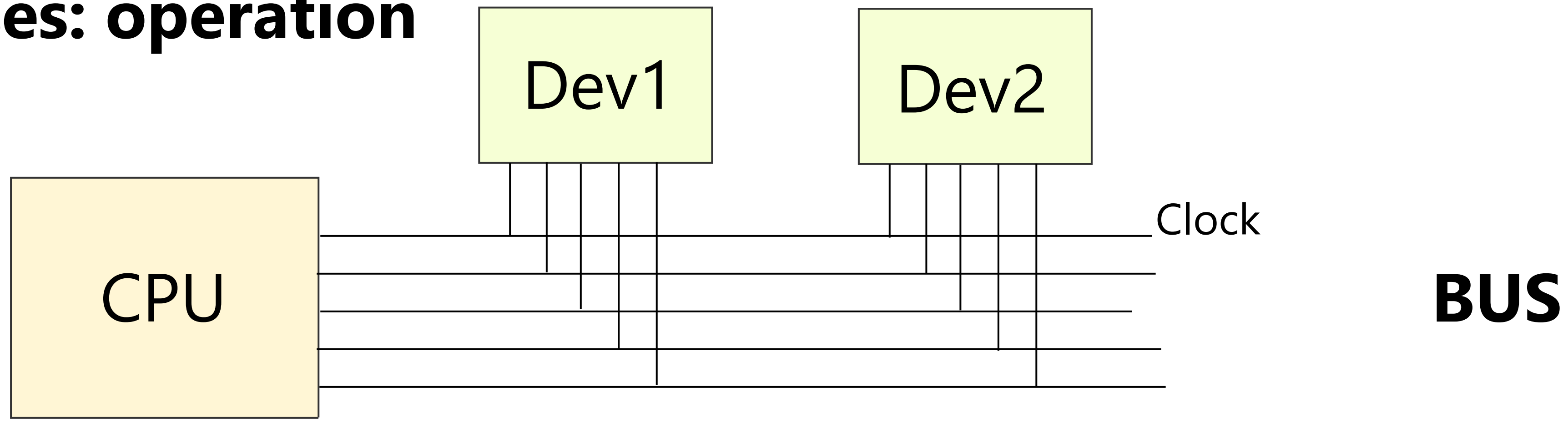
Buses and peripherals

Bus: How it is attached: specifies how the data should be exchanged over a medium.

Buses: operation



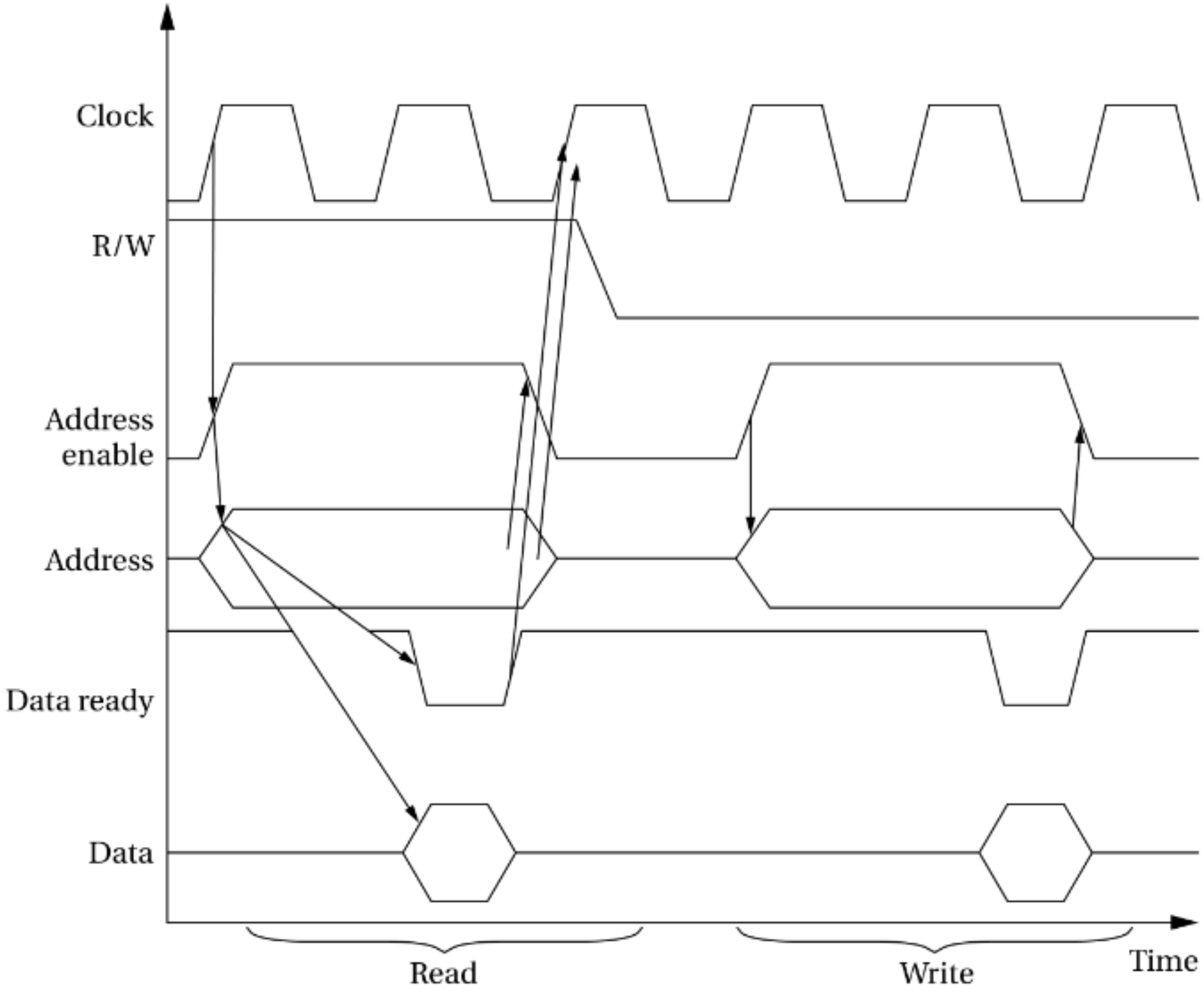
Dev1

Dev2

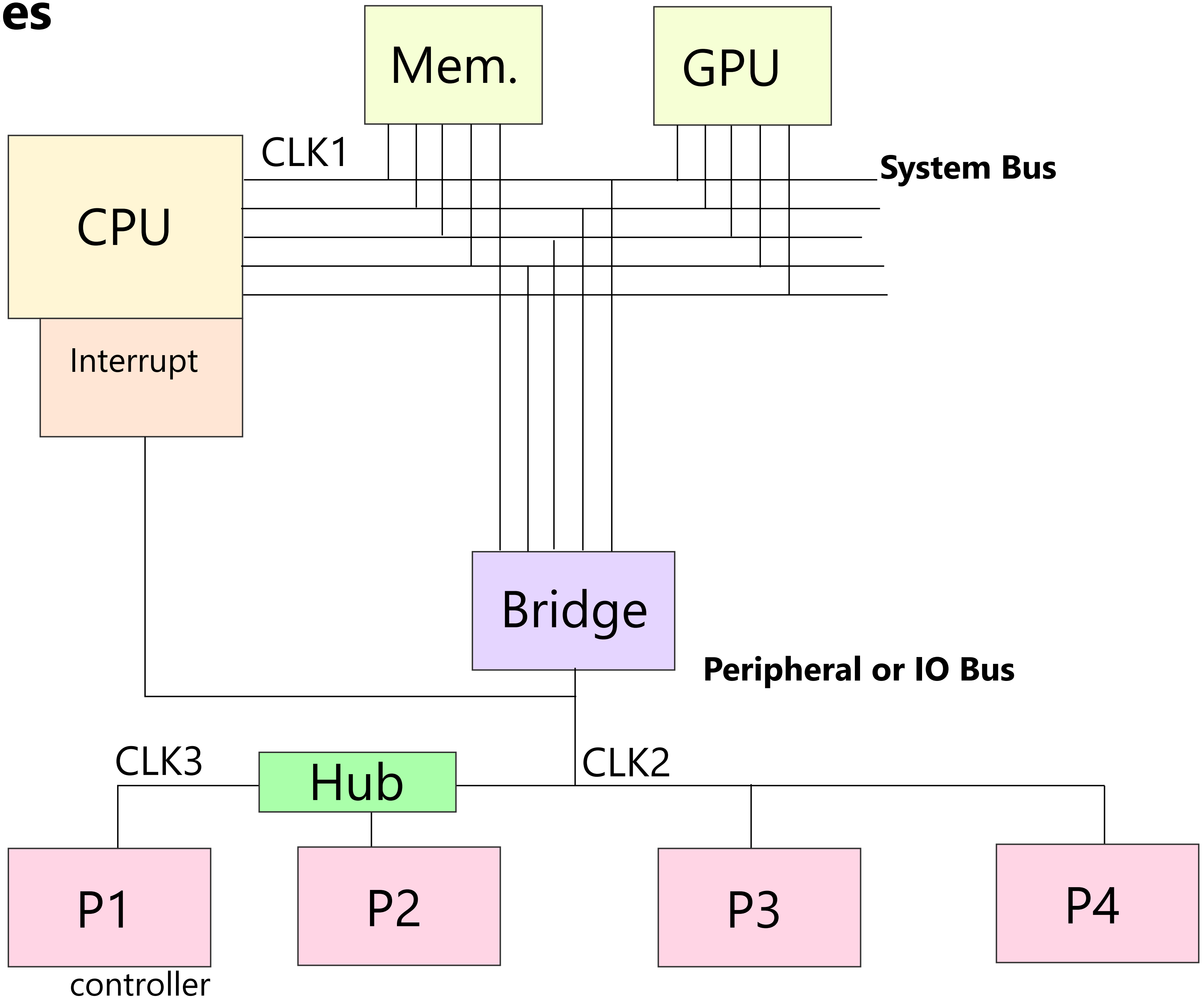
CPU

Clock

BUS



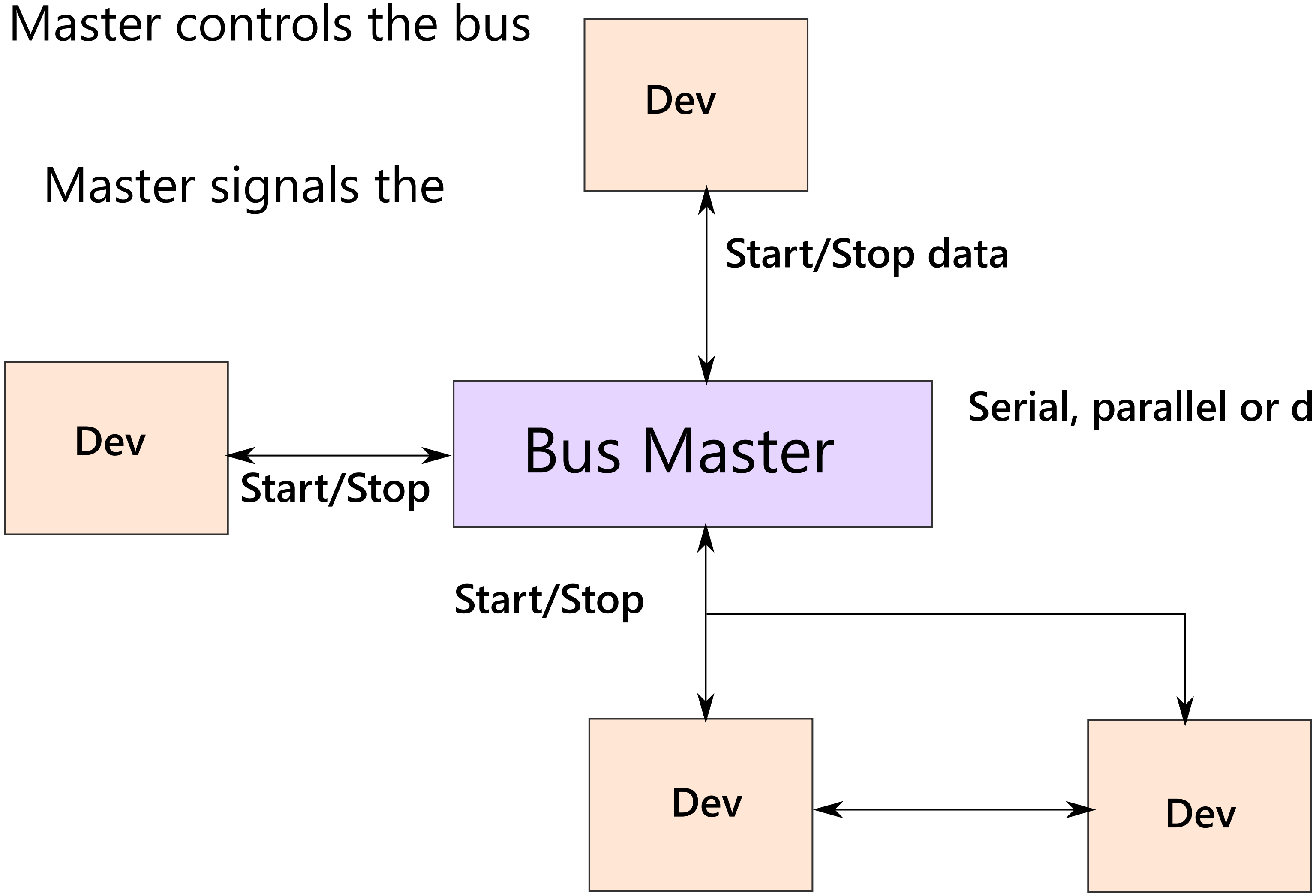
Buses


Mem.
GPU
CLK1
System Bus
CPU
Interrupt
Bridge
Peripheral or IO Bus
CLK3
Hub
CLK2
P1
P2
P3
P4
controller

