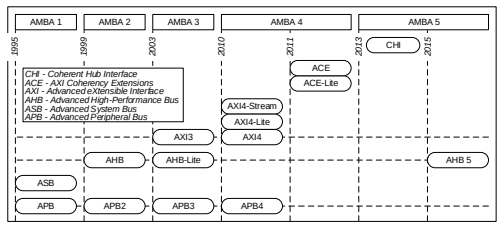
Introduction to AMBA 5 AHB5

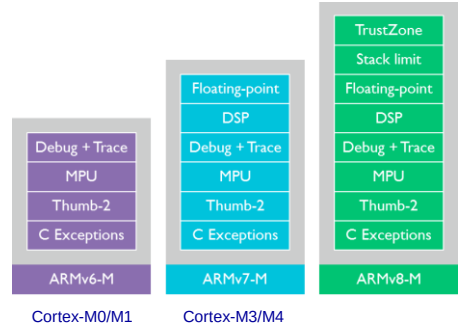
2016 - 2017

Ando Ki
(adki@future-ds.com)

AMBA 5 AHB5

- ARM® AMBA® 5 AHB Protocol Specification: AHB5, AHB-Lite, 2015.
- AMBA 5 AHB5 (Advanced High-performance Bus) is the latest addition to the AMBA family, complementing the ARMv8-M architecture to extend the TrustZone security foundation from the processor to the system.



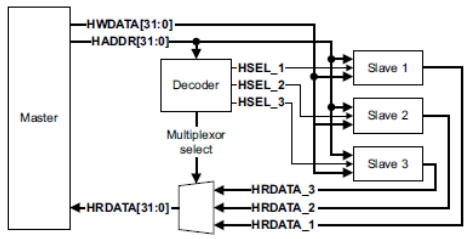


Cortex-M0/M1 Cortex-M3/M4

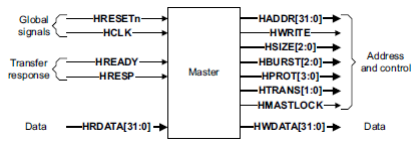
AHB5 new properties

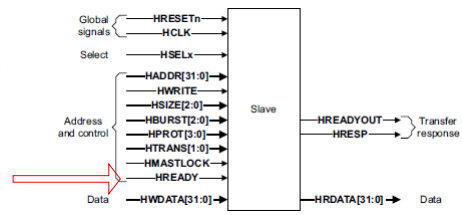
- Extended_Memory_Types
  - HPROT[6:4]
- Secure_Transfers
  - HNONSEC
- Endian
  - Little & Big endian
- Stable_Between_Clock
  - Glitch free between consecutive two rising edge of HCLK.
- Exclusive_Transfers
  - HEXCL, HEXOKAY
- Multi_Copy_Atomicity
  - The single-copy atomicity size defines the number of data bytes that a transfer is guaranteed to update atomically.
    - E.g., 32-bit data
  - Multi_Copy_Atomicity: Writes to the same location are observed in the same order by all agents.
  - Related to cache coherency

AHB blocks (1/2)



- Burst transfers.
- Single clock-edge operation.
- Non-tristate implementation.
- Wide data bus configurations, 64, 12 8, 256, 512, and 1024 bits.





AHB blocks (2/2)

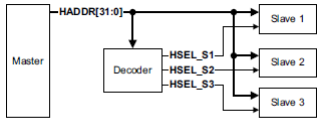
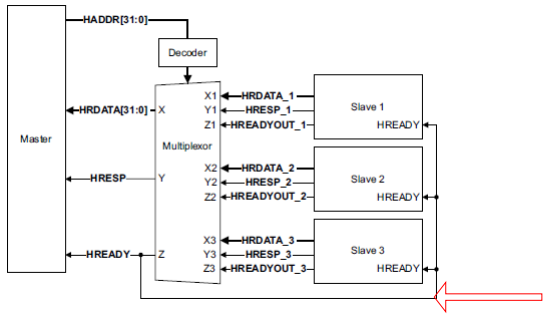


