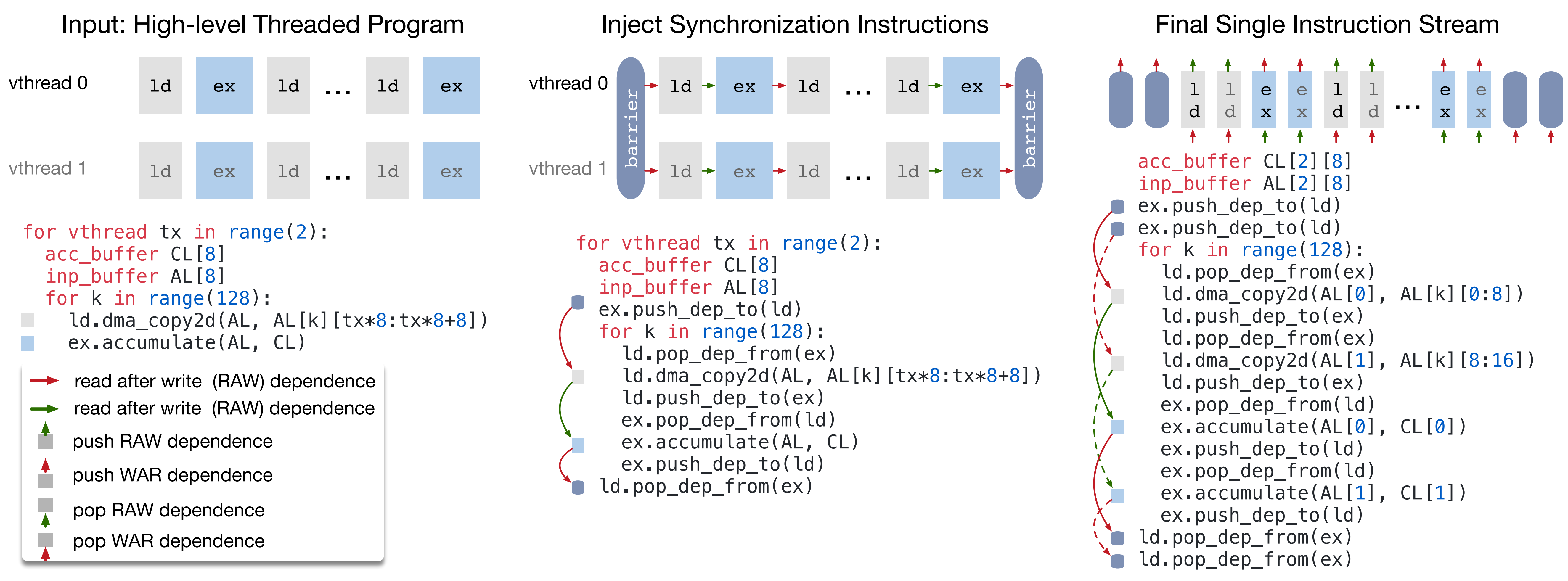

Input: High-level Threaded Program

```
for vthread tx in range(2):
  acc_buffer CL[8]
  inp_buffer AL[8]
  for k in range(128):
    ld.dma_copy2d(AL, AL[k][tx*8:tx*8+8])
    ex.accumulate(AL, CL)
```

- read after write (RAW) dependence
- read after write (RAW) dependence
- push RAW dependence
- push WAR dependence
- pop RAW dependence
- pop WAR dependence

Inject Synchronization Instructions

```
for vthread tx in range(2):
  acc_buffer CL[8]
  inp_buffer AL[8]
  ex.push_dep_to(ld)
  for k in range(128):
    ld.pop_dep_from(ex)
    ld.dma_copy2d(AL, AL[k][tx*8:tx*8+8])
    ld.push_dep_to(ex)
    ex.pop_dep_from(ld)
    ex.accumulate(AL, CL)
    ex.push_dep_to(ld)
  ld.pop_dep_from(ex)
```

Final Single Instruction Stream

```
acc_buffer CL[2][8]
inp_buffer AL[2][8]
ex.push_dep_to(ld)
ex.push_dep_to(ld)
for k in range(128):
  ld.pop_dep_from(ex)
  ld.dma_copy2d(AL[0], AL[k][0:8])
  ld.push_dep_to(ex)
  ld.pop_dep_from(ex)
  ld.dma_copy2d(AL[1], AL[k][8:16])
  ld.push_dep_to(ex)
  ex.pop_dep_from(ld)
  ex.accumulate(AL[0], CL[0])
  ex.push_dep_to(ld)
  ex.pop_dep_from(ld)
  ex.accumulate(AL[1], CL[1])
  ex.push_dep_to(ld)
ld.pop_dep_from(ex)
ld.pop_dep_from(ex)
```