Buses: controlling the shared resource

Single Master controls the bus

Master signals the



Serial, parallel or daisy chained

Cheap, simple to implement, low throughput. e.g. USB, SPI

transfer

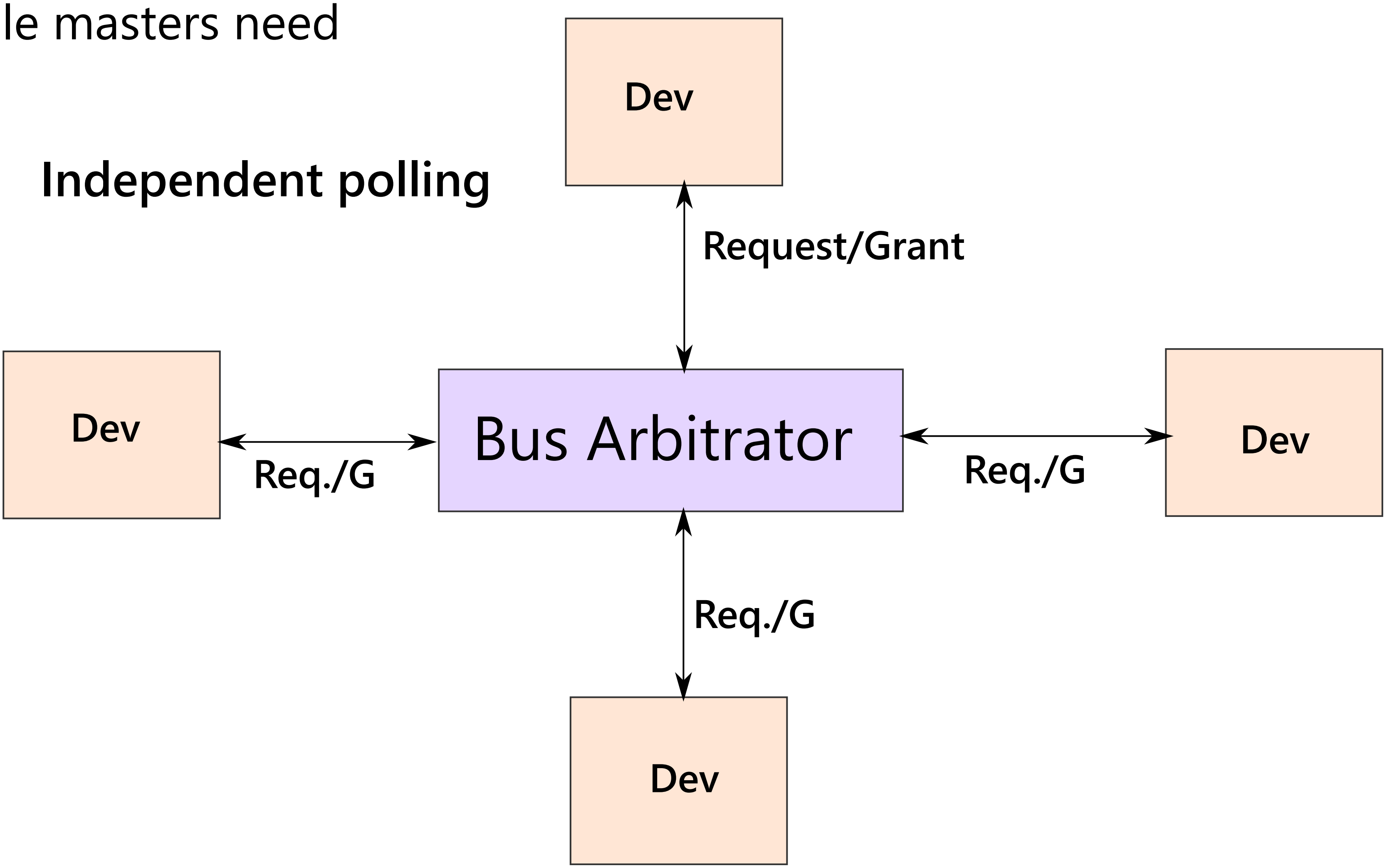
slaves to start or stop

access to the slaves

Buses: controlling the shared resource

Multiple masters need

Independent polling



arbitration for accessing bus.

e.g. ARM AMBA, PCI. High throughput, control logic with devices.

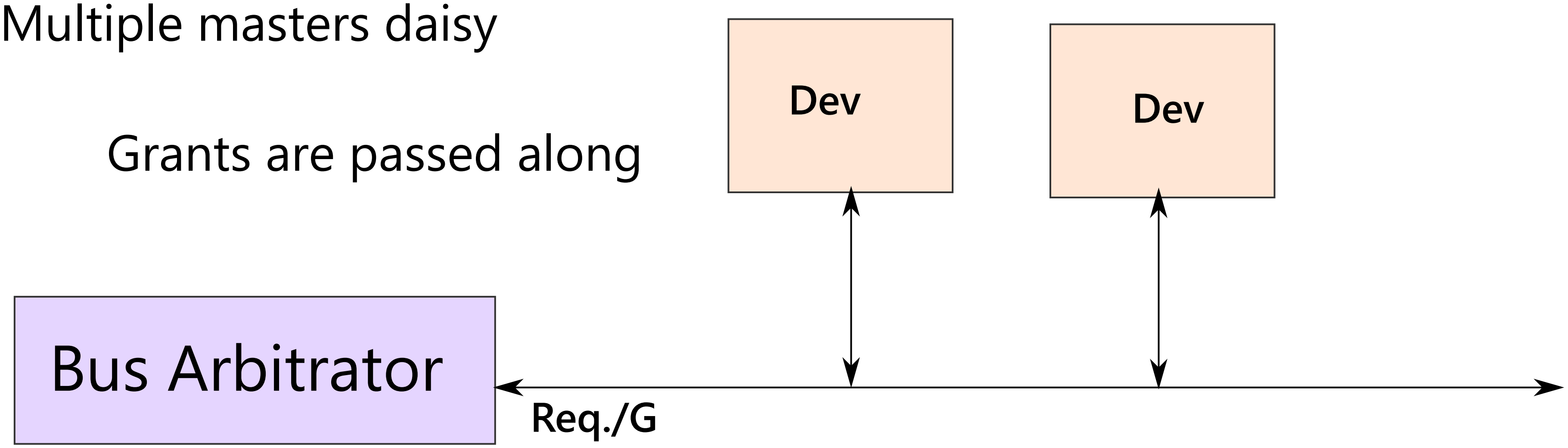
(all devices share same clock)

first come first serve or

Buses: controlling the shared resource

Multiple masters daisy

Grants are passed along



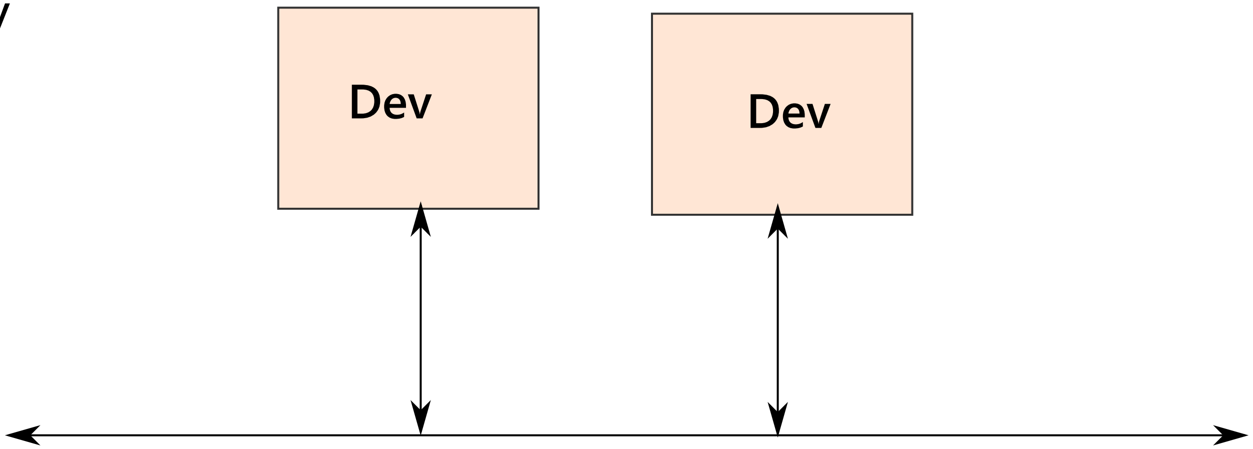
chained

the daisy chained link

Buses: controlling the shared resource

Multiple masters daisy

Distributed arbitration



Sense signal and assert request to own bus

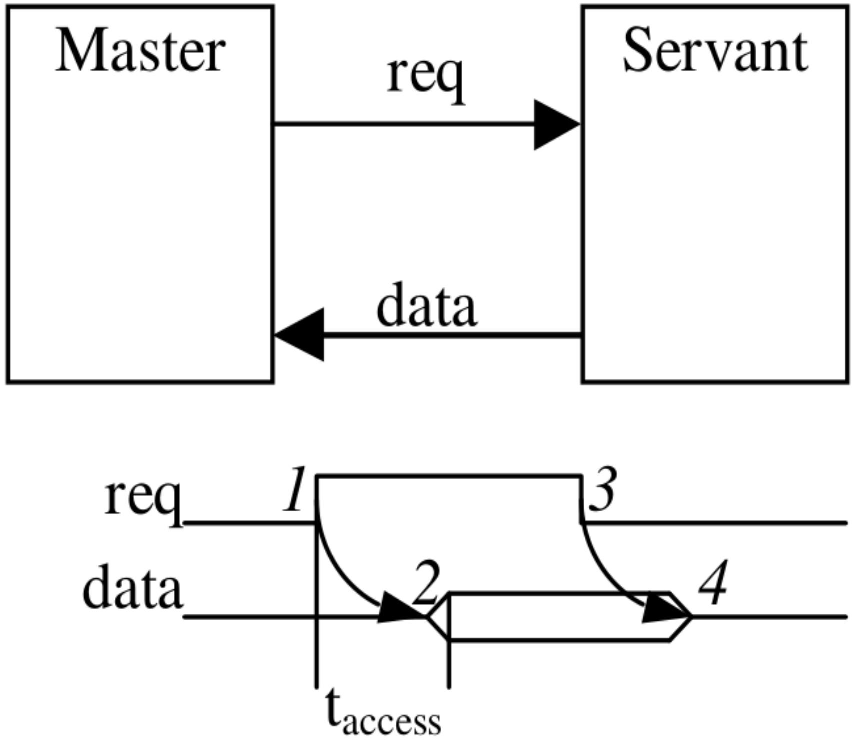
When a device becomes master, all others become slaves

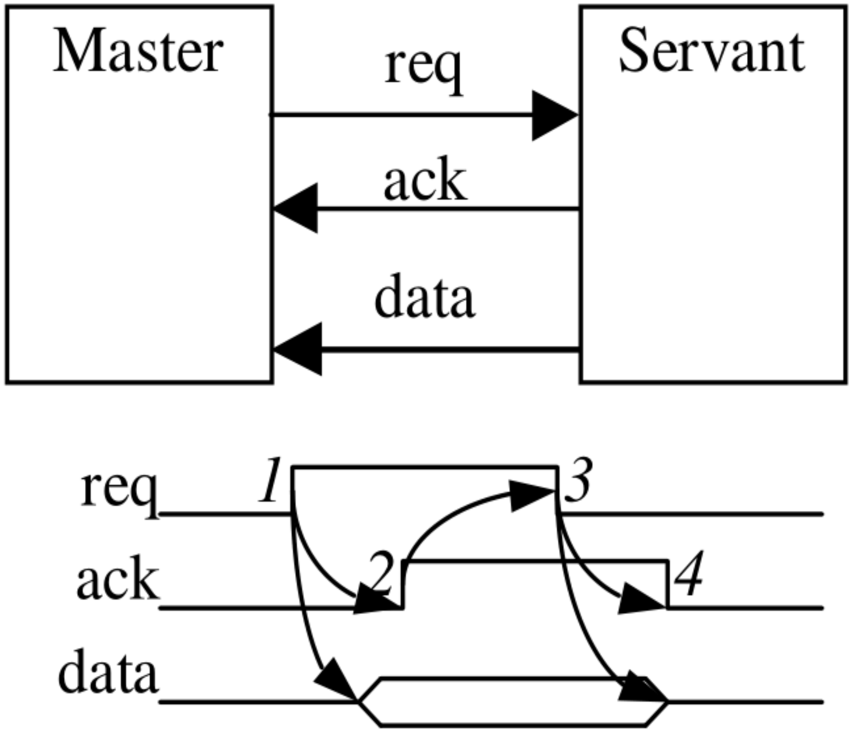
e.g. CAN and I2C. easy to implement, 1-2 wires sufficient, for slow