Figure 4-1 Slave select signals



Transfers

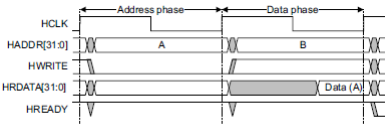


Figure 3-1 Read transfer

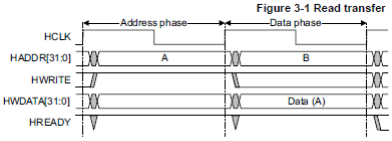


Figure 3-2 Write transfer

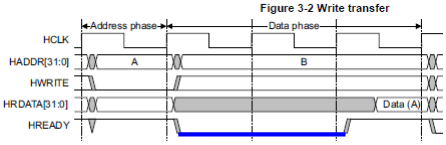


Figure 3-3 Read transfer with two wait states

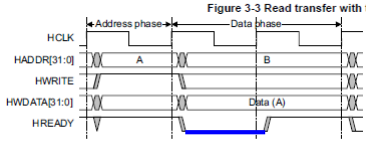


Figure 3-4 Write transfer with one wait state

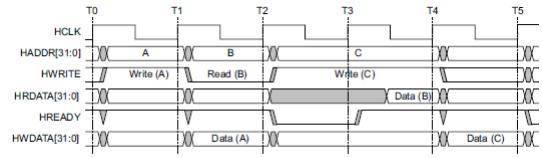


Figure 3-5 Multiple transfers

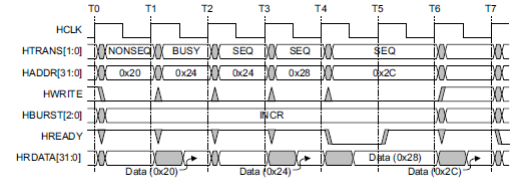


Figure 3-6 Transfer type examples

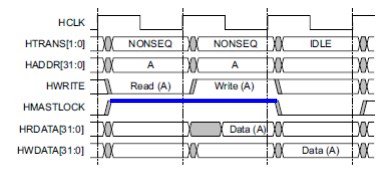


Figure 3-7 Locked transfer

Master signals

Name	Destination	Description
HADDR[31:0]	Slave and decoder	The 32-bit system address bus.
HBURST[2:0]	Slave	The burst type indicates if the transfer is a single transfer or forms part of a burst. Fixed length bursts of 4, 8, and 16 beats are supported. The burst can be incrementing or wrapping. Incrementing bursts of undefined length are also supported. See *Burst operation* on page 3-34.
HMASTLOCK	Slave	When HIGH, indicates that the current transfer is part of a locked sequence. It has the same timing as the address and control signals. See *Locked transfers* on page 3-32.
HPROT[3:0]	Slave	The protection control signals provide additional information about a bus access and indicate how an access should be handled within a system. The signals indicate if the transfer is an opcode fetch or data access, and if the transfer is a privileged mode access or a user mode access. See *Protection control* on page 3-44.
HPROT[6:4]	Slave	The 3-bit extension of the **HPROT** signal that adds extended memory types. This signal extension is supported if the AHB5 Extended_Memory_Types property is True. See *Memory types* on page 3-45.
HSIZE[2:0]	Slave	Indicates the size of the transfer, that is typically byte, halfword, or word. The protocol allows for larger transfer sizes up to a maximum of 1024 bits. See *Transfer size* on page 3-33.
HNONSEC	Slave and decoder	Indicates that the current transfer is either a Non-secure transfer or a Secure transfer. This signal is supported if the AHB5 Secure_Transfers property is True. See *Secure transfers* on page 3-50.
HEXCL	Exclusive Access Monitor	Exclusive Transfer. Indicates that the transfer is part of an Exclusive access sequence. This signal is supported if the AHB5 Exclusive_Transfers property is True. See *Exclusive access signaling* on page 8-72.
HMASTER[3:0]	Exclusive Access Monitor and slave	Master identifier. Generated by a master if it has multiple Exclusive capable threads. Modified by an interconnect to ensure each master is uniquely identified. This signal is supported if the AHB5 Exclusive_Transfers property is True. See *Exclusive access signaling* on page 8-72.
HTRANS[1:0]	Slave	Indicates the transfer type of the current transfer. This can be: • IDLE • BUSY • NONSEQUENTIAL • SEQUENTIAL See *Transfer types* on page 3-30.
HWDATA[31:0][a]	Slave	The write data bus transfers data from the master to the slaves during write operations. A minimum data bus width of 32 bits is recommended. However, this can be extended to enable higher bandwidth operation. See *Data buses* on page 6-60.
HWRITE	Slave	Indicates the transfer direction. When HIGH this signal indicates a write transfer and when LOW a read transfer. It has the same timing as the address signals, however, it must remain constant throughout a burst transfer. See *Basic transfers* on page 3-28.

