

Design Specifications

IP AMBA ABP Protocol

Table Content

- 1 Specifications
- 2 High Level Block Design
- 3 Low Level Block Design
- 4 RTL Verification

1. Specifications

1. Specifications

1.1 Overview of the AMBA specification

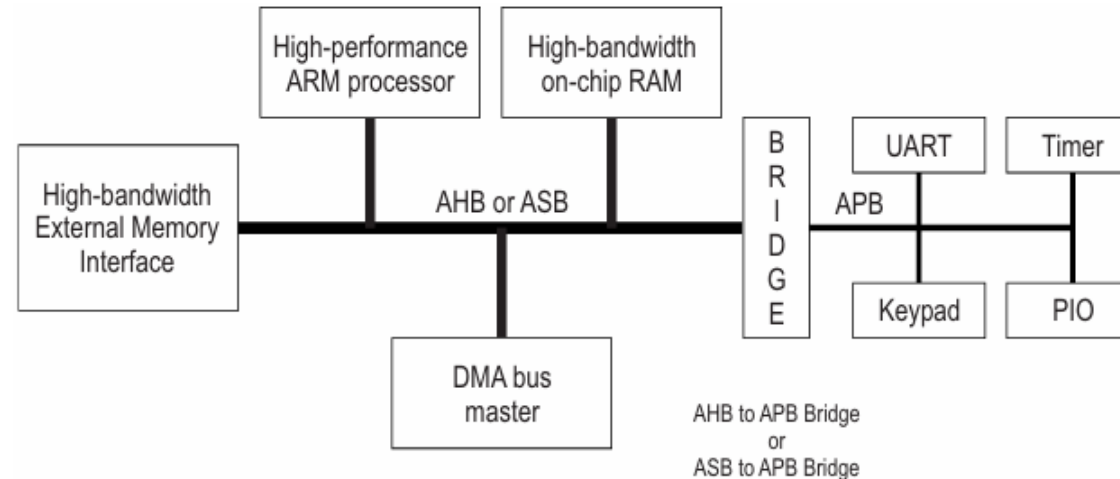
The Advanced Microcontroller Bus Architecture (AMBA) specification defines an on-chip communications standard for designing high-performance embedded microcontrollers.

Four distinct buses are defined within the AMBA specification:

- The Advanced High-performance Bus (AHB)
- The Advanced System Bus (ASB)
- The Advanced Peripheral Bus (APB).
- The Advanced eXtensible Interface (AXI)

1. Specifications

1.1 Overview of the AMBA specification



AMBA AHB	AMBA ASB	AMBA APB
High performance	High performance	Low power
Pipelined operation	Pipelined operation	Latched address and control
Multiple bus masters	Multiple bus masters	Simple interface
Burst transfers		Suitable for many peripherals
Split transactions		

1. Specifications

1.2 Key features of the AMBA Advanced Peripheral Bus (APB)

Low-cost interface: APB is designed with a simple architecture that minimizes hardware resources and implementation cost.

Low power consumption: The non-pipelined and simple transfer mechanism helps reduce power usage, making APB suitable for peripheral devices.

Synchronous protocol: All APB transactions are synchronized to the system clock.

Non-pipelined operation: Transfers are performed sequentially without overlapping phases.

Low bandwidth, optimized for control register access

APB is intended for accessing programmable control registers rather than high-throughput data transfers.

1. Specifications

1.2 Key features of the AMBA Advanced Peripheral Bus (APB)

Designed for simple peripheral devices: Commonly used for peripherals such as GPIO, Timers, UART, SPI, and I2C.

Accessed through an APB bridge: APB peripherals are typically connected to the main system bus through a bridge (e.g., AXI-to-APB bridge).

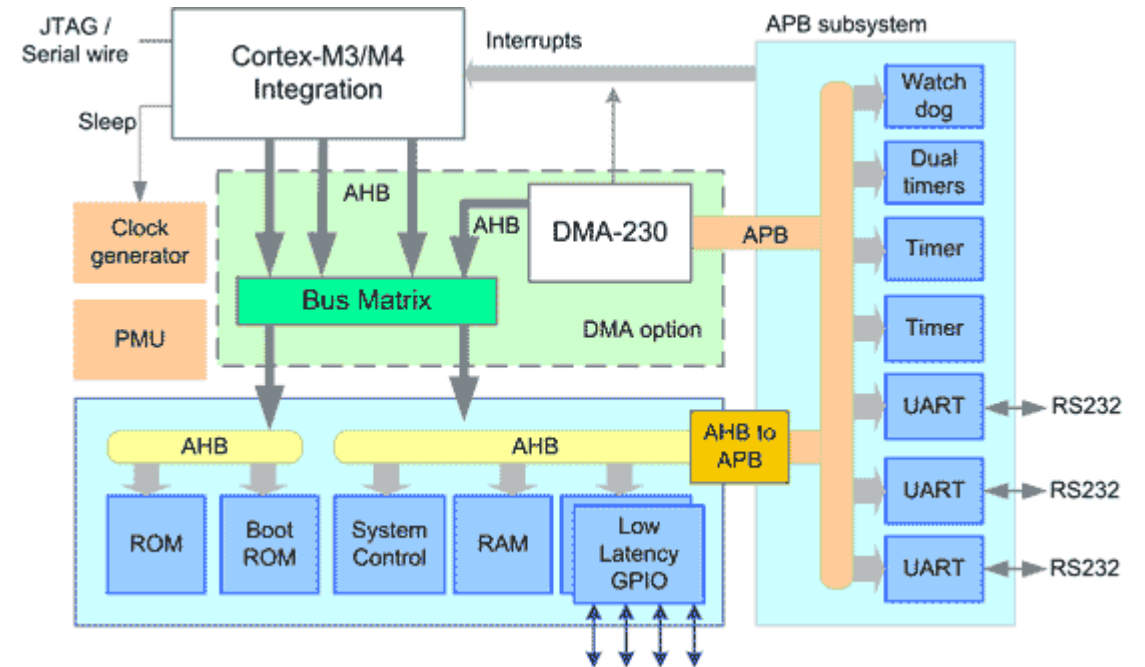
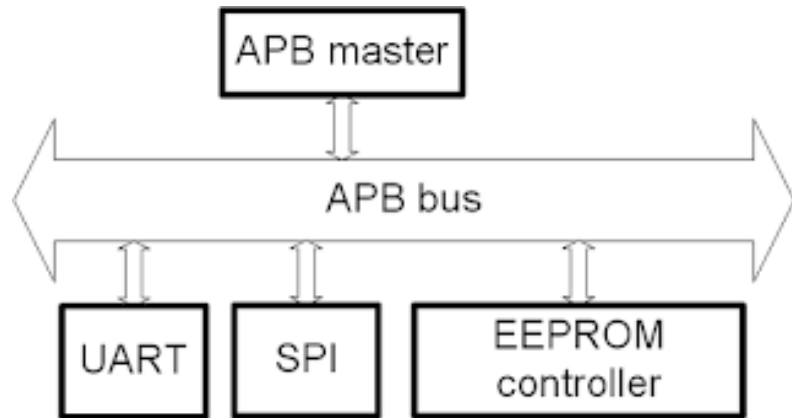
Clear Requester–Completer model:

- The APB bridge acts as the Requester, initiating transactions.
- The peripheral device acts as the Completer, responding to requests.

Well-suited for AMBA-based multi-bus systems: APB integrates seamlessly with other AMBA buses such as AXI and AHB via bridges.

