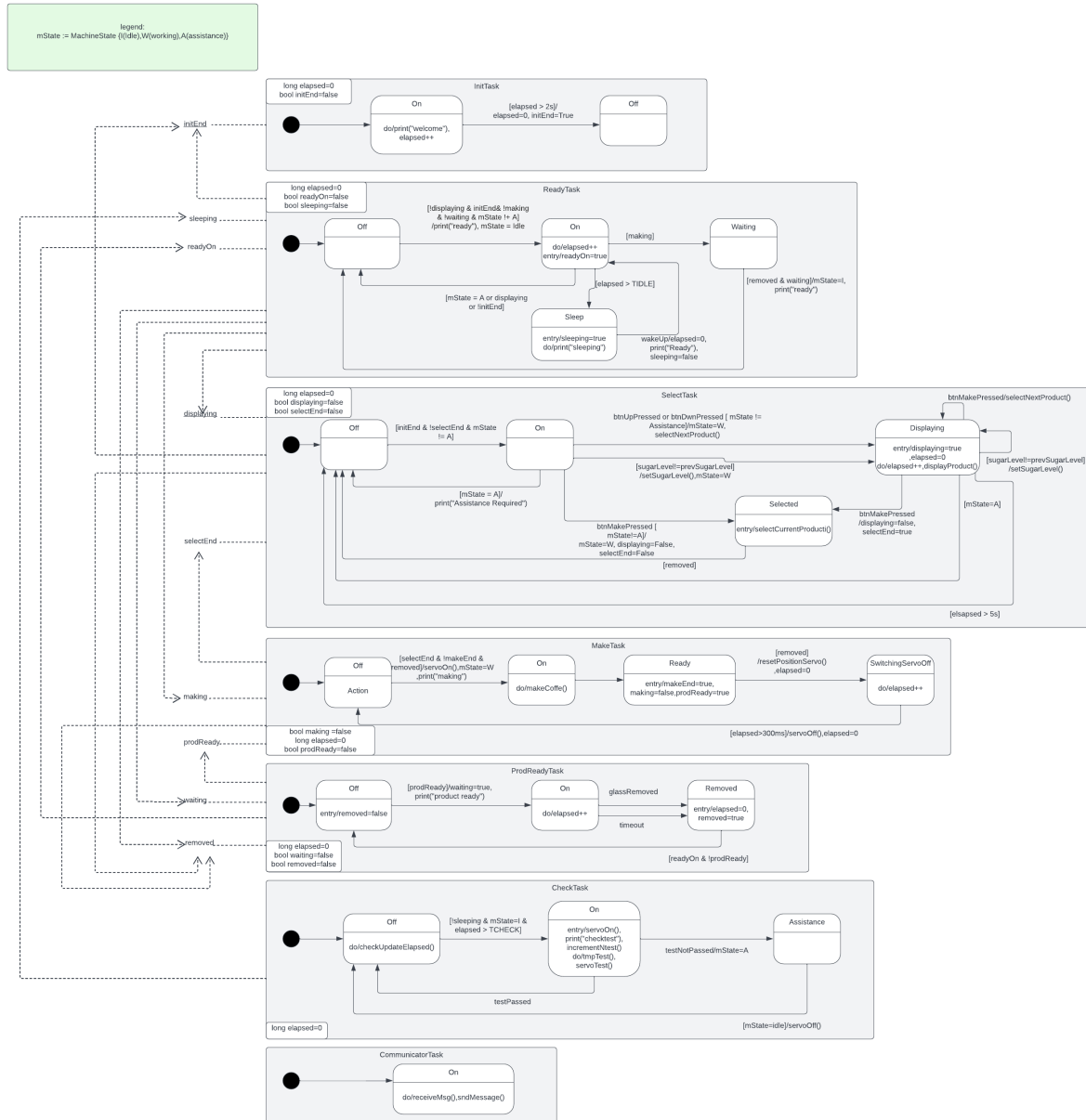


Report Assignment 2 -Sistemi Embedded & IoT- Manuel Luzietti

Ho deciso di scomporre il sistema in 7 task:

- task di inizializzazione InitTask:
 - si occupa di inizializzare il sistema all'avvio
- task ReadyTask:
 - si occupa di gestire in particolare il caso di sleep e copre gli scenari in cui la macchina é ferma
- task di Selezione SelectTask:
 - si occupa di gestire la selezione del prodotto
- task di Preparazione MakeTask:
 - si occupa della preparazione del prodotto selezionato
- task ProdReadyTask:
 - copre lo scenario in cui il prodotto è pronto e deve essere prelevato, oltre a sincronizzarsi con gli altri task per tornare allo stato iniziale Idle.
- task di check CheckTask:
 - si occupa di gestire i check periodici
- task di comunicazione seriale CommunicatorTask:
 - si occupa di gestire la comunicazione con il programma java di gestione e monitoraggio della macchina.

Segue diagramma a blocchi:



Le linee a sinistra identificano le dipendenze tra i vari task gestite con variabili modificabili solo dal task proprietario ma osservabili dai task indicati. Infatti il sistema in questione ha diverse dipendenze temporali.

L'architettura del sistema è task based e sincrona, basata su uno scheduler che esegue periodicamente i vari task.

Ho pensato che fosse una buona idea astrarre un modello del sistema che ha un suo stato a se stante che varia tra Idle, Working, Assistance. L'istanza del modello è ottenibile attraverso metodi statici della classe MachineModelSingleton, che è stata basata su pattern Singleton così che tutti i task possano operare sulla stessa istanza del modello. Ciò porta in parte a violare il principio di incapsulamento ma risulta in un minor numero di dipendenze tra i vari moduli. In più ciò non soffre dei problemi di

concorrenza soliti del pattern Singleton in quanto l'architettura, seppur logicamente concorrente, risulta fisicamente sequenziale.

Potrei aver scelto forse una granularità di modularizzazione troppo fine. Con qualche task in meno avrei avuto meno dipendenze.

Segue immagine del circuito:

