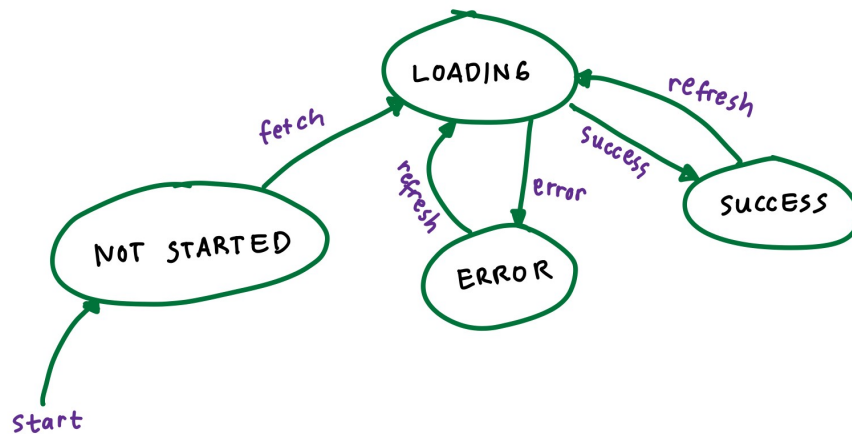


# Automi

[github.com/asdrubalini](https://github.com/asdrubalini)

February 7, 2022

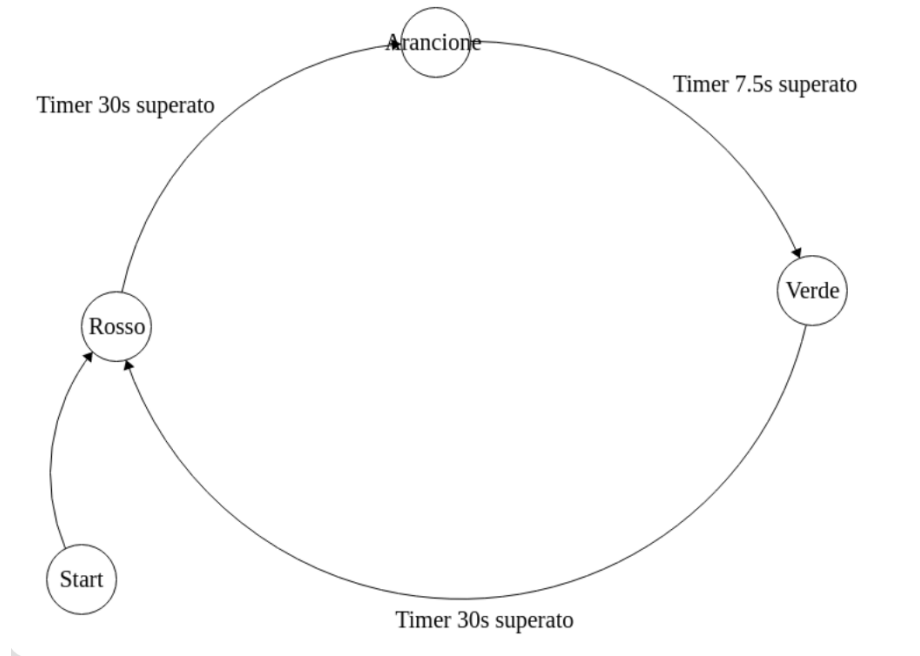
Automa: software generico che può funzionare su qualsiasi dispositivo programmabile. Un automa può essere progettato graficamente con l'utilizzo di due simboli: freccia e cerchio. Una volta che la progettazione è conclusa, l'automa deve essere programmato con un vero e proprio linguaggio di programmazione.



esempio di diagramma che descrive un'automa

Lo stato del sistema può mutare grazie ad un evento che può essere interno (un timer) o esterno (la pressione di un bottone). Lo stato di partenza è essenziale, e coincide con lo stato che viene eseguito dopo l'attivazione del programma.

