

C r u c e d e f e r r o c a r r i l e n

Statecharts

**Introducción a la
conurrencia**

**Requisitos temporales
simples**

Statecharts

- ❑ Lenguaje gráfico con semántica ejecutable
 - ❑ Se basa en el formalismo típico para describir FSM, aunque lo extiende de manera brillante.
- ❑ Imposible hablar de tipos
- ❑ Los conceptos básicos son eventos y estados; a ellos se agregan condiciones para las transiciones, super-estados, concurrencia, temporización, historia, etc.

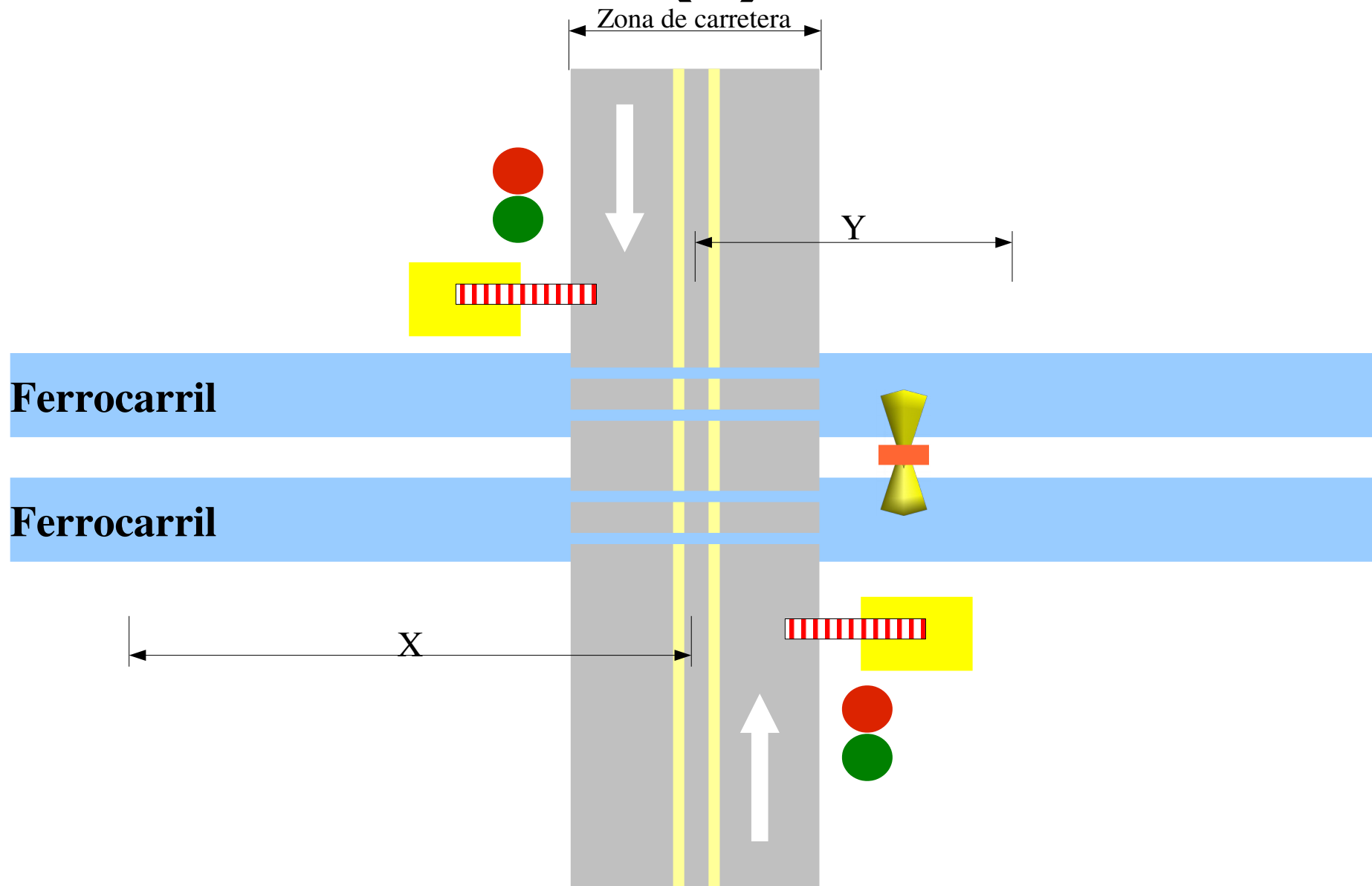
Cruce ferroviario

- ❑ Se desea automatizar el control de la barrera de un cruce de ferrocarril en dos sentidos.
- ❑ Por tal motivo, se instalará un *chip* que ejecutará un programa que recibirá datos desde varios sensores y actuará sobre las barreras, los semáforos y una alarma.
- ❑ Los requerimientos del cliente son los siguientes:
 - ❑ Si un convoy llega a x metros del centro del cruce, se deben bajar las barreras, poner los semáforos en rojo y debe sonar la alarma.

Cruce ferroviario (2)

- ❑ Una vez que el último vagón del convoy está a más de y metros del centro del cruce y alejándose, se debe subir la barrera, poner el semáforo en verde y apagar la alarma.
- ❑ Si por alguna razón un convoy llega a x metros del cruce pero nunca alcanza la zona de la carretera, se debe proceder como si el tren hubiese superado los y metros.
- ❑ Si, luego el convoy reanuda la marcha, se deberán activar las protecciones inmediatamente y el sistema continuará con su funcionamiento normal.

Cruce ferroviario (3)



Algunas designaciones

- ❑ La distancia de seguridad para bajar la barrera $\approx X$
- ❑ La distancia de seguridad para subir la barrera $\approx Y$
- ❑ Un tren cruza la distancia X acercándose al cruce por la vía $v \approx \text{Tren}X(v)$
- ❑ Un tren alcanza la zona de carretera por la vía $v \approx \text{Tren}C(v)$
- ❑ Un tren se detiene en la vía v en la zona de peligro $\approx \text{Tren}D(v)$
- ❑ El último vagón de un tren pasa la distancia Y alejándose del cruce por la vía $v \approx \text{Tren}Y(v)$

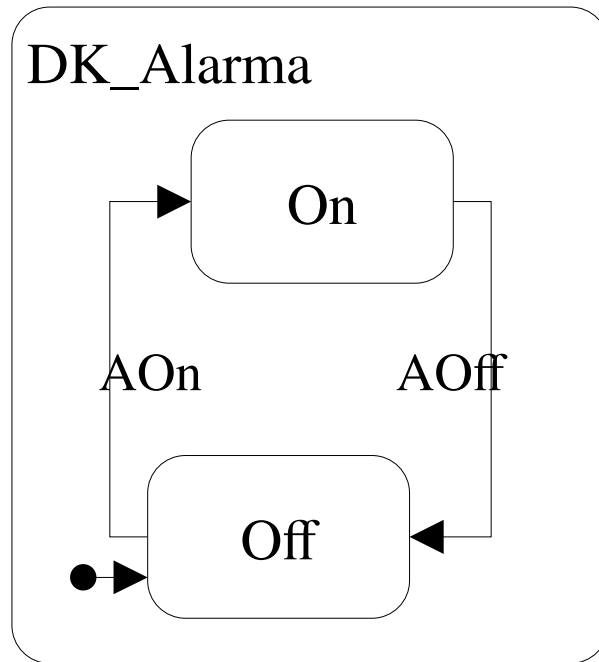
Algunas designaciones (2)

- ❑ El sistema enciende la alarma $\approx AOn$
- ❑ El sistema apaga la alarma $\approx AOff$
- ❑ El sistema baja la barrera $b \approx BBarrera(b)$
- ❑ El sistema sube la barrera $b \approx SBarrera(b)$
- ❑ El sistema pone el semáforo s en rojo $\approx Rojo(s)$
- ❑ El sistema pone el semáforo s en verde $\approx Verde(s)$

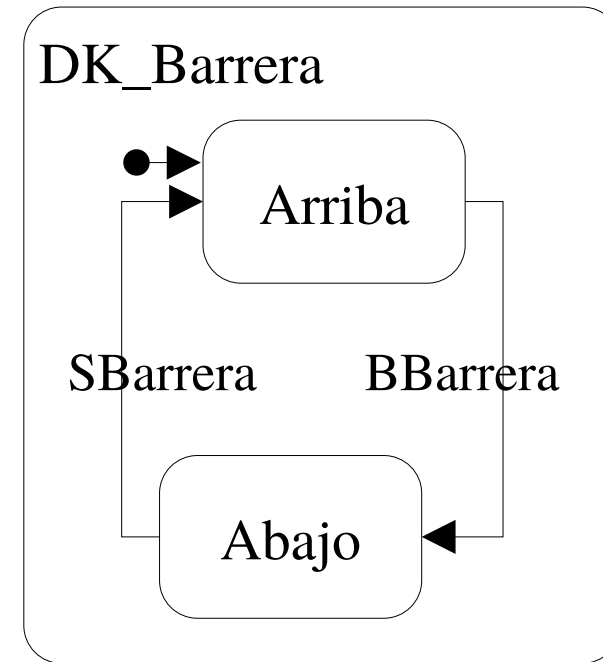
Control

F	MC	EC	U	S	F	MC	EC	U	S
X		■	■		AOn	■			■
Y		■	■		AOff	■			■
TrenX		■	■		BBarrera	■			■
TrenY		■	■		SBarrera	■			■
TrenC		■	■		Rojo	■			■
TrenD		■	■		Verde	■			■
TrenE		■	■		TrenS		■	■	

