Advanced Communication Devices Corp

ADVANCE INFORMATION

Data Sheet: ACD82224

ACD82224

24 Ports 10/100 Fast Ethernet Switch

Last Update: September 19, 2000

Please check ACD' s website for update

information before starting a design

Web site: http://www.acdcorp.com

or Contact ACD at:

Email: support@acdcorp.com

Tel: 510-354-6810

Fax:510-354-6834

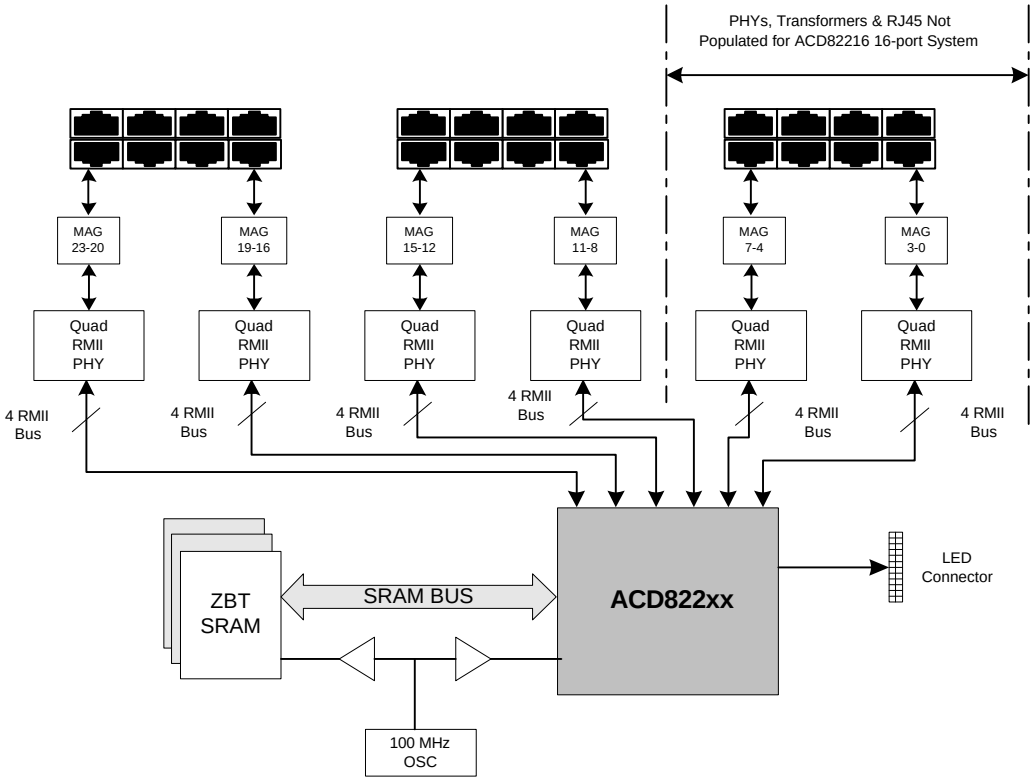
CONTENT LIST

1. GENERAL DESCRIPTION

The ACD82224 is a single chip implementation of 24-port 10/100 Ethernet switch system intended for IEEE 802.3 and 802.3u compatible networks. The device includes 24 independent 10/100 MACs. Each MAC interfaces with an external PMD/PHY device through a Reduced MII (RMII) interface. The last port is RMII and MII selectable. When in MII mode, this port becomes a shared port with the in-band management CPU. Link, Speed, and Duplex can be automatically configured through the MDIO port. Each port can operate at either 10Mbps or 100Mbps. The core logic of the ACD82224, implemented with patent pending BASIQ (Bandwidth Assured Switching with Intelligent Queuing) technology, can simultaneously process 24 asynchronous 10/100Mbps port traffic. The Queue Manager inside the ACD82224 provides the capability of routing traffic with the same order of sequence, without any packet loss.

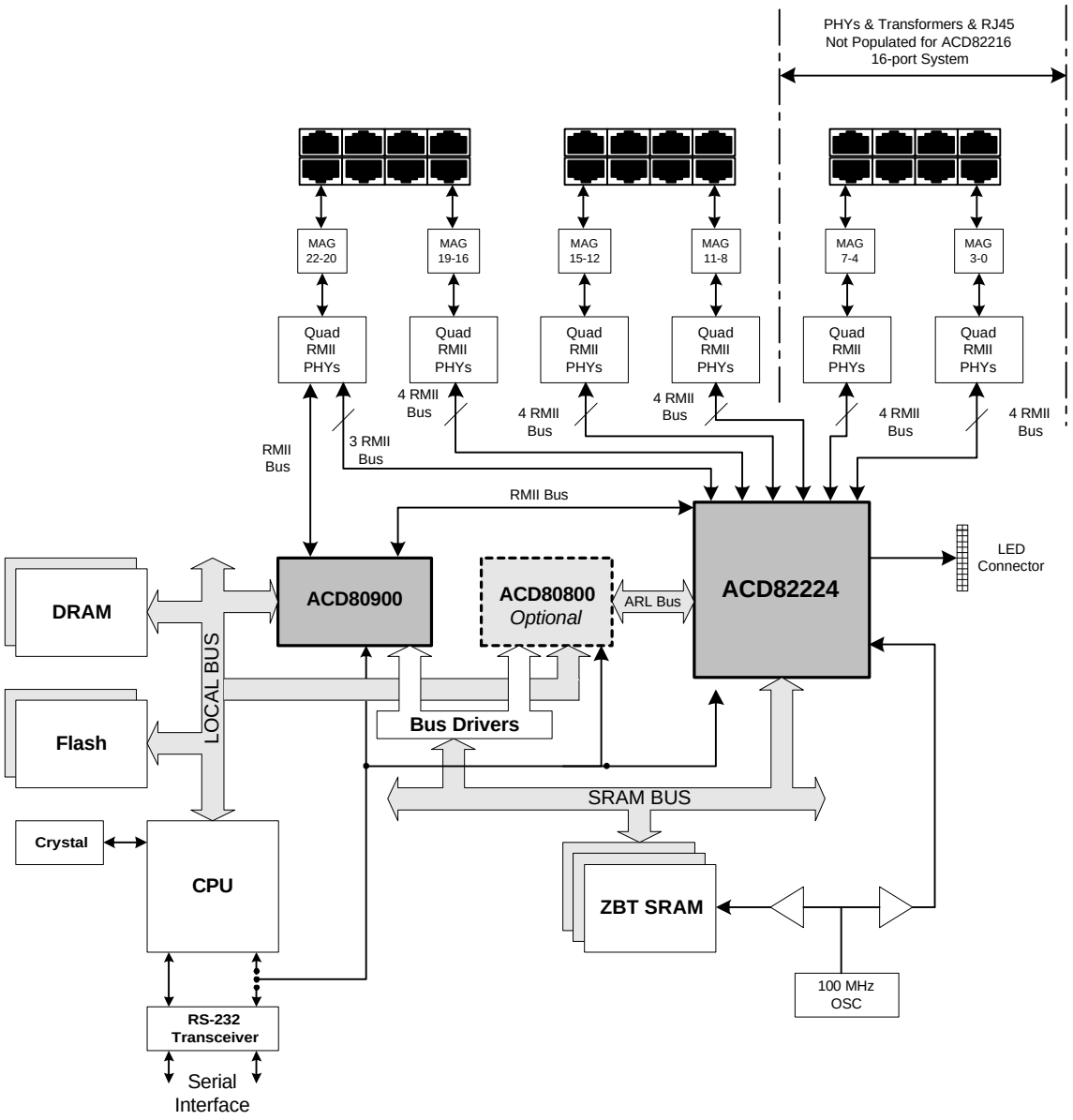
A complete 24-port 10/100 switch can be built with the addition of 10/100 RMII PHY and SRAM (ZBT[TM] [1]or compatible). An additional 11K MAC addresses can be supported with the use of ACD's Address Resolution Logic (ARL) chip, the ACD80800. Advanced network management features can be supported with the use of ACD's Management Information Base (MIB) chip, the ACD80900. The single universal 388-pin PBGA package for all 3 controllers makes One-PCB-For-ALL three systems very easy to implement, which significantly reduce the cost and time associated with multiple system product development.

Figure-1.1: ACD82224 Based 24 Ports Single Chip Un-managed 10/100 Switch System



[1] ZBT is the trade mark of IDT.