chained without arbiter

Buses: strobe and handshake

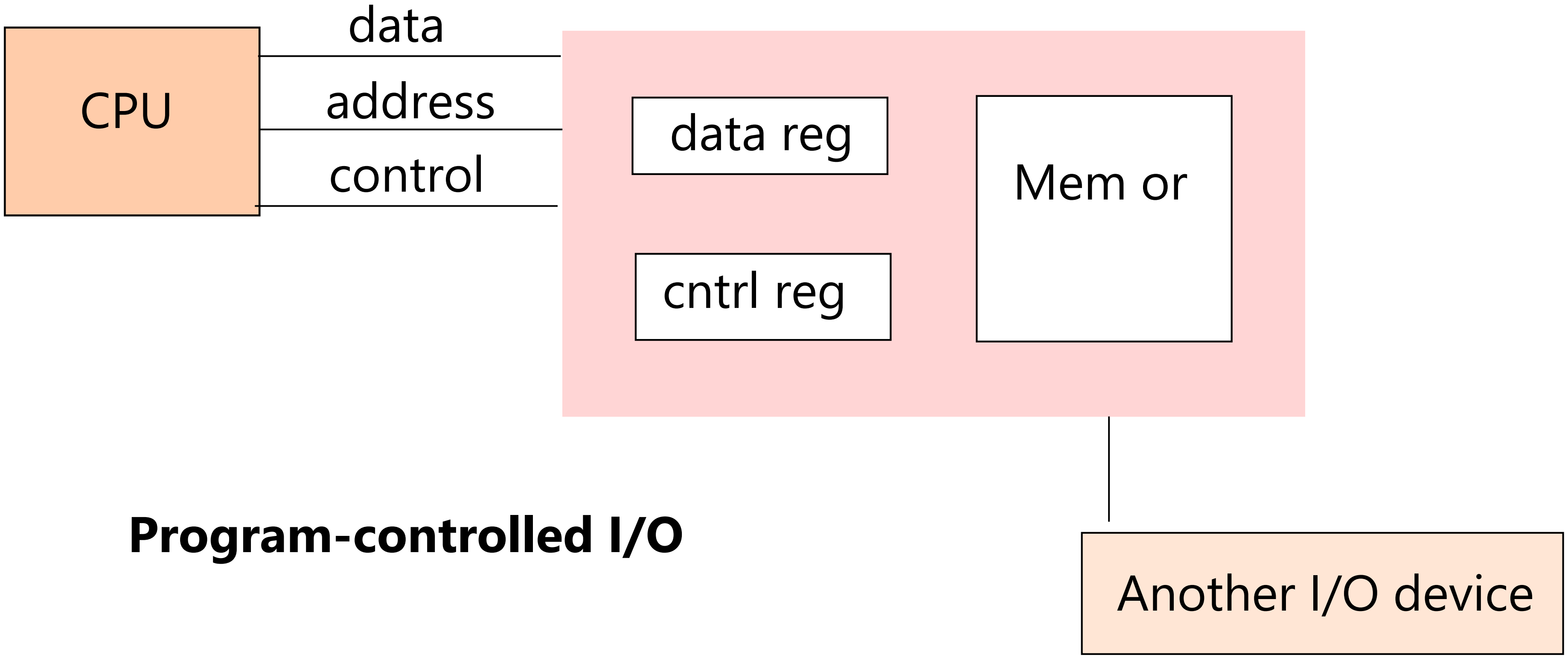
Grant type: atomic, split





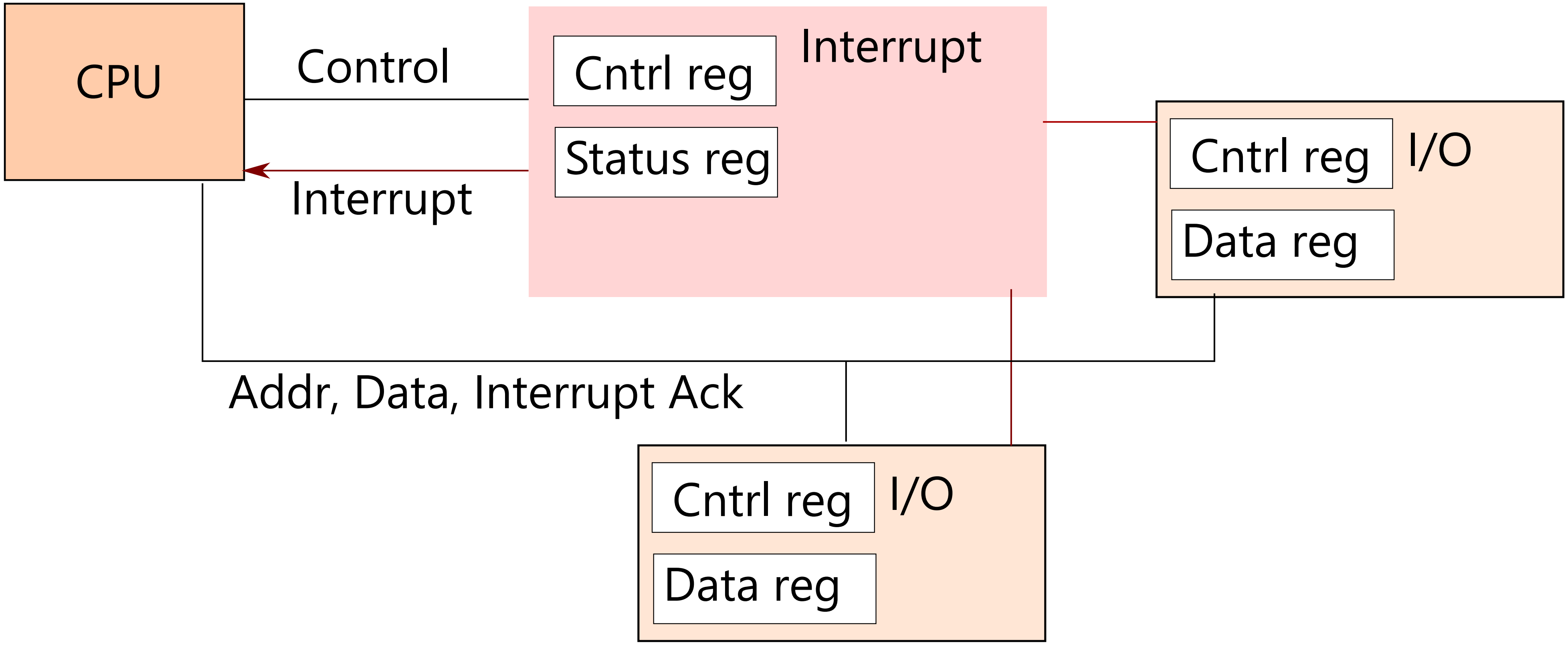
Modes of transfer

Program-controlled I/O



Program-controlled I/O

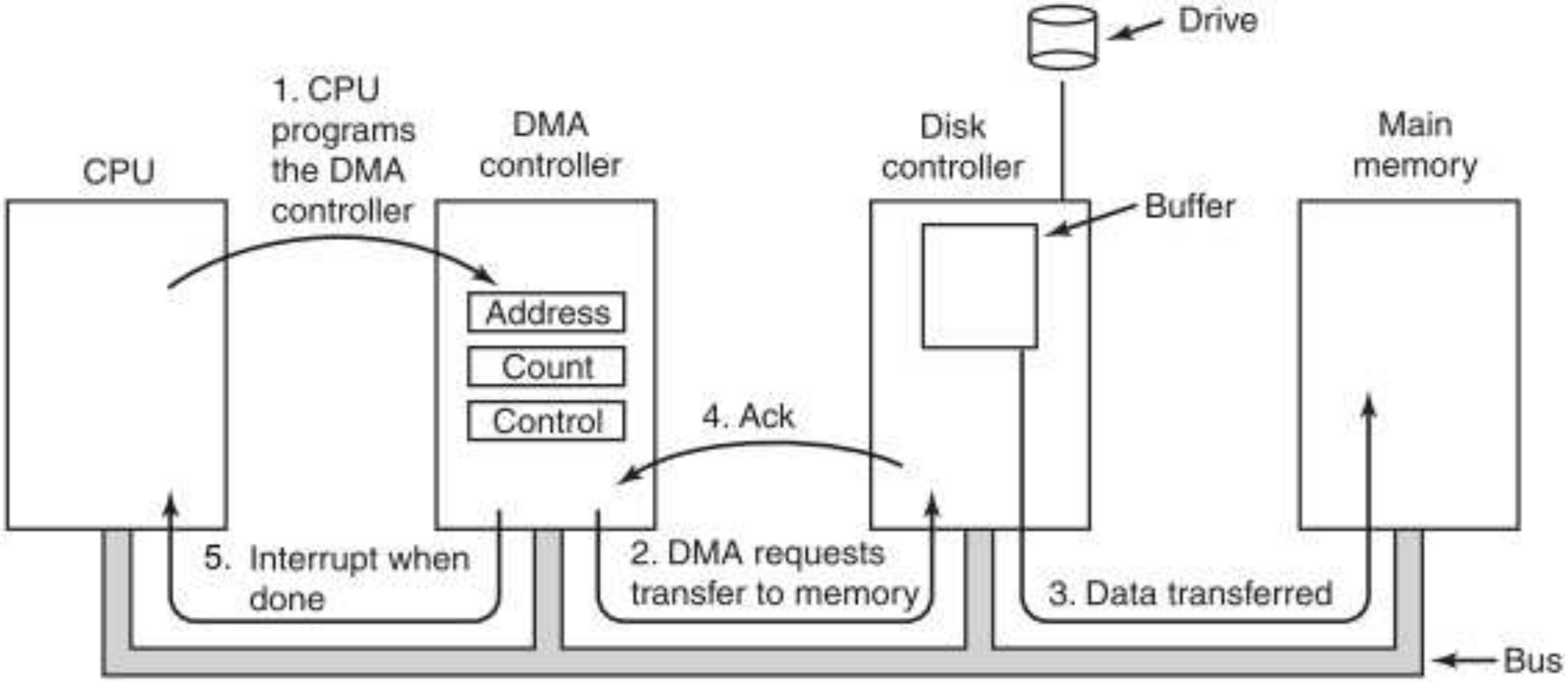
Interrupt Initiated I/O



Scalable

DMA Controller

CPU as a master of the system bus doesn't need to oversee data transfer

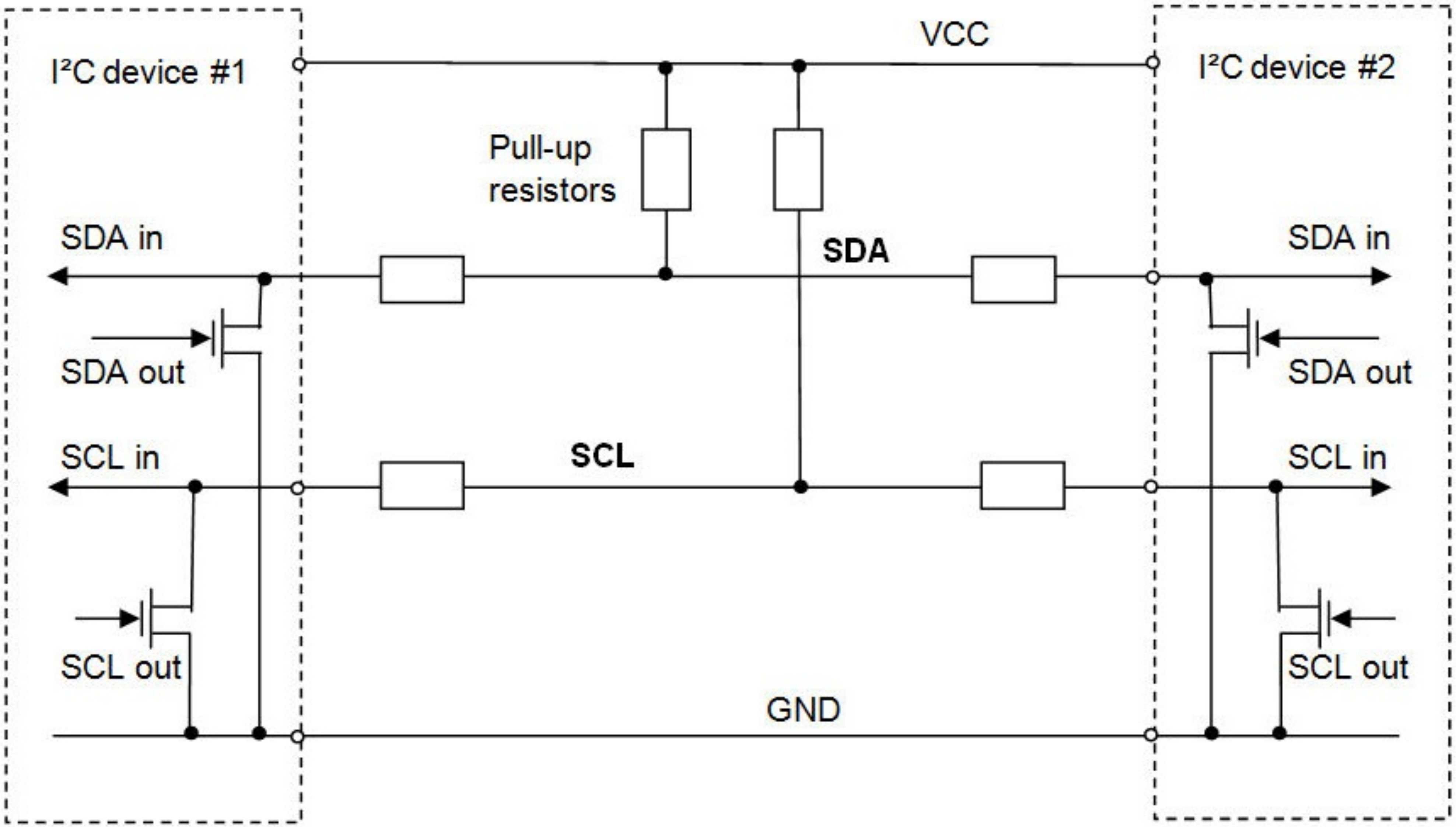


Here peripheral (.e.g GPU) and memory sit on same bus.