a. The write data bus width is not restricted to 32 bits. *Data bus width* on page 6-65 lists the other permitted data widths.

Slave, Decoder, and Mux signals

Slave signals

Name	Destination	Description
HRDATA[31:0][a]	Multiplexor	During read operations, the read data bus transfers data from the selected slave to the multiplexor. The multiplexor then transfers the data to the master. A minimum data bus width of 32 bits is recommended. However, this can be extended to enable higher bandwidth operation. See *Data buses* on page 6-60.
HREADYOUT	Multiplexor	When HIGH, the **HREADYOUT** signal indicates that a transfer has finished on the bus. This signal can be driven LOW to extend a transfer. See *Read data and response multiplexor* on page 4-54.
HRESP	Multiplexor	The transfer response, after passing through the multiplexor, provides the master with additional information on the status of a transfer. When LOW, the **HRESP** signal indicates that the transfer status is OKAY. When HIGH, the **HRESP** signal indicates that the transfer status is ERROR. See *Slave transfer responses* on page 5-56.
HEXOKAY	Multiplexor	Exclusive Okay. Indicates the success or failure of an Exclusive Transfer. This signal is supported if the AHB5 Exclusive_Transfers property is True. See *Exclusive access signaling* on page 8-72.

a. The read data bus width is not restricted to 32 bits. *Data bus width* on page 6-65 lists the other permitted data widths.

Decoder signals

Name	Destination	Description
HSELx[a]	Slave	Each slave has its own slave select signal **HSEL**x and this signal indicates that the current transfer is intended for the selected slave. When the slave is initially selected, it must also monitor the status of **HREADY** to ensure that the previous bus transfer has completed, before it responds to the current transfer. The **HSEL**x signal is a combinatorial decode of the address bus. See *Address decoding* on page 4-53.

a. The letter x used in **HSEL**x must be changed to a unique identifier for each slave in a system. For example, **HSEL_S1**, **HSEL_S2**, and **HSEL_Memory**.

Mux signals

Name	Destination	Description
HRDATA[31:0]	Master	Read data bus, selected by the decoder.[a]
HREADY	Master and slave	When HIGH, the **HREADY** signal indicates to the master and all slaves, that the previous transfer is complete. See *Read data and response multiplexor* on page 4-54.
HRESP	Master	Transfer response, selected by the decoder.[a]
HEXOKAY	Master	Exclusive okay, selected by the decoder.[a]

a. Because the **HRDATA**[31:0], **HRESP**, and **HEXOKAY** signals pass through the multiplexor and retain the same signal naming, the full signal descriptions for these three signals are provided in Table 2-3 on page 2-23.

Memory types

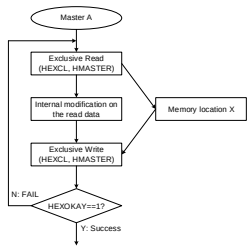
Table 3-4 Protection signal encoding

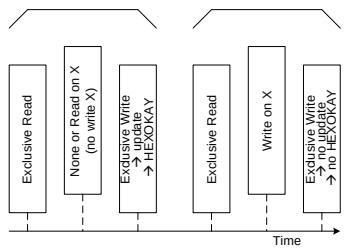
HPROT[3] Modifiable	HPROT[2] Bufferable	HPROT[1] Privileged	HPROT[0] Data/Opcode	Description
-	-	-	0	Opcode fetch
-	-	-	1	Data access
-	-	0	-	User access
-	-	1	-	Privileged access
-	0	-	-	Non-bufferable
-	1	-	-	Bufferable
0	-	-	-	Non-cacheable
1	-	-	-	Cacheable