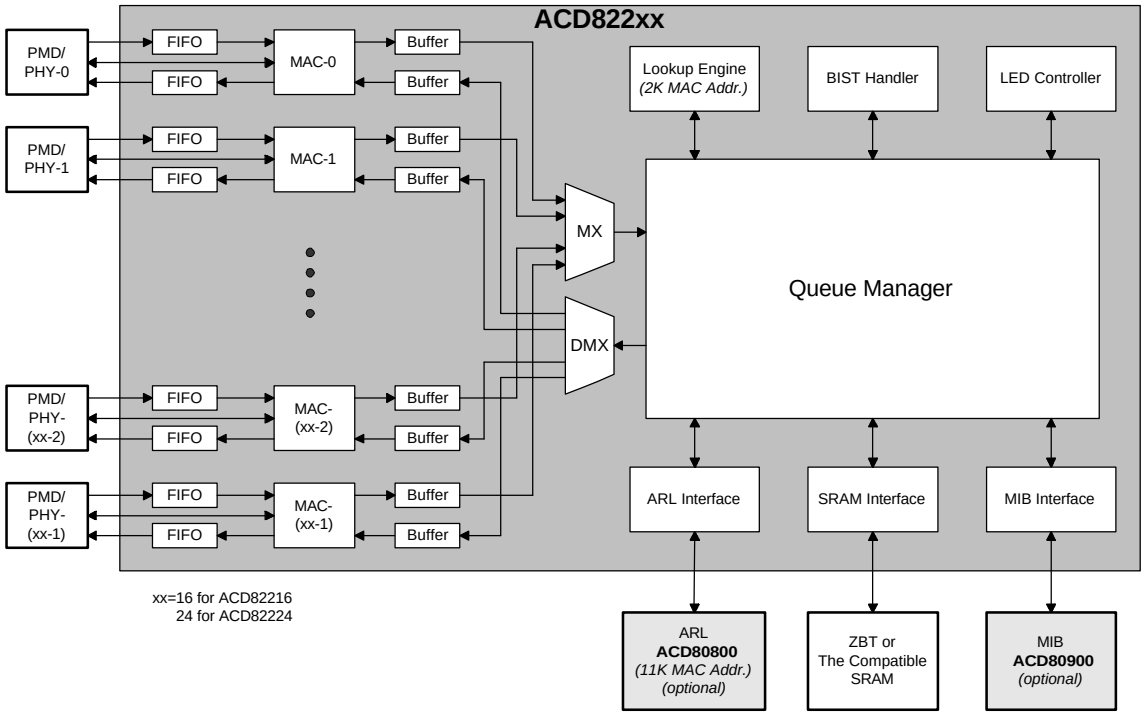
Figure-1.2: ACD82224 Based 24 Ports 3-Chip Managed 10/100 Switch System



2. MAJOR FEATURES

- 24 ports 10/100 Fast Ethernet Switch (auto-sensing or manual selection)
- Reduced MII interface, with selectable MII for the last port
- Capable of Trunking for up to 800 Mbps link
- Full & half duplex operation
- Speed auto negotiation through MDIO
- 4.8 Gbps aggregated throughput, true non-blocking switch architecture, full wire speed forwarding
- Built-in storage of 2,048 MAC address
- Supports up to 11K MAC addresses with the ACD80800
- Shared frame buffer with starvation control
- Memory interface with ZBT^{TM} or compatible SRAM at 100MHz
- Automatic source address learning
- Zero-Packet Loss back-pressure flow control under half duplex mode
- 802.3x pause frame flow control under full duplex mode
- Store-and-forward switch mode
- Port based V-LAN support for up to 4 VLANs
- UART type CPU management interface
- RMON and SNMP support with ACD80900
- Status LEDs: Link, Speed, Full Duplex, Transmit, Receive, Collision, and Frame Error
- 388-pin PBGA package
- Power: core 2.5V, I/O 3.3V with 5V tolerance

3. SYSTEM BLOCK DIAGRAM



4. SYSTEM DESCRIPTION

The ACD82224 is a single chip implementation of a 24-port Fast Ethernet switch. Together with external SRAM devices and transceiver devices, it can be used to build a complete 10/100 Mbps Fast Ethernet switch. Each port can be either auto-sensing or manually selected to run at 10 Mbps or 100 Mbps speed rates and under Full or Half-duplex mode.

There are four (4) major functional blocks inside the ACD82224:
(a) The Media Access Controller (MAC)
(b) The Queue Manager
(c) The Lookup Engine
(d) The Register file
(e) The MIB Engine interface

There are five (5) types of interfaces:
(a) RMII interfaces
(b) RMII/MII selectable interface
(c) Memory interface
(d) External-ARL interface
(e) External-MIB interface

MACs, RMII & MII Interfaces

There are 24 independent MACs within the ACD82224. The MAC controls the receiving, transmitting, and deferring process of each individual port, in accordance to the IEEE 802.3 and 802.3u standards. The MAC logic also provides framing, FCS checking, error handling, status indication and flow control functions (backpressure & pause-frame). Each MAC interfaces with an external transceiver through a RMII (Reduced MII) interface. The last MAC has a selectable RMII/MII interface. The MII mode allows direct connection with the ACD80900 (MIB), which also acts as a three-port switch for the management CPU to share the regular switch port for in-band management.

Queue Manager

The device utilizes ACD's proprietary BASIQ (Bandwidth Assured Switching with Intelligent Queuing) technology. It efficiently enforces the first-in-first-out rule of Ethernet Bridge-type devices. It also enables a true non-blocking frame switching operation at wire speeds for high throughput and high port density Ethernet switch design.

Built-in ARL & External-ARL Interface

The on-chip Lookup Engine implements a 2,048 entries MAC address lookup table. It maps each destination address with a corresponding port ID. Each MAC address is automatically learned by the LOOKUP ENGINE after an error-free frame is received. The address entries can also be managed for aging, locking, and forced filtering. Through the serial CPU interface of the ACD82224 switch, a management CPU can learn the address change in the lookup table. Hence, the ACD82224 alone can be used to build a complete Fast Ethernet switch with up to 2,048 host connections. *(See Appendix-A for detail)*

For workgroup or backbone switches, the ACD82224 can support more MAC addresses per port through the use of an external ARL chip, the ACD80800. The ACD82224 has a glueless ARL interface that allows a supporting chip (ACD80800) to provide up to 11K MAC addresses per

switch. System designers can also use this ARL interface to implement a vendor-specific address resolution algorithm.

Register & UART Interface

A System CPU can access various registers inside the ACD82224 through a serial CPU management interface (UART). The CPU can configure the switch by writing into the appropriate registers, or retrieve the status of the switch by reading the corresponding registers within the ACD82224 switch. The CPU can also access the registers of external transceiver (PHY) devices through the CPU management interface as well.

External-MIB & External-MIB Interface

The ACD82224 provides management support through the use of the ACD80900 (Management Information Base). The MIB interface can be used to monitor all traffic activities of the switch system. The supporting chip (the ACD80900) provides a full set of statistics counters to support both SNMP and RMON network management functions. In addition to the statistics counters, the ACD80900 also support all groups of RMON management, including Host, HostN, Matrix, Filtering and Capturing groups. System designers can also use the MIB interface to implement vendor-specific network management functionality.

5. FUNCTIONAL DESCRIPTION

The MAC controller performs transmitting, receiving, and deferring functions, in accordance to the 802.3 and 802.3u specification. The MAC logic also handles frame detection, frame generation, error detection, error handling, status indication and flow control functions. Under full-duplex mode, the flow control is implemented in compliance with IEEE 802.3x standard.

Frame Format

The ACD82224 assumes that the received data packet will have the following format:

Preamble	SFD	DA	SA	Type/Len	Data	FCS

Where,

- *Preamble* is a repetitive pattern of '1010….' of any length with nibble alignment.
- *SFD* (Start Frame Delimiter) is defined as an octet pattern of 10101011.
- *DA* (Destination Address) is a 48-bit field that specifies the MAC address of the destined DTE. For any frame with "1" in the first bit of the DA, with the exception of the BPDU address (the reserved group address described in table 3-5 of IEEE 802.1d), the ACD82224 will treat it as a broadcast/multicast frame. It will forward the frame to all ports within the source port's VLAN, except the source port itself.
- *SA* (Source Address) is a 48-bit field that contains the MAC address of the source DTE that is transmitting the frame to the ACD82224. After a frame is received with no error, the SA is learned as the port's MAC address.
- *Type/Len* field is a 2-byte field that specifies the type (DIX Ethernet frame) or length (IEEE 802.3 frame) of the frame. The ACD82224 does not process this information, unless it is a Pause-Frame
- *Data* is the encapsulated information within the Ethernet Packet. The ACD82224 does not process any of the data information in this field.
- *FCS* (Frame Check Sequence) is a 32-bit field of CRC (Cyclic Redundancy Check) value based on the destination address, the source address, the type/length and the data field. The ACD82224 will verify the FCS field for each frame. The procedure for computing FCS is described in the section "FCS Calculation."

Start of Frame Detection

When a port's MAC logic detects the assertion of the CRS_DV signal in the RMII interface, it will start a receiving process. The received data will come through a 2-bit wide data bus, clocked by the 50 MHz receiving-clock from the ACD82224. It will then pass a frame alignment circuit, which will convert the 2-bit signal into a single bit stream and detect the occurrence of the SFD pattern (10101011). All signals before the SFD are filtered out and the rest of the data frame will be stored into the frame buffer of the switch.