Alarma y barrera



El semáforo es idéntico

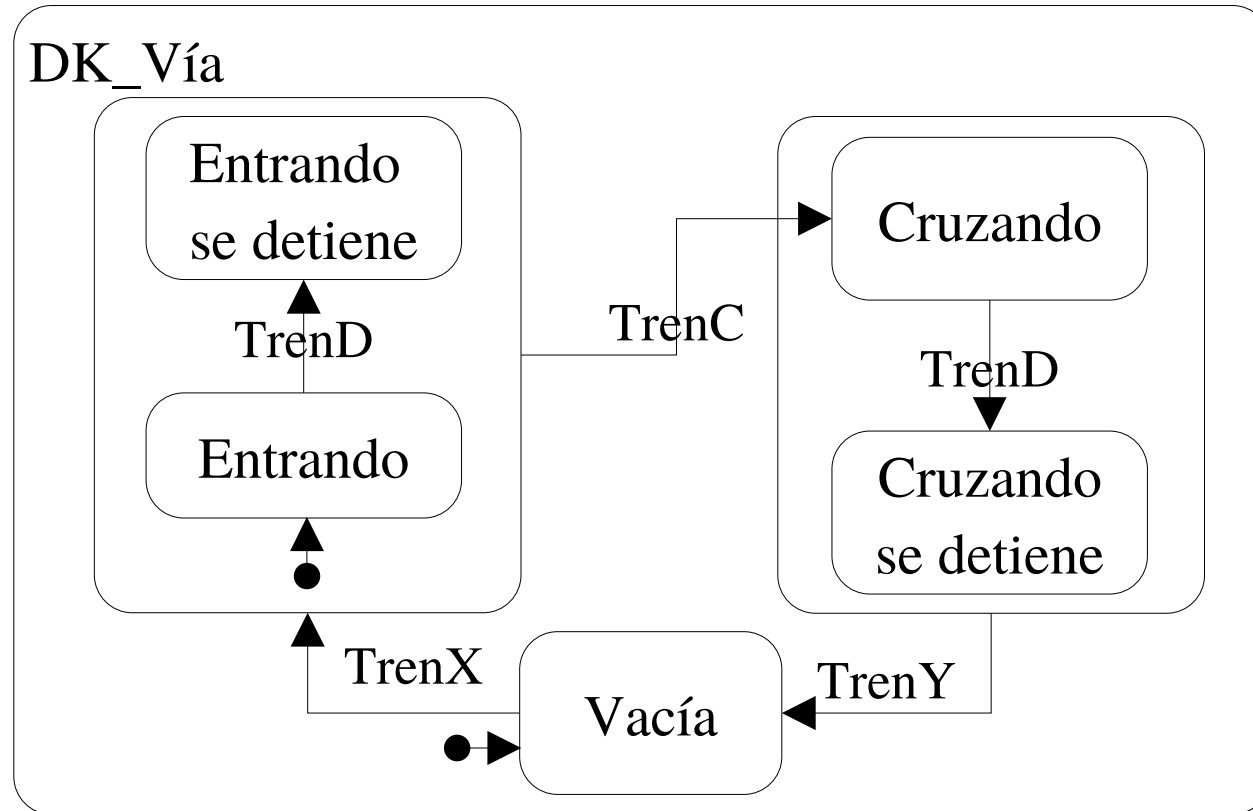


Super-estado

Estado por defecto

Eventos y designaciones

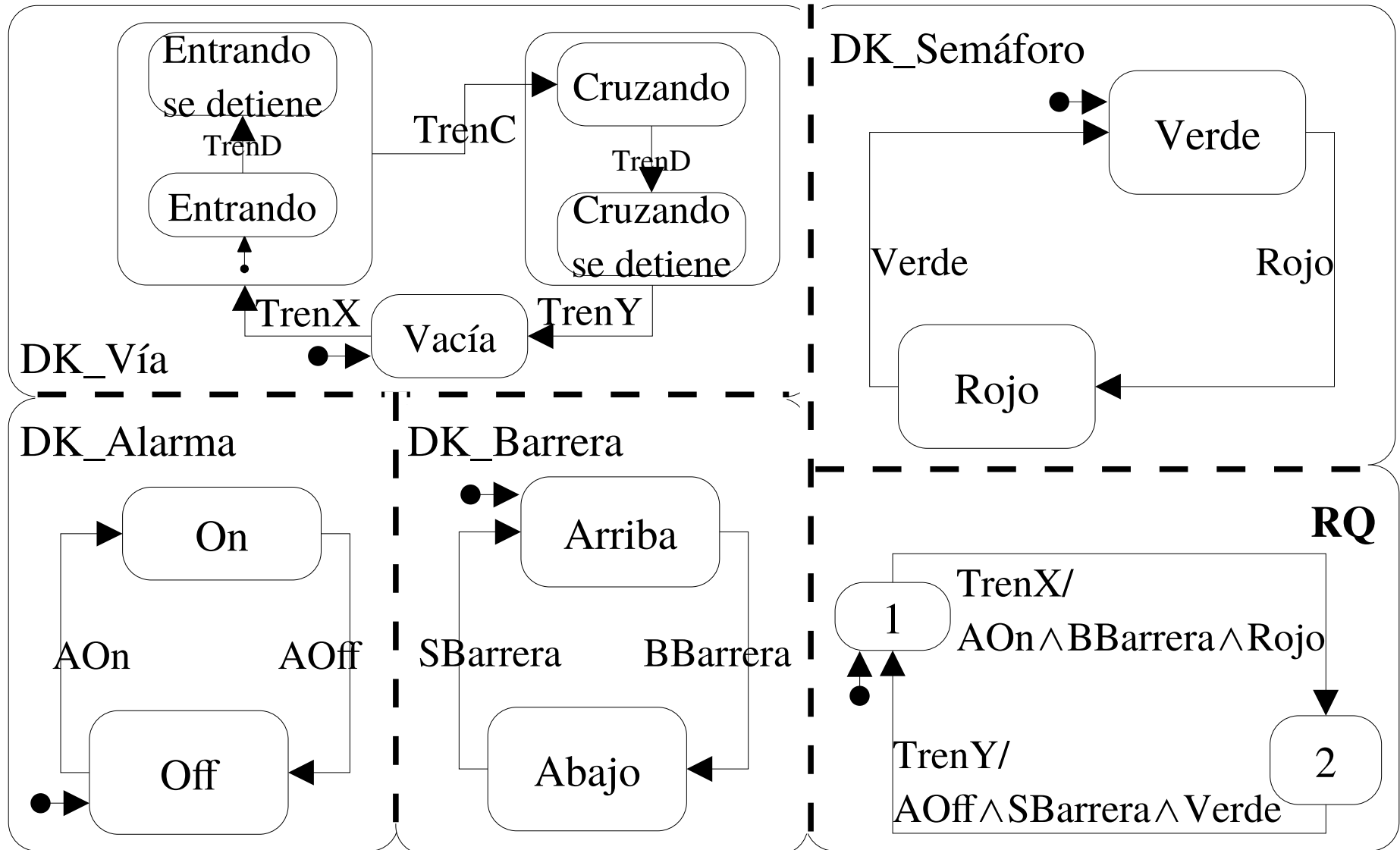
Una vía



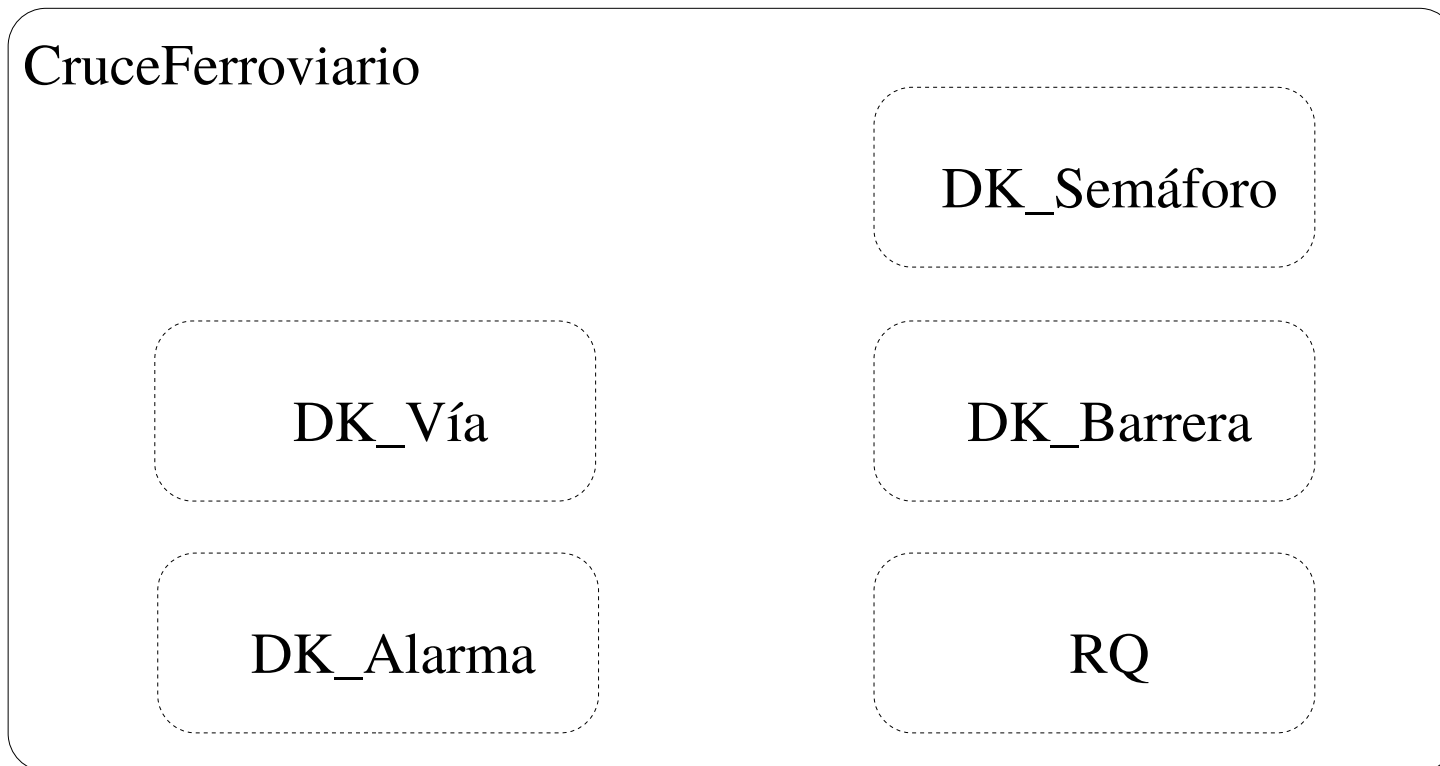
Salida común de un super-estado
Entrada común a un super-estado

RQ simple para una vía

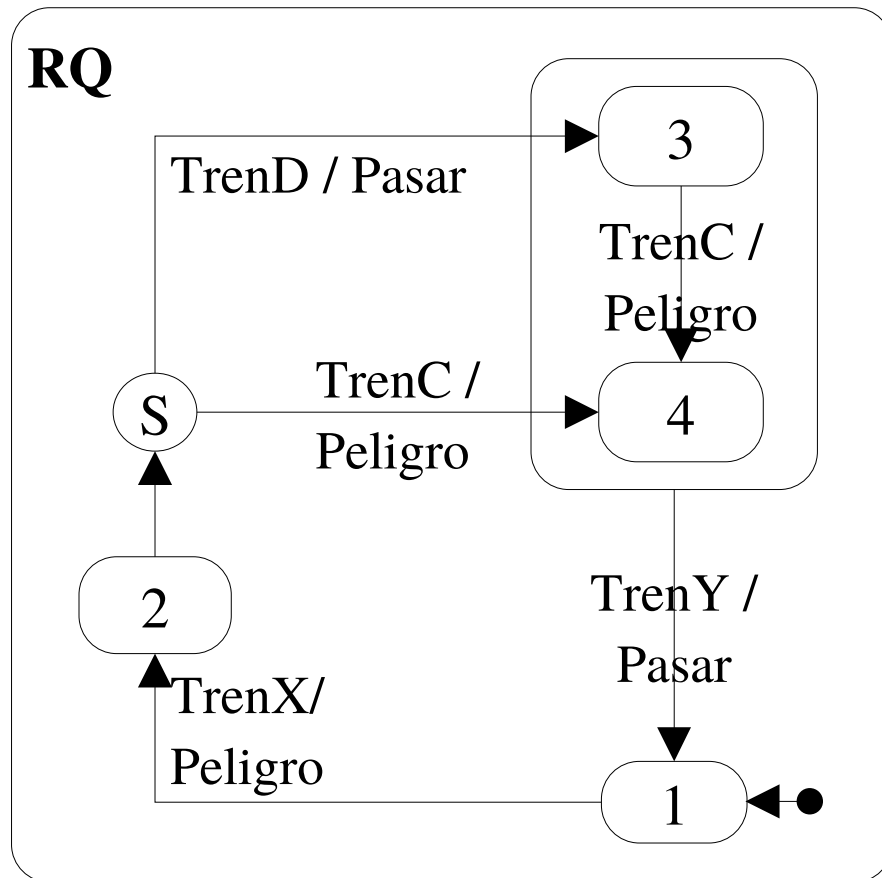
Eventos cerrados por \wedge y \vee
Concurrencia



Otra notación para estados-AND



RQ completo para una vía



$\text{Peligro} \stackrel{\text{def}}{=} A\text{On} \wedge B\text{Barrera} \wedge \text{Rojo}$

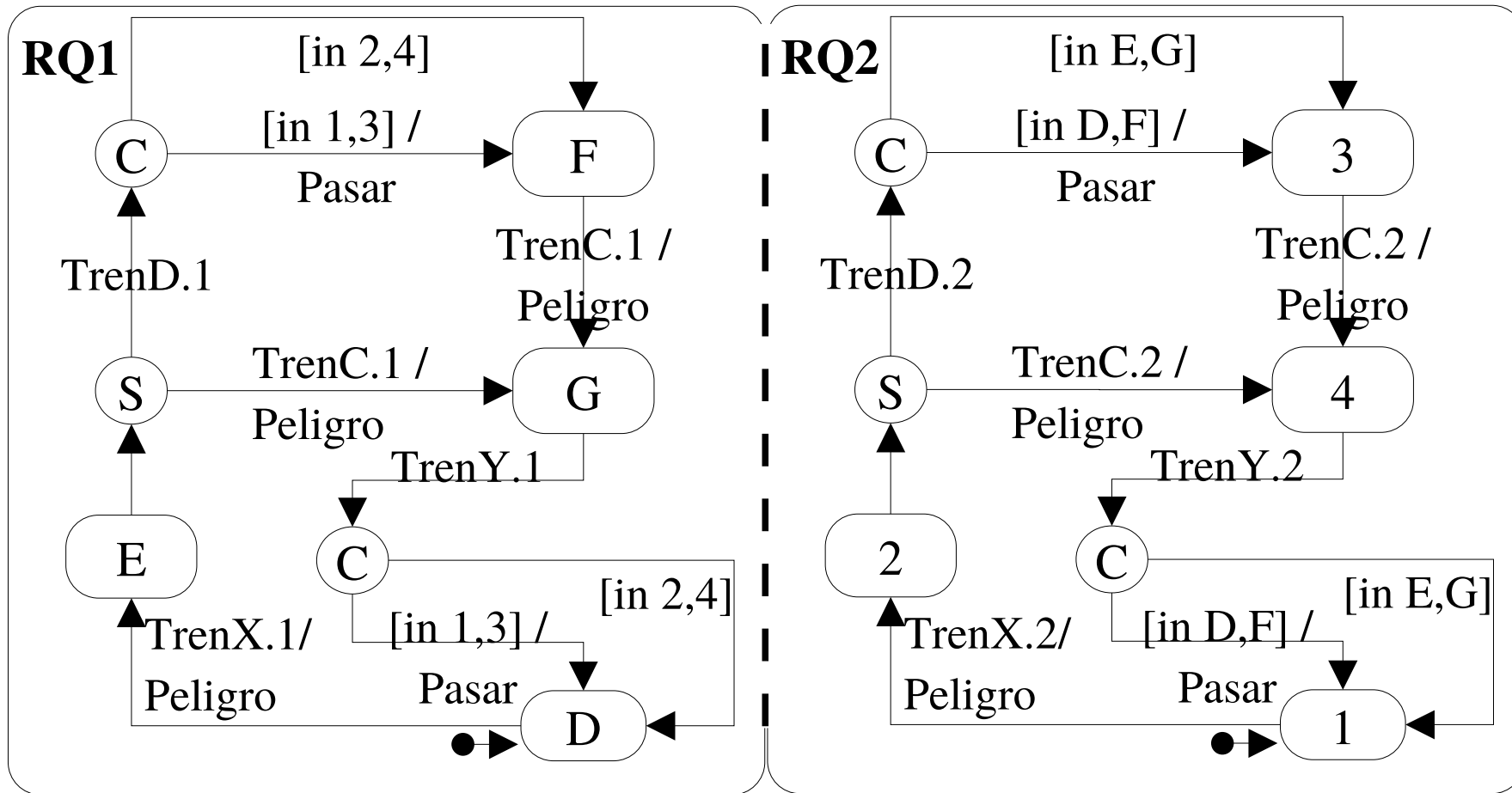
$\text{Pasar} \stackrel{\text{def}}{=} A\text{Off} \wedge S\text{Barrera} \wedge \text{Verde}$

