- Invariantes del monitor
  + NCARS = número de coches que están cruzando el puente
  + NPED = número de peatones que están cruzando el puente

Las fórmulas invariantes son

(1) $(NPED > 0 \Rightarrow NCARS = 0) \wedge (NCARS \leq 1) \wedge (NCARS = 1 \Rightarrow NPED = 0)$

(2) $\neg empty(Ped) \rightarrow NCARS \neq 0 \vee \neg empty(Coches)$

(3) $\neg empty(Coches) \rightarrow NPED \neq 0 \vee NCARS \neq 0$

Para demostrar que el puente es seguro basta ver que (1) es invariante. Para ello vamos a distinguir dos casos.

- Se ejecuta una operación del monitor desde el principio hasta el final
  + wants_enter_pedestrian esta operación aumenta el valor de NPED y el "if" asegura la operación acaba únicamente si NCARS = 0
  + leaves_pedestrian disminuye el valor de NPED haciendo que $R > 0$ y $R \neq 0$, pero esto no contradice al invariante
  + wants_enter_car aumenta el valor de NCARS y el "if" asegura que esta acaba si y solo si NCARS = 0 y NPED > 0
  + leaves_car disminuye el valor de NCAR, por lo $W = 0$ ó $W \leq 1$ se cumplen pero $W = 1$ no Esto tampoco contradice al invariante.

- Se ejecuta una operación que desbloquea un proceso que está en la cola de una variable condicional
  + signal(Ped) en wants_enter_pedestrian puede desbloquear a un peaton y esto solo pasa si NCARS = 0 gracias al "if" y el peatón continúa inmediatamente por lo que sigue NCARS = 0
  + signal(Ped) en leaves_car por hipótesis, NCARS ≤ 1 así que cuando el peatón cruce, NCARS = 0
  + signal(Coches) en leaves_pedestrian. signal solo se ejecuta si NPED = 1 una vez empezada la operación y cuando se completa NPED = 0, siendo ciertas la primera y tercera fórmulas de (1) Si NPED = 1, por hipótesis, W = 0 Por wants_enter_car del coche desbloqueado, W = 1, y (1) es cierto

  + signal(Coches) en leaves_car : $W \leq 1$ por hipótesis y por el coche ejecutando leaves_car, entonces W = 1 El coche desbloqueado acabará wants_enter_car manteniendo W = 1 y por tanto, $W \leq 1$ NPED no se ha modificado así que, (1) es cierto

Ahora, demostramos la ausencia de deadlocks Para ello, demostramos que (2) y (3) son invariantes.

(2) La primera parte solo puede ser cierta si ejecutamos wants_enter_pedestrian, pero el "if" asegura entonces que la consecuencia es también cierta. ¿Podría ocurrir que la consecuencia sea falsa y la condición cierta? Solo ejecutando

leaves_car en el último coche. Asumiendo que ¬empty(Ped) es cierto, signal(Ped) se ejecuta. Esto lleva a varios signal consecutivos y el último hace ¬empty(Ped) falso. Otra opción sería que leaves_pedestrian ejecute el último peatón, haciendo que signal(Coches) haga falso ¬empty(Coches), por tanto W ≠ 0 y la consecuencia sería siendo cierta

(3) Es cierta si al ejecutar wants_enter_car con el "if" ¿Puede ocurrir que la condición sea cierta y la consecuencia falsa? Al ejecutar leaves_pedestrian en el último peatón, NPED ≠ 0 es falso, pero como ¬empty(Coches) (por hipótesis), signal(Coches) hace que NCARS≠0 sea cierto. Ejecutando leaves_car en el último coche podría hacer NCARS ≠ 0 falso, pero un signalC haría o NPEP = 0 ó NCARS = 0

Por último, demostramos la ausencia de inanición

Si un peatón tiene inanición Ped tiene que bloquearlo. Vamos a demostrar que en algún momento, signal(Ped) va a ser ejecutado desbloqueando el primer peatón en la cola. Por la construcción de las colas tenemos que, por inducción en la posición de los peatones en la cola, el primer peatón se desbloqueará y pasará por el puente

Para demostrar

$$\neg empty(Ped) \rightarrow signalC(Ped)$$

asumimos lo contrario a:

$$\neg empty(Ped) \wedge \neg signalC(Ped)$$

Por (2), (NCARS ≠ 0) V ¬empty(Ped) Si NCARS≠0, leaves_cars será ejecutado signalC(Ped) será ejecutado al asumir ¬empty(Ped) es cierto ¡Contradicción!

Si ¬empty(Coches) es cierto, por (3) NPED≠0 V NCARS≠0. Hemos demostrado que NCARS ≠ 0 lleva a una contradicción así que solo consideramos el caso NPED≠0. Cuando no haya más peatones, todos acabarán ejecutando leaves_pedestrian y el último de ellos ejecuta signalC(Coches). Como hemos asumido que ¬empty(Coches) es cierto, un coche se desbloquea haciendo NCARS≠0 cierto y así, estamos en el mismo caso que antes

Para ver que los coches no tienen inanición la prueba es análoga