | Consente di visualizzare le informazioni per l'utente. |
| FSM | blocco di controllo del sistema, collega ad altri blocchi. |
| TIMER | Timer, creare latenza. |
| VELOCITA ' | Regolatori velocità del serpente movimento. |
| LIVELLO | giocatori Calcolo rango. |

1. particolari di disegno
   1. blocco VGA

**A proposito di CRT 640x480**

640x480 è il numero di pixel e linee che possono essere visualizzati su schermo hinh.Moi includono 640 linee orizzontali di pixel, 480 è il numero di linee di scansione orizzontali ma il fatto che ciascuna linea orizzontale comprende 800 pixel e il numero di linee di scansione orizzontali è 525 .quindi pixel e linee non riportate è chiamato bordo nero (bordo nero) L'attività .Tan è 25MHz.



**1.1.1. sincronizzazione asse orizzontale**

800 pixel asse orizzontale è diviso in 4 regioni



- Dislay: regioni in cui i pixel sono indicati con la lunghezza 640px.

- Retrace: le regioni in cui copertina elettrone posteriore sinistro, il segnale video devono essere spenti, la lunghezza 96px.

- margine destro (bordo destro) regioni che formano il bordo destro dell'area di visualizzazione sono chiamati veranda (cancello), il segnale dovrebbe essere spento, lunghezza 16px

- bordo sinistro (a sinistra): modulo di regione di confine sinistro dell'area di visualizzazione posteriore portico chiamato (portellone), il segnale deve essere spento, la lunghezza 48px.

Il codice seguente determina le quantità per l'asse orizzontale:

- i parametri di sincronizzazione VGA 640-by-480

*HD costante: integer: = 640; Area di visualizzazione --horizontal*

*costante HF: integer: = 16; -H. veranda*

*HB costante: integer: = 48; -H. veranda sul retro*

*costante HR: integer: = 96; -H. ripercorrere*

La lunghezza del bordo sinistro e destro può variare fra gli schermi differenti.

segnale Hsync bisogno di più contatore e un circuito di visualizzazione risoluzione di 800 pixel ma.Khi avviare il timer e iniziare il conteggio e segnalare la componente di segnale pixel\_x.Tin hsync segnale basso quando il segnale di uscita di contatori 656 e 751 intervalli.

Abbiamo usato per regolare il segnale video\_on mostra / non mostra quando il contatore ha un valore inferiore a 640

monitor CRT dovrebbero essere al nero nelle ali destra e sinistra e durante ritracciamento.

**1.1.2 sincronizzazione asse verticale**

Durante verticale, fascio di elettroni in movimento costante dall'inizio alla schermata finale, poi è tornato e ha iniziato hinh.Su stesso schermo richiede un certo tempo per aggiornare la struttura hinh.Cau Vsync simile a hsync.

Un ciclo del segnale vsync è di 525 linee ed è divisa in 4 aree come hsync.

Le caratteristiche di ciascuna regione sono hsync simili



linea di segnale Vsync 525 ad un conteggio e un soluzioni circuitali conteggio inizio precoce zona ma.Bat thi.tin segnale di uscita corrente del contatore è segnale vsync pixel\_y.Tin a livello basso quando il contatore sulla linea 490 o linea 491.

Come hsync, usiamo video\_on per vedere / non vedere quando il contatore ha un valore inferiore a 480.

**1.1.3 Calcolo del VGA segnali di sincronizzazione di tempo**

Nell'ambito di questo progetto abbiamo utilizzato la frequenza è 25MHz.Su questa selezione è deciso da tre quantità:

- p: il numero di pixel su una linea di scansione orizzontale. p = 800 pixel / linea

- L: Monitor totale di zuccheri. l = 525 linee / schermo

- s: numero di fotogrammi al secondo. s = 60 copie / secondo

Scegliere s = 60 qui è perché l'occhio umano funziona bene in questa cornice e contro il lampeggio.

Così pixel rate = p \* l \* s = 25M (pixel / secondo)

**1.1.4 Completa VGA grafica**

Sopra abbiamo progettato 2 set problema di progettazione dem.Van qui è che KIT DE1 supporta solo la frequenza 50MHz tale richiesta è 25MHz.Vi dai requisiti di progettazione così abbiamo creato uno dei 25 MHz permette di marcatura per disattivare o attivare il segnale è un segnale dem.Tin p\_tick fuori eseguire questo lavoro e coordinare il funzionamento del circuito di generazione di pixel.

Utilizzare 2 reale avanzamento h\_end e v\_end controlla la realizzazione di scansione orizzontale e verticale.

Anche per evitare interferenze situazione è necessario utilizzare più del buffer è stato inserito segnali HSYNC e VSYNC.

Il seguente codice utilizzato per creare il contatore mod-2 usato per segnare il passo:

*- mod-2 del circuito di generare 25 MHz consentono tick*

*mod2\_next <= non mod2\_reg;*

*- pixel tick 25 MHz*

*pixel\_tick <= '1' quando mod2\_reg = '1' altro '0';*

Il seguente codice utilizzato per identificare il completamento del orizzontale:

*h\_end <= - fine del contatore orizzontale*

*'1' quando h\_count\_reg = (HD + HF + HB + HR-1) --799 altro '0'; // sottrarre 1 perché contiamo da 0 //*

Il codice seguente viene utilizzato per eliminare il rumore:

*- sincronizzazione orizzontale e verticale, tamponata per evitare inconveniente*

*h\_sync\_next <= '1', quando (h\_count\_reg> = (HD + HF)) --656*

*e (h\_count\_reg <= (HD + HF + HR-1)) else -751*

*'0';*

* 1. blocco PS2

Circuito include un ricevitore che può inserire un blocco contro interferenza del segnale, un blocco FSM di trasmissione cane codice di scansione, blocco FIFO per scrivere i dati e trasmette fuori meccanismo first in first out ( first in first out).

**2.1 Blocco ricezione dei dati (PS2\_rx)**

Utilizzare il seguente codice per evitare interferenze con il segnale:

ps2cdeb <= '0' quando ps2creg = "1111111111111111" else // restituisce 0 se tutti i bit è 1

'1' quando ps2creg = "0000000000000000" else // restituisce 1 quando tutti i bit sono 0;

Per capire il meccanismo della PS2 ricevere dati consideriamo il seguente diagramma di flusso:



**grafico ASMD del ricevitore porta PS2**

Quando il segnale ricevuto consente segnali positivi rx\_en e costolette scorrimento allo stato tradurranno ad iniziare bit e spostare affermare DPS.

Poiché i dati sono ricevuti in un blocco è stato controllato, tradurremo 10 bit in uno stato separato piuttosto che utilizzare uno stato separato come dati, parità, stop.

Allora il circuito si sposterà carico stato che ha un ciclo del segnale di clock ulteriormente distribuiti per completare il processo di traduzione del bit di stop, e rx\_done\_tick segnale inserito dopo il primo ciclo di notare aver ricevuto la dati.

* 1. ARTICOLI blocco
     1. Descrizione generale



Figura 2. ARTICOLI blocchi

ARTICOLI compito blocchi nati preda di serpenti, con la posizione esca che soddisfa le condizioni:

* Nuova esca viene creata se l'esca è mangiato, o non appena creano uno nuovo ha coinciso alcuni combustibili solidi, o centrato intorno alle ostruzioni.
* Bait è generato in posizioni casuali, non dipende dal luogo in precedenza esca.
* Nuovo Predator nato nella regione devono spostare i serpenti.
* Nuovo primer è la posizione non coincide con qualsiasi combustibile solido, se si tiene su un solido, nuova esca di ricreare.
* Nuovi ostacoli esca nati non devono essere identici, se non, per ricreare una nuova esca.
* Le esche speciali scompaiono dopo un certo periodo di tempo.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| tipo | forma | tempo scompare | punto | Le caratteristiche supplementari |
| 1 | mele |  | 1 |  |
| 2 | come | 5s | 1 | decelerazione |
| 3 | cuore | 5s | 1 | Aumento risuona |
| - | - | - | - | - |

Gli ingressi e le uscite sono dichiarati come segue:

I prodotti che entità è

porte (

- orologio principale

clk: in std\_logic;

reset: in std\_logic;

