

Sistemas e Automação 2022/23

Trabalho Prático 1

Solução

SOLUÇÃO

A solução para este problema depende do nível de otimização que se pretende implementar. Podemos identificar os seguintes níveis crescentes de otimização:

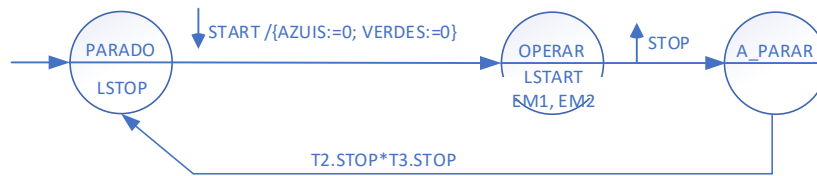
1. Só é gerada uma nova peça quando a última tiver saído do 'sistema' (i.e. ultrapassando ST₂ / ST₃).
2. Sempre que for transportada uma peça de T₁→T₂ (T₄→T₃), pode ser gerada uma nova peça em T₁ (T₄). Contudo, T₂ e T₃ são tratados como um único tapete.
3. Idem anterior, mas T₂ e T₃ podem funcionar de forma independente desde que não existam conflitos. Exemplo: transportar em 'simultâneo' uma peça Azul de T₁→T₂ e uma peça Verde de T₄→T₃.
4. Idem anterior, mas permitir que mesmo com algum tipo de conflito T₂ e T₃ possam funcionar de forma independente. Exemplo: transportar em 'simultâneo' uma peça Azul de T₁→T₂ e uma peça Azul de T₄→T₃. Esta última peça aguarda em T₃ até T₂ ficar livre para fazer o transporte.

O modelo apresentado aqui cobre o nível de mais elevado (4) e são compostos pelas seguintes máquinas de estados:

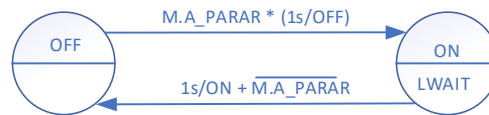
- MODOS: controla os modos do sistema
- T₁, T₂, T₃ e T₄: controla cada um dos tapetes
- PISCA: controla a luz LWAIT
- LT₂ e LT₃: semáforos que controlam o acesso a T₂ e T₃.