Single, block, demand transfers can be supported

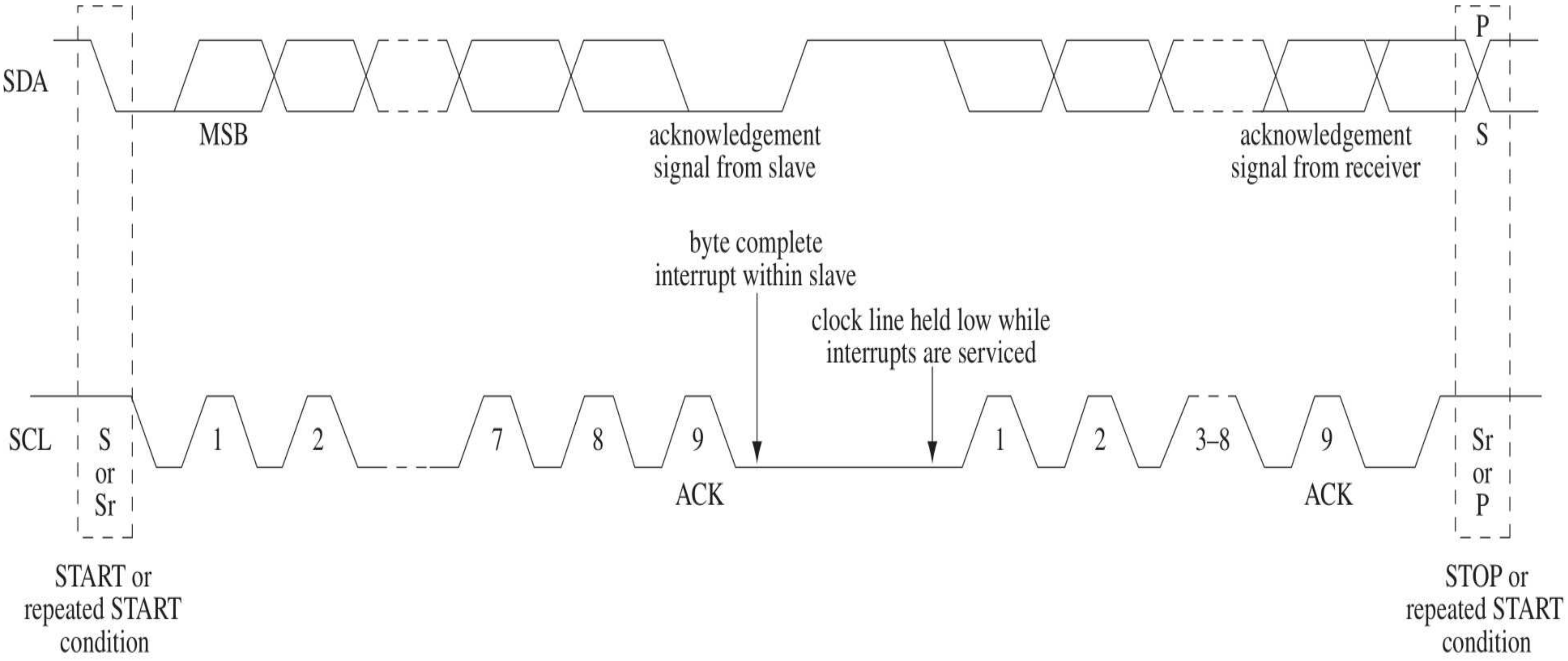
I2C Bus

2-wire, 7 or 10 bit address, various transfer speeds, non-extensible



I2C Bus

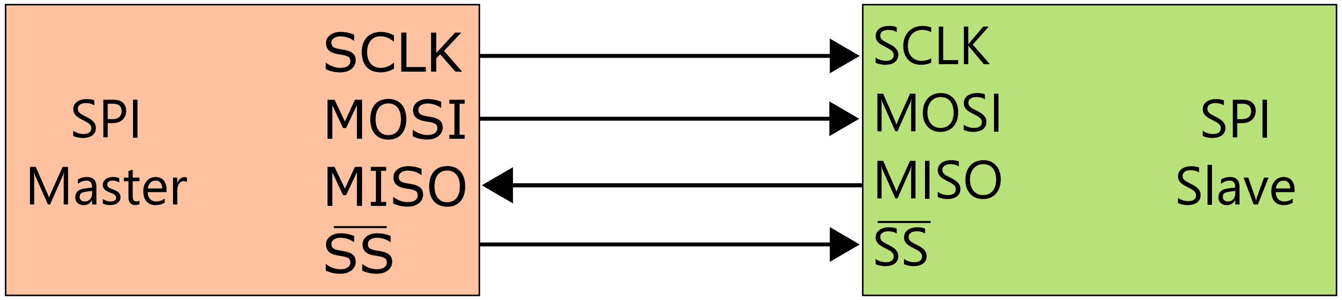
Transfers: single/combined


P
SDA
MSB
acknowledgement signal from slave
acknowledgement signal from receiver
S
byte complete interrupt within slave
clock line held low while interrupts are serviced
SCL
S or Sr
1
2
7
8
9
ACK
1
2
3–8
9
ACK
Sr or P
START or repeated START condition
STOP or repeated START condition

Clock stretching, arbitration

SPI bus

Full duplex, 4-wire, synchronous, master-slave bus (each slave needs its enable)