- coordinate del pixel sottoposti a scansione

pixel\_x: in std\_logic\_vector (9 downto 0);

pixel\_y: in std\_logic\_vector (9 downto 0);

- livello attuale appaia diverso livello adeguato fondo

Livello: nel campo dei numeri interi da 1 a 7;

item\_x: fuori senza segno (9 downto 0);

item\_y: fuori senza segno (9 downto 0);

pausa: in std\_logic;

item\_on\_wall: in std\_logic;

item\_on\_snake: in std\_logic;

item\_ate: in std\_logic;

timer2\_reset: out std\_logic;

timer2\_start: out std\_logic;

timer2\_up: in std\_logic;

item\_type: fuori campo dei numeri interi da 1 a 7: = 1;

item\_on: out std\_logic;

item\_rgb: out std\_logic\_vector (2 downto 0)

);

articoli finiti;

Creare rom articoli:

Descriviamo ROM attraverso la matrice di bit, come segue:

Generatore di numeri casuali:



Per generare numeri casuali usiamo il contatore valori di rotazione e contatori vengono recuperati in qualsiasi momento che vogliamo, per garantire la casualità, i contatori di tempo di ritorno è molto più piccolo rispetto al tempo medio necessario valore dal contatore

Usiamo prezioso banco piccola, ma la frequenza di ripetizione veloce, che è l'orologio 50Mhz.

Contatore 1 viene utilizzato per la gamma di valori della coordinata orizzontale x preda

Contatori 2 utilizzati per creare valore y coordinate dell'esca verticalmente

Combina 2 valori stanno per ottenere le coordinate della nuova esca nasce, e sembra quasi casuale.

codice:

* 1. blocco SNAKE
     1. diagrammi
     2. codice
  2. blocco MAPPE
  3. testi di blocco
     1. font

- ROM con lettura synchonous (inferire Block RAM)

- ROM carattere

- - 8-by-16 (8-by-2 ^ 4) carattere

- - 128 (2 ^ 7) caratteri

- - dimensione ROM: 512-by-8 (2 ^ 11 x 8) bit

- 16K bit: 1 BRAM

IEEE biblioteca;

ieee.std\_logic\_1164.all utilizzare;

ieee.numeric\_std.all utilizzare;

entità font\_rom è

porte (

clk: in std\_logic;

addr: in std\_logic\_vector (10 downto 0);

Dati: out std\_logic\_vector (0-7)

);

font\_rom fine;

architettura arco di font\_rom è

ADDR\_WIDTH costante: integer: = 11;

DATA\_WIDTH costante: integer: = 8;

segnale addr\_reg: std\_logic\_vector (1 downto ADDR\_WIDTH-0);

Tipo rom\_type è array (0 e 2 \*\* ADDR\_WIDTH-1)

di std\_logic\_vector (0 a DATA\_WIDTH-1);

- Definizione ROM

costante ROM: rom\_type: = (- 2 ^ 11-by-8

"00000000", - 0

"00000000" - 1

"00000000" - 2

"00000000", - 3

"00000000", - 4

"00000000" - 5

"00000000" - 6

"00000000" - 7

"00000000", - 8

"00000000", - 9

"00000000" - un

"00000000", - b

"00000000", - c

"00000000", - d

"00000000", - e

"00000000" - f

- il codice x01

"00000000", - 0

"00000000" - 1

"01111110" - 2 \*\*\*\*\*\*

"10000001", - 3 \*

"10100101", - 4 \* \* \* \*

"10000001", - 5 \*

"10000001" - 6 \*

"10111101" - 7 \* \*\*\*\* \*

"10011001", - 8 \* \*\* \*

"10000001" - 9 \*

"10000001" - a \* \*

"01111110", - b \*\*\*\*\*\*

"00000000", - c

"00000000", - d

"00000000", - e

"00000000" - f

...  
 );

iniziare

- addr registrati per inferire blocco di RAM

processo (CLK)

iniziare

se (clk'event e clk = '1') allora

addr\_reg <= addr;

end if;

fine processo;

Dati <= ROM (to\_integer (senza segno (addr\_reg)));

end arco;

* 1. blocco FSM
     1. diagrammi
     2. codice
  2. sottoblocco
     1. timer\_1

Si tratta di blocchi 2s timer per creare la latenza tra gli stati della macchina a stati FSM, utile quando si desidera annunciare qualcosa sullo schermo. Ecco l'annuncio GAME OVER, STAGE CLEAR LEVEL UP ...