MÁQUINAS DE ESTADOS

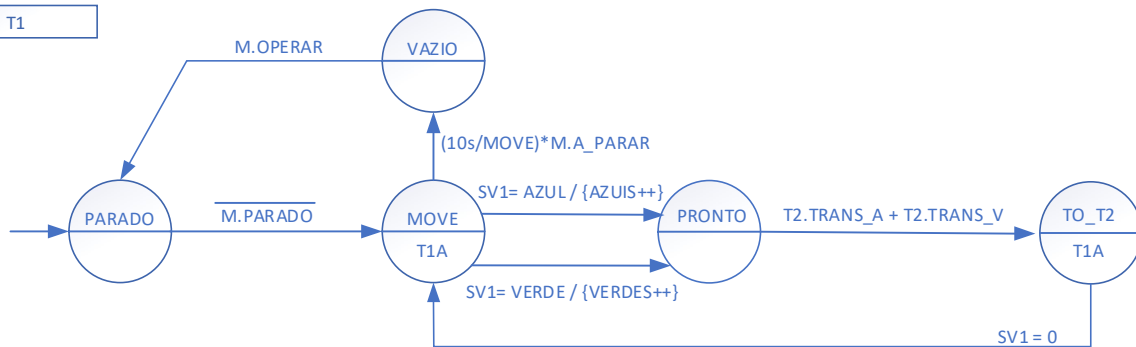
MODOS



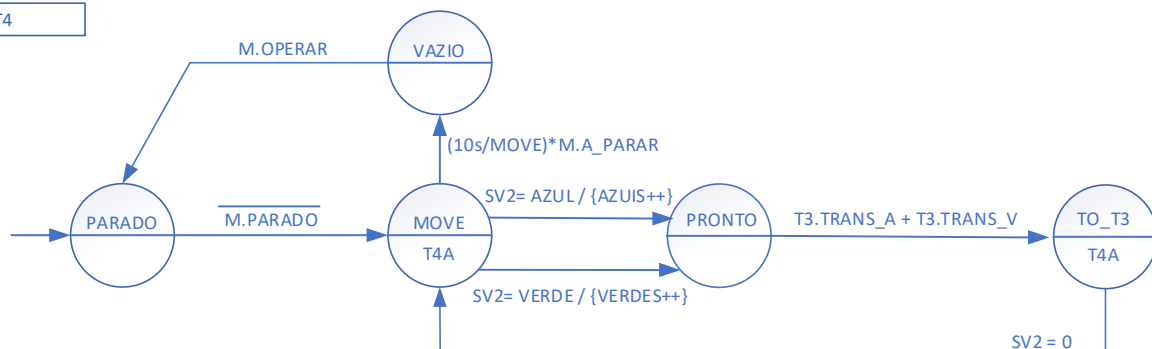
PISCA

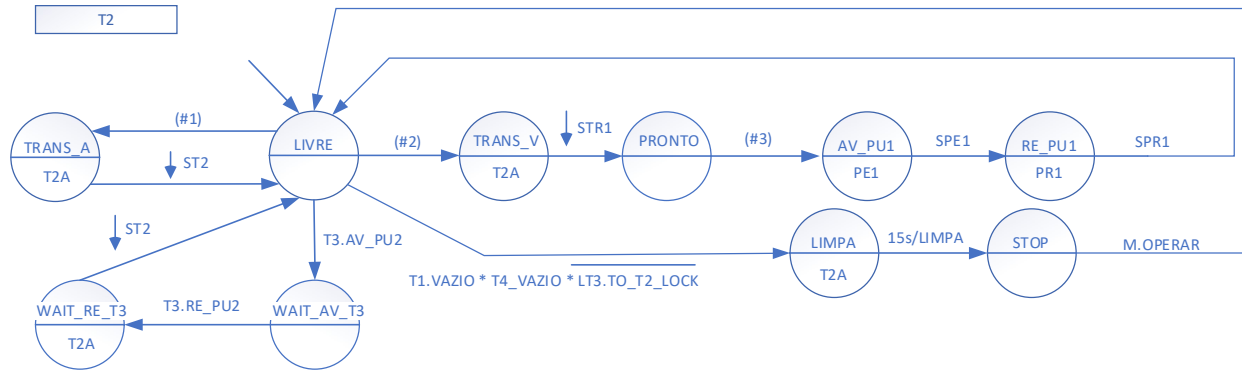


T1



T4



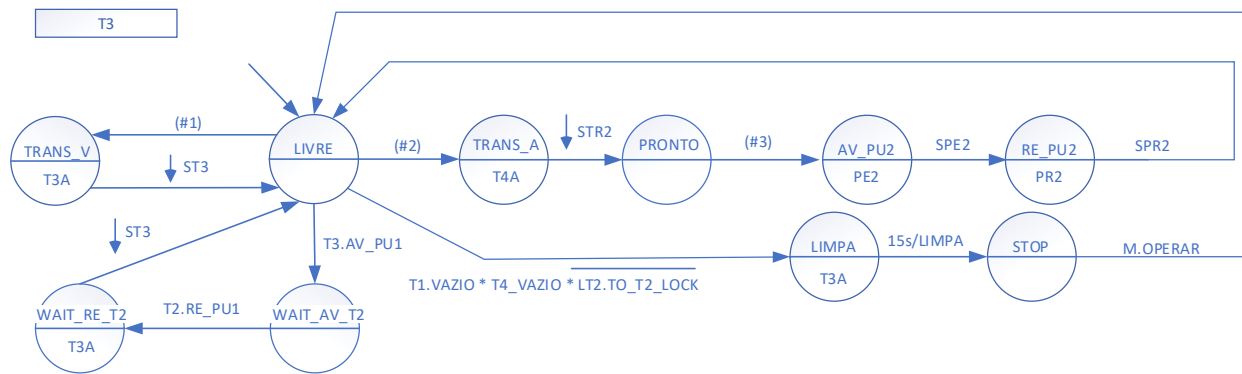


(#1) = T1.PRONTO * SV1=AZUL * LT2.UNLOCK * (LT3.UNLOCK + LT3.LOCK)

vermelho = evitar deadlock

(#2) = T1.PRONTO * SV1=VERDE * LT2.UNLOCK * (LT3.UNLOCK + LT3.LOCK) * (T4.PRONTO * SV2=AZUL * LT3.UNLOCK * (LT2.UNLOCK + LT2.LOCK))

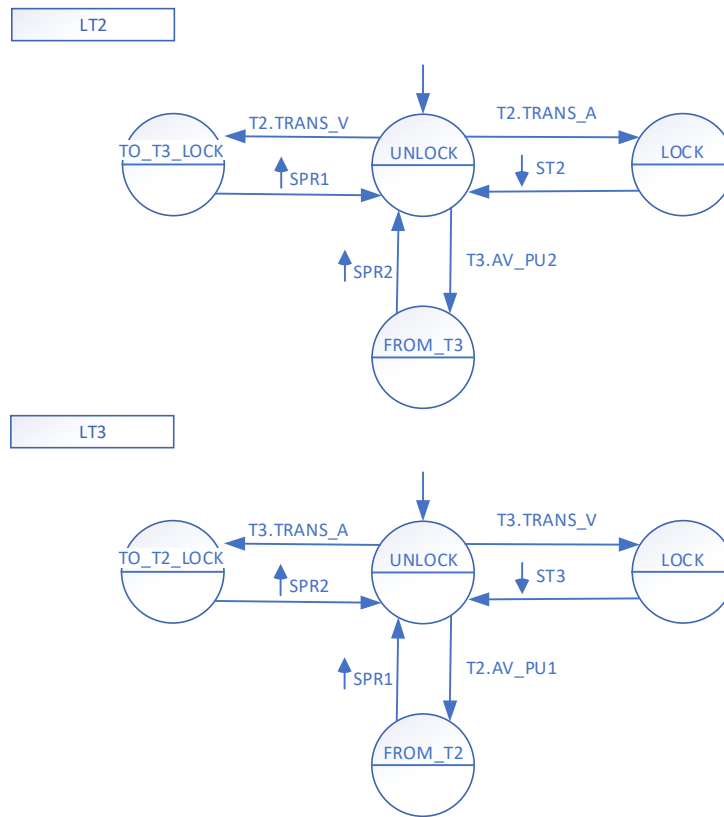
(#3) = LT3.UNLOCK * (T2.PRONTO + LT2.UNLOCK)



(#1) = T4.PRONTO * SV2=AZUL * LT3.UNLOCK * (LT2.UNLOCK + LT2.LOCK)

(#2) = T4.PRONTO * SV2=VERDE * LT3.UNLOCK * (LT2.UNLOCK + LT2.LOCK)

(#3) = LT2.UNLOCK



FIM