SPI Master

SCLK

MOSI

MISO

$\overline{SS}$

SPI Slave

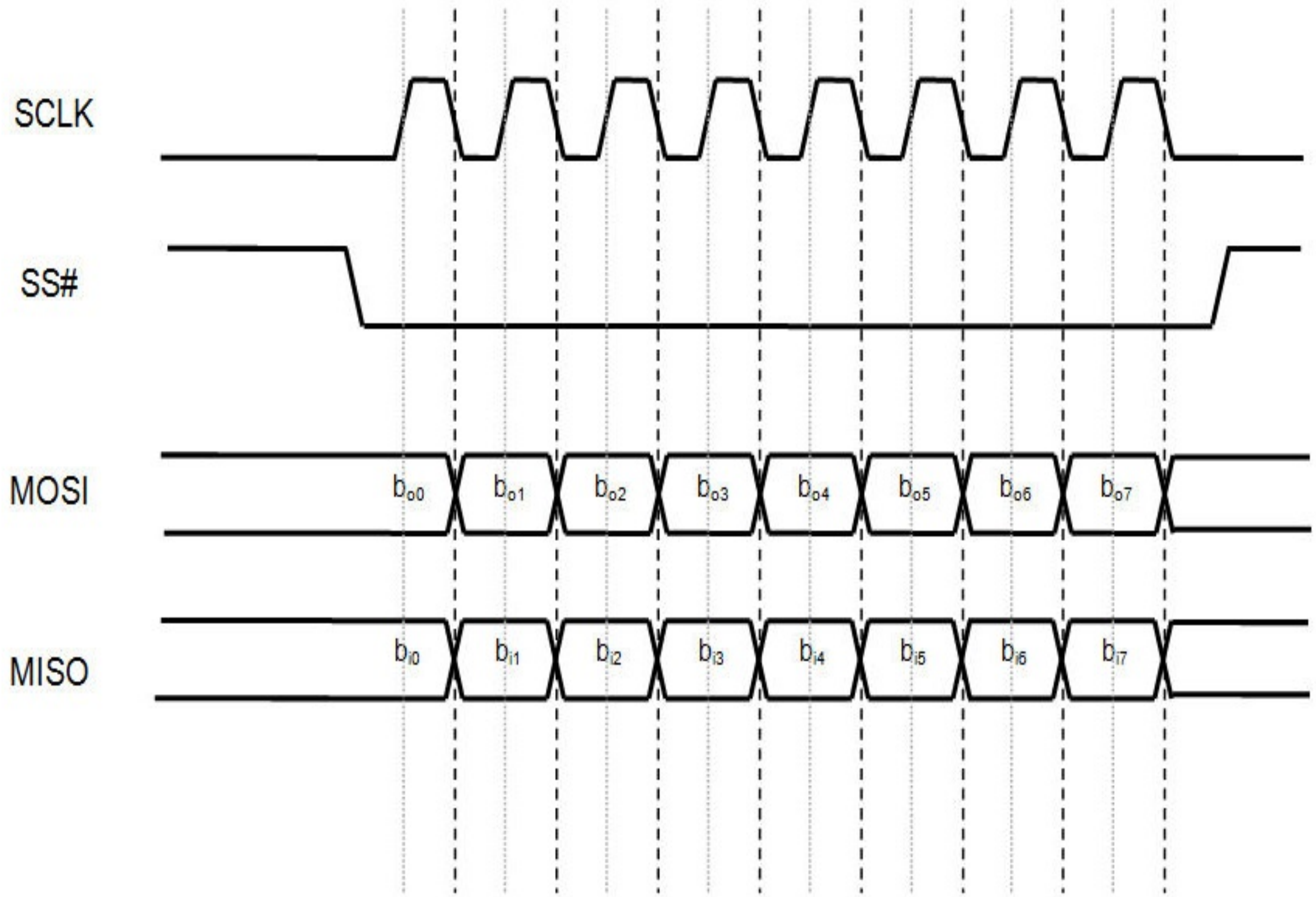
SCLK

MOSI

MISO

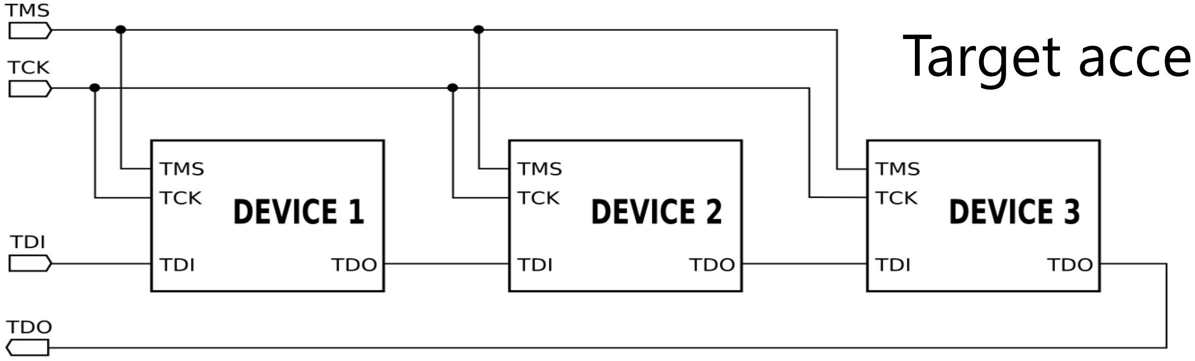
$\overline{SS}$

Simple transmission,

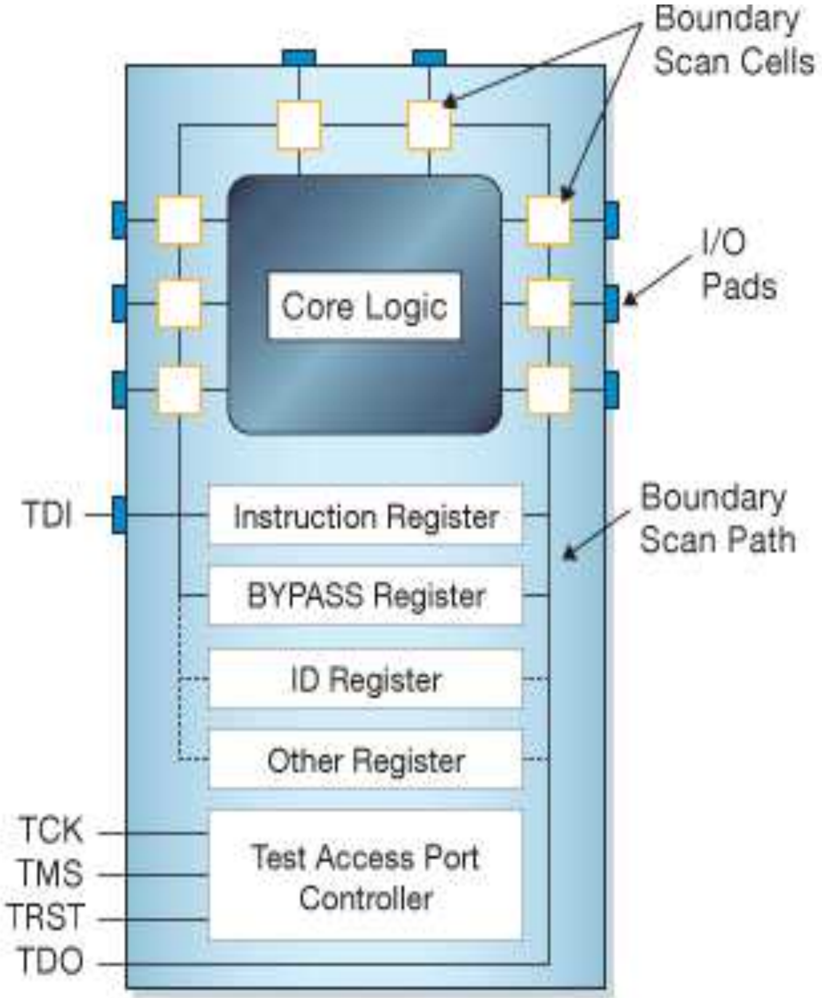


JTAG bus

Testing, diagnosis and fault isolation: de facto standard for embedded

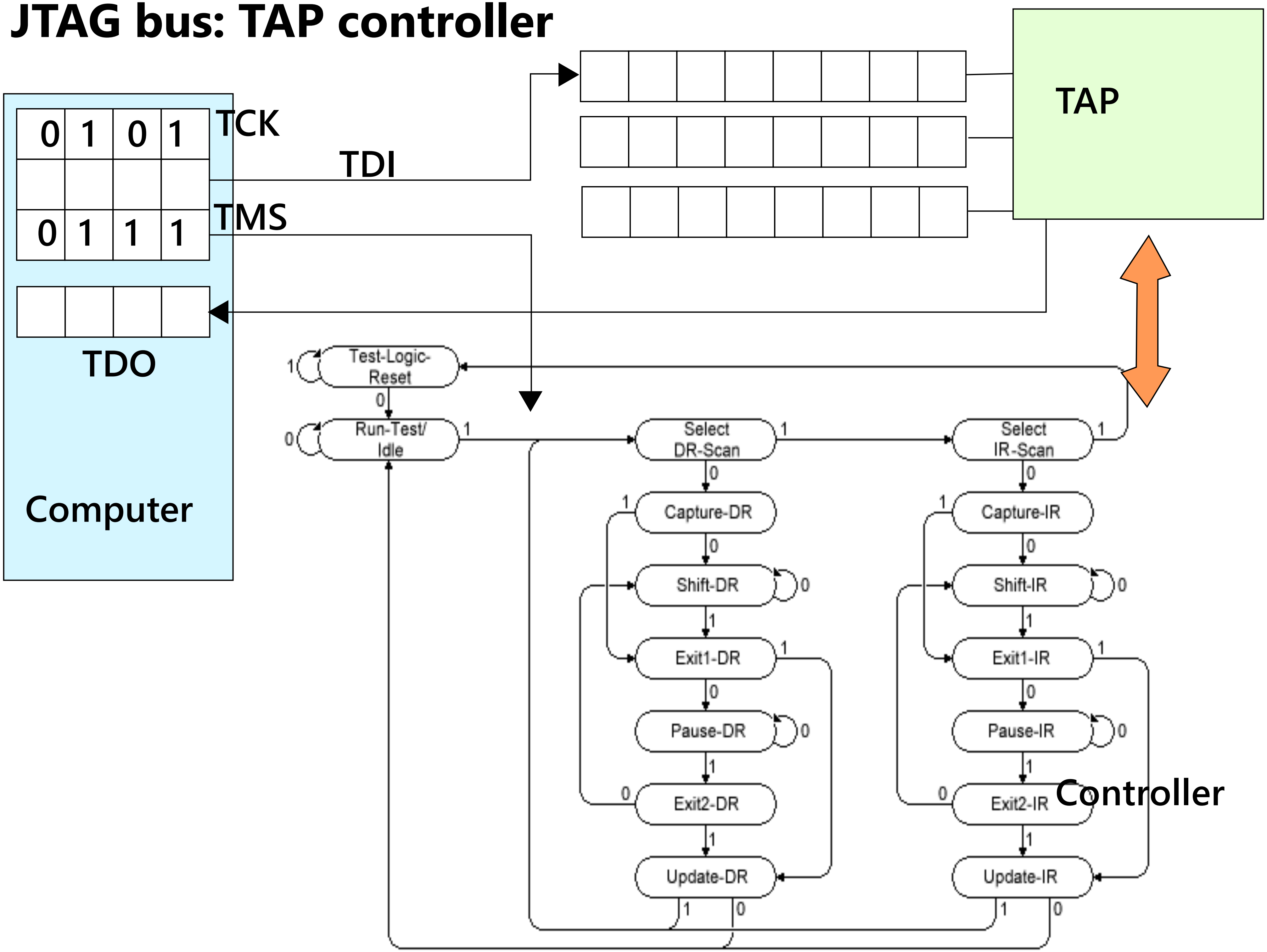


Target access ports (TAPs)



TMS and TCK are used together to change

JTAG bus: TAP controller

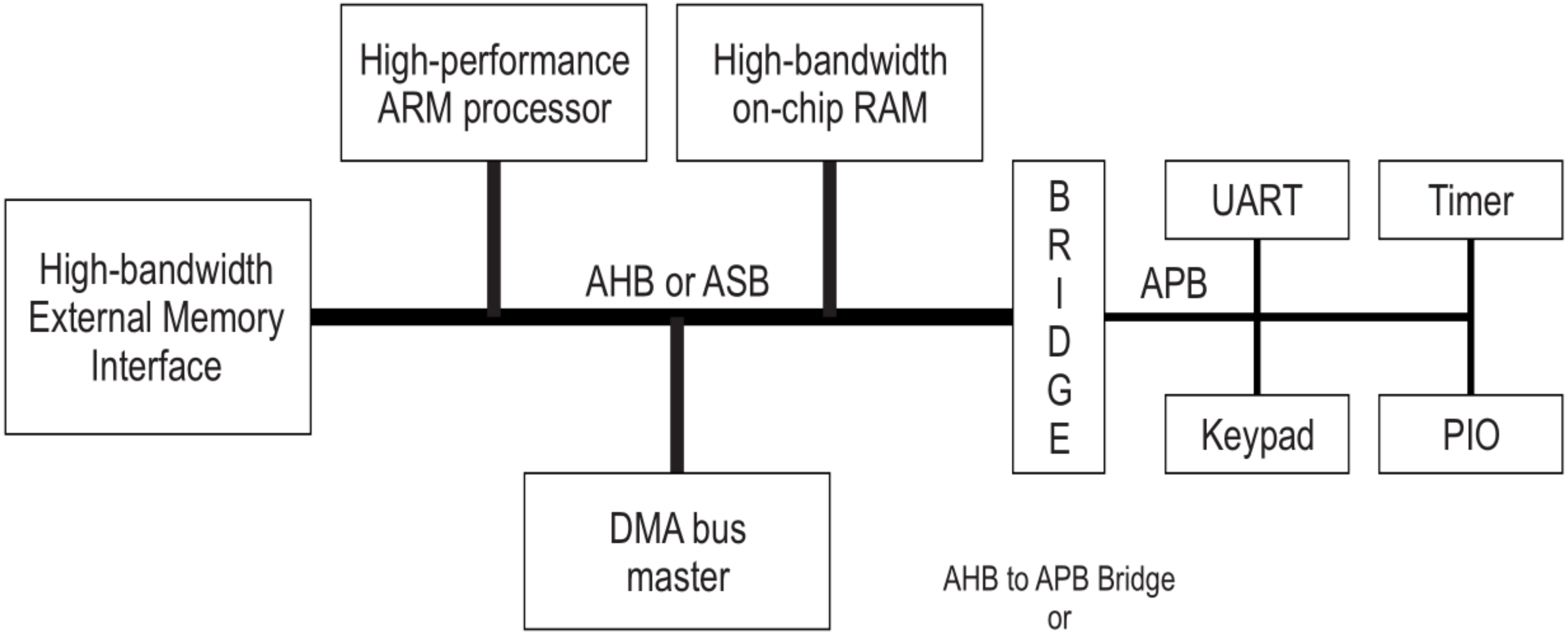


JTAG bus: instructions

```
attribute INSTRUCTION_LENGTH of EP1C3T100 : entity is 10;
```

ARM AMBA

Advanced Microcontroller Bus Architecture: defacto standard in u-controllers


High-performance ARM processor
High-bandwidth on-chip RAM
High-bandwidth External Memory Interface
AHB or ASB
DMA bus master
BRIDGE
APB
UART
Timer
Keypad
PIO

AHB to APB Bridge
or
ASB to APB Bridge

AMBA AHB

* High performance
* Pipelined operation
* Multiple bus masters
* Burst transfers
* Split transactions

AMBA ASB

* High performance
* Pipelined operation
* Multiple bus masters

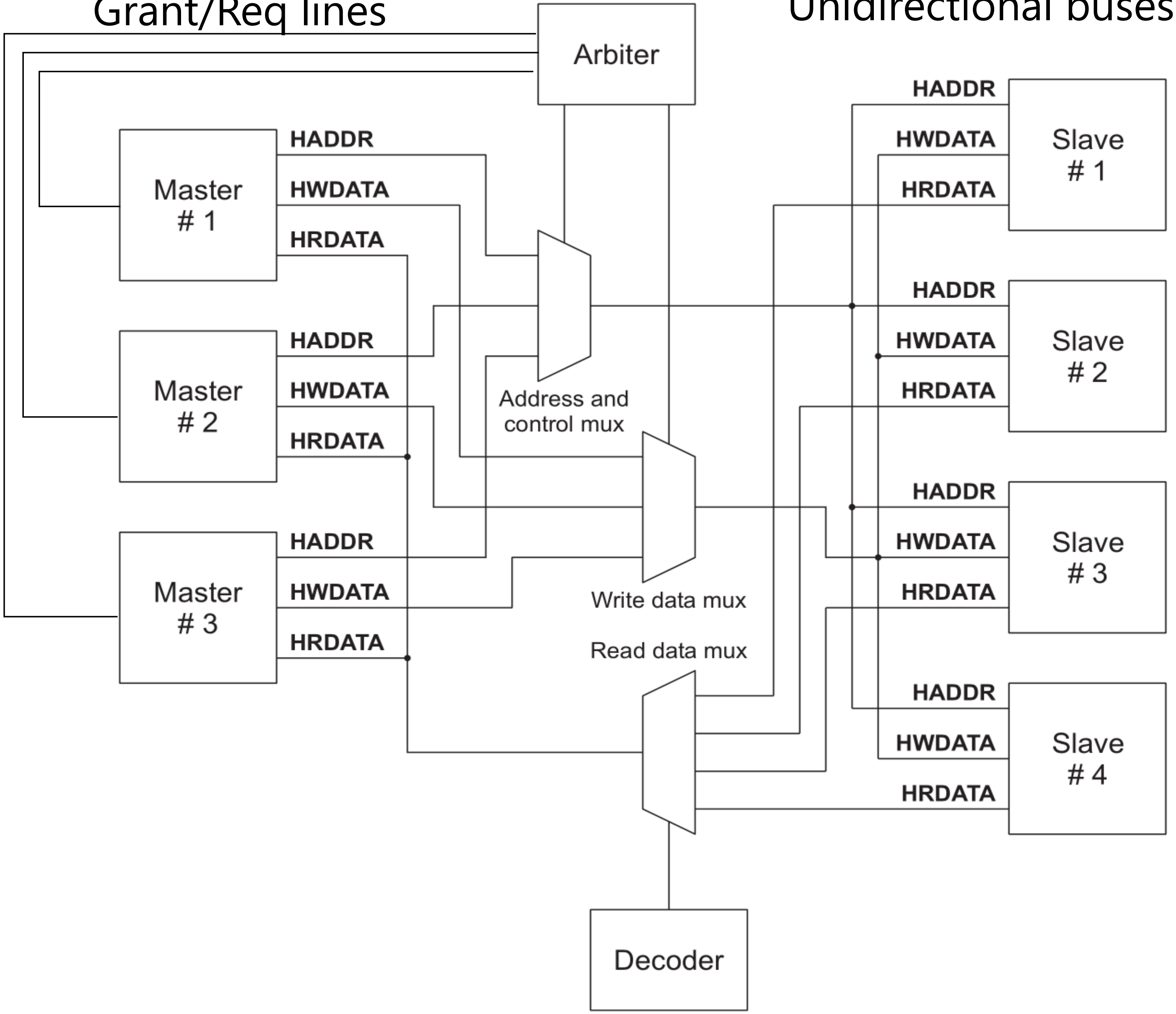
AMBA APB

* Low power
* Latched address and control
* Simple interface
* Suitable for many peripherals