Conector de selección

RQ para dos vías

- ❑ No es tan simple pues hay que considerar el caso en que dos trenes pasan uno por cada vía y casi al mismo tiempo.
- ❑ La propiedad que hay que preservar es que se permite el paso recién cuando el último de los trenes terminó de pasar.
- ❑ La coordinación entre las dos vías la lograremos utilizando transiciones condicionales.

RQ para dos vías (2)



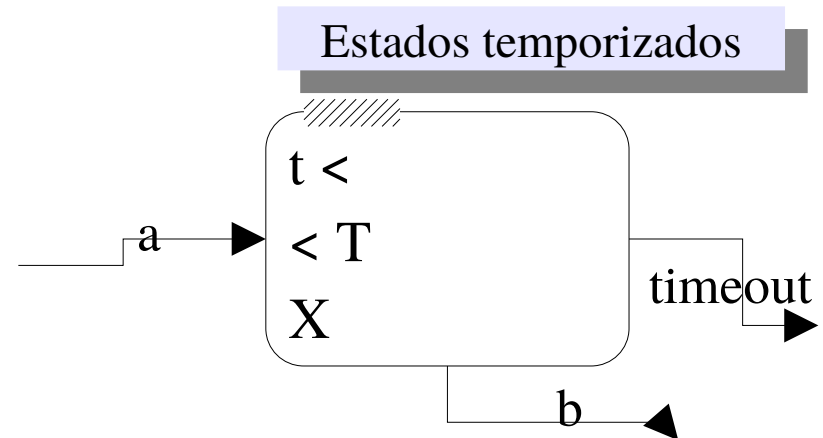
$\text{Peligro}.i \stackrel{\text{def}}{=} A_{on} \wedge B_{Barrera.i} \wedge Rojo.i$

$\text{Peligro} \stackrel{\text{def}}{=} \text{Peligro}.1 \wedge \text{Peligro}.2$

$\text{Pasar}.i \stackrel{\text{def}}{=} A_{off} \wedge S_{Barrera.i} \wedge Verde.i$

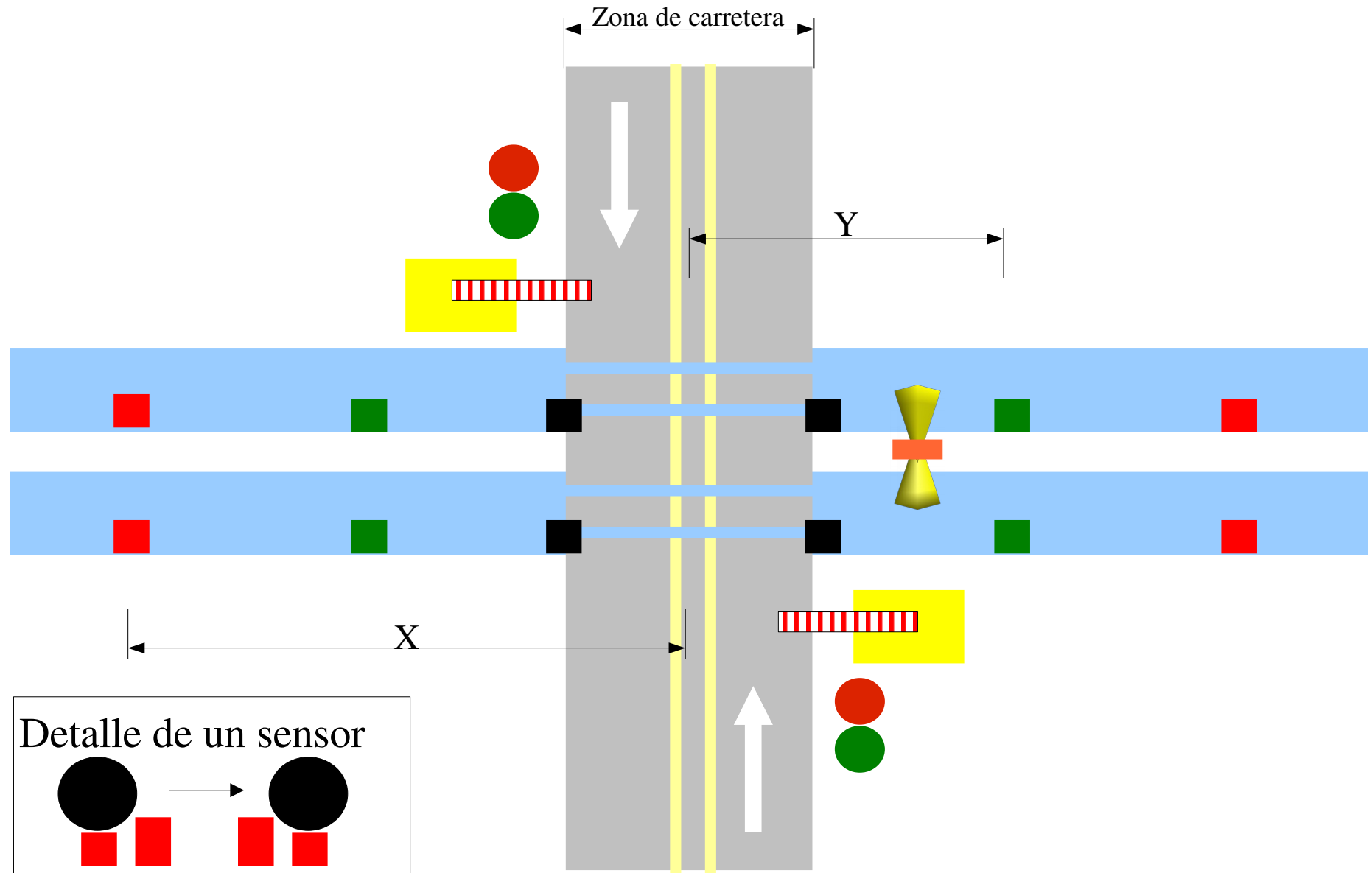
$\text{Pasar} \stackrel{\text{def}}{=} \text{Pasar}.1 \wedge \text{Pasar}.2$

