

Sistema semaforico avanzato

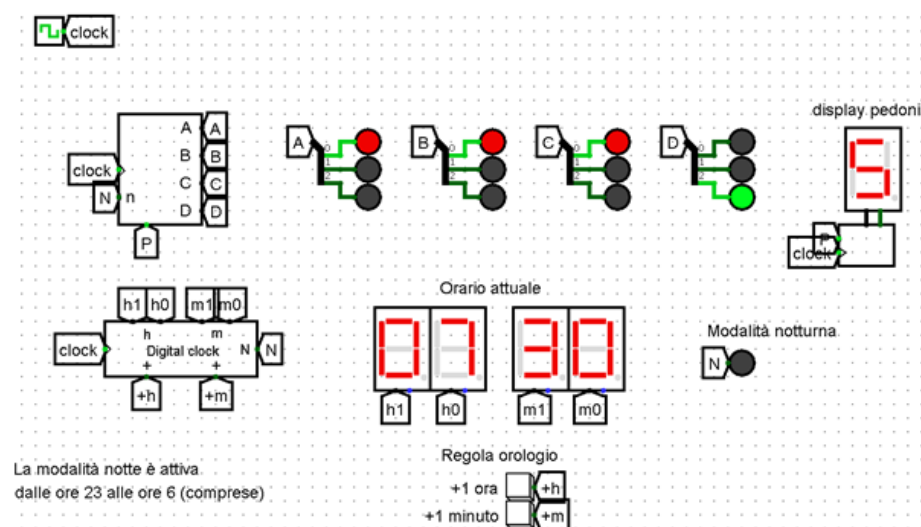
Progetto d'esame di Architettura degli Elaboratori I

Marco Aceti

marco.aceti@studenti.unimi.it

Primo appello di febbraio 2021

1 Introduzione



Partendo dall'attenta osservazione di un incrocio realmente presente nella mia cittadina, ho implementato in un circuito il **controllore** di un sistema semaforico avanzato utilizzando le porte logiche di base e i componenti offerti dal software Logisim. Come da specifica, il progetto supporta il controllo di 4 semafori, un timer per i pedoni, un orologio (regolabile) e la modalità notturna. Per un'esperienza realistica, è consigliato impostare il clock a 2Hz.

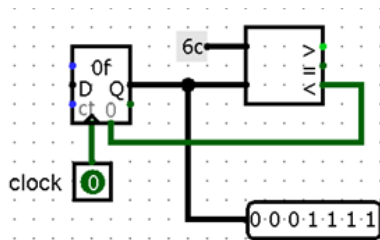
2 Analisi dei circuiti

2.1 Circuito main

Il circuito **main** è il **principale**. Esso presenta:

- **4 semafori A, B, C, D**: si comportano come indicato in specifica. Ogni semaforo ha tre lanterne corrispondenti ai tre stati che può assumere: *rosso*, *giallo*, *verde*.
- **Display pedoni**: si attiva solo quando il semaforo dei pedoni D è *verde* o *giallo*.
 - Se il semaforo dei pedoni **D** è *verde*, mostra un countdown da 9 a 0 corrispondente al numero dei secondi rimanenti ai pedoni per attraversare la strada.
 - Se il semaforo dei pedoni **D** è *giallo*, mostra uno 0 fisso con il punto lampeggiante per avvisare ai pedoni che devono liberare l'incrocio. Idealmente, il segnale del punto lampeggiante può essere utilizzato per controllare un dispositivo di segnalazione acustico.
- **Orario attuale**: mostra l'orario attuale, in formato 24 ore con 4 display per ore e minuti. L'orologio è regolabile tenendo premuti i bottoni sotto che aggiungono un'ora o un minuto a ogni ciclo di clock.
- **Modalità notturna**: attiva un LED giallo quando la modalità notturna è attiva.