Frame Reception

Under normal operating conditions, the ACD82224 expects a received frame to have a minimum inter frame gap (IFG). The minimum IFG required by the ACD82224 is 64 BT.

If a packet comes with an IFG less than 64 BT, the ACD82224 will not guarantee the reception of that frame. The packet may be dropped if it is not properly received.

The ACD82224 will check all received frames for errors such as symbol error, FCS error, short event, runt, long event, jabber, etc. Frames with any kind of error will not be forwarded to any port.

Preamble Bit Processing

The preamble bit in the header of each frame will be used to synchronize the MAC logic with the incoming bit stream. There is no limit on the minimum length or the maximum length of preamble bits. After the receive-signal CRS_DV is asserted by the external PHY device, the MAC will wait for the SFD pattern (10101011) to trigger a frame receiving process.

Destination Address Processing

As a frame comes in, the embedded Destination Address (DA) is passed to the Address Resolution Logic (ARL). The ARL will compare it with the MAC address entries stored in the address lookup table. A destination port is identified if a match of address is found. If external ARL is used, the ACD82224 will indicate the present of 48-bit DA through the ARL interface. The external ARL will use the value of DA for address comparison and return a result to the ACD82224.

Source Address Processing

As a frame comes in, the embedded Source Address (SA) will be passed to the ARL. At the end of the frame, if no error is detected, the SA will be used to update the address lookup table. If an external ARL is used, the ACD82224 will indicate the presence of a SA on the ARL interface, so that the external ARL can learn the address. The address table will be cleared after a Hardware-Reset, but a Software-Reset will not clear the address table.

Frame Data

Frame data are transparent to the ACD82224. The ACD82224 will forward the data to destination port(s) without interpreting the content of the frame data field.

FCS Calculation

Each port of the ACD82224 has a CRC checking logic to verify if the received frame has the correct FCS value. An incorrect FCS value is an indication of a fragmented frame or a frame with frame bit error. The method of calculating the CRC value is by using the following polynomial,

$$G(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^{8} + x^{7} + x^{5} + x^{4} + x^{2} + x + 1$$

as a divider to divide the bit sequence of the incoming frame, beginning with the first bit of the destination address field, to the end of the data field. The result of the calculation, which is the

residue after the polynomial division, is the value of the frame check sequence. This value should be equal to the FCS field appended at the end of the frame. If the value does not match the FCS field of the frame, the Frame Bit Error LED of the port will be turned on once and the packet will be dropped.

Illegal Frame Length & Extra-long Frame

During the receiving process, the MAC will monitor the length of the received frame. Legal Ethernet frames should have a length of not less than 64 bytes and no more than 1518 bytes. If the carrier-sense signal (CRS_DV) of a frame is asserted for less than 84 BT, the frame is flagged with short event error. If the length of a frame is less than 512 BT, the frame is flagged with runt error.

In order to support an application where extra byte length is required, an Extra long frame option is provided. When the Extra long frame option is enabled *(bit-11 of Register 25)*, only frames longer than 1530 bytes are marked with a long event error. Frame length is measured from the first byte of DA to the last byte of FCS.

Frame Filtering

Frames with any kind of error will be filtered. Error types include CRC, alignment, false carrier sense, short event, runt, long event and jabber. An error frame will still be displayed on the MIB interface, along with the error status indication.

Any frame heading to its own source port will be filtered.

When external ARL is used, the filtering decision will be made by the ARL. The ACD82224 will act in accordance with the ARL's decision.

If the Spanning Tree Support option (Bit 1 of Register 16) is set, a frame with DA equal to 01-80-C2-00-00-00 will be forwarded to port-23 (the default CPU port) as a BPDU frame. If Spanning Tree Support is not enabled, a frame with reserved group address specified in Table-3.5 of the IEEE802.1d will be treated as a broadcast frame and will be forwarded to all the ports in the same VLAN of the source port.

Jabber Lockup Protection

If a receiving port is active continuously for more than 50,000 bit times, the port is considered to be jabbering. A jabbering port will automatically be partitioned from the switch system in order to prevent it from impairing the performance of the network. The partitioned port will be re-activated as soon as the offending signal discontinues.

Excessive Collision

In the event that there are more than 16 consecutive collisions, the ACD82224 will reset the counter to zero and re-transmit the packet. This implementation insures there is no packet loss even under channel capture situation. However, the ACD82224 has an option to drop the packet on excessive collisions. When this option is enabled *(bit-15 of Register 25)*, the frame will be dropped after 16 consecutive collisions.

Frame Forwarding

If the first bit (bit-40) of the destination address is 0, the frame is handled as a unicast frame. The destination address is passed to the Address Resolution Logic; which returns a destination port number to identify which port the frame should be forwarded to. If the Address Resolution Logic cannot find any match for the destination address, the frame will be treated as a frame with unknown DA. The frame will be processed in one of two ways upon *bit-12of Register 25*.

1. If the option flood-to-all-port is set, the switch will forward the frame to all ports within the same VLAN of the source port, except the source port itself.

2. If the option is not set, the frame will be forwarded to the 'dumping port' of the source port VLAN only. The dumping port is determined by the VLAN ID of the source port. If the source port belongs to multiple VLANs, a frame with unknown DA will then be forwarded to multiple dumping ports of the VLANs.

If the first bit of the destination address is a 1, the frame is handled as a multicast or broadcast frame. The ACD82224 does not differentiate a multicast packet from a broadcast packet except for the reserved bridge management group address, as specified in Table-3.5 of IEEE 802.1d standard. The destination ports of a broadcast frame are all ports within the same VLAN except the source port itself.

The order of all broadcast frames with respect to the unicast frames is strictly enforced by the ACD82224.

Frame Transmission

The ACD82224 transmits all frames in accordance to IEEE 802.3 standard. The ACD82224 will send the frames with a guaranteed minimum inter frame gap of 96 BT, even if the received frames have an IFG less than the minimum requirement. Before the transmitting process is started, the MAC logic will check if the channel has been silent for more than 64 BT. Within the 64 BT silent window, the transmission process will defer on any receiving process. If the channel has been silent for more than 64 BT, the MAC will wait an additional 32 BT before starting the transmitting process. In the event that the carrier sense signal is asserted by the RMII during the wait period, the MAC logic will generate a JAM signal to cause a forced collision.

The MAC logic will abort the transmitting process if a collision is detected. Re-transmission of the frame is scheduled in accordance to the IEEE 802.3's truncated binary exponential back-off algorithm. If the transmitting process has encountered 16 consecutive collisions, an excessive collision error is reported, and the ACD82224 will try to re-transmit the frame, unless the drop-on-excessive-collision option of the port is enabled. It will first reset the number of collisions to zero and then start the transmission after a 96 BT of inter frame gap. If drop-on-excessive-collision is enabled, the ACD82224 will not try to re-transmit the frame after 16 consecutive collisions. If collision is detected after 512 BT of the transmission, a late collision error will be reported and the frame may or may not be retransmitted.

Shared Buffer

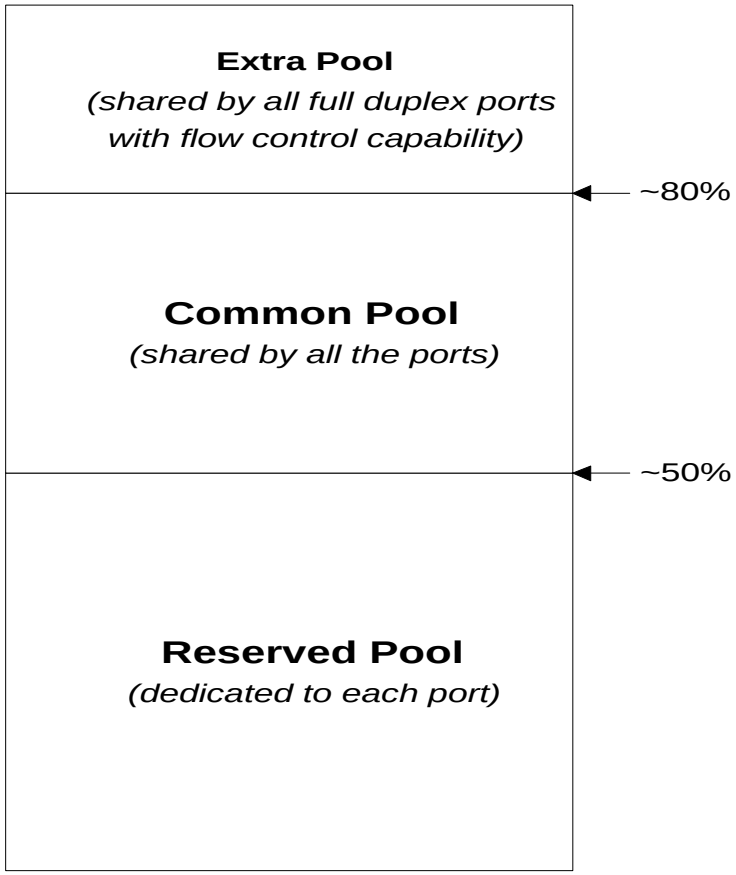
All ports of the ACD82224 work in Store-And-Forward mode so that all ports can support both 10Mbps and 100Mbps data speeds. The ACD82224 utilizes a global memory buffer pool, which is shared by all ports. The device has a unique architecture that inherits the advantages of both output buffer-based and input buffer-based switches: short latency of an output-buffer based switch which only store the received data once into the memory and efficient flow control of an input-buffer based switch.