Modello:



Dichiarare ingresso, di uscita:

Timer entità è

porte (

- Main-orologio in ingresso 50MHz

CLK: in std\_logic;

- reset sincrono

reset: in std\_logic;

- Tasso di 60Hz-clock del monitor è di input

refr\_tick: in std\_logic;

- segnale di avvio

timer\_start: in std\_logic;

- uscita

timer\_up: std\_logic out

);

end timer;

Useremo il contatore per quest a tempo, quindi abbiamo salvato il valore di conteggio in un registro, può contare salita, conteggio diminuita arbitraria ma contando ridotta vantaggio quando si confronta il valore della barra incisa con un valore 0.

Qui abbiamo bisogno di circa due secondi il timer, il segnale di ingresso è un aggiornamento dello schermo, a circa 60 Hz a 2 secondi corrisponderà al valore del registro 120.

Abbiamo scelto registri 7-bit, cioè il suo valore massimo è 127, che significa un ritardo è 127/60 = 2,1 secondi, incertezza ancora accettabile.

codice:

architettura arco del timer è

- valore massimo nel registro è 127, il valore massimo è di circa 2 contatore dei secondi

segnale timer\_reg, timer\_next: senza segno (6 downto 0);

iniziare

La sezione di codice tempo di lavorazione successivo conteggio, primi registri del processore, utilizzando orologio 50Mhz quando l'orologio principale per aumentare l'uniformità del sistema

Abbiamo schierato forma contatore di macchine a stati, ma non hanno alcun ram stato creato proprio questo valore registro diviso in due registri: solo timer\_reg contatore di valori ora, e solo timer\_next contatore di valori poi, se vi capita di clock.

codice:

- registri

processo (clk, reset)

iniziare

se (clk'event e clk = '1') allora

Se il reset = chiave '1'

timer\_reg <= (altri => '1');

altro

timer\_reg <= timer\_next;

end if;

end if;

fine processo;

- la logica dello stato successivo

processo (timer\_start, timer\_reg, refr\_tick)

iniziare

if (timer\_start = '1') allora

timer\_next <= (altri => '1');

elsif refr\_tick = '1' e timer\_reg / = 0 allora

timer\_next <= timer\_reg - 1;

altro

timer\_next <= timer\_reg;

end if;

fine processo;

- uscita

timer\_up <= '1' quando timer\_reg = 0 altrimenti '0';

end arco;

* + 1. timer\_2

Analogamente timer\_1, timer\_2 anche il contatore, ma utilizzato per altri scopi, come ad esempio il tempo di visualizzazione di un esche e soprattutto nei giochi, come LIVE\_UP, SPEED\_DOWN ... Il display esca in un certo periodo di tempo e poi scompare se in quel periodo, ma non poteva mangiare cibo solido.



Come timer\_1, contiamo l'ultima volta con il registro 60Hz orologio battere attraverso il valore del registro. L'unica differenza è la lunghezza del registro significa che i registri max stoccaggio valore.

Qui registra sarà:

segnale timer2\_reg, timer2\_next: senza segno (9 downto 0);

10 valore di bit memorizzato max = 511, così il più grande ritardo è 511/60 = 9 secondi.

La movimentazione timer\_2 timer\_1 identico non deve dire qui nuovamente.

* + 1. LIVELLO

A livello di blocco monitorare il numero di prede aveva mangiato il serpente, in modo che quando il serpente mangia una quantità certa preda, sarà aumentato il rango del giocatore, accompagnato dalla velocità di movimento di aumenti solidi mirato aumentare il livello di difficoltà la schermata di gioco.

Una volta che i giocatori iniziano a livello 1 e raggiungere il livello 5, la schermata di gioco cambierà, abbiamo bisogno di controllare quando cambia la schermata di gioco, ora è necessario aggiungere del codice per un corrispettivo di concessione livello è stato raggiunto.