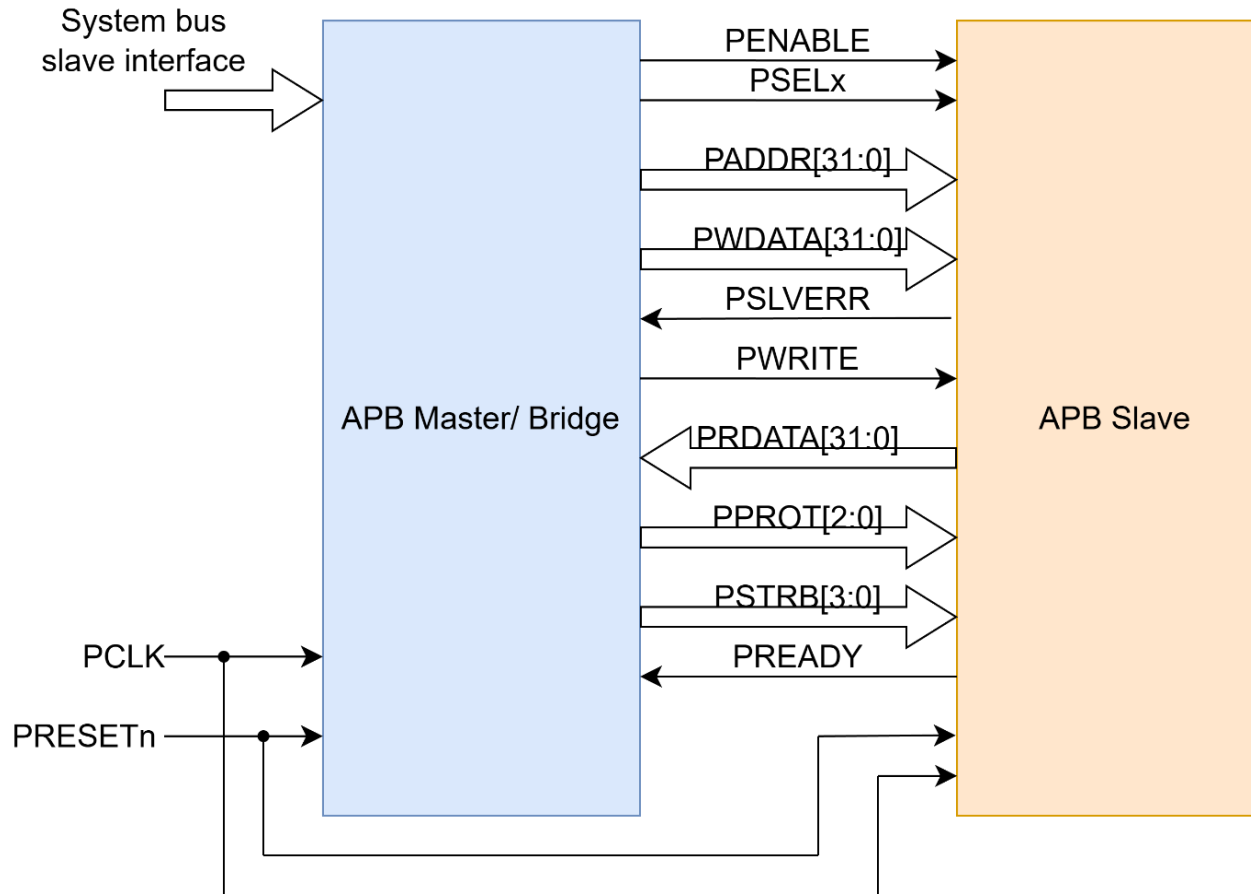
1. Specifications

1.2 Key features of the AMBA Advanced Peripheral Bus (APB)



1. Specifications

1.3 AMBA ABP signals



Block Diagram & Signal Description of APB

1. Specifications

1.3 AMBA ABP signals

Signal	Source	Width	Description
PCLK	Clock Source	1	Clock. The rising edge of PCLK times all transfer on the APB
PRESETn	System Bus	1	Reset. The APB reset signal is active LOW. This signal is normally connected directly to the system bus reset signal.
PADDR	APB bridge	ADDR_WIDTH	Address. This is APB address bus. It can be up to 32 bits wide and is driven by the peripheral bus bridge unit.
PPROT	APB bridge	3	Protection type. This signal indicates the normal, privileged, or secure protection level of the transaction and whether the transaction is a data access or an instruction access.
PSELx	APB bridge	1	Slave select signal, there will be one PSEL signal for each slave connected to master.. It's an active high signal.
PENABLE	APB bridge	1	Enable. It indicates the 2 nd cycle of a data transfer. It's an active high signal.

1. Specifications

1.3 AMBA ABP signals

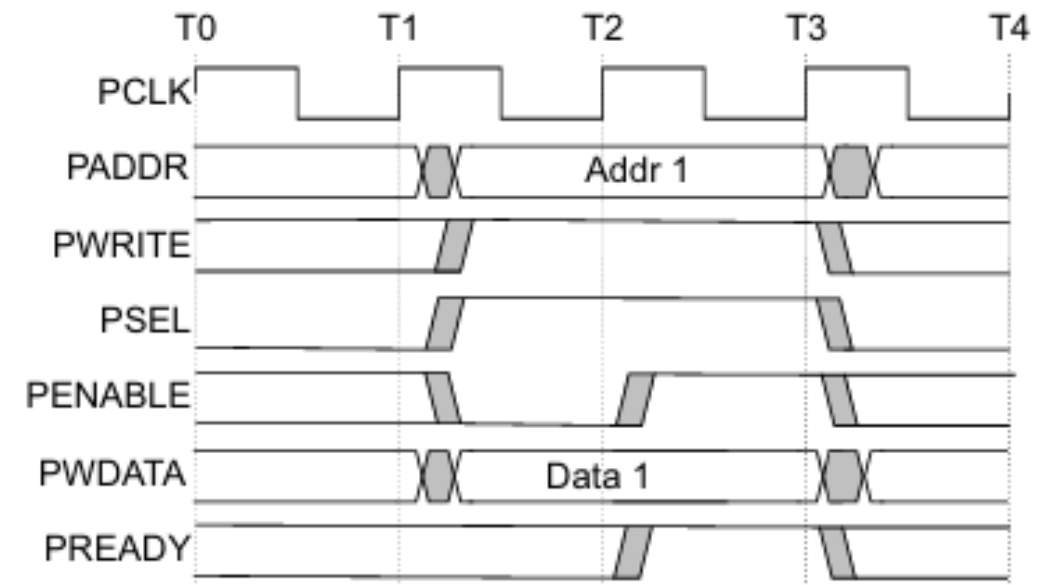
Signal	Source	Width	Description
PWRITE	APB bridge	1	Direction. PWRITE indicates an APB write access when HIGH and an APB read access when LOW.
PWDATA	APB bridge	DATA_WIDTH	Write data. Write data bus from Master to Slave, can be up to 32 bit wide
PSTRB	APB bridge	DATA_WIDTH/ 8	Write strobe. PSTRB indicates which byte lanes to update during a write transfer. There is one write strobe for each 8 bits of the write data bus. PSTRB[n] corresponds to PWDATA[(8n + 7):(8n)]. PSTRB must not be active during a read transfer.
PREADY	Slave Interface	1	It is used by the slave to include wait states in the transfer. i.e. whenever slave is not ready to complete the transaction, it will request the master for some time by de-asserting the PREADY.
PSLVERR	Slave Interface	1	Indicates the Success or failure of the transfer. HIGH indicates failure and LOW indicates Success
PRDATA	Slave Interface	DATA_WIDTH	Read data us from Slave to Master, can be up to 32 bit wide

1. Specifications

1.3 APB Transfers – Write transfers

WRITE Transfer – Without Wait States

- At T1, a write transfer with address **PADDR**, **PWDATA**, **PWRITE** and **PSEL** starts.
- They will be registered at the next rising edge of **PCLK**, T2.
- This is Setup Phase of Transfer.
- After T2, **PENABLE** and **PREADY** are registered at the rising edge of **PCLK**.
- When asserted, **PENABLE** indicates starting of ACCESS Phase
- When asserted, **PREADY** indicates that slave can complete the transfer at the next rising edge of **PCLK**.
- **PADDR**, **PDATA** and control signals all should remain valid till the transfer completes at T3.
- **PENABLE** signal will be de-asserted at the end of transfer.
- **PSEL** is also de-asserted, if next transfer is not to the same slave.

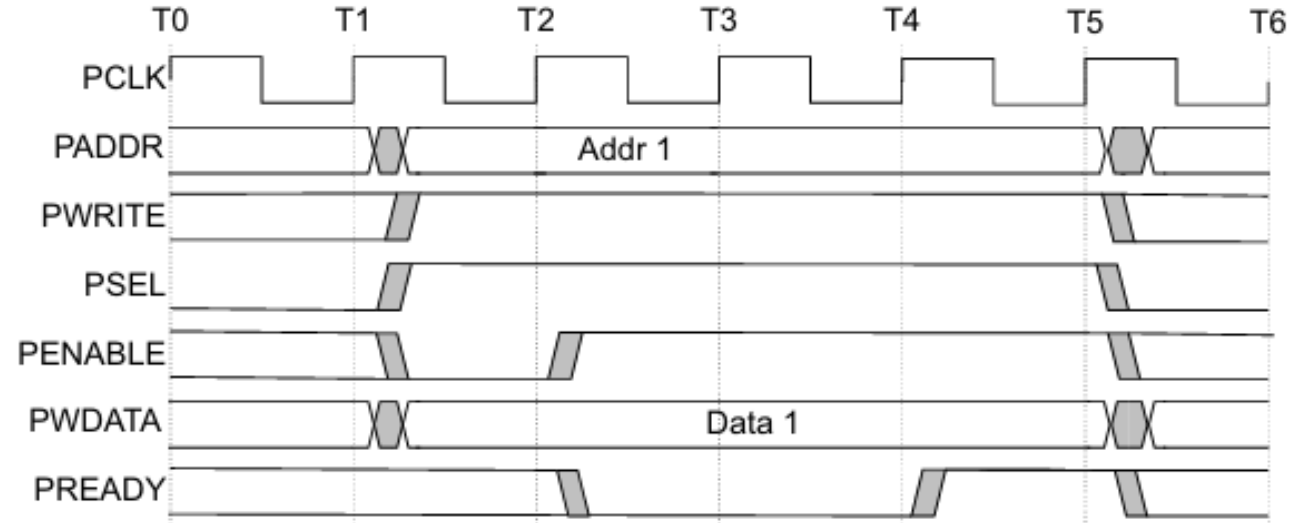


1. Specifications

1.3 APB Transfers – Write transfers

WRITE Transfer – With Wait States

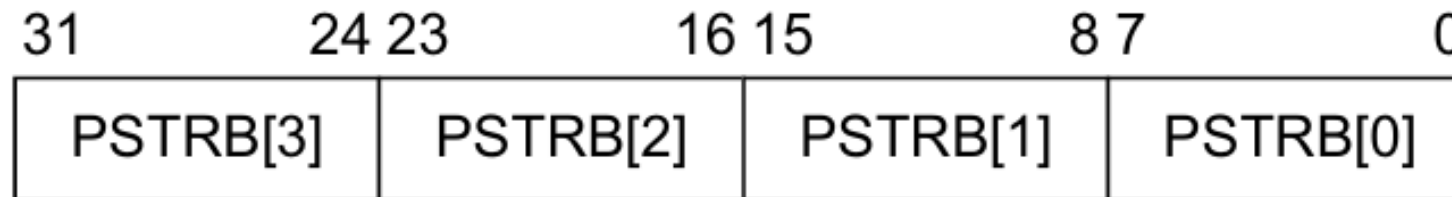
- During the ACCESS Phase, when **PENABLE** is high, the slave extends the transfer by driving **PREADY** low.
- The **PADDR**, **PWRITE**, **PSEL**, **PENABLE**, **PWDATA**, **PSTRB**, **PPROT** signals should remain unchanged while **PREADY** is low
- **PREADY** can take any value when **PENABLE** is low.
- It is recommended that the address and write signals are not changed immediately after a transfer, but remain stable until another access occurs.