2.2 Circuito loop-counter

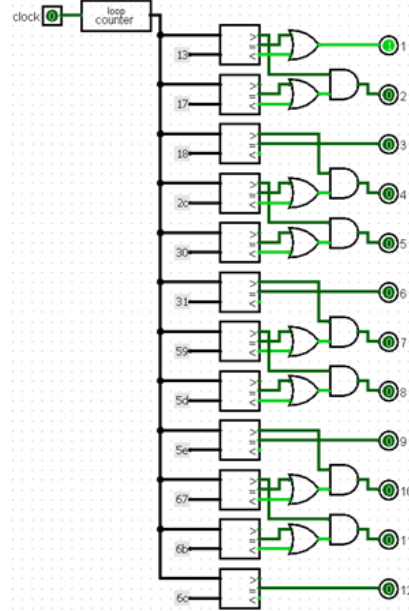


Il circuito **loop-counter** è molto semplice: utilizzando un contatore e un comparatore, conta da 0 a $6C$ ($= 108$). Se il valore del contatore è strettamente maggiore di $6C$, il comparatore attiva un segnale per resettarlo. L'input è il segnale di clock e l'output è equivalente al valore del contatore.

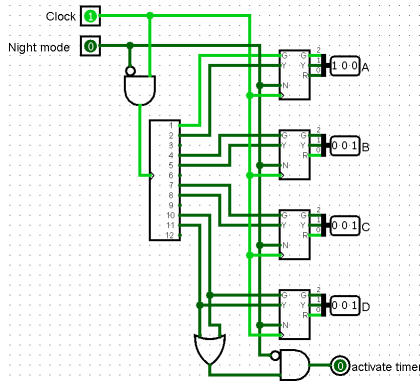
2.3 Circuito state-selector

Il circuito **state-selector** dà come output lo *stato* che deve assumere il controllore in funzione del valore del contatore **loop-counter**; così facendo possiamo assicurarci che ogni *stato* duri il tempo stabilito. La seguente tabella mostra a sinistra tutti vari *range* di valori che può assumere il contatore **loop-counter** ed a destra i relativi stati con la durata associata; riferirsi alla specifica per ulteriori dettagli.

Valori di loop-counter			
inizio	fine	stato	durata
0x00	0x13	1	20
0x14	0x17	2	4
0x18	0x18	3	1
0x19	0x2C	4	20
0x2D	0x30	5	4
0x31	0x31	6	1
0x32	0x59	7	40
0x5A	0x5D	8	4
0x5E	0x5E	9	1
0x5F	0x67	10	9
0x68	0x6B	11	4
0x6C	0x6C	12	1



