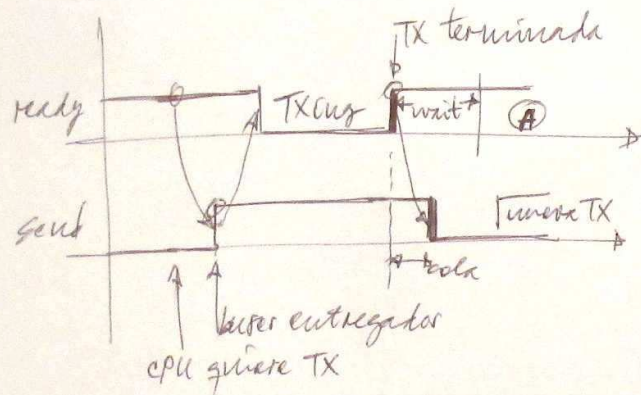
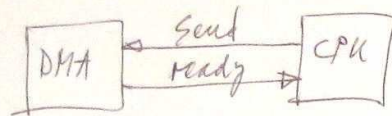
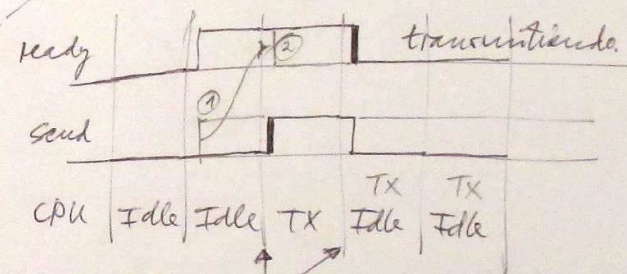
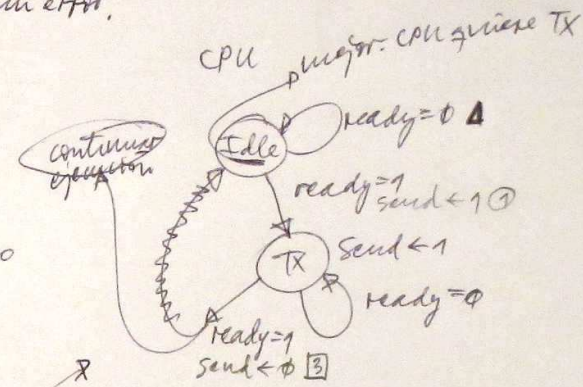
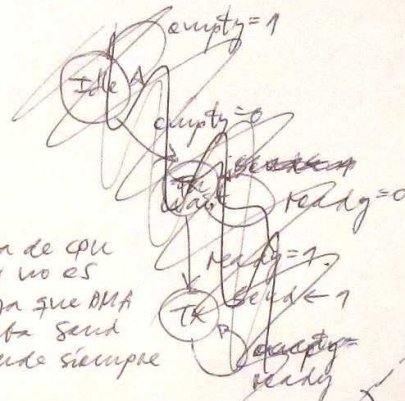


(E) en algunos casos no se puede estar indef.  $\rightarrow$  superado un timeout habrá que alzar un error.



La espera de CPU por ready  $\neq 0$  no es necesaria, ya que DMA sólo comprueba send en Idle, donde siempre está libre.



Si DMA ready es Ready Moore send  $\leftarrow 0$  en un ciclo.

CPU no se bloquea en TX  $\rightarrow$  MAL (\*)

SOLUCIONES:

- 1) send es Ready
- 2) ready es Ready

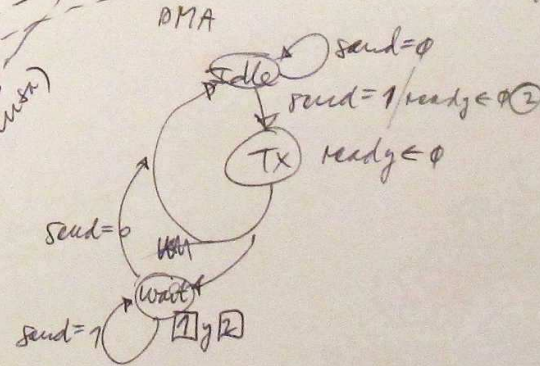
angelph  
dic-2014

A para la siguiente TX hay dos posibilidades:

1. DMA espera un  $\phi$  en send  $\rightarrow$  esto obliga a CPU a pasar por  $\phi$  antes de volver a TX.
2. DMA espera un tiempo acordado con CPU para asegurar que CPU quiere la TX.

Si no se siguen los anteriores, DMA puede reiniciar TX erróneo por la falta del 1 en send

PROB. 1A



(\*) realmente no es un problema  $\rightarrow$  CPU puede seguir operando  $\rightarrow$  si quiere TX de nuevo se bloqueará en la espera porque ready=0. para esto habría que cambiar CPU TX  $\rightarrow$  en vez de esperar ready=1 debería esperar ready=0, para asegurar que hay respuesta del DMA. (E)