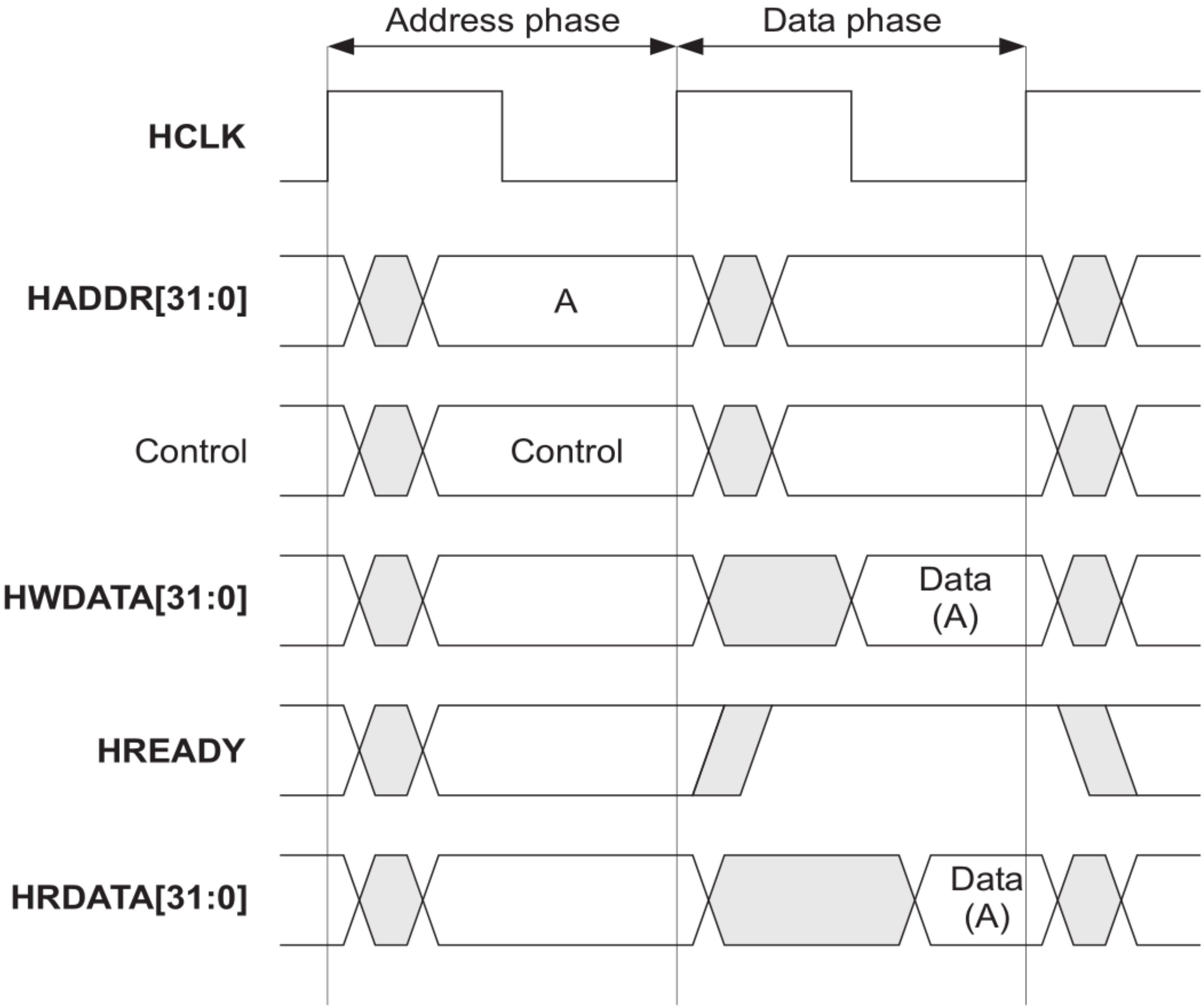
ARM AHB: connections

Grant/Req lines

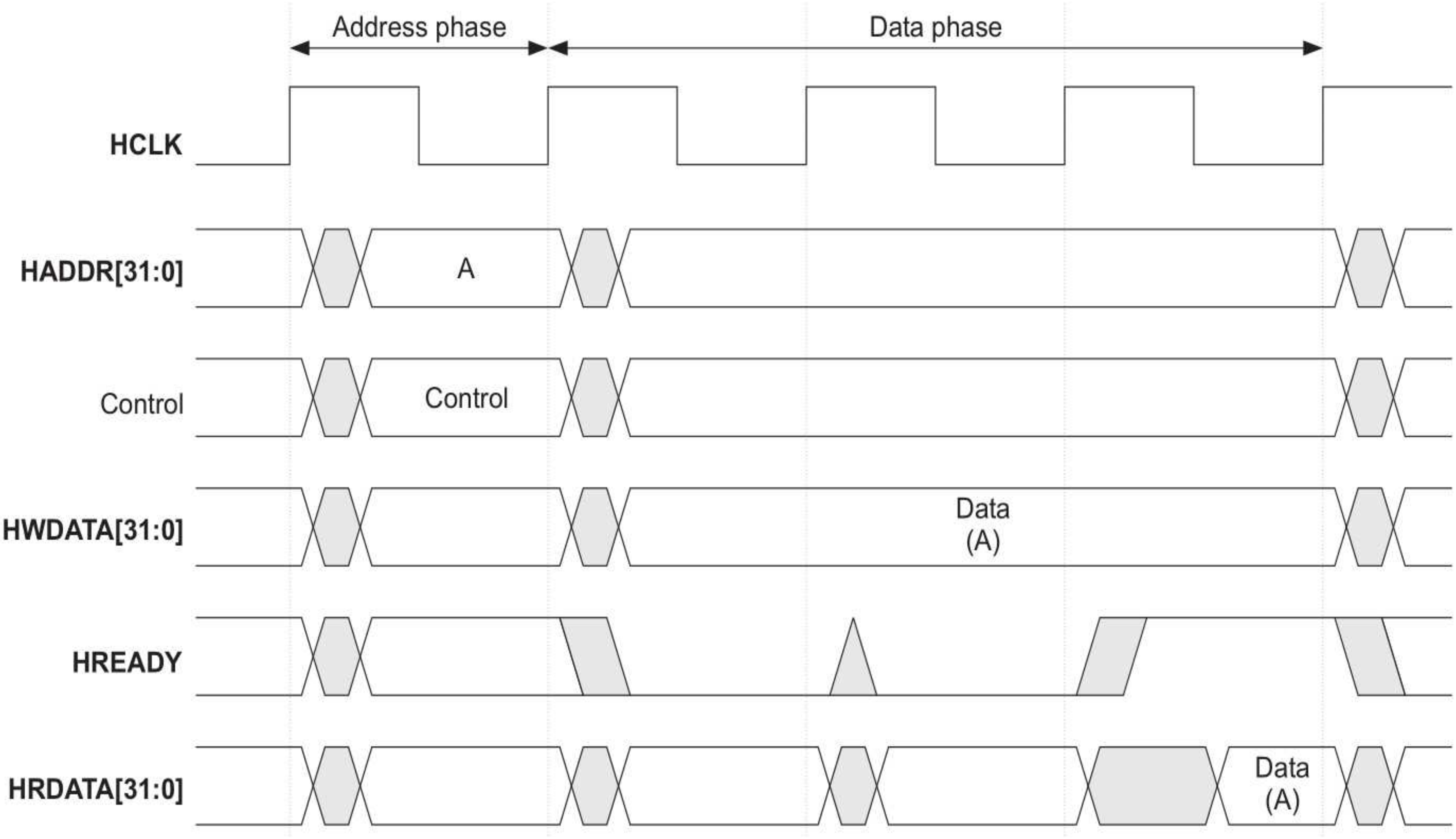
Unidirectional buses



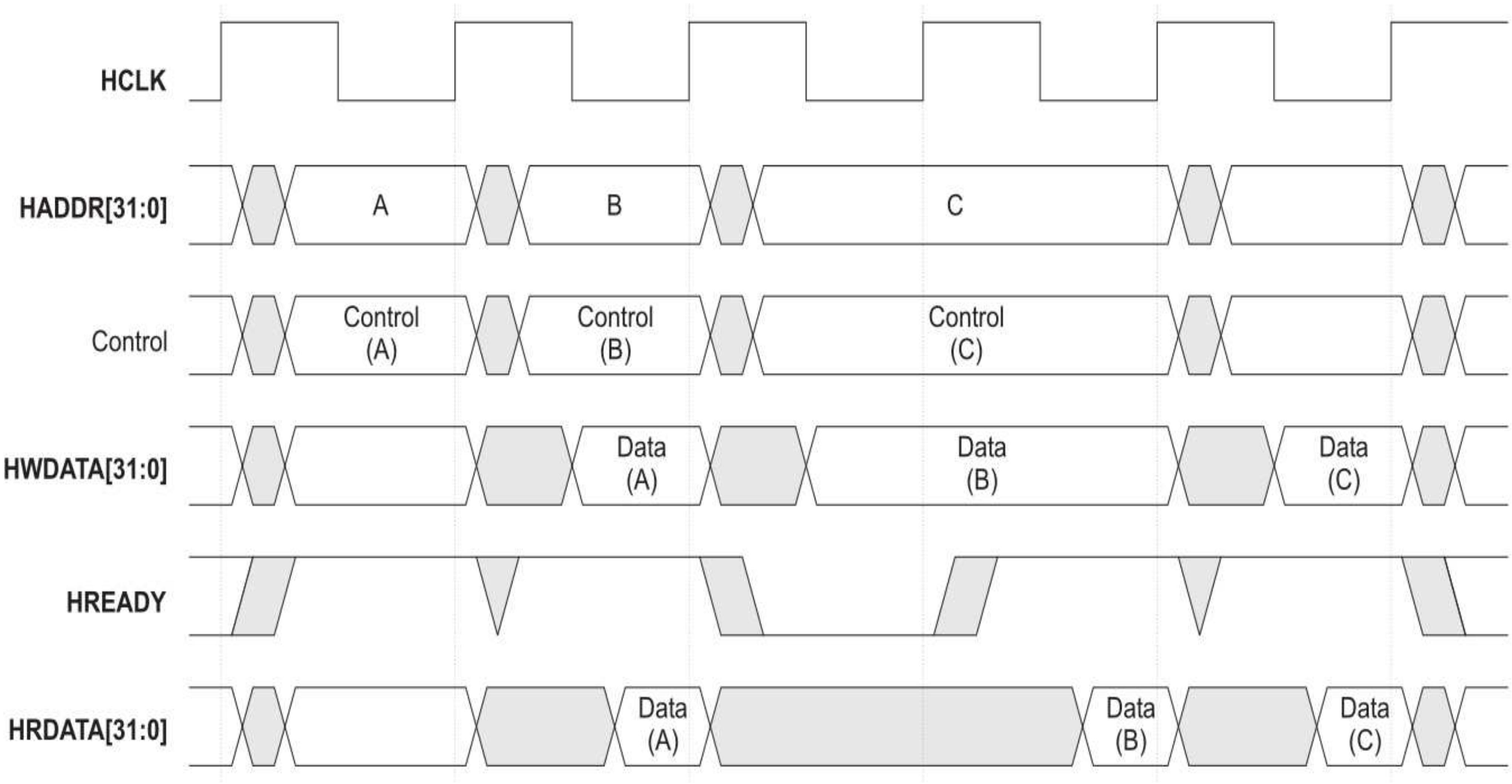
ARM AHB: simple transactions



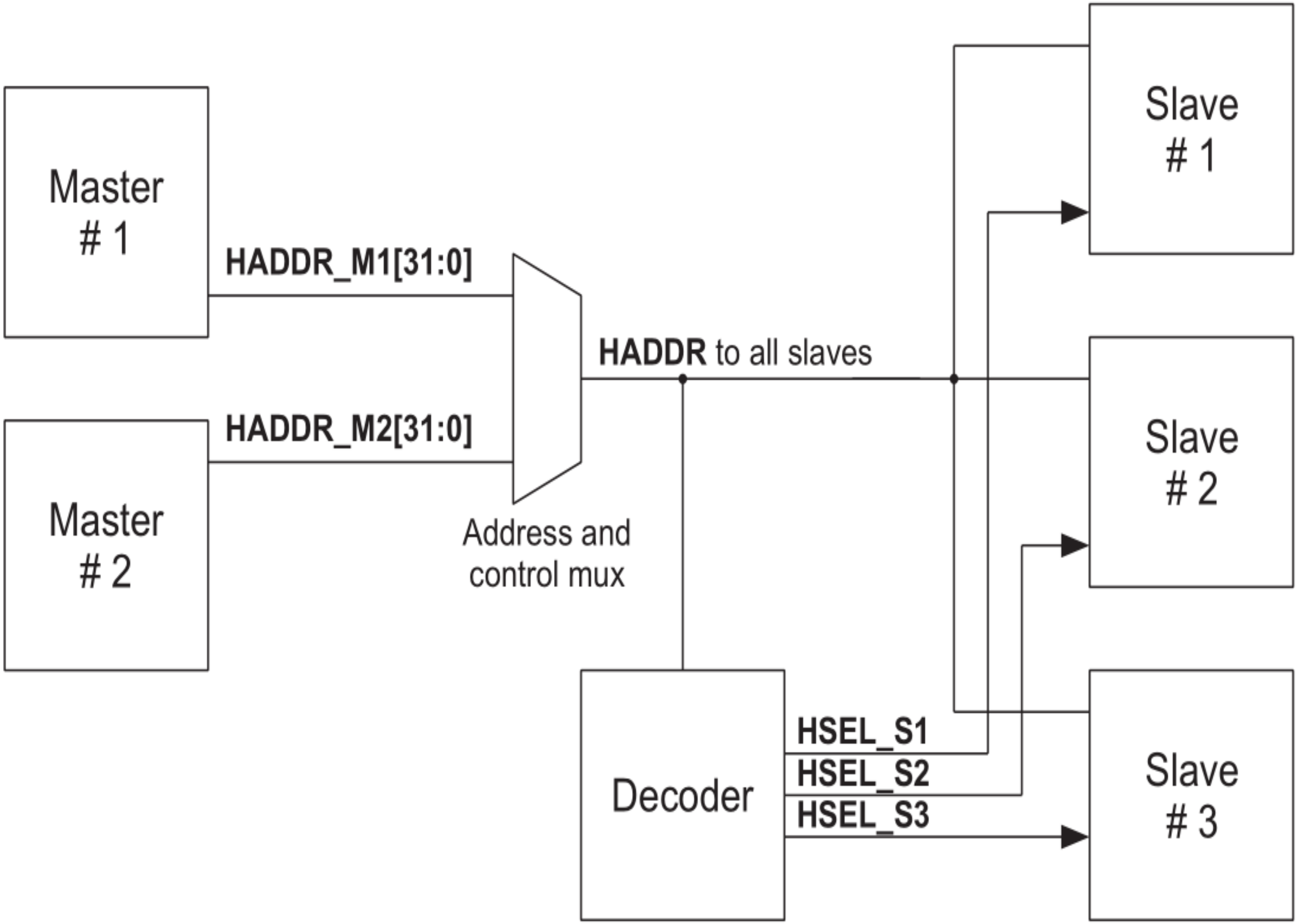
ARM AHB: simple transactions with delay



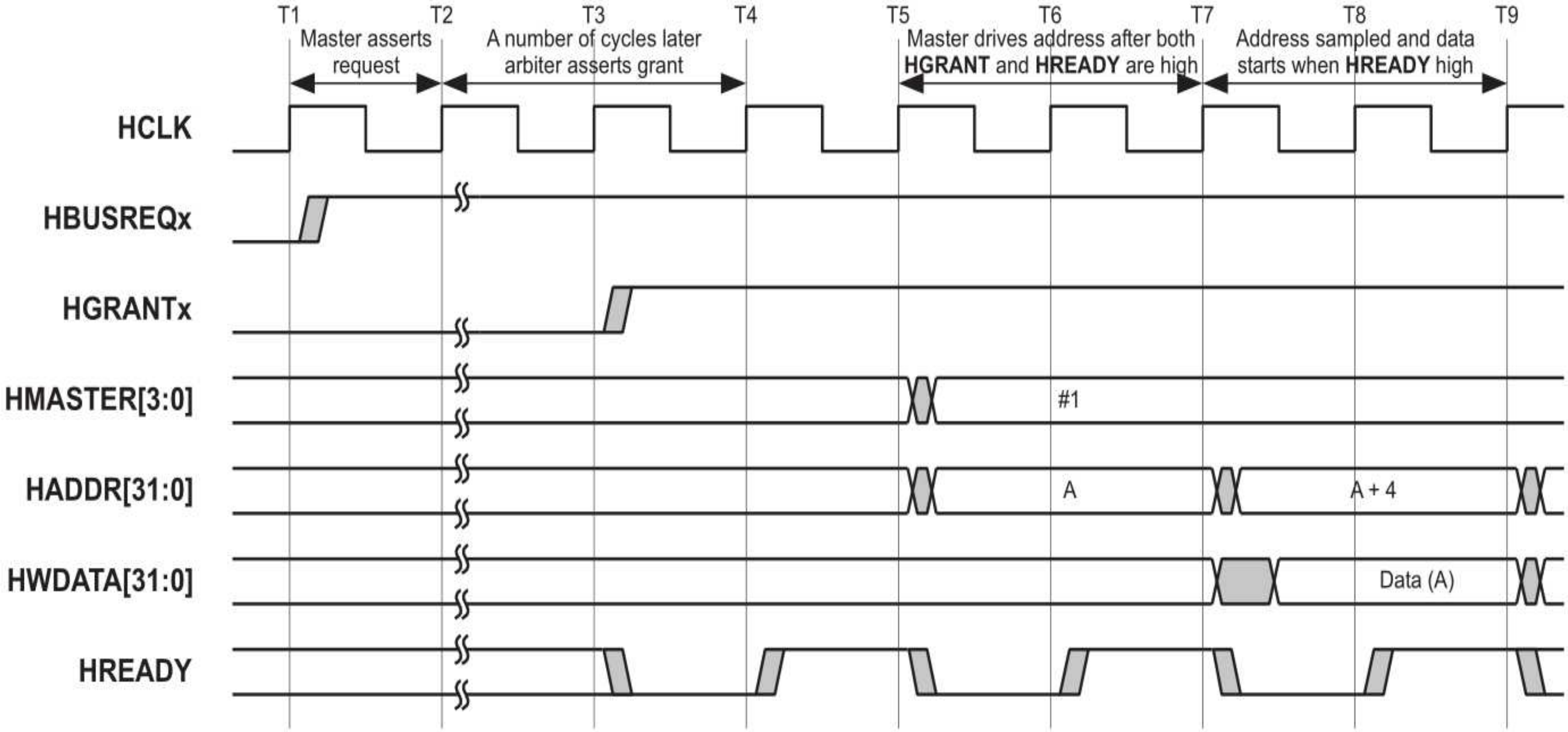
ARM AHB: pipelining for high bandwidth



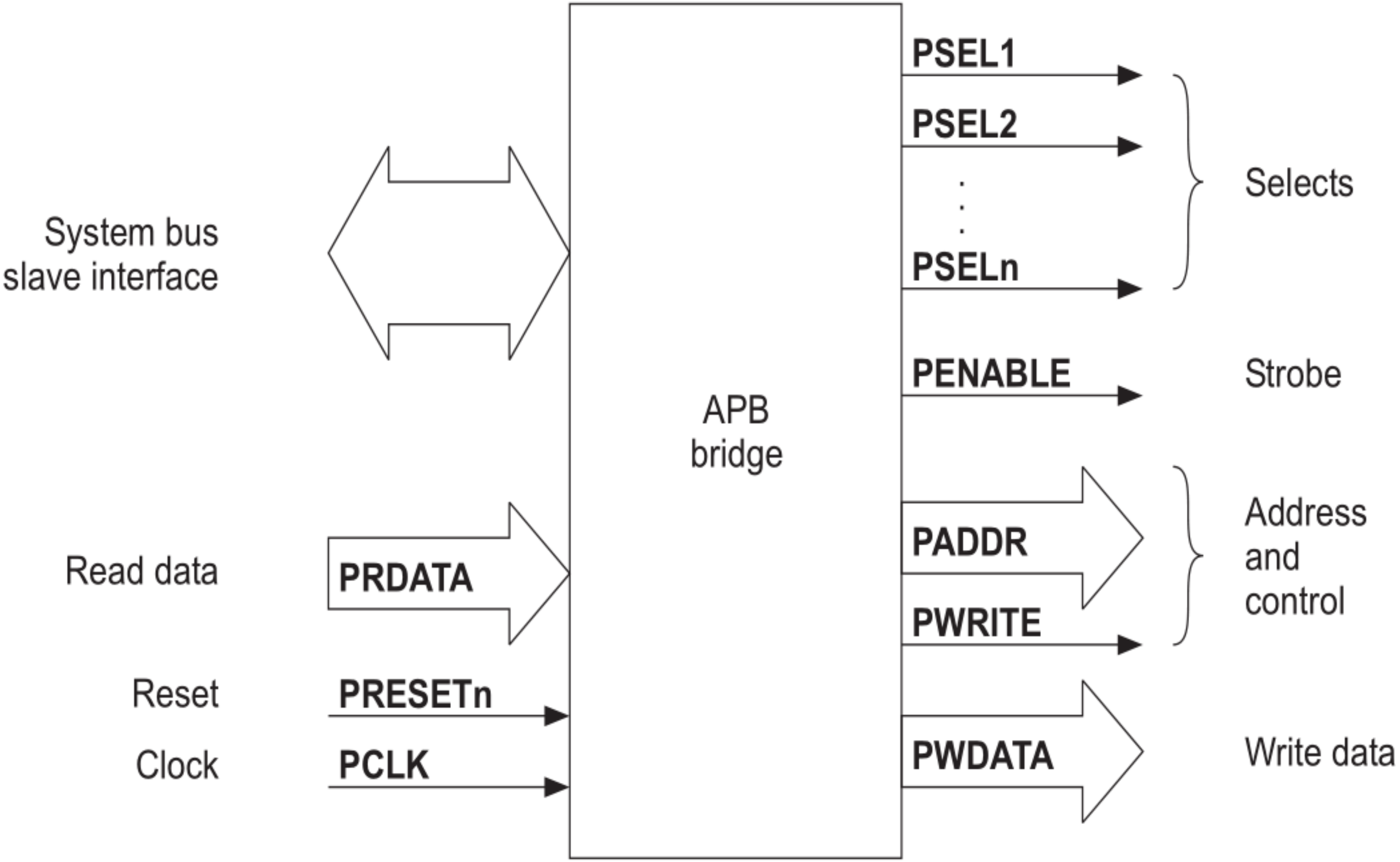