Starvation Control Scheme

All frames received by the ACD82224 will be stored into a common physical frame buffer pool. In order to make sure all ports have fair access to the network, a buffer allocation scheme (starvation control) is used to prevent active ports from occupying all the buffers and starving off the less active ports.

The frame buffer pool is divided into 3 portions: the *reserved poo*l, the *common pool* and the *extra pool*, as shown in *Figure-5.1:*

Figure-5.1: Buffer Partition



The *reserved pool* guarantees each port will have a fair network access possibility, even under the worst traffic congestion situation. It takes about 50% of the total buffer and is evenly allocated to each port as its dedicated buffer slot. The dedicated slot is not shared with other ports.

The *common pool* provides a deep buffer for the busy ports (e.g. server port) to serve multiple low speed ports (e.g. client port) simultaneously. It helps to avoid head-of-line blocking. It takes about 30% of the total buffer and is shared by all ports. It stores the congested traffics before the flow control mechanism is triggered.

The *extra pool* is reserved only for ports with pause frame based flow control capacity. It takes the remaining 20% of the total buffer. It is used to minimize the chance of frame dropping by buffering for the latency of the pause frame based flow control scheme. It is used only after a flow control mechanism is triggered.

Flow Control Scheme

Flow control activity is triggered when the buffer utilization exceeds certain thresholds specified by the dedicated registers. *Register-10* is used to specify the Upper and the Lower Thresholds of the reserved buffer slot for each port. *Register-11* is used to specify the Upper and the Lower thresholds of the broadcast queue.

For full duplex with pause frame capability operation:

(1) If the buffer utilization of the reserved buffer slot for the source port has exceeded the "Upper Threshold", and the common pool has been used up.

(2) Then a max-pause-frame (a pause frame with a maximum time interval of FFFFh) will be sent to the sending port to stop it from sending any new frames. If pause-frame based flow control is not enabled at that port, the frame will be dropped.

(3) Once a Max-Pause-Frame is sent, if the utilization of the reserved buffer slot of the port drops below the lower threshold, a Mini-Pause-Frame (a pause frame with minimum time interval of 0) will be sent to the sending port to enable new frame transmission. But if the utilization of the reserved buffer slot of the port does not drop below the lower threshold for the maximum time interval, ACD82224 will not send another Max-Pause-Frame to the sending port to prevent the Buffer Capturing by other ports.

For half duplex operation:

(1) If the buffer utilization of a port has exceeded the upper threshold of the reserved buffer slot, and the common pool has been used up.

(2) The port will execute backpressure based flow control by sending a jam pattern on each incoming frame. If backpressure flow control of the port is not enabled, the frame will be dropped.

If the broadcast flow control is enabled (when bit-17 of register-25 is cleared), flow control will be triggered when the broadcast queue is larger than the "Upper Threshold" in *Register-11*. All full duplex ports with pause-frame capability will send a *Max-Pause-Frame* to its linking party. All half-duplex ports with backpressure capability will jam incoming frames. After *a Max-Pause-Frame* is sent, and if the broadcast queue is below the "Lower Threshold" in *Register-11*, a *Mini-Pause-Frame* will be sent to release the hold on transmission.

Port-based VLAN Support *(Registers 23 & 24)*

The ACD82224 can support up to 4 port-based security VLANs. Each port of the ACD82224 can be assigned to up to four VLAN. On power up, every port is assigned to VLAN-I as the default

VLAN. Frames from the source port will only be forwarded to destination ports within the same VLAN domain. A broadcast/multicast frame will be forwarded to all ports within the VLAN(s) of the source port. A unicast frame will be forwarded to the destination port only if the destination port is in the same VLAN as the source port. Otherwise, the frame will be treated as a frame with unknown DA. Each VLAN can be assigned with a dedicated dumping port. Multiple VLANs can also share a dumping port. Unicast frames with unknown destination addresses will be forwarded to the dumping port of the source port VLAN.

A Leaky VLAN can be enabled by setting the corresponding bit in the system configuration register (bit-8 of register-16). A VLAN becomes a broadcast domain. Each broadcast domain VLAN can still be assigned with a dedicated dumping port.

Port Trunking

Security VLAN can be disabled by setting the corresponding bit in the system configuration register (bit 8 of *Register 16, see Table 7.15*). When security VLAN is disabled, each VLAN becomes a Leaky VLAN and is equivalent to a broadcast domain. Four dumping ports of four different Leaky VLANs can be grouped together to form a fat pipe uplink (for example, port 0, port 1, port 2, and port 3 can be grouped to form an 800 Mbps uplink port). When multiple dumping ports are grouped as a single pipe, each port has to be assigned to one and only one VLAN. A unicast frame with a matched DA will be forwarded to any destination port, even if the VLAN ID is different. All unmatched DA packets will be forwarded to the designated dumping port of the source port VLAN. The broadcast and multicast packets will only be forwarded to the ports in the same VLAN of the source port. Therefore, a 200 to 800 Mbps pipe can be established by carefully grouping the dumping ports, and directly connecting with any segmentation switches.

Dumping Port

Each VLAN can be assigned with a dedicated dumping port. Multiple VLANs can share a single dumping port. Each dumping port can be used for an up-link connection or for a DTE connection. That is, the dumping port can be used to connect the switch with a computer repeater hub, a workgroup switch, a router, or any type of interconnection device compliant with IEEE 802.3 standard. ACD82224 will direct the following frames to the dumping port:

(1) A frame with unknown unicast destination address.
(2) A frame with a unicast destination address that does not match with any port's source address within the VLAN of the source port.
(3) A frame with a broadcast/multicast destination address (See Spanning Tree Support).

If the device is configured to work under Flood-to-All-Port mode *(Register 25, bit 12)*, the frames with unknown DA will be forwarded to all the ports in the VLAN(s) of the source port except the source port itself.

Mode of Operation

Each port can be configured or auto-sensed to work under either half-duplex or full-duplex mode. Under half duplex mode, the CSMA/CD will be enforced, and the flow control is achieved through backpressure. Under full duplex mode, the receiving process and the transmitting process of the port are independent, and the flow control is done by pause-frame based flow control scheme specified by the IEEE 802.3x.

Spanning Tree Support

The ACD82224 supports the Spanning Tree protocol. When Spanning Tree Support is enabled *(Register-16 bit-1, see Table 7.15)*, frames from the CPU port (port-23) having a DA value equal to the reserved Bridge Management Group Address for BPDU will be forwarded to the port specified by the CPU. Frames from all other ports with a DA value equal to the Reserved Group Address for BPDU will be forwarded to the CPU port if the port is in the same VLAN of the CPU port. Port 23 is designed as the default CPU port. When Spanning Tree Support is disabled *(Register-22 nPORT Register)*, all reserved group addresses for Bridge Management is treated as broadcast addresses, with the exception of the reserved multicast addresses for pause frame specified by IEEE802.3x.

Every port of the ACD82224 can be set to block-and-listen mode *(Register-21 nBP Register)* through the CPU interface. In this mode, incoming frames with a DA value equal to the reserved Group Address for BPDU will be forwarded to the CPU port. Incoming frames with all other DA values will be dropped. Outgoing frames with a DA value equal to the Group Address for BPDU will be forwarded to the attached PHY device; all other outgoing frames will be filtered.

Queue Management

Each port of the ACD82224 has its own individual transmission queue. All frames coming into the ACD82224 are stored into the shared memory buffer, and are lined up in the transmission queues of the corresponding destination port. The order of all frames, unicast or broadcast, is strictly enforced by the ACD82224. The ACD82224 is designed with a non-blocking switching architecture. It is capable of achieving wire speed forwarding rates and can handle maximum traffic loads.

PHY Management

The ACD82224 supports PHY device management through the serial MDIO and MDC signal lines. The ACD82224 can continuously poll the status of the PHY devices through the serial management interface if Register-25 bit-16 is cleared. The ACD82224 will also configure the PHY capability field like Link, Speed, and Duplex status to ensure proper operation of the link. The ACD82224 also enables the CPU to access any registers in the PHY devices through the CPU interface (See Register-32 example). The ID of the PHY device can start from either “0” or “1”, depending on the setting of bit-14 of register-25.