La especificación



- ❑ RQ no es S por dos motivos:
 - ❑ hay varios fenómenos no compartidos ($TrenX$)
 - ❑ hay referencia al futuro ($TrenY$)
- ❑ Los fenómenos no compartidos se reemplazarán por señales enviadas por detectores.
- ❑ Las referencias al futuro se reemplazarán por tiempos de espera.
- ❑ Para describir los tiempos de espera utilizaremos los estados temporizados de Statecharts.

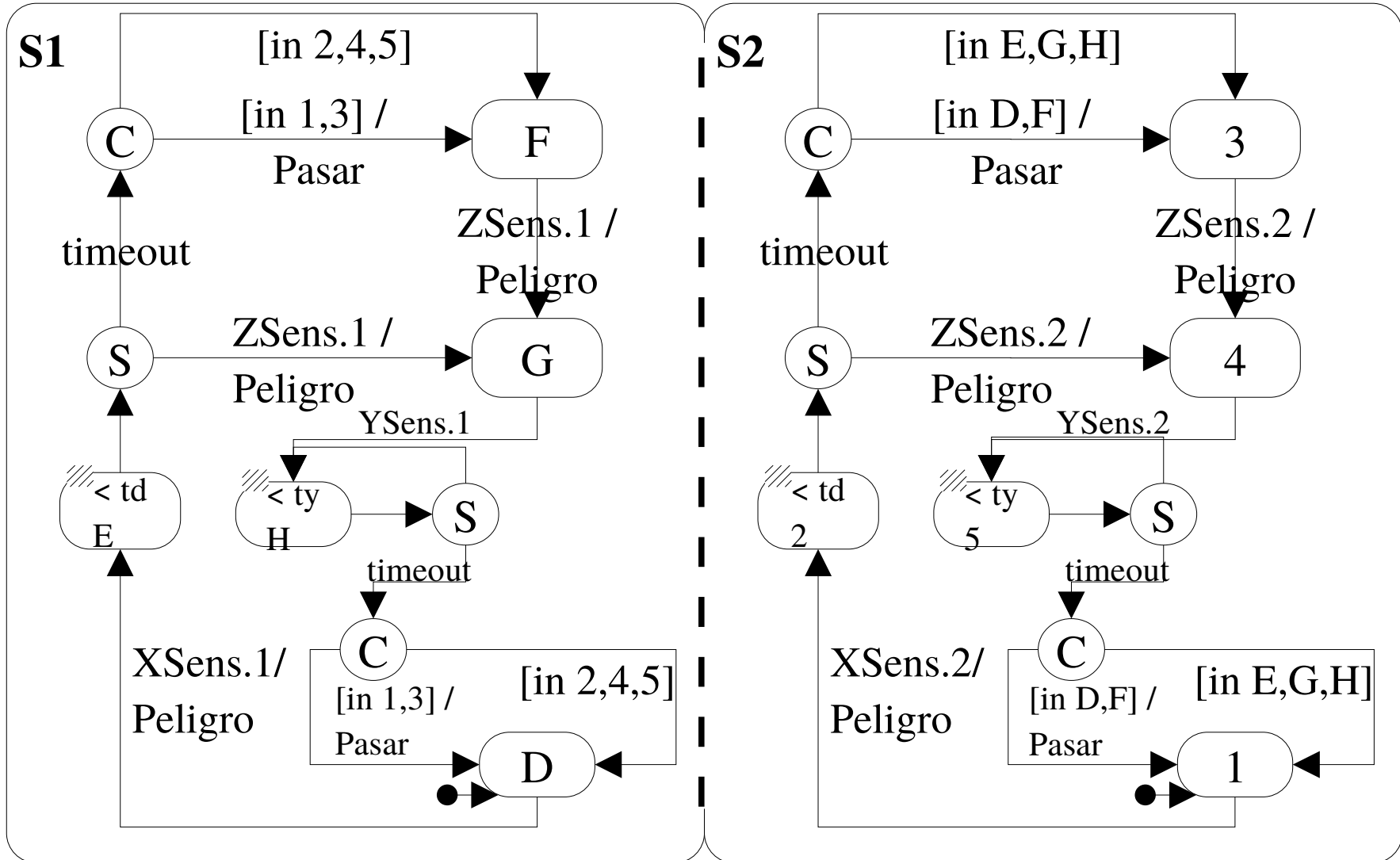
Cruce ferroviario con sensores



Más designaciones

- ❑ Los sensores X y Z envían una señal sólo cuando el convoy se dirige hacia el cruce; los sensores Y sólo cuando el convoy se aleja.
- ❑ El sensor de distancia X de la vía v envía una señal $\approx XSens(v)$
- ❑ El sensor de distancia Y de la vía v envía una señal $\approx YSens(v)$