ARM AHB: address decoder



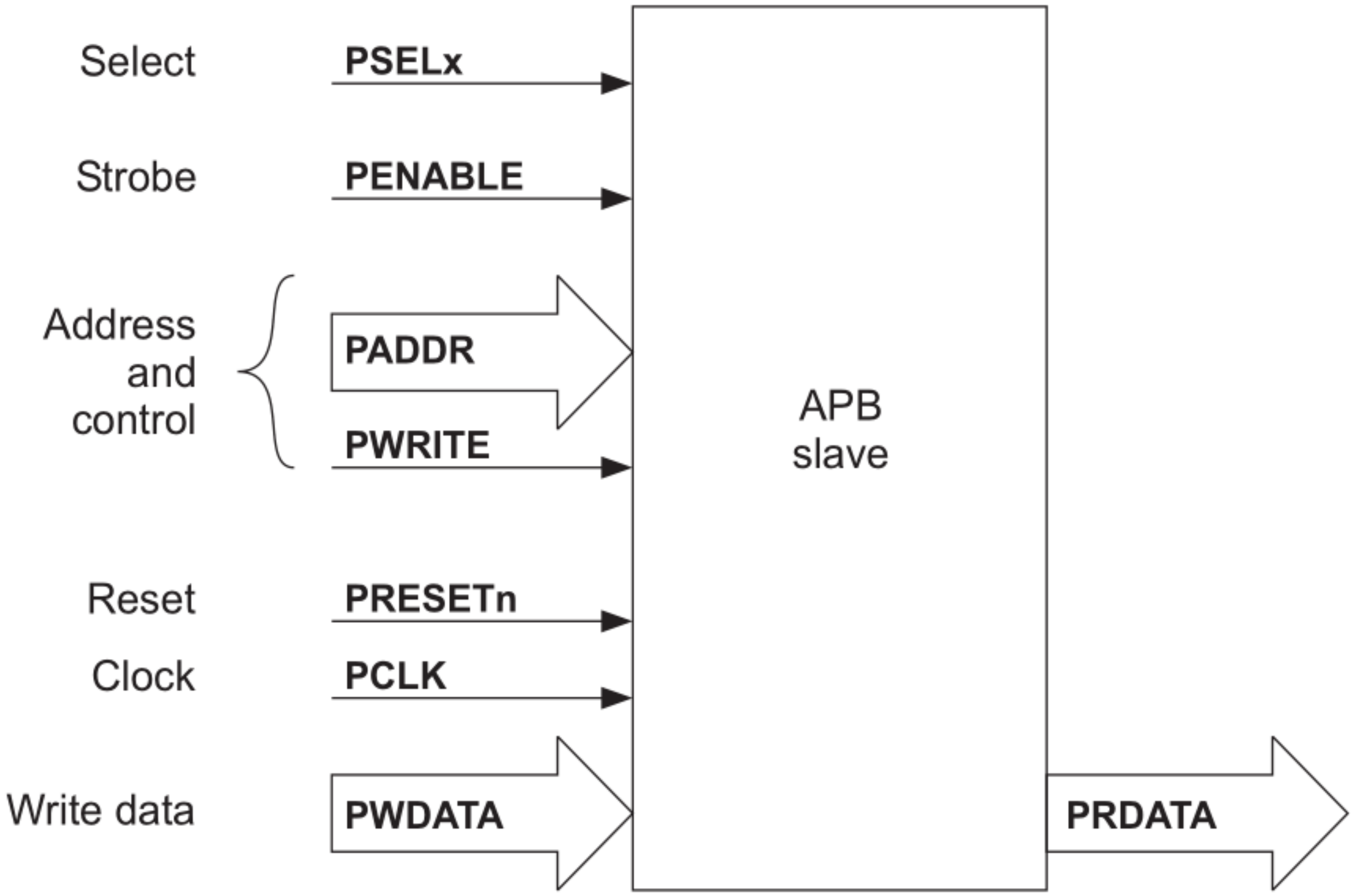
ARM AHB: arbiter


T1
T2
T3
T4
T5
T6
T7
T8
T9
Master asserts request
A number of cycles later arbiter asserts grant
Master drives address after both **HGRANT** and **HREADY** are high
Address sampled and data starts when **HREADY** high
HCLK
HBUSREQx
HGRANTx
HMASTER[3:0]
#1
HADDR[31:0]
A
A + 4
HWDATA[31:0]
Data (A)
HREADY