1. Specifications

1.3 APB Transfers – Write strobes

- **PSTRB** enables sparse data transfer on the write data bus. Each **PSTRB** corresponds to 1 byte of the write data bus. When asserted HIGH, **PSTRB** indicates that the corresponding byte lane of the write data bus contains valid information.
- There is one write strobe for each 8 bits of the write data bus, so **PSTRB[n]** corresponds to **PWDATA[(8n + 7):(8n)]**.



1. Specifications

1.3 APB Transfers – Write strobes

PSTRB is an optional signal. An APB peripheral might support a limited set of access types, which must be documented for the programmer. This means that all combinations of **PSTRB** presence might be compatible, if this document states that sparse writes are not supported.

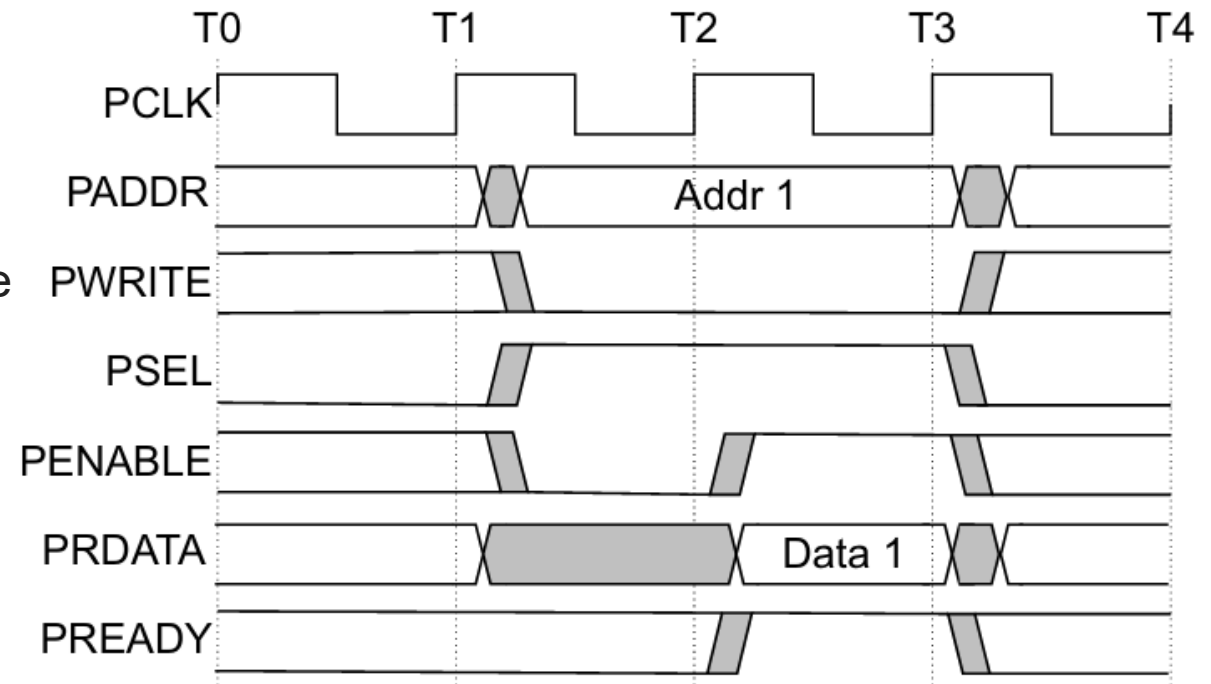
PSTRB	Completer: signal not present	Completer: signal present
Requester: signal not present	Compatible. Sparse writes are not supported.	Compatible. All write data byte lanes are valid for a write. Tie the PSTRB inputs to the PWRITE output from the Requester.
Requester: signal present	Compatible. Sparse writes are not supported.	Compatible.

1. Specifications

1.3 APB Transfers – Read transfers

Read Transfer – Without Wait States

- At T1, a READ transfer with address **PADDR**, **PWRITE** and **PSEL** starts.
- They will be registered at rising edge of **PCLK**.
- This is SETUP Phase of the transfer.
- After T2, **PENABLE** and **PREADY** are registered at the rising edge of **PCLK**.
- When asserted, **PENABLE** indicates the starting of ACCESS phase.
- When asserted, **PREADY** indicates that slave can complete the transfer at next rising edge of **PCLK** by providing the data on **PRDATA**.
- Slave must provide the data before the end of read transfer. i.e. before T3.

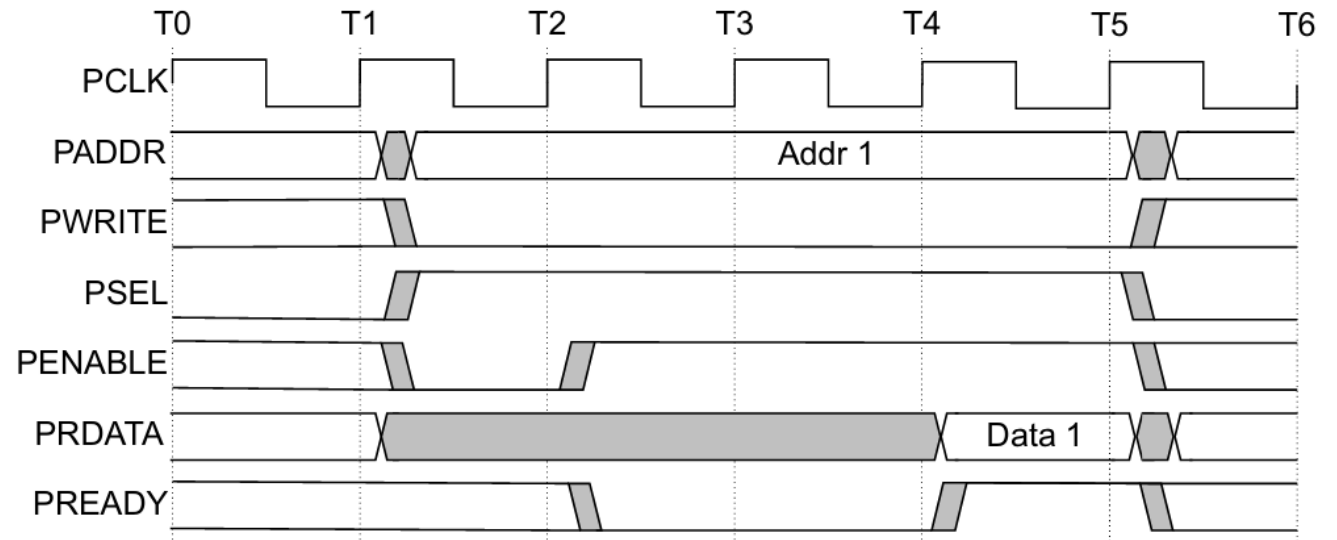


1. Specifications

1.3 APB Transfers – Read transfers

Read Transfer – With Wait States

- During the ACCESS Phase, when **PENABLE** is high, the slave extends the transfer by driving **PREADY** low.
- The **PADDR**, **PWRITE**, **PSEL**, **PENABLE**, **PPROT** signals should remain unchanged while **PREADY** is low



1. Specifications

1.3 APB Transfers – Error response

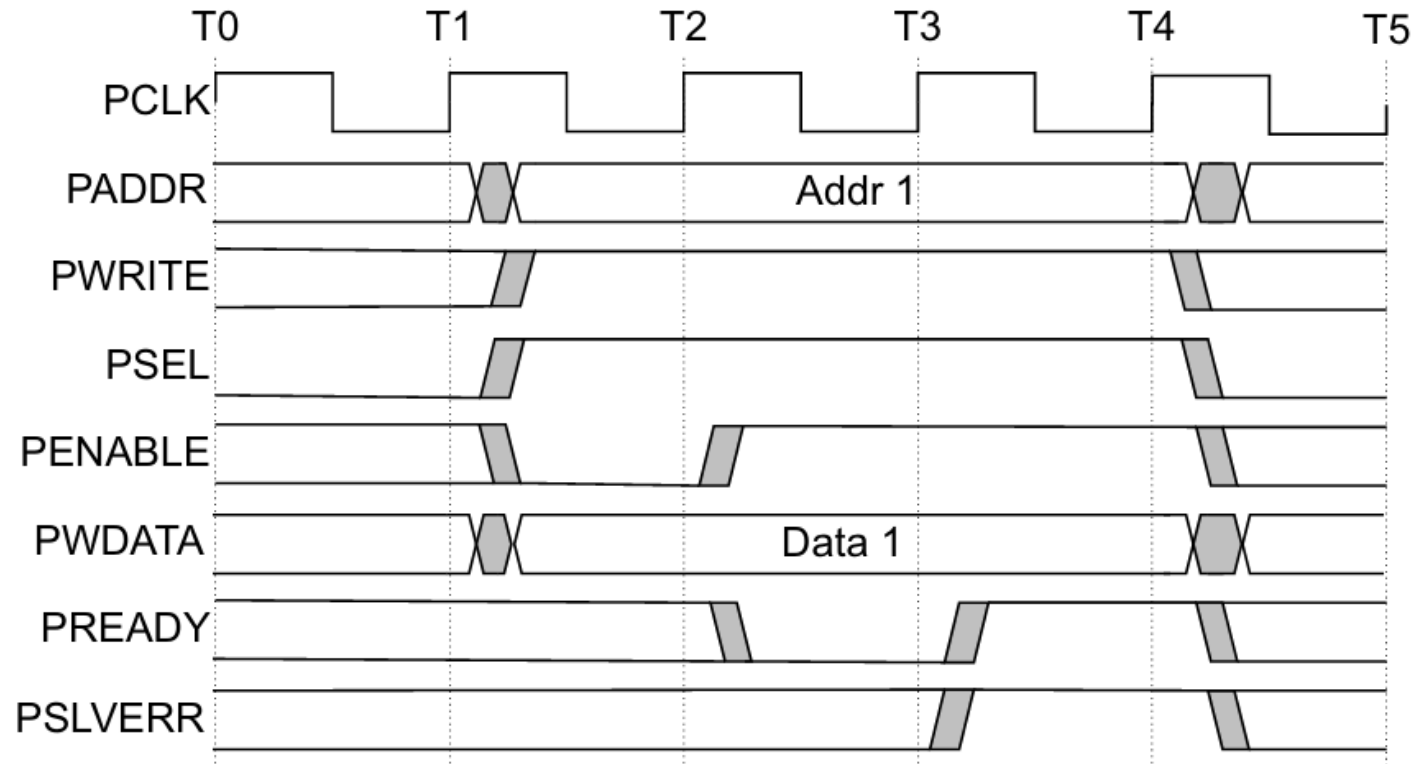
Whenever there is a problem in the transfer, Slave indicates the error response for the transfer by asserting the **PSLVERR** signal. **PSLVERR** is only considered valid during the last cycle of an APB transfer, when **PSEL**, **PENABLE** and **PREADY** are all HIGH. It is recommended, but not mandatory that you drive **PSLVERR** low when it is not being sampled.