I pin di segnale necessari sono descritti come segue:

Entità è

porte (

- orologio principale

clk: in std\_logic;

riavviare: in std\_logic\_vector (1 downto 0);

- esca è stato mangiato

item\_ate: in std\_logic;

- Notifica segnale di livello alto

level\_up: out std\_logic;

- segnali su schermi.

stage\_clear: out std\_logic;

- monitorare riproduzione corrente del giocatore.

stage\_select\_in: nel campo dei numeri interi da 1 a 7;

stage\_select\_out: fuori campo dei numeri interi da 1 a 7;

- comunicare al livello attuale del giocatore.

Livello: fuori campo dei numeri interi da 1 a 7

);

end livello;

Codice a livello di blocco per controllare il lettore sulla base del esca è stato mangiato.

Alcuni esca per mangiare il livello successivo è 5 volte superiore rispetto ai livelli attuali, per esempio dal livello 2 al livello 3 vuole mangiare 2 \* 5 = 10 primer.

elsif item\_ate = chiave '1'

se counter\_next\_level = 5 \* level\_tmp Athens

level\_tmp: = level\_tmp + 1;

Questo codice controllerà la schermata di gioco se il giocatore ha raggiunto il gioco a schermo livello 6, il giocatore sarà sollevato più difficile

se level\_tmp = 6 Athens

se stage\_sel\_tmp <2 bar

stage\_sel\_tmp: = stage\_sel\_tmp + 1;

end if;

stage\_clear\_tmp: = '1';

e gradi contenute nella nuova schermata di gioco sarà avviato di nuovo dal livello 1.

level\_tmp: = 1;

end if;

level\_up\_tmp: = '1';

counter\_next\_level: = 0;

altro

counter\_next\_level: = counter\_next\_level + 1;

end if;

end if;

Livello <= level\_tmp;

level\_up <= level\_up\_tmp;

stage\_clear <= stage\_clear\_tmp;

stage\_select\_out <= stage\_sel\_tmp;

end if;

fine processo;

end arco;

* + 1. PUNTEGGI
       1. diagrammi



* Questa unità è responsabile per la classificazione del giocatore, e il punteggio ricevuto dipenderà dal tipo di esca che si snoda mangiare, livello attuale del serpente (maggiore è la velocità, maggiore è il livello, più punti ricevuti capiscono).
* Blocchi ricevono segnali da parte degli investitori compresi gli oggetti Item\_ate, e dopo aver calcolato item\_type blocco Add\_ind punto nguii in cui si aggiunge il gioco, si genereranno impulsi Aggiungi al blocco del contatore segnato. Per esempio, quando Item\_ate = 1, item\_type = 3, Livello = 2, quindi il punteggio è 3 + 2 persone = 5, questa volta genererà 5 segnale Add\_Ind cubic Aggiungi al contatore.
* Score\_lcr ripristino mirato quando il punteggio del 0000 appena iniziato schermata di gioco
  + - 1. Counter\_9999

Come menzionato sopra, più segnali di blocco ricevuti attraverso COUNTER\_9999 d\_inc piede, ogni volta che questo segnale da basso ad alto, il valore del contatore aumenta di 1.

Le gambe del blocco sono descritti come segue:

entità counter9999 è

porte (

- orologio principale

clk: in std\_logic;

- Reset

reset: in std\_logic;

- aggiunta indicare

d\_inc: in std\_logic;

- uscita è il numero di vantaggi per l'introduzione di testi variabili per le cifre, serve la visualizzazione sullo schermo.

dig0: out std\_logic\_vector (3 downto 0);

DIG1: out std\_logic\_vector (3 downto 0);

DIG2: out std\_logic\_vector (3 downto 0);

DIG3: out std\_logic\_vector (3 downto 0)

);

end counter9999;

-------------------------------------------------- ----------

architettura arco di counter9999 è

- registri salvare il valore corrente e il successivo conteggio 4.