There are two ways to disable Automatic PHY Management as follows:

(1) Set Register-25 bit-16 to disable Automatic PHY Management for all ports.

(2) Set the specific port in Register-29 to disable the port from Automatic PHY Management.

PHY Interface

The PHY interface for port-0 through port-22 can only be RMII. The RMII CLK (50MHz clock) will be used as the RXCLK and TXCLK. There are three wires on the receiving side (RXD[1:0] and CRS_DV), and three wires on the transmitting side (TXD[1:0] and TXEN). Port-23 can be configured as either RMII or MII *(Register-16 bit-15).* The MII option is used for direct connection with the ACD80900 only supporting MII interface.

ACD82224 supports PHY management through the MDIO and MDC signal lines. ACD82224 can continuously poll the status of the PHY devices through the serial management interface, without

using a CPU. ACD82224 also allows the CPU to access any registers in the PHY devices through the CPU interface.

SRAM Interface

The ACD82224 uses pipeline ZBT™ (zero-bus- turn-around) or compatible types of SRAM. The speed should be 100MHz or faster. Each read or write cycle should take no more than 10 ns. The SRAM interface contains a 52-bit data bus (48-bit data and 4-bit status), a 19-bit address bus, and 2 control signals.

CPU Interface

The ACD82224 does not require any microprocessor for operation. Initialization and most configurations can be done with pull-up or pull-down of designated hardware pins. A CPU interface is provided for a microprocessor to access the internal control registers and status registers. The microprocessor can send a read command to retrieve the status of the switch, or send a write command to configure the switch through the interface. The interface is a commonly used UART type interface. The CPU interface can also be used to access the registers inside each PHY device connected to the ACD82224.

ARL Interface

The ACD82224 has a built-in MAC address storage for up to 2,048 source addresses. If more than 2,048 addresses are needed, an external ARL (e.g. ACD80800) can be used to expand the address space to 11K entries.

The external ARL is connected through the ARL interface *(Table-6.9)*. It can tap the value if DA out of the memory interface bus, and execute a lookup process to map the value of the DA into a port number. It can also learn the SA values embedded in the received frames. The value of SA is used to build the address lookup table inside the ACD80800. If the new addressed are over the maximum number of look-up table, ARL will not learn the newer address until there is any available entry again (i.e. the learned address aged).

MIB Interface

Traffic activities on all ports of the ACD82224 can be monitored through the MIB interface. Through the MIB interface, a MIB device can view the frames transmitted from or received by any port. Therefore, the MIB device can maintain a record of traffic statistics for each port to support network management. Since all received data are stored into the memory buffer, and all transmitted data are retrieved from the memory buffer, the data of the activities can also be captured from the memory interface data bus. The status of each data transaction between the ACD82224 and SRAM is displayed by dedicated status signals of the ACD82224 interface *(Table-6.9)*.

LED Interface

The ACD82224 provides a wide variety of LED indicators for simple system management. The update of the LED is completely autonomous and merely requires low speed TTL or CMOS

devices as LED drivers. The status display is designed to be flexible to allow the system designer to choose those indicators appropriate for the specification of the end equipment.

There are two LED control signals, LEDVLD0 and LEDVLD1. They are used to indicate the start and end of the LED data signal presented on nLED0-nLED3. The LEDCLK signal is a ***2.5 MHz*** clock signal. The rising edge of LEDCLK should be used to latch the LED data signal into the LED driver circuitry.

The LED data signals contain Lnk, Xmt, Rcv, Col, Err, Fdx and Spd, which represent Link status, Transmit status, Receive status, Collision indication, Frame error indication, Full duplex operation and Operational Speed status respectively. These status signals are sent out sequentially from port 23 to port 0, once every ***50 ms***. For details about the timing diagrams of the LED signals, refer to the chapter of “Timing Description ”

Life Pulse

The ACD82224 will generate Life Pulses at WCHDOG pin once every 800 nsec to indicate normal operation status. Absence of the Life Pulses is an indication that, the ACD82224 has encountered some fatal error and needs to be reset. The life pulses are used to reset a watchdog counter, such that, if the watchdog counter is not reset and has reached a predetermined value, a system reset signal can be generated to reset the ACD82224.

6. INTERFACE DESCRIPTION

Figure-6.1 shows all the interfaces of the ACD82224 controller and their relations to other modules in a typical switch system. Table-6.1 list all function pins grouped by types.

Figure-6.1 Major Interface Group

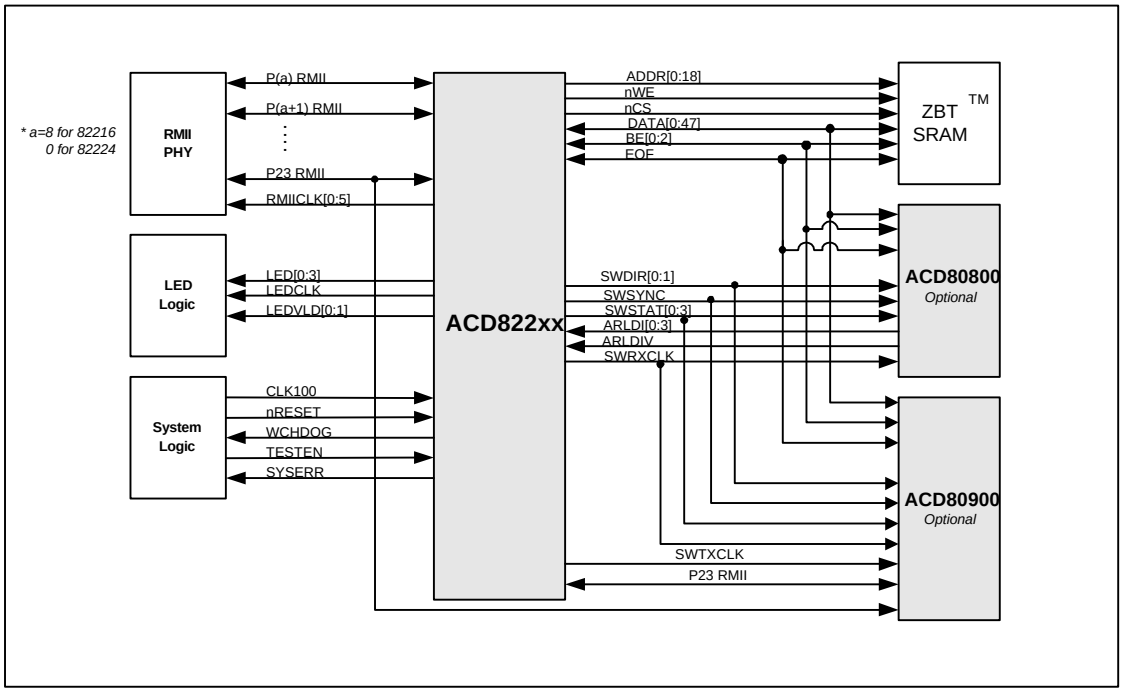


Table-6.1 Interface Group

Interface	Pin Name	82224
RMII Port[0:22]	PxCRS_DV, PxRXD0, PxRXD1, PxTXEN, PxTXD0, and PxTXD1	138
MII Port-23	P23CRS*, P23RXDV*, P23RXCLK, P23RXERR, P23RXD0*, P23RXD1*, P23RXD2, P23RXD3, P23COL,P23TXEN, P23TXD0*, P23TXD1*, P23TXD2, P23TXD3, and P23TXCLK (*: shared with RMII signals in Port23)	15
RMII Clock	RMIICLK[5:0]	6
PHY Management	MDC, and MDIO	2
Memory Address	ADDR[0:18]	19
Memory Data	DATA[47:0], BE[2:0], and EOF	52
Memory Control	nWE, and nCS	2
ARL and MIB	SWDIR[1:0], SWPID[4:0], SWSYNC, SWSTAT[4:0], SWRXCLK, SWTXCLK, ARLDI[3:0], and ARLDIV	19
LED	LEDVLD[1:0], LEDCLK, and nLED[3:0]	7
CPU	CPUDI, CPUDO, and CPUIRQ	3
System Control	CLK100, nRESET, WCHDOG, and SYSERROR	4
Factory Test	EN16P, CLKSEL, TESTEN, and ZBTCLK	4
2.5V	VDD (for Core)	17
3.3V	VDDQ (for I/O)	33
Ground	VSS (including 36 center thermal ground)	67
Total Number of pins		**388**

RMII Interface (RMII)

The ACD82224 communicates with the external 10/100 Ethernet transceivers through standard RMII interface. The signals of RMII interface are described in *Table-6.2a*:

Table-6.2a: RMII Interface

Name	Type	Description
PxCRSDV	I	Carrier Sense/Receive data valid
PxRXD0	I	Receive data bit 0
PxRXD1	I	Receive data bit 1
PxTXEN	O	Transmit enable
PxTXD0	O	Transmit data bit 0
PxTXD1	O	Transmit data bit 1
RMIICLK[5:0]	O	Reduced MII clock (50 MHz)