Bit	Name	Description
HPROT[0]	Data/Inst	When asserted, this bit indicates the transfer is a data access. When deasserted this bit indicates the transfer is an instruction fetch.
HPROT[1]	Privileged	When asserted, this bit indicates the transfer is a privileged access. When deasserted this bit indicates the transfer is an unprivileged access.
HPROT[2]	Bufferable	If both of HPROT[4:3] are deasserted then, when this bit is: • Deasserted, the write response must be given from the final destination. • Asserted, the write response can be given from an intermediate point, but the write transfer is required to be made visible at the final destination in a timely manner.
HPROT[3]	Modifiable	When asserted, the characteristics of the transfer can be modified. When deasserted the characteristics of the transfer must not be modified.
HPROT[4]	Lookup	When asserted, the transfer must be looked up in a cache. When deasserted, the transfer does not need to be looked up in a cache and the transfer must propagate to the final destination.
HPROT[5]	Allocate	When asserted, for performance reasons, this specification recommends that this transfer is allocated in the cache. When deasserted, for performance reasons, this specification recommends that this transfer is not allocated in the cache.
HPROT[6]	Shareable	When asserted, indicates that this transfer is to a region of memory that is shared with other masters in the system. A response for the transfer must not be provided until the transfer is visible to other masters. When deasserted, indicates that this transfer is Non-shareable and the region of memory is not shared with other masters in the system. A response for the transfer does not guarantee the transfer is visible to other masters.

The master sets HPROT to 0b0011 to correspond to a Non-cacheable, Non-bufferable, privileged, data access.
Slaves do not use HPROT unless absolutely necessary.

Exclusive Transfer





- This sequence ensures that the memory location is only updated if, at the point of the store to memory, the location still holds the same value that was used to calculate the new value to be written to the location.
- If the Exclusive Write transfer fails, it is expected that the master will repeat the entire Exclusive access sequence.
- 'Exclusive access monitor'
  - It monitors which master is performing exclusive transfer.

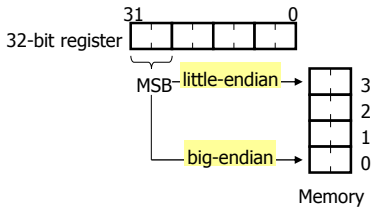
Exclusive signals

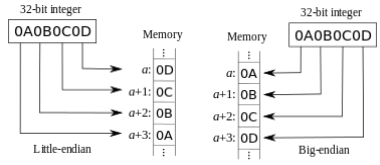
HEXCL	Exclusive Transfer. Indicates that the transfer is part of an Exclusive access sequence. This signal is an address phase signal and has the same validity constraints as **HADDR**.
HMASTER[m:0]	Master Identifier. A master that has multiple Exclusive capable threads must generate this signal to differentiate between the threads. The **HMASTER** value generated by the master will be combined with an interconnect generated value to ensure the resultant **HMASTER** value presented to the Exclusive Access Monitor is unique. This signal is an address phase signal and has the same validity constraints as **HADDR**.
HEXOKAY	Exclusive Okay. An additional response signal is added to indicate the success or failure of an Exclusive Transfer.

- HEXOKAY must only be asserted in the same cycle as HREADY is asserted.
- HEXOKAY must not be asserted in the same cycle as HRESP is asserted.
  - HEXOKAY should not be driven when ERROR, i.e., HRESP is 1.

Endianness

- Little-endian and big-endian





- AMBA AHB usually uses little-endian.
- AHB5 also allows big-endian in two forms.
  - BE8: Byte-invariant big-endian (i.e, address invariant)
    - 메모리에 있는 byte 단위 데이터는 little-endian이든 big-endian이든 동일한 주소를 갖는다.
  - BE32: Word-invariant big-endian
    - 4-byte로 참조할 경우, little-endian이든 big-endian이든 동일한 값으로 참조된다.