segnale dig0\_reg, dig1\_reg, dig2\_reg, dig3\_reg: senza segno (3 downto 0);

dig0\_next segnale, dig1\_next, dig2\_next, dig3\_next: senza segno (3 downto 0);

iniziare

- registri

processo (clk, reset)

iniziare

se (clk'event e clk = '1') allora

Se il reset = chiave '1'

dig0\_reg <= (altri => '0');

dig1\_reg <= (altri => '0');

dig2\_reg <= (altri => '0');

dig3\_reg <= (altri => '0');

altro

dig0\_reg <= dig0\_next;

dig1\_reg <= dig1\_next;

dig2\_reg <= dig2\_next;

dig3\_reg <= dig3\_next;

end if;

end if;

fine processo;

- la logica dello stato successivo per il contatore decimale

processo (reset, d\_inc, dig0\_reg, dig1\_reg, dig2\_reg, dig3\_reg)

iniziare

Se il reset = chiave '1'

dig0\_next <= (altri => '0');

dig1\_next <= (altri => '0');

dig2\_next <= (altri => '0');

dig3\_next <= (altri => '0');

altro

dig0\_next <= dig0\_reg;

dig1\_next <= dig1\_reg;

dig2\_next <= dig2\_reg;

dig3\_next <= dig3\_reg;

end if;

se (d\_inc = '1') allora

se dig0\_reg = 9 bar

dig0\_next <= (altri => '0');

se dig1\_reg = 9 bar

dig1\_next <= (altri => '0');

se dig2\_reg = 9 bar

dig2\_next <= (altri => '0');

se dig3\_reg = 9 bar

dig3\_next <= (altri => '0');

altro

dig3\_next <= dig3\_reg + 1;

end if;

altro

dig2\_next <= dig2\_reg + 1;

end if;

altro

dig1\_next <= dig1\_reg + 1;

end if;

altro

dig0\_next <= dig0\_reg + 1;

end if;

end if;

fine processo;

- uscita

dig0 <= std\_logic\_vector (dig0\_reg);

DIG1 <= std\_logic\_vector (dig1\_reg);

DIG2 <= std\_logic\_vector (dig2\_reg);

DIG3 <= std\_logic\_vector (dig3\_reg);

end arco;

* + - 1. unità punteggi

Questa massa è la natura massa del controllo blocco counter999 e comunicare con il blocco superiore.

Dichiarare i pin di segnale come segue:

IEEE biblioteca;

ieee.std\_logic\_1164.all utilizzare;

punteggio entità è

porte (

- orologio principale

clk: in std\_logic;

- punti di segnale eliminazione su 0000

score\_clr: in std\_logic;

- Giornale esca è stato mangiato

item\_ate: in std\_logic;

- esche

item\_type: nel campo dei numeri interi da 1 a 7;

- il livello attuale del giocatore

Livello: nel campo dei numeri interi da 1 a 7;

- mettere le cifre per i testi

dig0: out std\_logic\_vector (3 downto 0);

DIG1: out std\_logic\_vector (3 downto 0);

DIG2: out std\_logic\_vector (3 downto 0);

DIG3: out std\_logic\_vector (3 downto 0)

);

punteggio finale;

dichiariamo un componente per il blocco counter9999

ed eseguire segnale morsettiera.

counter\_u0: work.counter9999 Entity

Mappa porto (...);

Parte importante qui è come creare un corrispondente ritmo contando i centri saranno più punti per i giocatori, questo tasso di conteggio tiene counter9999 tramite contatore indicatori di direzione e d\_inc.

Abbiamo gestito attraverso un processo come segue:

processo (clk, livello, item\_ate, item\_type)

- condizioni di flusso variabili

aggiungere variabili: std\_logic;

- salvare il valore di comunità

variabile contatore: campo dei numeri interi da 0 a 63;

iniziare

se clk'event e clk = chiave '1'

- se l'esca è mangiato

se item\_ate = chiave '1'

- segnali di uscita per i contatori

aggiungere: = '1';

end if;

- se il segnale per il contatore

se add key = '1'

- calcolare la quantità più

se il contatore = livello + item\_type Athens

- se lo scrittore più sufficiente avviso pubblico

aggiungere: = '0';

- mettere contatore a 0

contatore: = 0;

altro

- in caso contrario, contare fino a

contatore: = counter + 1;

end if;

end if;

- e ha dato segnali per counter9999

d\_inc <= aggiungere;

end if;

fine processo;

end arco;

* + 1. costi
    2. SNAKE\_COUNTER