Transactions that receive an error response, might or might not have changed the state of peripheral. For example, If APB master performs a write transaction to an APB slave and received an error response, it is not guaranteed that the data is not written on the slave peripheral.

1. Specifications

1.3 APB Transfers – Error response

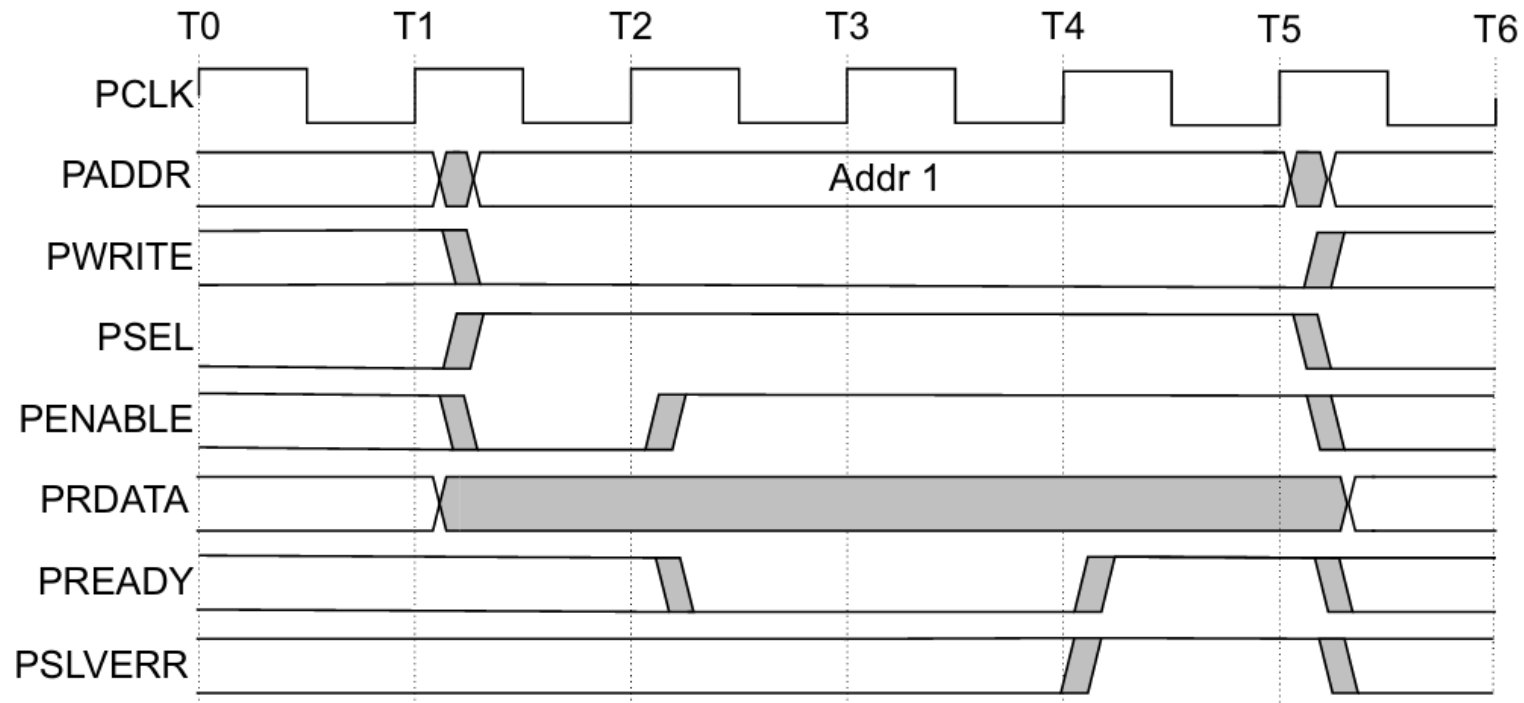
Write transfers



1. Specifications

1.3 APB Transfers – Error response

Read transfers



1. Specifications

1.3 APB Transfers – Protection until support

To support complex system designs, it is often necessary for both the interconnect and other devices in the system to provide protection against illegal transactions. It is provided by Protection Unit in APB Protocol. The signals indicating the protection unit are PPROT[2:0].

PPROT	Protection	Description	Comments
PPROT[0]	Normal or Privileged	PPROT[0] is used by Requesters to indicate processing mode. A privileged processing mode typically has a greater level of access within a system.	<ul style="list-style-type: none">• LOW indicates normal access.• HIGH indicates privileged access.
PPROT[1]	Secure or Non-secure	PPROT[1] is used in systems where a greater degree of differentiation between processing modes is required.	<ul style="list-style-type: none">• LOW indicates secure access.• HIGH indicates non-secure access
PPROT[2]	Data or Instruction	PPROT[2] gives an indication if the transaction is a data or instruction access. The transaction indication is provided as a hint and might not be accurate in all cases.	<ul style="list-style-type: none">• LOW indicates data access.• HIGH indicates instruction access.

1. Specifications

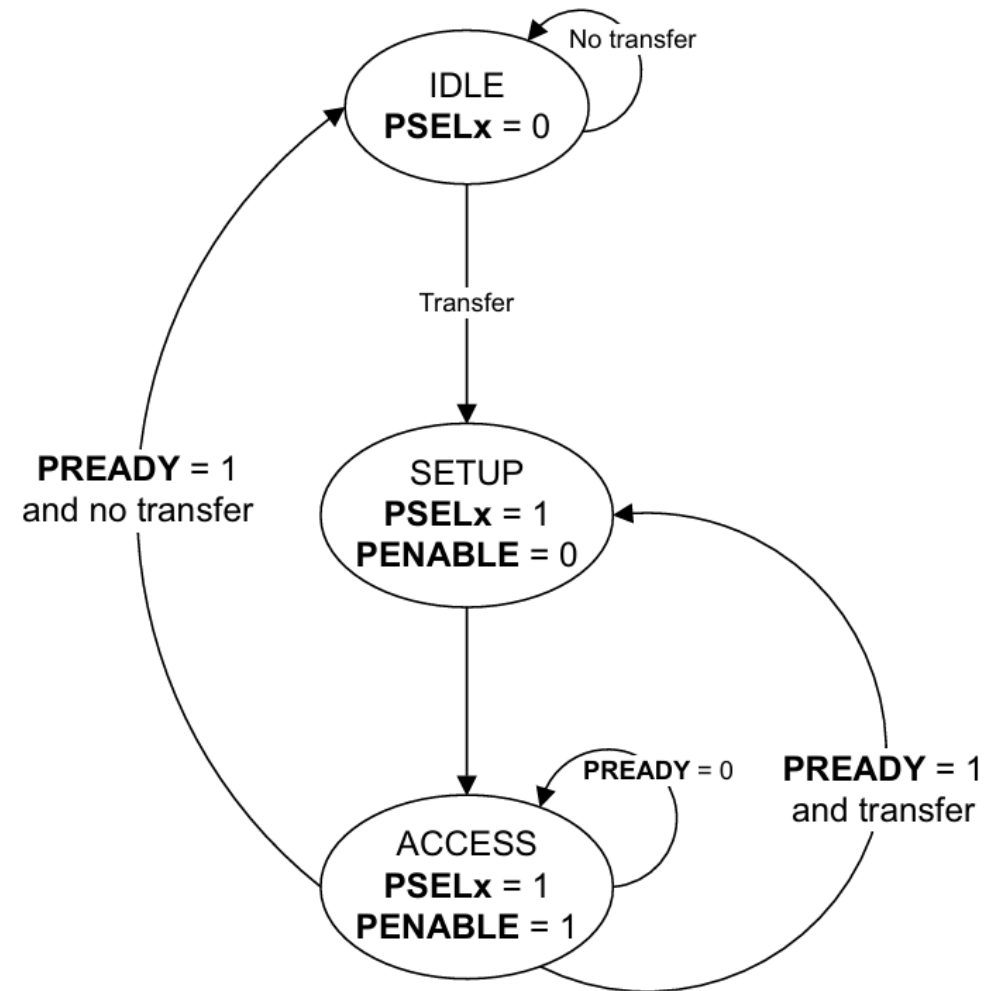
1.4 Operation States

IDLE

This is the default state of the APB interface.