- ❑ El sensor de zona de carretera de la vía v envía una señal $\approx ZSens(v)$
- ❑ Tiempo de espera para determinar la detención de un tren $\approx td$
- ❑ Tiempo de espera desde que se recibió la "última" señal de un sensor $Y \approx ty$

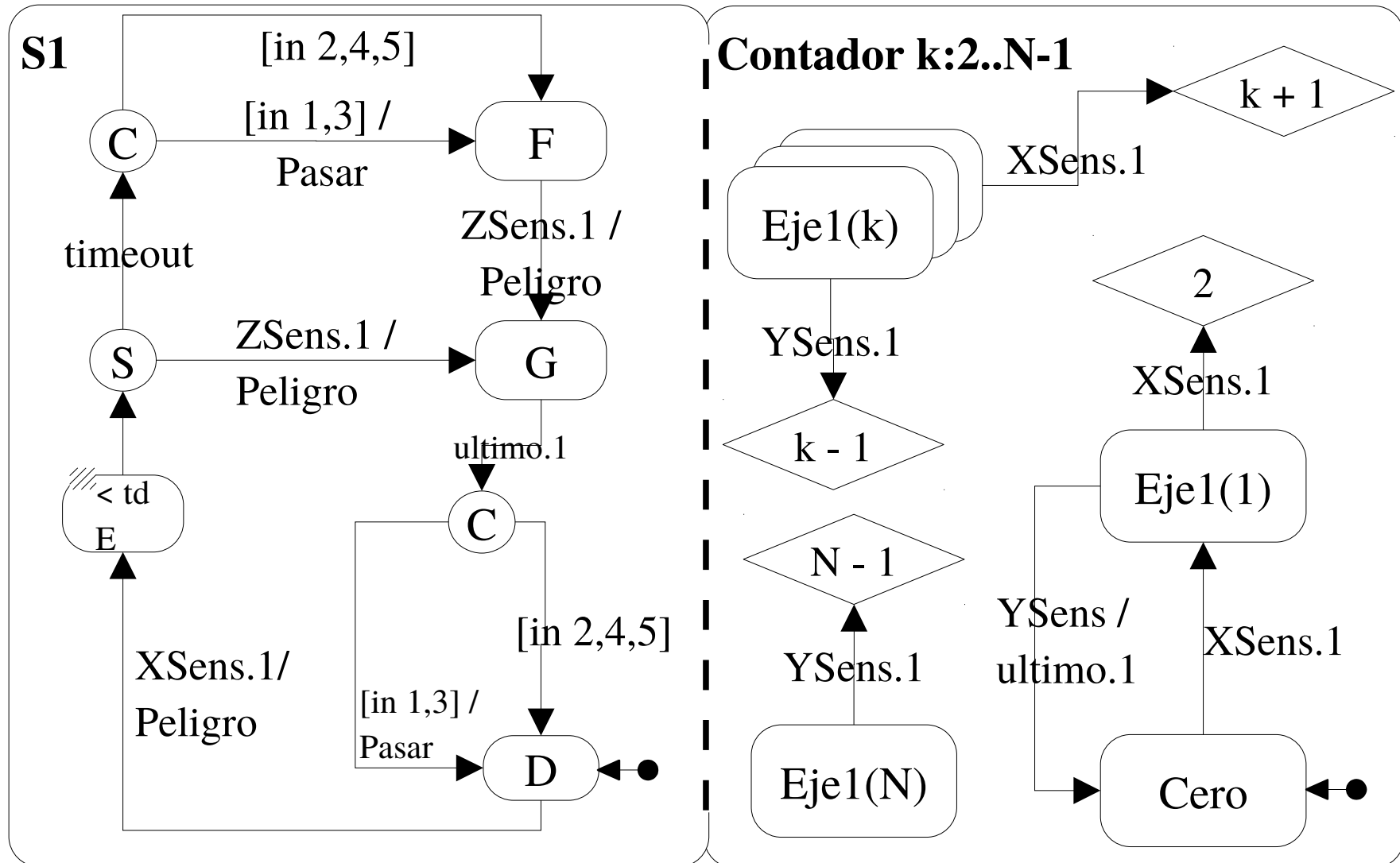
La especificación



Otra especificación posible

- ❑ Otra forma de transformar la referencia a futuro generada por el evento *TrenY* en algo implementable, consiste en contar la cantidad de señales *XSens* e *YSens* hasta que ambas coinciden.
- ❑ Para describir esto con Statecharts es necesario utilizar estados parametrizados.
- ❑ Además, habría que designar la cantidad máxima de ejes que puede tener un tren (N).
- ❑ Mostraremos la solución para una sola vía.

La especificación (2)



Contador expandido (N = 5)

