Veamos que se cumple todo:

• Puente seguro:

Se tiene como consecuencia del invariante.
Los casos no seguros son:

- $c0>0 \wedge c1>0$
- $(p>0 \wedge c0>0) \vee (p>0 \wedge c1>0)$

pero ambos contradicen el invariante.

• Deadlocks:

Habría un deadlock si todos los peatones están esperando en can_enter_ped, todos los coches c0 en can_enter_c0 y todos los c1 en can_enter_c1. Sin embargo, si no hay nadie cruzando, se tendría $p=c1=c0=0$. Pero self.turn.value $\in \{1,2,3\}$ y alguna de las condiciones se cumplirá y algún proceso será desbloqueado, y podrá cruzar. No se podría dar un deadlock.

• Inanición:

Veamos los 3 posibles casos de inanición:

- c0 sufre inanición → se queda esperando a can_enter_c0. Supongamos primero que se tiene $c1>0$. Por el avance del programa, todos los c1 cruzando acabarán ejecutando leaves_car, cuando el primero lo haga, se cambiará el turno a 2 y cuando el último lo haga se ejecutará notify(can_enter_ped y can_enter_c0)

En caso de que no haya peatones esperando, se desbloquea un coche c0 (y eventualmente todos), en caso contrario se desbloquea un c1.
Si por el contrario tenemos que p>0, repitiendo el razonamiento llegamos a que se cambia el turno a 0 y se hacen los notify (can_enter_c0 y can_enter_c1) con lo que se desbloquea un c0 (y eventualmente todos). Así, c0 no puede sufrir inanición.

- <u>c1 sufre inanición</u> → análogo a la demostración anterior.

- <u>ped sufre inanición</u> → ocurre cuando se queda esperando a can_enter_ped. Supongamos primero que es porque c0>0. Por el avance del programa, los coches que hay acabarán ejecutando leaves_car, cuando el primero lo haga, el turno cambiará a 1 y cuando el último lo haga, se hará notify(can_enter_c1) y notify(can_enter_ped). Si no hay c1 esperando, se desbloqueará a un peatón (y eventualmente todos), en caso contrario se desbloquea un c1. En ese caso, repitiendo el mismo proceso llegamos a la ejecución de notify(can_enter_ped) y notify(can_enter_c0) y con el turno = 2, lo cual desbloque un peatón (y eventualmente todos).
Si suponemos c1>0 estamos en el último caso ahora visto.
Luego, ped no puede sufrir inanición.