SETUP

When a transfer is required, the interface moves into the SETUP state, where the appropriate select signal, PSELx, is asserted. The interface only remains in the SETUP state for one clock cycle and always moves to the ACCESS state on the next rising edge of the clock.

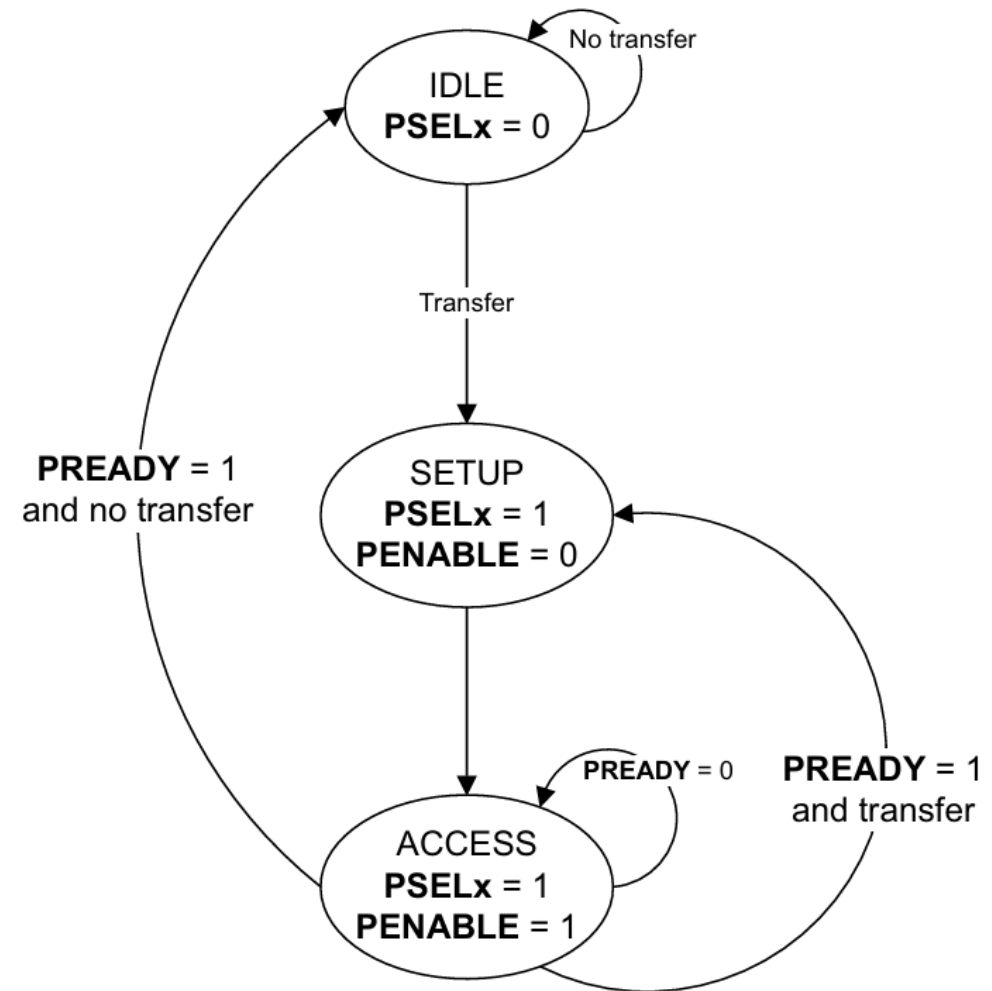


1. Specifications

1.4 Operation States

ACCESS

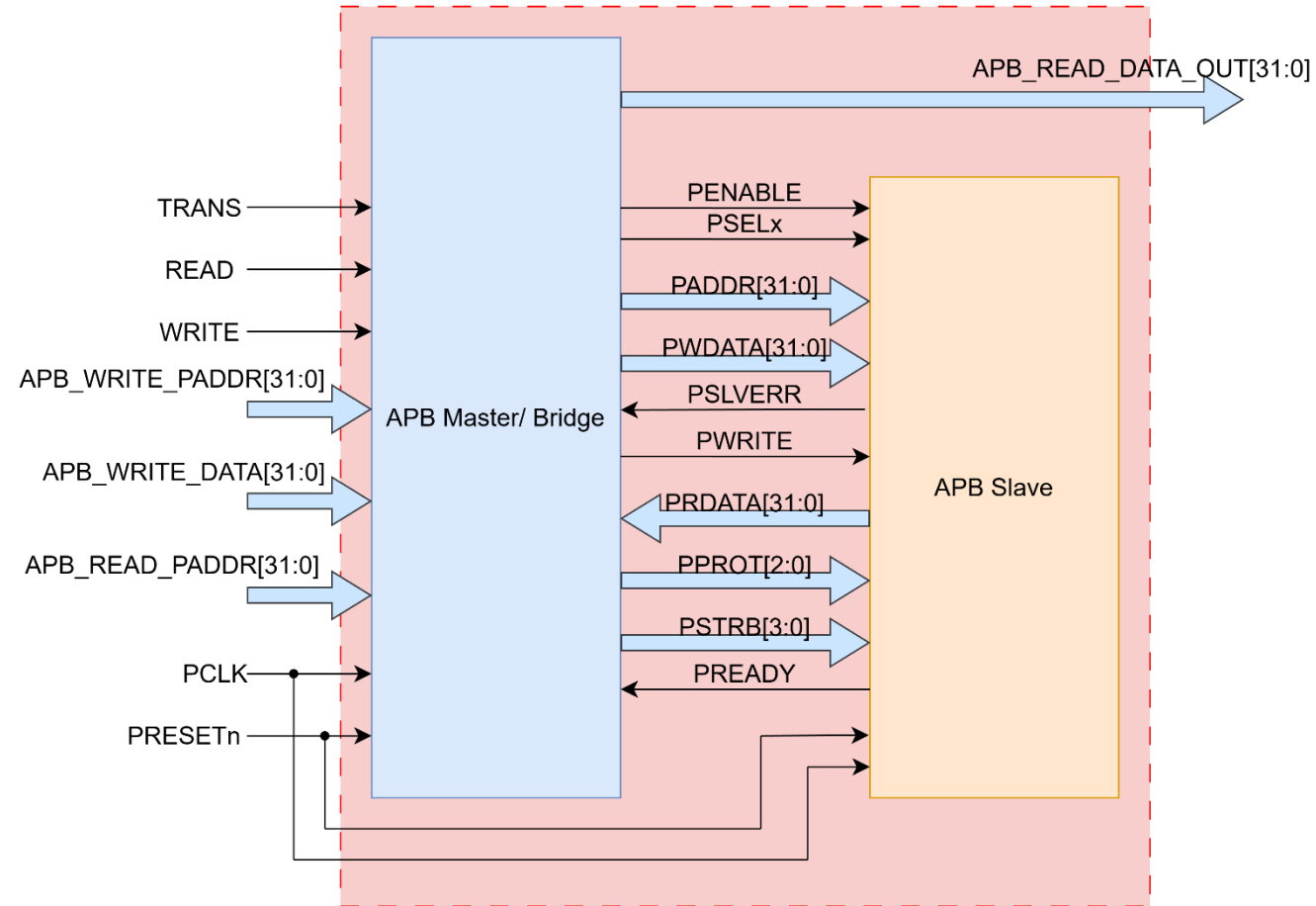
- **PENABLE** is asserted in the ACCESS state. During the transition from SETUP to ACCESS and throughout the ACCESS state, the following signals must remain stable: **PADDR**, **PPROT**, **PWRITE**, **PSTRB**, **PAUSER**, **PWUSER**, and **PWDATA** (for write transfers only).
- The exit from the ACCESS state depends on **PREADY** from the Completer: if **PREADY** is LOW, the bus **stays in ACCESS**; if **PREADY** is HIGH, the bus **leaves ACCESS** and returns to IDLE or moves directly to SETUP for the next transfer.