For RMII interface, signal PxRXDV, PxRXD0, and PxRXD1 are sampled by the rising edge of RMIICLK. Signal PxTXEN, PxTXD0, and PxTXD1 are clocked out by the falling edge of RMIICLK. The detailed timing requirement is described in the chapter of "Timing Description"

MII Interface (MII)

The last port (e.g. Port-23 on the ACD82224) can be selected to act in MII mode. The MII mode is used to connect the ACD80900 for in-band management function. The ACD80900 acts as a three-way switch to allow the management CPU to share the regular port. The signals of MII interface are described in *Table-6.2b*:

Table-6.2b: MII Interface

MII Mode	RMII Mode	Type	Description
P23CRS	P23CRSDV	I	Carrier sense
P23RXDV	NC	I	Receive data valid
P23RXCLK	NC	I	Receive clock (25/2.5 MHz)
P23RXERR	NC	I	Receive error
P23RXD0	P23RXD0	I	Receive data bit 0
P23RXD1	P23RXD1	I	Receive data bit 1
P23RXD2	NC	I	Receive data bit 2
P23RXD3	NC	I	Receive data bit 3
P23COL	NC	I	Collision indication
P23TXEN	P23TXEN	O	Transmit data valid
P23TXCLK	NC	I	Transmit clock (25/2.5 MHz)
P23TXD0	P23TXD0	O	Transmit data bit 0
P23TXD1	P23TXD1	O	Transmit data bit 1
P23TXD2	NC	O	Transmit data bit 2
P23TXD3	NC	O	Transmit data bit 3

Under the MII mode, signal P23RXDV, P23RXER and P23RXD0 through P23RXD3 are sampled by the rising edge of P23RXCLK. Signal P23TXEN, and P23TXD0 through P23TXD3 are clocked out by the falling edge of P23TXCLK. The detailed timing requirement is described in the chapter of "Timing Description"

PHY Management Interface

All control and status registers of the PHY devices are accessible through the PHY management interface. The interface consists of two signals: MDC and MDIO, which are described in *table-6.3*.

Table-6.3: PHY Management Interface Signals

Name	Type	Description
MDC	O	PHY management clock (**1.25MHz**)
MDIO	I/O	PHY management data

Frames transmitted on MDIO has the following format (*Table-6.4*):

Table-6.4: MDIO Format

Operation	PRE	ST	OP	PHY-ID	REG-AD	TA	DATA	IDLE
Write	1…1	01	01	A[4:0]	R[4:0]	10	D[15:0]	Z
Read	1…1	01	10	A[4:0]	R[4:0]	Z0	D[15:0]	Z

Prior to any transaction, the ACD82224 will output thirty-two bits of '1' as preamble signal. After the preamble, a 01 signal is used to indicate the start of the frame.

For a write operation, the device will send a '01' to signal a write operation. Following the '01' write signal will be the 5 bit ID address of the PHY device and the 5 bit register address. A '10' turn around signal is then follows the "write" signal. After the turn around, the 16 bit of data will be written into the register. After the completion of the write transaction, the line will be left in a high impedance state.

For a read operation, the ACD82224 will output a '10' to indicate read operation after the start of frame indicator. Following the '10' read signal will be the 5-bit ID address of the PHY device and the 5-bit register address. Then, the ACD82224 will cease driving the MDIO line, and wait for one bit time. During this time, the MDIO should be in a high impedance state. The ACD82224 will then synchronize with the next bit of '0" driven by the PHY device, and continue on to read 16 bits of data from the PHY device.

The system designer can set the ID of the PHY devices as 0 for port-0, 1 for port-1, … and 23 for port-23, when the PHYID option (Bit-14 of Register-25) is set to "0". If the PHYID option is set to "1", the corresponding PHY ID should set to 1 through 24. The detailed timing requirements on PHY management signals are described in the chapter of "Timing Description."

CPU Interface

The ACD82224 includes a CPU UART interface to enable an external CPU to access the internal registers of the ACD82224. The baud rate of the UART can be from 19200 to 38400 bps. The ACD82224 automatically detects the baud rate for each command, and returns the result at the same baud rate. The signals in the CPU interface are described in *Table-6.5*.

Table-6.5: CPU Interface Signals

Name	Type	Description
CPUDI	I	CPU data input
CPUDO	Tri-state	CPU data output
CPUIRQ	O	CPU interrupt request

A command sent by the CPU through the CPUDI line consists of 7 octets. Command frames transmitted on CPUDI have the format shown in *Table-6.6*:

Table-6.6: CPUDI Format

Operation	Command	Address	Index	Data	Checksum
Write	0010XX11	A[7:0]	I[7:0]	D[23:0]	C[7:0]

Read	0010XX01	A[7:0]	I[7:0]	D[23:0]	C[7:0]

The byte order of data in all fields follows the big-endian convention, i.e. most significant octet first. The bit order is the least significant order first. The Command octet specifies the type of the operation.

The Bit-7, bit-6, and bit-5 of the command octet are specified the Device Type.
(1) Switch Controller, the device type is 001.
(2) ARL Controller, the device type is 010.
(3) Management Controller, the device type is 100.

The Bit-2 and bit-3 of the command octet are used to specify the device ID of the chip. They are set by bit 20 and bit 21 of the *Register 25* at power on strobe. The address octet specifies the number of the register. The index octet specifies the index of the register in a register array.

For write operation, the Data field is a 3-octet value to specify what to write into the register. For read operation, the Data field is a 3-octet 0 as padded data. If the data of register is less than 24-bit, it is aligned to bit-0 of Data field.

The checksum value is an 8-bit value of exclusive-OR of all octets in the frame, starting from the Command octet.

For each valid command received, the ACD82224 will always send a response. Response from the ACD82224 is sent through the CPUDO line. Response frames sent by the ACD82224 have the following format (*Table-6.7*):

Table-6.7: Switch Response Format

Response	Command	Result	Data	Checksum
Write	00000011	R[7:0]	D[23:0]	C[7:0]
Read	00000001	R[7:0]	D[23:0]	C[7:0]

The command octet specifies the type of the response. The result octet specifies the result of the execution.

The Result field in a response frame is defined as:
- “0” for no error
- “1” for Access Violation Error

For response to a read operation, the Data field is a 3-octet value to indicate the content of the register. For response to a write operation, the Data field is 32 bits of 0. If the data of register is less than 24-bit, it is align to bit-0 of Data field.

The checksum value is an 8-bit value of exclusive-OR of all octets in the response frame, starting from the Command octet.

CPUIRQ is high active and used to notify the CPU that some special status has been encountered by the ACD82224, like port partition, and fatal system error, etc. By clearing the appropriate bit in the interrupt mask register, the specific source interrupt can be stop. Reading the interrupt source register retrieves the source of the interrupt request and clears the interrupt source register. CPUIRQ keeps high if the interrupt source is still existed.

SRAM Interface

All received frames are stored into the shared frame buffer through the memory interface. When the destination port is ready to transmit the frame, data is read from the shared memory buffer through the memory interface. The memory interface signals are described in the following table:

Table-6.8: ZBT SRAM Interface

Name	Type	Description
DATA[47:0]	I/O	Memory data bus
BE[2:0]	I/O	Byte enable
EOF	I/O	End of frame
ADDR[18:0]	O	Memory address bus
nWE	O	Write enable: 0=Write 1=Read
nCE	O	Chip enable, low active

The synchronous clock input of the External SRAM's should connect to 100 MHz system clock. Data is written into the SRAM or read from the SRAM in 52-bit wide words. ADDRx specifies the address of each word.

- DATA[47:0] are used to transfer up to 6 octets of data each time.
- BE[2:0] indicate the valid bytes in 6 octet of data in DATA[47:0].
- EOF indicates the last word of a frame.
- ADDR[18:0] specifies up to 512K word addresses.
- nWE specifies the type of operation for each clock cycle.
- nCE selects the SRAM chip associated with the word address.

The timing requirement for SRAM access is described in the chapter of "Timing Description."

ARL & MIB Interfaces

The ARL interface provides a communication path between the ACD82224 and an external ARL device such as the ACD80800, which can provide up to 11K of address lookup. The MIB interface provides a communication path between the ACD82224 and an external MIB device such as the ACD80900 for management function implementation. Both the ACD80800 and the ACD80900 collect traffic information by monitoring the data bus of the buffer memory. The two interfaces share many common pins, as shown in *Figure-6.1*and *Table-6.9.*

When the ACD82224 receives a frame, it will store it into the frame buffer memory through data bus DATA[47:0]. The external ARL extracts the destination address and source address of the frame posted on the Data [0:47] while it is written into the memory. It then finds the corresponding destination port and returns the result through the ARLDI[3:0] lines to the ACD82224. The timing requirement on ARL signals is described in Chapter 9, "Timing Description". At the same time, the external MIB also grabs the frame into its internal buffer for further management processing. Please see the ACD80800 and the ACD80900 Data Sheets for details.