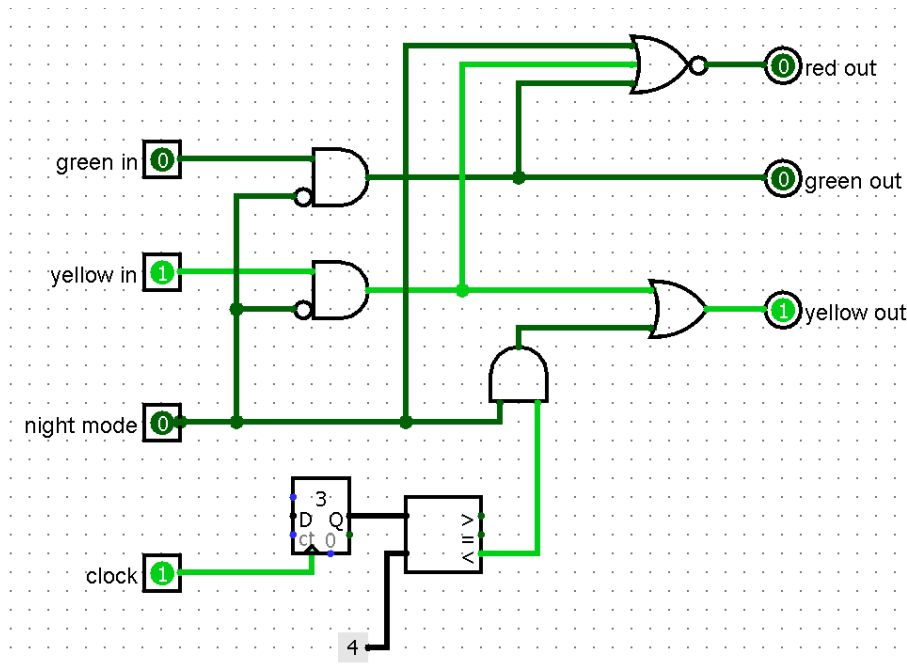
2.4 Circuito controller



Il **controller** è responsabile del controllo dei 4 semafori e del segnale *activate timer*. Oltre al segnale di clock, esso prende in ingresso anche il segnale *night mode*, attivato dal circuito **digital-clock**. Viene utilizzato il circuito **state-selector** per controllare i circuiti **semaphore** e il segnale *activate timer*. Essendo i semafori di default rossi, gli stati 3, 6 e 9 sono ignorati. Se la il segnale *night mode* è attivo, con le due porte AND del circuito viene disattivato il segnale di clock di **state-selector** e l'output *activate timer*, in quanto devono essere entrambi inattivi di notte.

2.5 Circuito semaphore

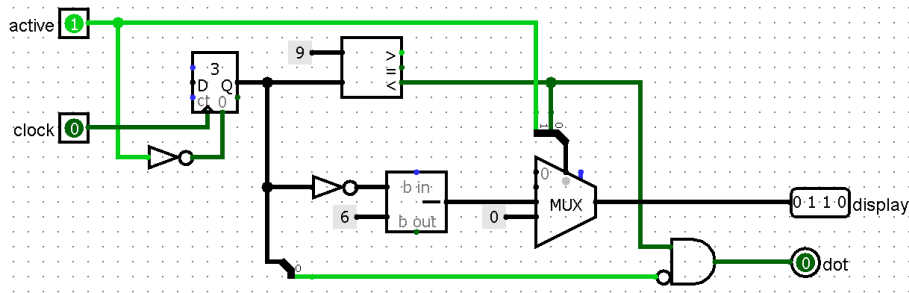
Il circuito **semaphore** rappresenta un semaforo. Ha come 4 ingressi (*green in*, *yellow in*, *night mode* e il clock) e tre uscite (*green out*, *yellow out* e *red out*).



La necessità di questo circuito è data dalla complessità aggiunta dalla modalità notte. Di seguito alcune considerazioni a riguardo:

- Se i segnali *green in*, *yellow in* e *night mode* sono disattivati l'output *red out* viene attivato. Si può anche dire che se è giorno e il semaforo non è né verde né giallo, allora è rosso.
- I segnali *green in* e *yellow in* sono subordinati dal segnale *night mode*. Se quest'ultimo è attivo, infatti, il semaforo smetterà di comportarsi normalmente.
- Se la *night mode* è attiva, l'output *yellow out* sarà acceso solo quando il valore del contatore aumentato a ogni ciclo di clock sarà minore di 4. Il contatore a 3 bit si resetterà dopo altri 4 cicli di clock, causando un effetto di accensione alternata dell'uscita.