2. High Level Block Design

2. High Level Block Design

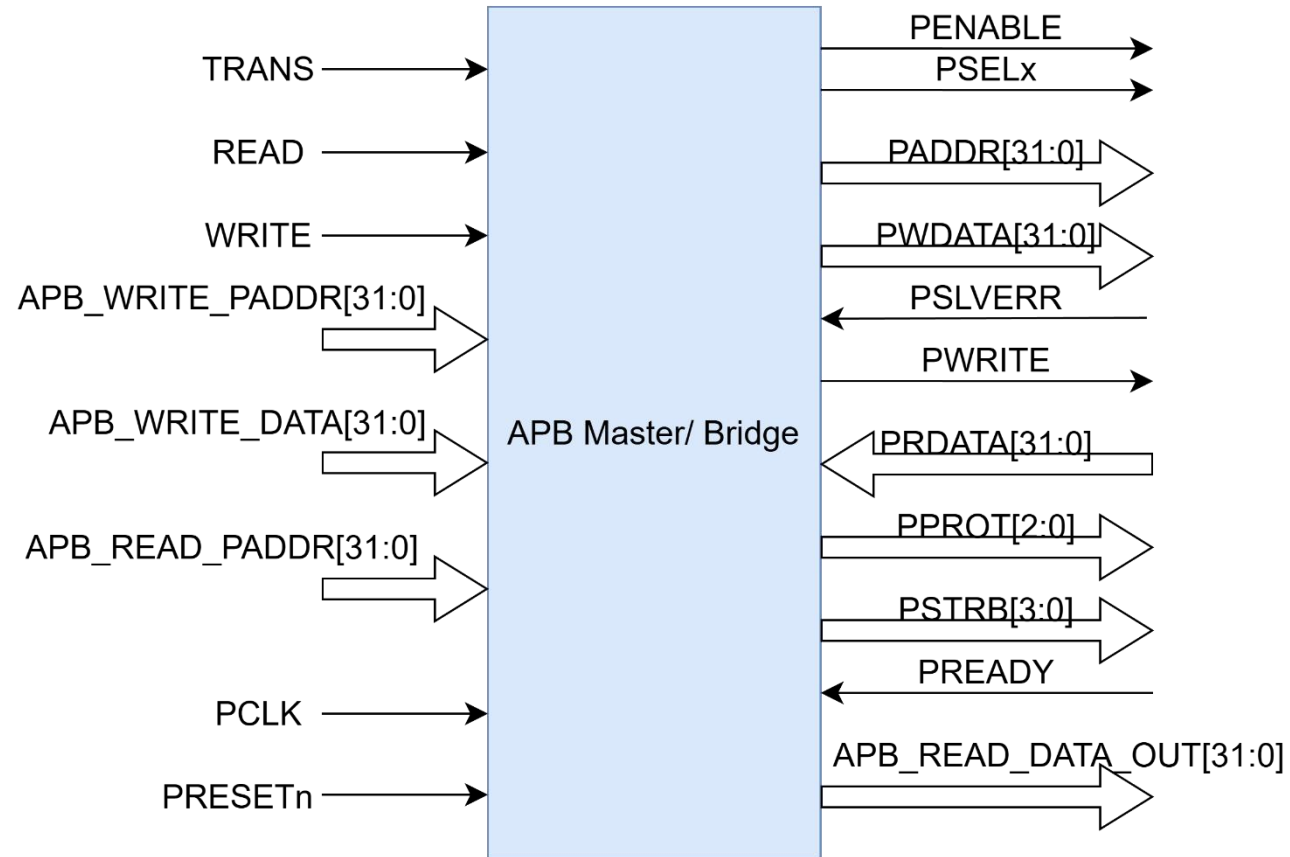
2.1 APB Interface



APB Interface

2. High Level Block Design

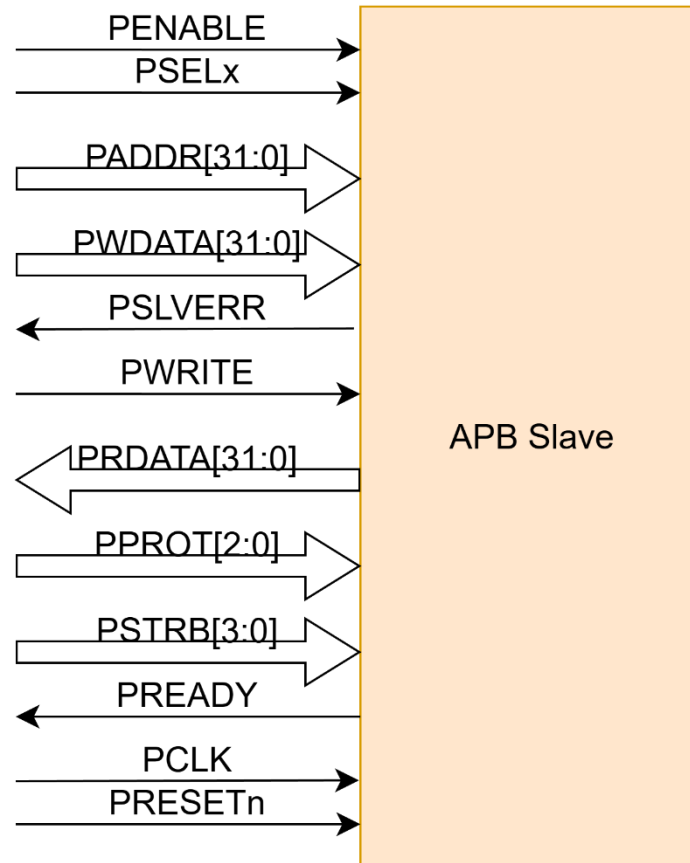
2.2 APB Master



APB Master block

2. High Level Block Design

2.3 APB Slave



APB Slave block

2. High Level Block Design

2.4 AMBA ABP Interface signals

Signal	Source	Width	Description
TRANS	System BUS	1	Start signal. When the TRANS signal is set to HIGH, the system will begin operation. This signal may remain active for one clock cycle or multiple cycles.
READ	System BUS	1	Read signal.
WRITE	System BUS	1	Write signal.
APB_WRITE_PADDR	System BUS	ADDR_WIDTH	Address. This is the APB write address used by the APB master to write data to the slave. The address can be up to 32 bits wide.
APB_WRITE_DATA	System BUS	DATA_WIDTH	Data Write. This is the data sent from the APB master to the APB slave during a write operation. The data width can be up to 32 bits.
APB_READ_PADDR	System BUS	ADDR_WIDTH	Address. This is the APB read address used by the APB master to read data from the slave. The address can be up to 32 bits wide.
APB_READ_DATA_OUT	System BUS	DATA_WIDTH	Data Out. This is the data which is read from slave. The data width can be up to 32 bits.

2. High Level Block Design

2.4 AMBA ABP Interface signals

Signal	Source	Width	Description
PCLK	Clock Source	1	Clock. The rising edge of PCLK times all transfer on the APB
PRESETn	System Bus	1	Reset. The APB reset signal is active LOW. This signal is normally connected directly to the system bus reset signal.
PADDR	APB bridge	ADDR_WIDTH	Address. This is APB address bus. It can be up to 32 bits wide and is driven by the peripheral bus bridge unit.
PPROT	APB bridge	3	Protection type. This signal indicates the normal, privileged, or secure protection level of the transaction and whether the transaction is a data access or an instruction access.
PSELx	APB bridge	1	Slave select signal, there will be one PSEL signal for each slave connected to master.. It's an active high signal.
PENABLE	APB bridge	1	Enable. It indicates the 2 nd cycle of a data transfer. It's an active high signal.