Table-6.9: ARL & MIB Interfaces Signals

Name	Type	ARL Interface	MIB Interface
DATA[47:0]	O	Frame Data	
BE[2:0]	O	Byte Enable: BE[0:2]	
EOF	O	End of Frame: EOF	
SWDIR[1:0]	O	Data Direction Indicator : 00 = idle, 01 = receive, 10 = transmit, 11 = control	
SWSYNC	O	Port synchronization: indicating when Port-0 is driving DAT[0:47]	
SWSTAT[3:0]	O	Data state indicator: 0000 – Idle 0001 – First word (DA) 0010 – Second word (SA) 0011 – Third through last word 0100 – Filter Event 0101 – Drop Event 0110 – Jabber 0111 – False Carrier (receive) - Deferred Transmission (transmit) 1000 – Alignment error (receive) - Single Collision (transmit) 1001 – Flow Control (receive) - Multiple Collision (transmit) 1010 – Short Event (receive) - Excessive Collision (transmit) 1011 – Runt (receive) - Late Collision (transmit) 1100 – Symbol Error 1101 – FCS Error 1110 – Long Event 1111 – Reserved	
SWRXCLK	O	Receive Clock	
SWTXCLK	O	NA	Transmit Clock
P-23 MII		NA	The default CPU port, share with regular Port-23 traffic
ARLDI[3:0]	I	ARL Data Input: Returns 12-bit (3-cycles) ARL look-up result to the switch: •Bit[11:7] - Source Port ID (0 - 23) •Bit[6:5] – Look-up Result: 00 – reserved 01 – matched 10 – not matched 11 – forced discard •Bit[4:0] - Destination Port ID (0 - 23)	NA
ARLDIV	I	ARL Data Input Valid: Assertion to indicate start of a new result on ARLDI[0:3]	NA

LED Interface

The status of each port is displayed on the LED interface for every 50ms. LEDVLD0 and LEDVLD1 are used to indicate the start and end of the LED data. LED data is clocked out by the

falling edge of LEDCLK, and should be sampled by the rising edge of LEDCLK. LED data of port-23 are clocked out first, followed by port-22 down to port-0. All LED signals are low active. The signals in the LED interface is described in *Table-6.10*:

Table-6.10: LED Interface Signals

Name	Type	Description	Group 0	Group 1
LEDVLD0	O	LED signal valid 0	1	0
LEDVLD1	O	LED signal valid 1	0	1
LEDCLK	O	2.5 MHz LED clock	-	-
nLED0	O	Dual purpose indicator	NA	frame error indicator
nLED1	O	Dual purpose indicator	full duplex indication	collision indication
nLED2	O	Dual purpose indicator	port speed (1=10Mbps,0=100Mbps)	receiving activity
nLED3	O	Dual purpose indicator	Link status	transmit activity

Configuration Interface

The default values of certain register bits are set by internal pull-high/pull-low 75K Ohm resistors. These default values can be overwritten by external pull-high/pull-low with 4.7K Ohm resistors. *Table-6.11* lists all the available pins. The meanings of the register bits are described in the chapter of "Register Description."

Table-6.11: Configuration Interface

POS#	Pin Name	Register	Bit#	Default
0	P20TXD0	Register-25	0	Table-7.25
1	P20TXD1		1	
2	P21TXD0		2	
3	P21TXD1		3	
4	P17TXD0		4	
5	P17TXD1		5	
6	P18TXD0		6	
7	P18TXD1		7	
8	P20TXEN		8	
9	P21TXEN		9	
10	P17TXEN		10	
11	P18TXEN		11	
12	LEDCLK		12	
13	LEDVLD0		13	
14	LEDVLD1		14	
15	nLED3		15	
16	nLED2		16	
17	nLED1		17	
18	nLED0		18	
19	P15TXEN		19	
20	P12TXEN,		20	
21	P14TXEN		21	
22	P23TXEN		22	
23	P11TXEN		23	
24	P03TXEN	Register-16	15	Table-7.16
25	P23TXD0	Register-31	0	Table-7.31
26	P23TXD1		1	
27	P23TXD2		2	
28	P23TXD3		3	
29	P05TXEN		4	
30	P06TXEN		5	
31	P08TXEN		6	
32	P09TXEN		7	
33	P00TXD0	Register-30	0	Table-7.30
34	P00TXD1		1	
35	P02TXD0		2	
36	P02TXD1		3	
37	P03TXD0		4	
38	P03TXD1		5	
39	P05TXD0		6	
40	P05TXD1		7	
41	P00TXEN	Register-20 of the built-in ARL	1	0
42	P02TXEN		2	0

Note: POS41 and POS42 are belonging to Build-In ARL'S *PosCfg Register (ARL Register 20).*

Other Interface

Table-6.12: Other Signals

Name	Type	Description
CLK100	I	100 MHz clock input
NRESET	I	Hardware reset, low active
WCHDOG	O	Watch dog life pulse
SYSERROR	O	system error indication
ZBTCLK	O	Factory use only
CLKSEL, EN16P, TESTEN	I	Factory use only
VDD	I	3.3V power
VDDC	I	2.5V power
VSS	I	Ground

The CLK100 should come from 100MHz clock oscillator, with 3.3Volt or 5Volt, 40/60 Duty Cycle, and 50 ppm accuracy.

The nRESET is a low-active hardware reset pin. Assertion of this pin will cause the ACD82224 to go through a power-up initialization process. All registers are set to their default value after reset.

The WCHDOG pin is used to handle exceptional cases. A normal working ACD82224 sends out continuous life pulses from the WCHDOG pin, which can be monitored by an external watchdog circuit. If no life pulse is detected, the external watchdog circuit may force reset of the switch system. It is a safeguard against unforeseeable situations.

The SYSERROR is System error indication and it's high active. When there is any error occurs in SYSERR Register it's asserted to HIGH until the SYSERR Register been read.

The CLKSEL, EN16P, and TESTEN are input pins for factory test use only. These three pins must be connected to ground directly. The ZBTCLK is output pin and also for factory use. Do not connect the ZBTCLK.

7. REGISTER DESCRIPTION

Registers in the ACD82224 are used to define the operational mode of various function modules of the switch controller and the peripheral devices. Default values at power-on are predefined. The management CPU (optional) can read the content of all registers and modify some of the registers to change the operational mode. Table-7.0.1 lists all the registers inside the switch controller.

Table-7.0.1: Register List

Address	Name	Type	Size	Index	Description
0	DEVID	R	16 Bit	1	Device ID is 0601h
1	INTSRC	R	8 Bit	1	Interrupt Source
2	SYSERR	R	9 Bit	1	System Error
3	PAR	R	24 Bit	1	Port Partition Indication
4	PMERR	R	24 Bit	1	PHY Management Error
5	ACT	R	24 Bit	1	Port Activity
6	RSVD	R	-	-	-
7	RSVD	R	-	-	-
8	SAL	R/W	24 Bit	1	Source Address, bit 23:0
9	SAH	R/W	24 Bit	1	Source Address, bit 47:24
10	UTH	R/W	16 Bit	1	Unicast Threshold
11	BTH	R/W	16 Bit	1	Broadcast Threshold
12	MAXL	R/W	16 Bit	1	FCS of *Max-Pause-Frame*, bit 15:0
13	MAXH	R/W	16 Bit	1	FCS of *Max-Pause-Frame*, bit 31:16
14	MINL	R/W	16 Bit	1	FCS of *Mini-Pause-Frame*, bit 15:0
15	MINH	R/W	16 Bit	1	FCS of *Mini-Pause-Frame*, bit 31:16
16	SYSCFG	R/W	16 Bit	1	System Configuration
17	INTMSK	R/W	8 Bit	1	Interrupt Mask
18	SPEED	R/W	24 Bit	1	Port Speed
19	LINK	R/W	24 Bit	1	Port Link
20	nFWD	R/W	24 Bit	1	Port Forward Disable
21	nBP	R/W	24 Bit	1	Port Back Pressure Disable
22	nPORT	R/W	24 Bit	1	Port Disable
23	PVID	R/W	4 Bit	24	Port VLAN ID
24	VPID	R/W	5 Bit	4	VLAN Dumping Port
25	POSCFG	R/W	24 Bit	1	Power-On-Strobe Configuration
26	PAUSE	R/W	24 Bit	1	Port Pause Frame Disable
27	DPLX	R/W	24 Bit	1	Port Duplex Mode
28	RSVD	R	-	-	-
29	nPM	R/W	24 Bit	1	Port PHY Management Disable
30	ERRMSK	R/W	8 Bit	1	Error Mask
31	CLKADJ	R/W	8 Bit	1	ARL Clock Delay Adjustment
32~63	PHYREG	R/W	16 Bit	32	Registers in PHY device with ID is 0~31

Many registers have particular bit designated to a particular port, so that the status of each port can be changed or monitored independently. The mapping of Register-Bit and Port-ID for each controller is listed in Table-7.0.2.