Esercitazione: progettare l'automa che fa funzionare un semaforo. Verde e rosso devono avere la stessa durata (30s) mentre l'arancione deve avere un quarto della durata (7.5s).



PLC: Programmable Logic Controller  
 Nati per risolvere in modo automatico reti elettriche e automi elettromeccanici. Da un punto di vista elettronico sono reti combinatorie e automi con flip flop.

Oggi i PLC sono connessi in rete ma non è necessario che sia così. Originariamente i PLC venivano programmati in un linguaggio particolare chiamato KOP o Ladder che si basa sulle porte logiche, interruttori e bobine.

DSP permettono di eseguire azioni in tempi rapidissimi.

## 1 Progettazione centralina

Tutte le combinazioni logiche sono state radunate e influenzano un marker. Esiste un contatto SpecialMerker0.0 (SM00) che indica lo stato di accensione e quindi è sempre ON. SpecialMerker0.1 (SM01) è ON solo durante il primo ciclo. Ciclo = lettura degli ingressi. Possiamo usarlo per passare allo stato di accensione.

Per ogni stato serve un flag (ovvero un merker).

1. Merker 0.0 (rete AND)
2. Merker 1.0 (fermo)

3. Merker 1.1 (irrigazione)

4. Merker 1.2 (avaria)

5 network per risolvere l'automa + 1 per la rete AND



(S) = bobina di settaggio

Ogni volta che cambio lo stato, setto quello nuovo e resetto quello precedente.

In tutto 6 network.

Aggiungere anche un allarme acqua che viene attivato quando non c'è abbastanza acqua.

Tabella di stato (stato -> uscita)

1. Irriga → Q1.0

2. Allarme → Q1.1

3. Livello → Q1.2

Mettere tutto in bella e consegnare in PDF

## 2 Perchè usiamo i sensori:

1. Perchè sono una tecnologia emergente

2. Perché potrebbero essere utili a chi frequenterà l'università

3. Perché potrebbero essere richiesti nell'esame di Stato

Solitamente una rete di sensori è composta da sensori wireless. Spesso, uno dei problemi da affrontare è l'alimentazione.

### 3 Come lavora un PLC

Nomenclatura:

1. Merker: M

2. Ingressi: I

3. Uscite: U

4. Temporizzatori: T

Tutte queste cose occupano risorse nella memoria del PLC.

Il PLC lavora sempre ciclicamente, ogni ciclo è fatto da questi step:

1. Lettura ingressi → Immagine di processo degli ingressi d'ora in poi gli ingressi possono cambiare ma il PLC fa affidamento a quelli appena letti

2. Esecuzione programma → Elabora il programma scritto e va a modificare qualcosa nelle risorse ma finché non arrivo in fondo al programma e scrivo le uscite, le uscite non cambiano

3. Scrittura delle uscite

Un sistema realtime assicura il fatto che ogni millisecondo gli ingressi vengono letti.

#### 3.1 Temporizzatori

Esistono tre tipi di temporizzatori:

1. TON → dopo un certo tempo si accende

2. TOFF → accende subito qualcosa e poi lo spegne

3. TONR

Un timer può essere visto come una funzione lineare che ogni base tempi viene incrementata. Quando il conteggio supera il tempo che ho impostato, l'uscita viene modificata

Un temporizzatore occupa un'area di memoria di 2 bytes. Un bit contiene l'informazione del contatto.

Fino alla terza generazione dei PLC, le basi tempi erano fisse per ogni temporizzatore. Nei PLC della nuova generazione posso decidere io la base tempi, essendo puramente software.

Traccia: costruire un'automa che faccia ping-pong. Uno stato chiamato ping ed uno chiamato pong.

1. Desc automa
2. Tabella ingressi
3. Tabella risorse
4. Tabella temporizzatori
5. Tabella uscite

Nell'esercitazione base del timer bisogna usare necessariamente 9 righe.  
Versione aggiornata con il lampeggio di un led alla fine

## 4 Reti di sensori

Le reti di sensori predittive vengono create dagli ingegneri e permettono, attraverso funzioni matematiche, di predire la rottura di qualcosa.