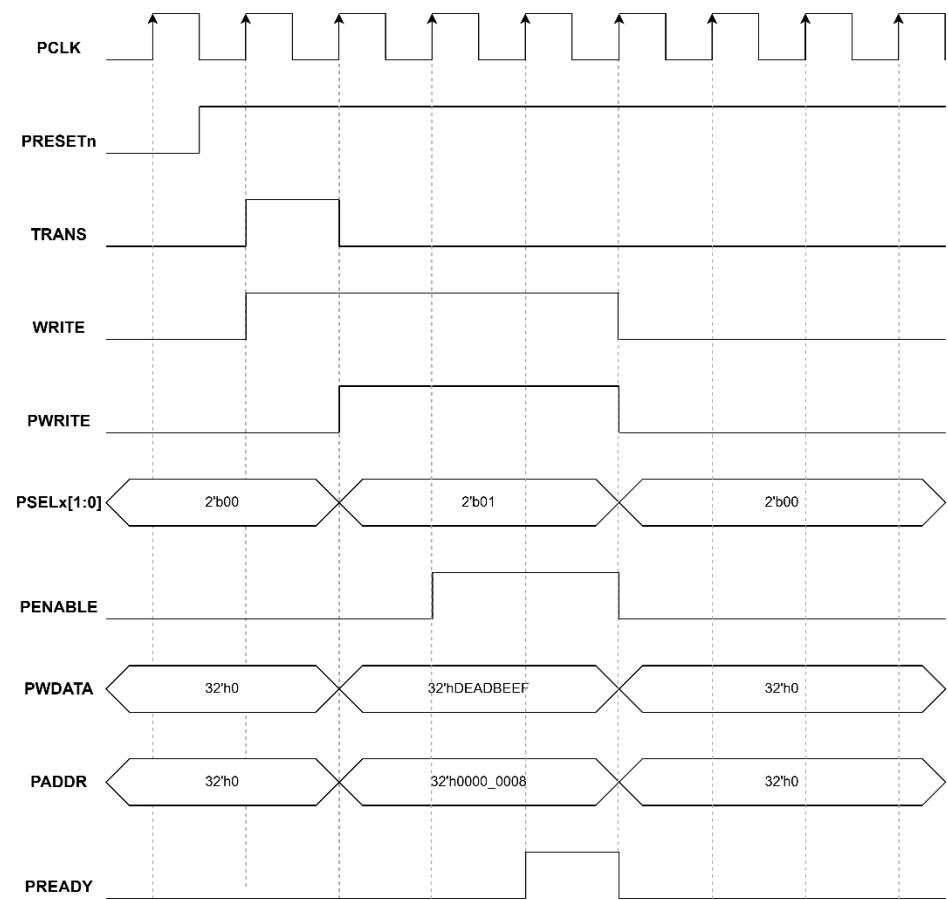
2. High Level Block Design

2.4 AMBA ABP Interface signals

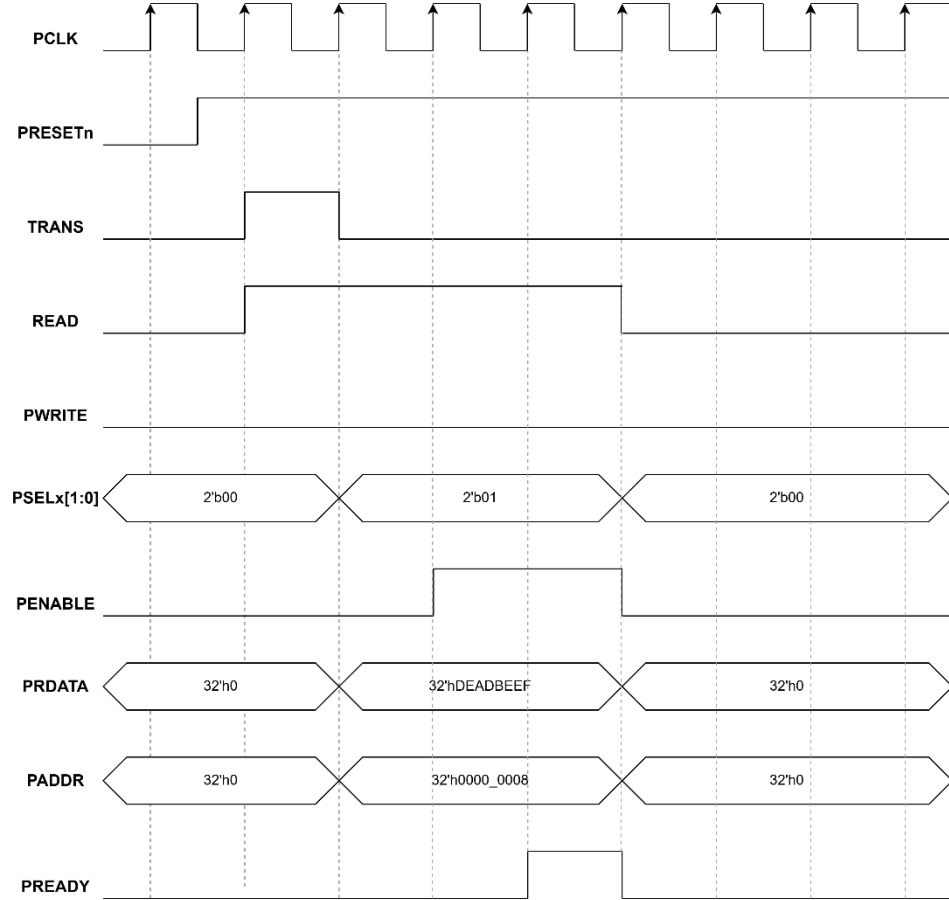
Signal	Source	Width	Description
PWRITE	APB bridge	1	Direction. PWRITE indicates an APB write access when HIGH and an APB read access when LOW.
PWDATA	APB bridge	DATA_WIDTH	Write data. Write data bus from Master to Slave, can be up to 32 bit wide
PSTRB	APB bridge	DATA_WIDTH/8	Write strobe. PSTRB indicates which byte lanes to update during a write transfer. There is one write strobe for each 8 bits of the write data bus. PSTRB[n] corresponds to PWDATA[(8n + 7):(8n)]. PSTRB must not be active during a read transfer.
PREADY	Slave Interface	1	It is used by the slave to include wait states in the transfer. i.e. whenever slave is not ready to complete the transaction, it will request the master for some time by de-asserting the PREADY.
PSLVERR	Slave Interface	1	Indicates the Success or failure of the transfer. HIGH indicates failure and LOW indicates Success
PRDATA	Slave Interface	DATA_WIDTH	Read data us from Slave to Master, can be up to 32 bit wide

2. High Level Block Design

2.4 AMBA ABP Interface signals



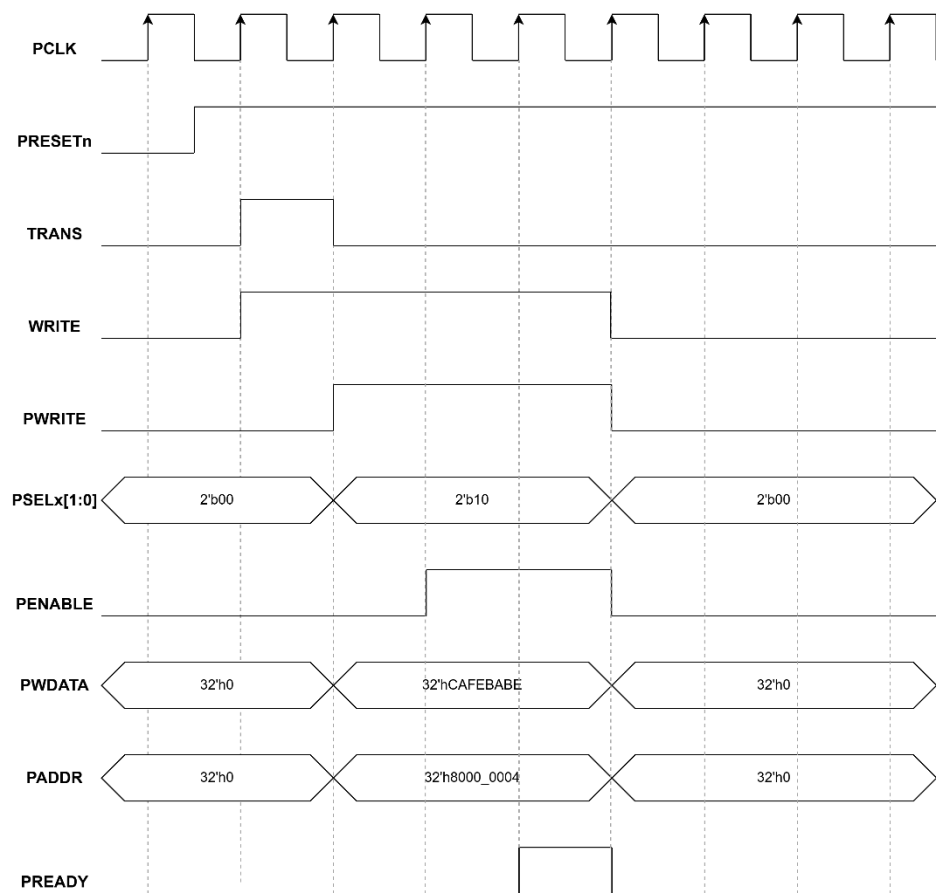
APB Write transfer without waiting state