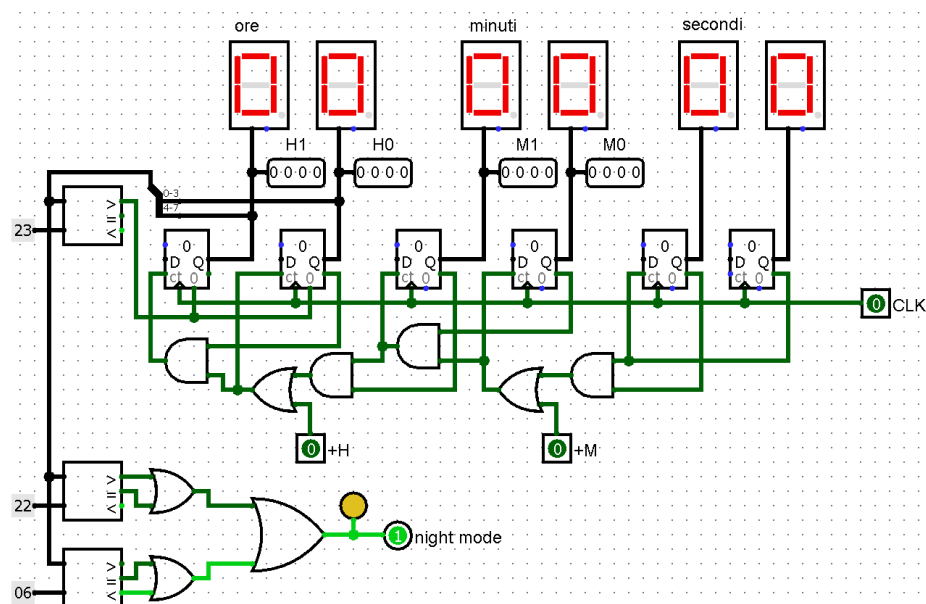
2.6 Circuito pedestrians-timer



Il circuito **pedestrains-timer** serve per controllare il timer dei pedoni attivato quando il semaforo **D** è verde o è giallo (negli stati 10 e 11). Ha due ingressi (*active* e il *clock*) e due uscite (*display* e *dot*). L'uscita è sempre nulla se il circuito non è attivo; quando viene attivato, il display deve mostrare un countdown da 9 a 0 (stato 10) e fare lampeggiare il *dot* due volte (stato 11). Analizziamo i componenti da sinistra verso destra:

- Il **contatore** serve per tenere traccia dello stato del circuito. Oltre al clock, in ingresso ha il segnale *active* **negato** collegato al **CLEAR**; in questo modo possiamo assicurarci che ad ogni attivazione il circuito partirà sempre dallo stato iniziale. Notare come il contatore faccia un countup da 0 a 15, piuttosto che un countdown.
- L'uscita del contatore è collegata a **tre componenti**:
 - Ad un **comparatore**, cui output servirà per stabilire quando attivare il punto lampeggiante.
 - Ad una **porta NOT** collegata ad un **sottrattore**, che sottrae 6. Questo è necessario per avere rispettivamente un countdown e un countdown che parte da 9 (al posto da 15).
 - Ad uno **splitter**, che seleziona il bit più a destra del numero.
- Il **multiplexer** è necessario per selezionare cosa mostrare sul display:
 - **Niente**, se l'ingresso *active* è spento.
 - L'**uscita del sottrattore**, se il valore del contatore è ≤ 9 .
 - **0**, se il valore del contatore è > 9 .
- L'ultima porta **AND** attiva l'uscita *dot* solo se:
 - Il valore del contatore è > 9 (ovvero il countdown è arrivato a 0).
 - Il valore del contatore è **dispari**. In questo modo l'uscita, al posto di rimanere fissa, lampeggia.

2.7 Circuito digital-clock



Il circuito **digital-clock** memorizza, mostra ed incrementa l'orario attuale in formato 24 ore. Inoltre, stabilisce se è bisogno attivare la *night mode* oppure no. Per migliorare la leggibilità, ci sono 6 display (non necessari) che mostrano in modo trasparente l'orario attuale, comprendente anche i secondi.

Il circuito è essenzialmente un insieme contatori collegati in serie con vari meccanismi per gestire il riporto, basandoci sull'uscita di **CARRY** che viene attivata quando il contatore raggiunge il suo massimo. Da destra verso sinistra:

- Il primo contatore viene incrementato ad ogni ciclo di clock; il suo valore massimo è 9.
- Ogni volta che il primo contatore raggiunge il suo massimo (9), il secondo contatore viene incrementato. Il suo valore massimo è 5.
- Ogni volta che sia il primo che il secondo contatore hanno raggiunto il loro massimo (59), il terzo contatore viene incrementato.
- e così via...

Se il valore degli ultimi due contatori (delle ore) è > 23 , vengono entrambi resettati a 0. Se il loro valore è compreso tra 22 e 06 allora viene attivata la *night mode*.

Per incrementare ore e minuti, sono presenti due ulteriori input **+H** e **+M** che, messi in OR con i riporti, aumentano artificialmente i valori del terzo e del penultimo contatore.