1. Problemi di alimentazione: ad esempio se installo un sensore che può rilevare in anticipo le valanghe, dobbiamo tenere conto di quanto può durare la batteria
2. La comunicazione wireless ha un raggio limitato. Se voglio coprire distanze ampie devo trovare il modo di avere dei modi che fanno da ponte tra una rete e quella successiva

Tante volte le innovazioni tecnologiche partono da esigenze di tipo militari. MEMS: elementi sensitivi che forniscono il dato in forma digitale.

## 5 Esercitazione semaforo

Centro di lavoro: una macchina che esegue delle operazioni in modo sequenziale. Creare un piccolo sistema che riusciamo a programmare che poi potremo anche gestire da remoto.

Progettazione di un sistema semaforico per due vie, quella principale e quella secondaria.

Ogni stato cambia con il passare del tempo. Sarà necessario definire le durate. Inoltre, durante la notte i semafori non sono normalmente operativi ma lampeggiano di giallo. Abbiamo quindi due cicli distinti, un ciclo giorno ed un

ciclo notte che si occupa semplicemente di far lampeggiare i semafori. A volte il ciclo notte coincide anche con il "fuori servizio".

## 5.1 Possibili stati

Abbiamo 5 stati in cui il sistema si può trovare in ogni momento:

- 1.
- 2.
- 3.
- 4.
5. Lampeggiante: attivabile da interruttore fuori servizio o durante la notte.

## 5.2 Contatti

Contatti:

1. Fuori servizio (attivo alto quando il sistema viene messo fuori servizio)
2. Orologio notte (attivo alto quando l'orologio segna notte)

Il passaggio tra notte e giorno non può essere fatto a caso ma dovrà essere fatto in condizioni di sicurezza. Se sta lampeggiando devo decidere come entrare nel ciclo normale, magari con entrambe le luci rosse accese.

## 5.3 Struttura documentazione

Passi da svolgere:

1. Descrivo
  2. Definisco ingressi ed uscite
  3. Scelgo le risorse che uso nel PLC
  4. Diagramma degli stati
  5. Tabella dei temporizzatori
  6. Registrare un video del funzionamento
- Da consegnare entro Lunedì prossimo.

## 6 Rete di sensori wireless

Obiettivi di una rete di sensori wireless: quali obiettivi ci si deve porre affinché una rete svolga il suo lavoro in efficacia ed efficienza.

Le informazioni raccolte devono essere affidabili.

Sono importanti le certificazioni.

Se prendo un Arduino qualsiasi e metto in piedi una rete LoraWAN, sono affidabile nella raccolta delle informazioni? No, perchè Arduino non ha nessuna certificazione industriale.

Un'azienda non può utilizzare una board non certificata.

Certezza dei dati = nel momento in cui vengono forniti più parametri, i valori forniti devono essere coerenti. I dati che vengono forniti devono essere compatibili tra loro.

I dati devono essere sempre disponibili.

Ci sono anche dei motivi secondari, che non significa che sono di secondo ordine.

Configurazione statica della rete = far sì che i dispositivi riescano a dialogare in maniera efficiente.

Supponiamo che stia utilizzando uno standard di trasmissione. Dopo due anni quello standard riceve un upgrade. Devo stare attento ad aggiornare tutto. Minimizzazione del costo economico dei nodi.

Il progettista deve bilanciare efficienza energetica ed efficacia applicativa.

Devo essere bravo a scrivere il codice in modo tale che stia all'interno della memoria del dispositivo.

Devono essere anche energeticamente efficienti.

Se ho delle reti di sensori che raccolgono dei dati, il flusso delle informazioni prevalentemente va dai sensori al gateway, anche se in alcune situazioni il flusso funziona nel verso opposto.

Alcuni microcontrollori hanno i sensori integrati. Grazie alla comunicazione tra nodi, un sensore può comunicare a più lunga distanza.

Argomenti affrontati:

1. Obiettivi
2. Caratteristiche
3. Problematiche

Fare una scaletta