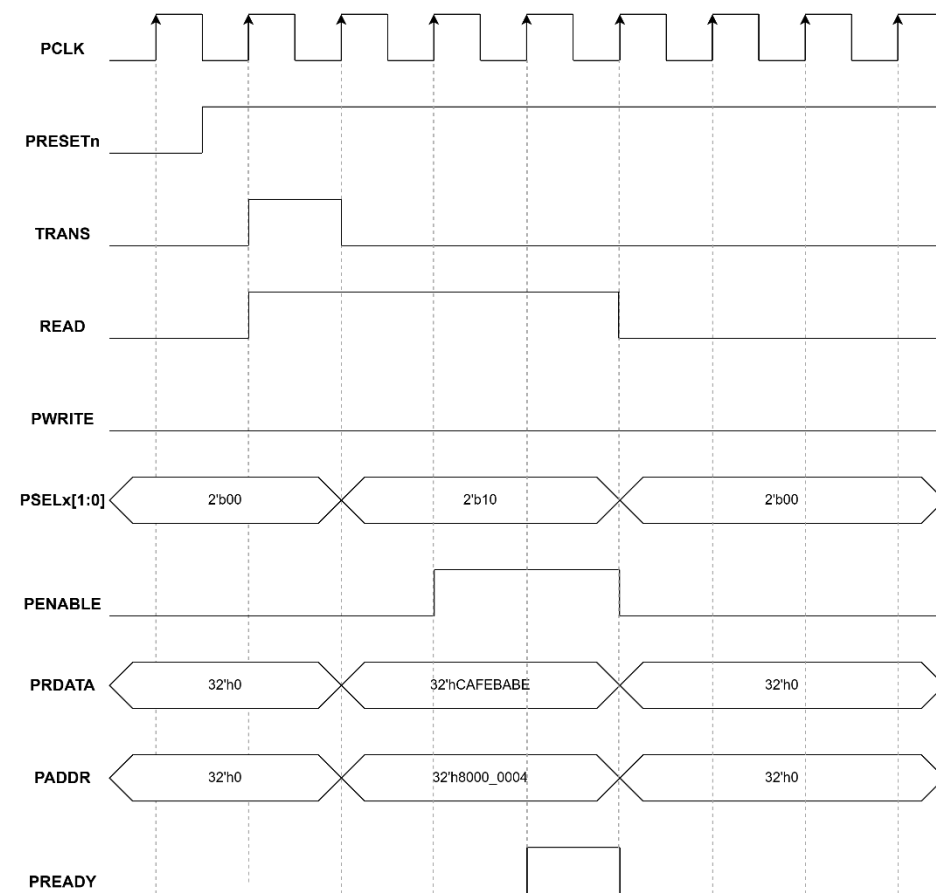
APB Read transfer without waiting state

2. High Level Block Design

2.4 AMBA ABP Interface signals



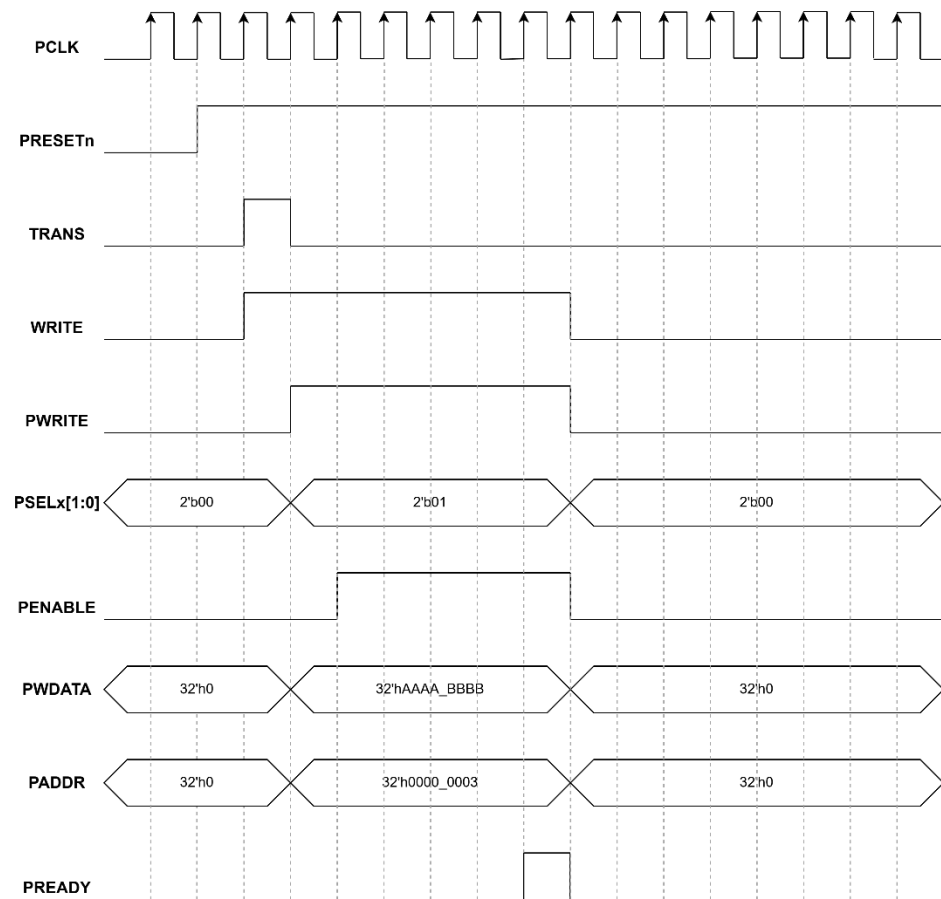
APB Write with Slave 1 access



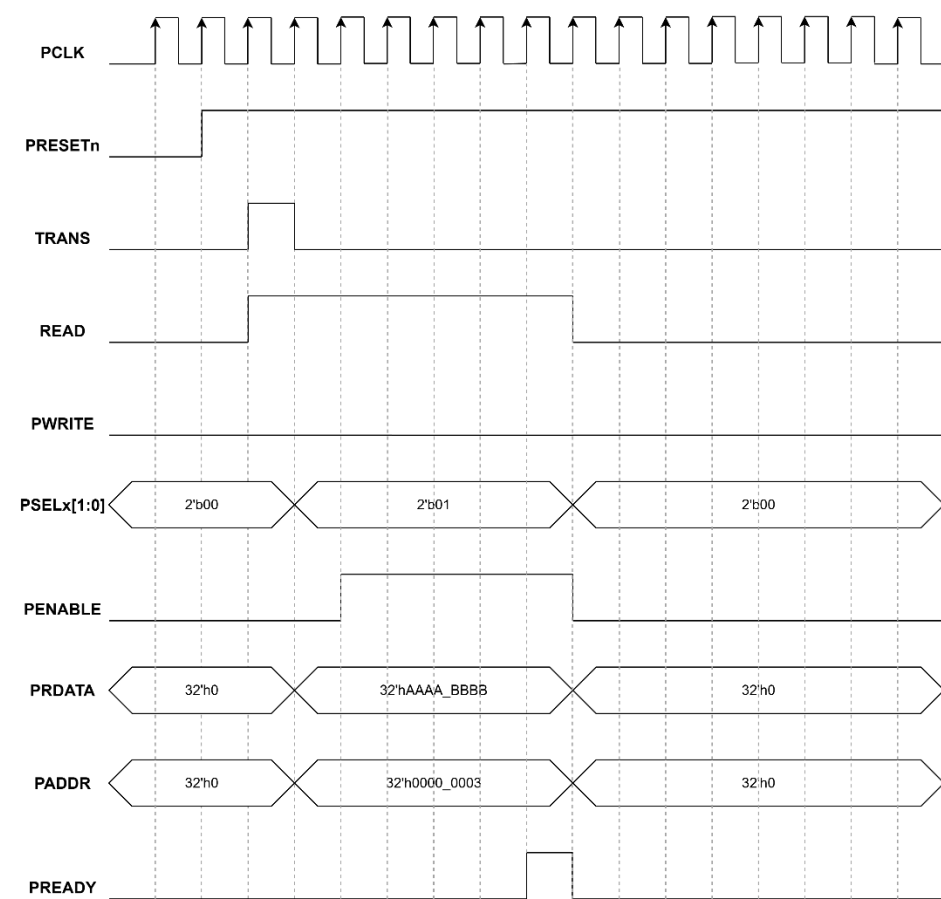
APB Read with Slave 1 access

2. High Level Block Design

2.4 AMBA ABP Interface signals



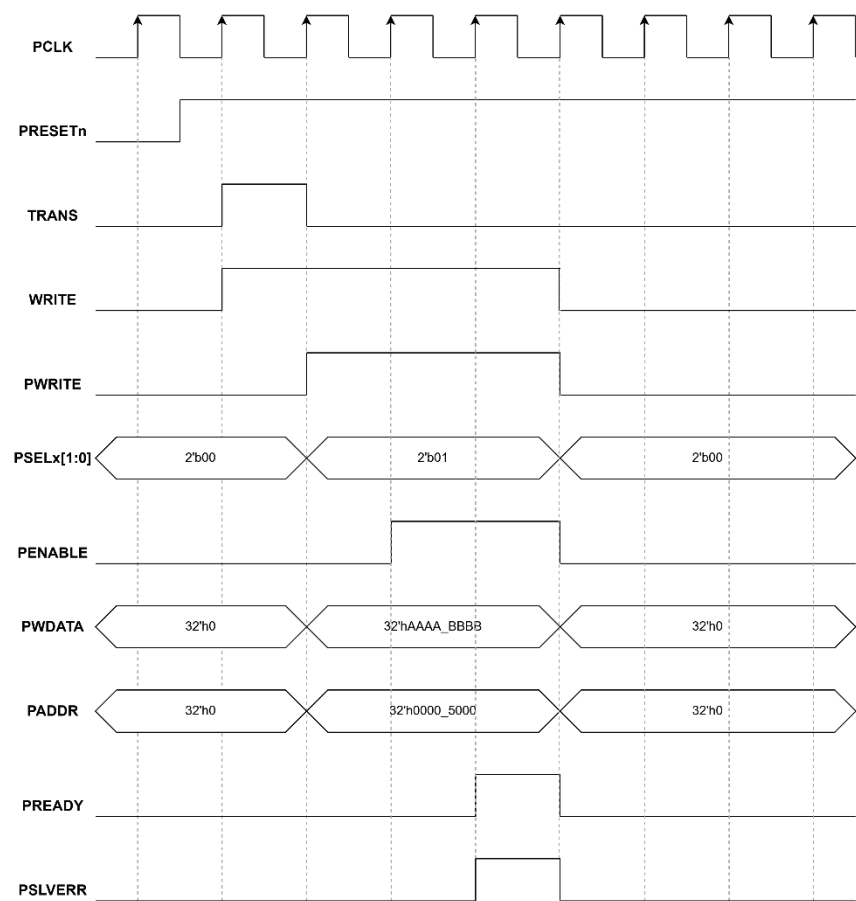
APB Write transfer with waiting state



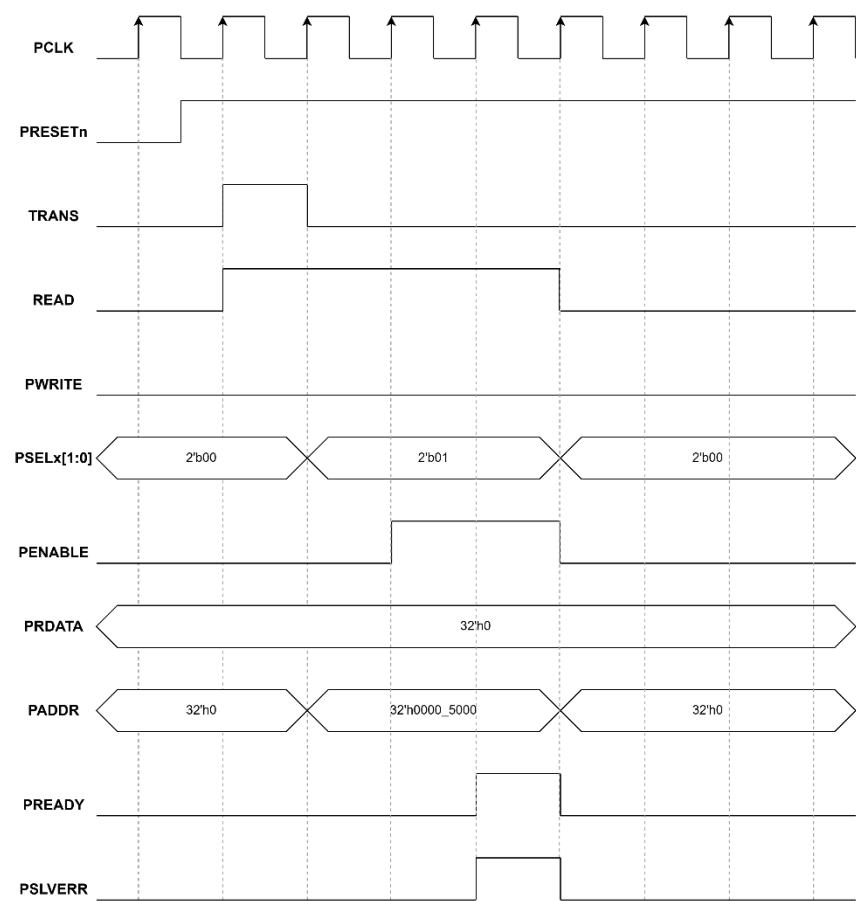
APB Read transfer with waiting state

2. High Level Block Design

2.4 AMBA ABP Interface signals



APB Write with address range error



APB Read with address range error