Middle-endian for the case of PDP-11:
Big-endian among 16-bit, while little-endian within 16-bit.

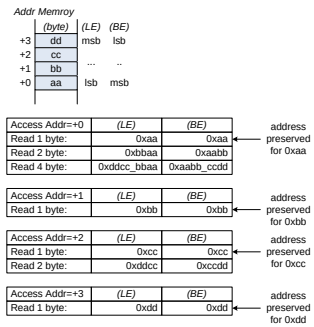
Byte-invariant

Table 6-1 Active byte lanes for a 32-bit little-endian data bus

Transfer size	Address offset	DATA[31:24]	DATA[23:16]	DATA[15:8]	DATA[7:0]
Word	0	Active[MS]	Active	Active	Active[LS]
Halfword	0	-	-	Active[MS]	Active[LS]
Halfword	2	Active[MS]	Active[LS]	-	-
Byte	0	-	-	-	Active
Byte	1	-	-	Active	-
Byte	2	-	Active	-	-
Byte	3	Active	-	-	-

Table 6-2 Active byte lanes for a 32-bit byte-invariant big-endian data bus

Transfer size	Address offset	DATA[31:24]	DATA[23:16]	DATA[15:8]	DATA[7:0]
Word	0	Active[LS]	Active	Active	Active[MS]
Halfword	0	-	-	Active[LS]	Active[MS]
Halfword	2	Active[LS]	Active[MS]	-	-
Byte	0	-	-	-	Active
Byte	1	-	-	Active	-
Byte	2	-	Active	-	-
Byte	3	Active	-	-	-



- Address invariance (byte invariance)
  - When 4-byte data is read by big- or little-endian fashion.
    - The 4-byte are stored at the same address (invariant), but its significance varies depending on access size.
- 위 표에서, 같은 byte는 같은 주소 같은 byte-lane을 사용
  - 특정 주소에 특정 byte가 little/big에 무관하게 동일해야 하므로
  - 혼용을 위해서는 conversion 기능이 버스에 구현되어야 함
- 옆 예제에서,
  - 'aa'는 little-endian이든 big-endian이든 +0 주소에 저장된다.
    - 따라서 significant가 바뀐다.

Word-invariant

Table 6-1 Active byte lanes for a 32-bit little-endian data bus

Transfer size	Address offset	DATA[31:24]	DATA[23:16]	DATA[15:8]	DATA[7:0]
Word	0	Active[MS]	Active	Active	Active[LS]
Halfword	0	-	-	Active[MS]	Active[LS]
Halfword	2	Active[MS]	Active[LS]	-	-
Byte	0	-	-	-	Active
Byte	1	-	-	Active	-
Byte	2	-	Active	-	-
Byte	3	Active	-	-	-

Table 6-3 Active byte lanes for a 32-bit word-invariant big-endian data bus

Transfer size	Address offset	DATA[31:24]	DATA[23:16]	DATA[15:8]	DATA[7:0]
Word	0	Active[MS]	Active	Active	Active[LS]
Halfword	0	Active[MS]	Active[LS]	-	-
Halfword	2	-	-	Active[MS]	Active[LS]
Byte	0	Active	-	-	-
Byte	1	-	Active	-	-
Byte	2	-	-	Active	-
Byte	3	-	-	-	Active

- Data invariance
  - 32-bit data invariance (word invariance)
    - The datum of 32-bit word always the same value independent of endianness.
    - This scheme makes it possible to inter-mix big- and little-endian system without any treatment.
      - However, accesses should keep its access size.
        - E.g., 32-bit data invariance only guarantees data-invariance for 32-bit wide data access.

References

- ARM, AMBA Specification, Chapter 3 AMBA AHB, ARM IHI 0011A, 1999.
- ARM, AMBA3 AHB-Lite Protocol Specification, v1.0, ARM IHI 0033A, 2006.
- ARM, Multi-layer AHB Overview, ARM DVI 0045B, 2004.
- ARM, AMBA 5 AHB Protocol Specification, AHB5, AHB-Lite, ARM IHI 0033B, 2015.