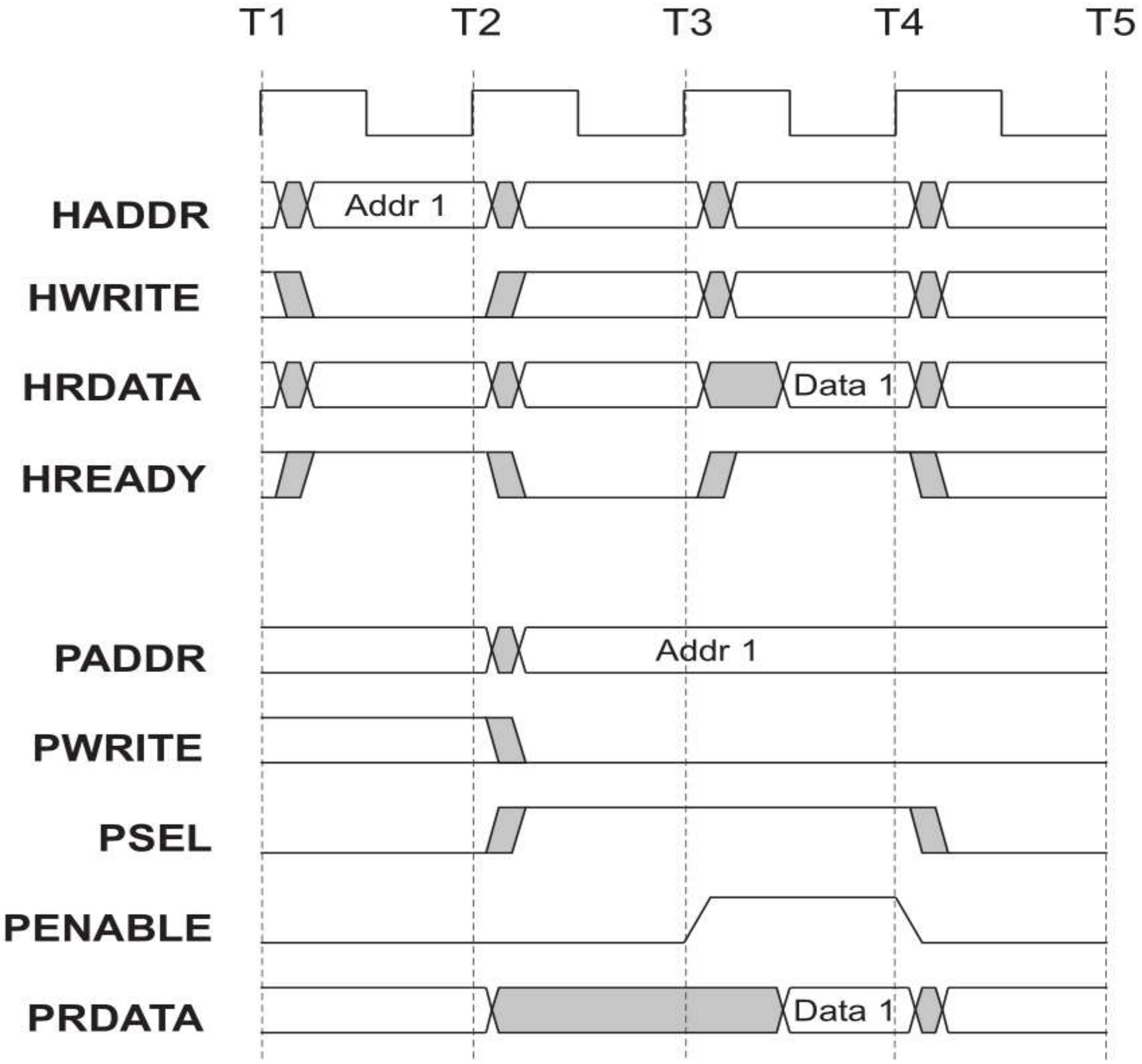
ARM APB: bridge



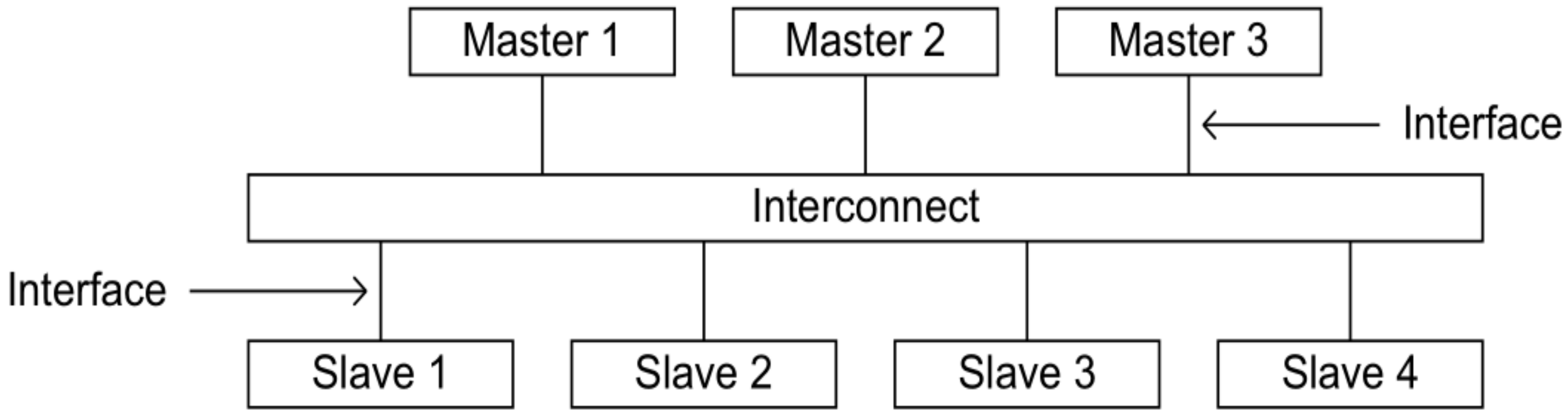
ARM APB: slave

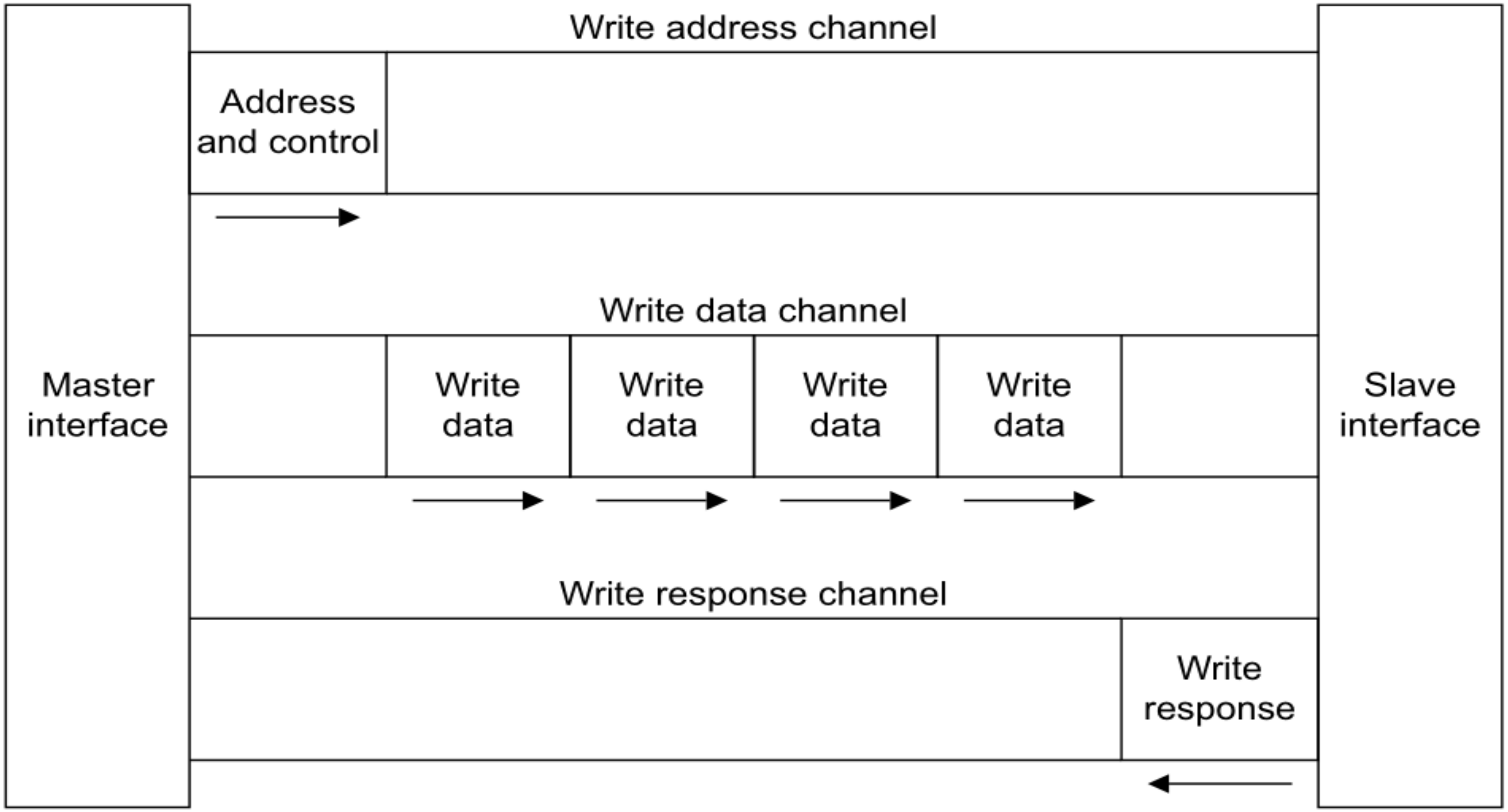


ARM APB: transactions

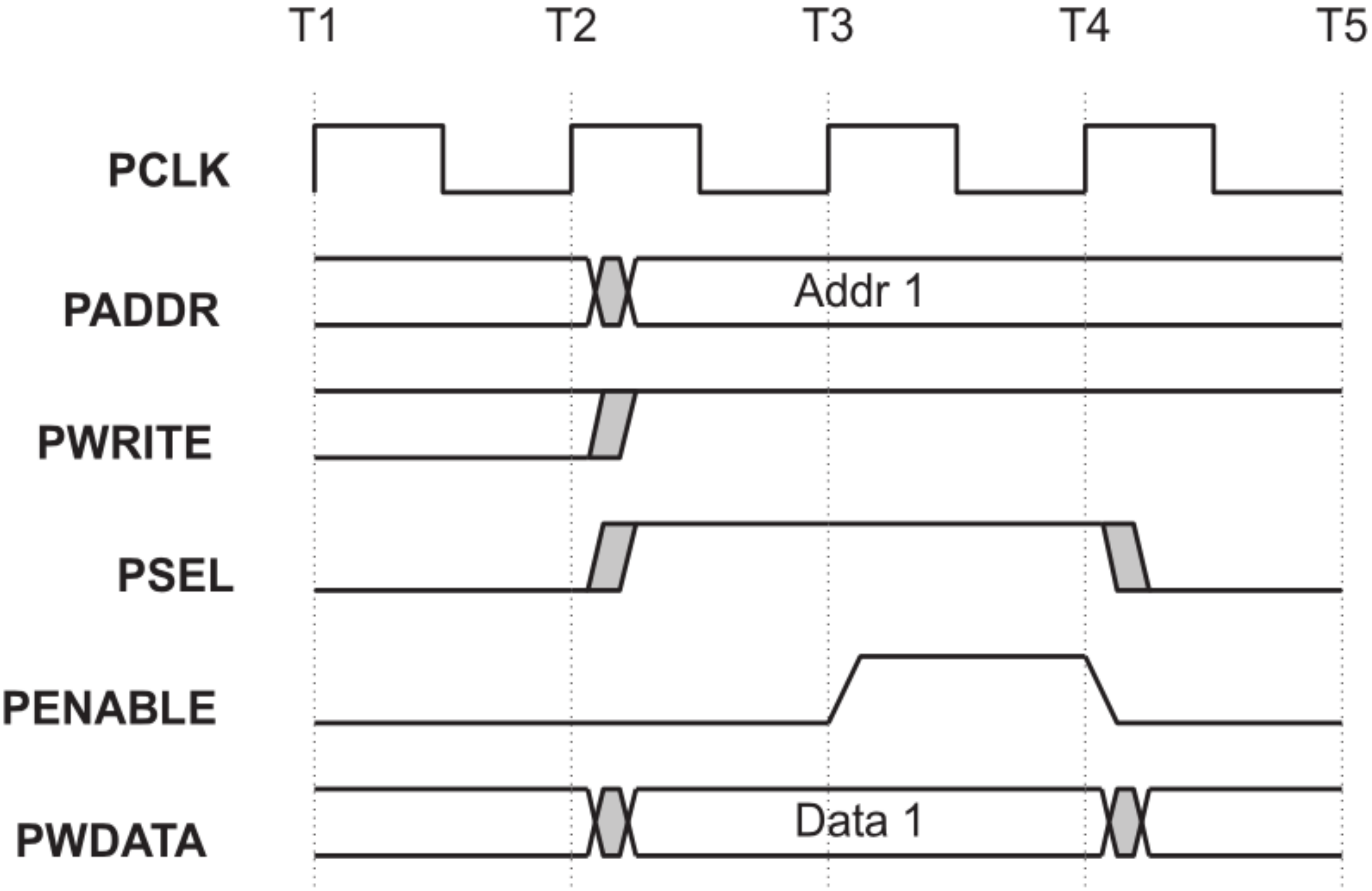


ARM AXI4


Master 1
Master 2
Master 3
Interface
Interconnect
Interface
Slave 1
Slave 2
Slave 3
Slave 4


Write address channel
Address and control
Write data channel
Master interface
Write data
Write data
Write data
Write data
Slave interface
Write response channel
Write response

ARM APB: write cycle


T1
T2
T3
T4
T5
PCLK
PADDR
Addr 1
PWRITE
PSEL
PENABLE
PWDATA
Data 1