Table-7.0.2: Register-Bit/Port-ID Mapping

Register –Bit	Port –ID	Port Number
		ACD82224
0	0	Port 0
1	1	Port 1
2	2	Port 2
3	3	Port 3
4	4	Port 4
5	5	Port 5
6	6	Port 6
7	7	Port 7
8	8	Port 8
9	9	Port 9
10	10	Port 10
11	11	Port 11
12	12	Port 12
13	13	Port 13
14	14	Port 14
15	15	Port 15
16	16	Port 16
17	17	Port 17
18	18	Port 18
19	19	Port 19
20	20	Port 20
21	21	Port 21
22	22	Port 22
23	23	Port 23

INTSRC register (register 1)

The INTSRC register indicates the source of the interrupt request. Before the CPU starts to respond to an interrupt request, it should read this register to find out the interrupt source. This register is automatically cleared after each read. Table-7.1 lists all the bits of this register.

Table-7.1: INTSRC Register

Bit	Description	Default
0	System initialization completed	0
1	System error occurred	
2	Port partition occurred	
3	ARL Interrupt	
4	Reserved	
5		
6		
7		

Note: The source interrupt for bit-3 ARL interrupt is referred to ARL Register-13.

SYSERR register (register 2)

The SYSERR register indicates the presence of system errors. It is automatically cleared after each read. Table-7.2 lists the system error reported.

Table-7.2: SYSERR Register

Bit	Description	Default
0	BIST failure indication	0
1	Reserved	
2		
3		
4		
5		
6		
7		
8		

PAR register (register 3)

The PAR register indicates the presence of the partitioned ports and the port ID. A port can be automatically partitioned if there is a consecutive false carrier event, an excessive collision or jabber. This register is automatically cleared after each read. Table-7.3 lists the bits of this register.

Table-7.3: PAR Register

Bit	Description	Default
[0:23]	0 – Port X is not partitioned. 1 – Port X is partitioned.	0

PMERR register (register 4)

The PMERR register indicates the presence of PHYs that have failed to respond to the PHY Management command issued through the MDIO line. This register is automatically cleared after each read. Table-7.4 describes the bits of this register.

Table-7.4: PMERR Register

Bit	Description	Default
[0:23]	0 – Port X's PHY responded 1 – Port X's PHY failed to respond	0

ACT register (register 5)

The ACT register indicates the presence of transmitting or receiving activities of each port since the register was last read. This register is automatically cleared after each read. Table-7.5 describes all the bits of this register.

Table-7.5: ACT register

Bit	Description	Default
[0:23]	0 – Port X no activity 1 – Port X has activity	0

SAL & SAH register (register 8,9)

The SAL and SAH registers together contain the complete Source Address for pause frame generation. SAL contains the least significant 24 bit of the MAC address. SAH contains the most significant 24 bit of the MAC address. The default locally managed source address for pause frame generation is FEh-FFh-FFh-FFh-FFh-FFh a. Table-7.8 and table-7.9 describes all the bits of these two registers.

Table-7.8: SAL Register

Bit	Description	Default
7:0	Bit 47:40 of the switch's MAC address.	FEh
15:8	Bit 39:32 of the switch's MAC address.	FFh
23:16	Bit 31:24 of the switch's MAC address.	

Table-7.9: SAH Register

Bit	Description	Default
7:0	Bit 23:16 of the switch's MAC address.	FFh
15:8	Bit 15:8 of the switch's MAC address.	
23:16	Bit 7:0 of the switch's MAC address.	

UTH register (register 10)

The UTH register contains the unicast buffer thresholds for each port. When the upper threshold is exceeded, the MAC may generate a *Max-Pause-Frame*. When the lower threshold is crossed, the MAC may generate a *Mini-Pause-Frame*. Table-7.10 describes each bit in this register.

Table-7.10: UTH Register

Bit	Description	Total Frame Buffer Depth (52-bit wide word)	Default*
7:0	Lower threshold of unicast utilization.	64K 128K 256K 512k	2 4 8 16
15:8	Higher threshold of unicast utilization.	64K 128K 256K 512k	4 8 24 64

*Note: The value is related to the memory size specified by bit[9:8] of register 25.

BTH register (register 11)

The BTH register contains the broadcast queue buffer threshold for each port. When the upper threshold is exceeded, the MAC may generate a *Max-Pause-Fra*me. When the lower threshold is crossed, the MAC may generate a *Mini-Pause-Frame*. Table-7.11 describes each bit in this register.

Table-7.11: BTH Register

Bit	Description	Default
7:0	Lower threshold of broadcast queue	16
15:8	Higher threshold of broadcast queue	48

MINL & MINH register (register 12,13)

The MINL and MINH registers together contain the 32-bit Frame Check Sequence (FCS) of the mini-pause-frame. MINL contains the least significant 16 bit of the FCS. MINH contains the most significant 16 bit of the FCS. The default FCS value assumes the default source address for the *Mini-Pause-Frame*. Table-7.12 and table-7.13 describe all the bits of these two registers.

Table-7.12: MINL Register

Bit	Description	Default
7:0	Bit 31:24 of the mini-pause-frame's FCS	89
15:8	Bit 23:16 of the mini-pause-frame's FCS	O3

Table-7.13: MINH Register

Bit	Description	Default
7:0	Bit 15:18 of the mini-pause-frame's FCS	D7
15:8	Bit 7:0 of the mini-pause-frame's FCS	A9

MAXL & MAXH register (register 14,15)

The MAXL and MAXH registers together contain the 32-bit Frame Check Sequence (FCS) of the max-pause-frame. MAXL contains the least significant 16 bit of the FCS. MAXH contains the most significant 16 bit of the FCS. The default FCS value assumes the default source address for the *Max-Pause-Frame*. Table-7.14 and table-7.15 describe all the bits of these two registers.

Table-7.14: MAXL Register

Bit	Description	Default
7:0	Bit 31:24 of the max-pause-frame's FCS	0D
15:8	Bit 23:16 of the max-pause-frame's FCS	68

Table-7.15: MAXH Register

Bit	Description	Default
7:0	Bit 15:8 of the max-pause-frame's FCS	D8
15:8	Bit 7:0 of the max-pause-frame's FCS	D0

SYSCFG register (register 16)

The SYSCFG register specifies certain system configurations. The system options are described in the chapter of "Function Description." Table-7.16 describes all the bits of this register.

Table-7.16: SYSCFG Register

Bit	Description	Default
0	0 – BIST enabled; 1 – BIST disabled.	0
1	0 – Spanning Tree support disabled; 1 – Spanning Tree support enabled	
2	0 – External ARL result latched by rising edge; 1 – External ARL result latched by falling edge;	
3	Reserved.	
4	Reserved.	
5	0 – wait for CPU. 1 – system ready to start **This bit is used by the CPU when bit-15 of register-25 is set as "0" (for system with control CPU). The system will wait for CPU to set this bit.* *Must set this bit before CPU programs any register.*	
6	0 – PHY Management not completed 1 – PHY Management completed. **This bit is used by the CPU when bit-15 of register-25 is set as "0" (for system with a control CPU). The MAC will not start until this bit is set by the CPU.*	
7	0 – Watchdog function enabled. 1 – Watchdog function disabled.	
8	0 – Secure VLAN checking rule enforced. 1 – Leaky VLAN checking rule enforced.	
9	Reserved.	
10	0 – Late Back-Pressure scheme disabled 1 – Late Back-Pressure scheme enabled **When enabled, the MAC will generate back-pressure only after reading the first bit of DA*	
11	0 – special handling of broadcast frames disabled 1 – special handling of broadcast frames enabled **When enabled, all broadcast frames from Port0~Port22 are forwarded to the Port23 only, and all broadcast frames from the Port23 are forwarded to all other ports.*	
12	Software Reset: Set "1" to start a system reset to initialize all state machines. It will not re-start PHY's Auto-Negotiation.	
13	Hardware Reset: Set "1" to stop the life pulse on the watchdog pin, which in turn will trigger the external watchdog circuitry to reset the whole system.	
14	Reserved	
15	0 – Port 23 is MII 1 – Port 23 is RMII (POS shared with P03TXEN)	1

If the bit-19 of Register-25 is set (CPU start mode), ACD82224 will stop the initialized procedures after it is completed with self-test. CPU must set bit-5 of register-16 to enable access internal registers. CPU set bit-6 of register-16 to enable MAC and Queue manager. Then ACD82224 will start switching based on CPU's configuration.

INTMSK register (register 17)

The INTMSK register defines the valid interrupt sources allowed to assert interrupt request pin. Table-7.17 lists all the bits of this register.

Table-7.17: INTMSK Register

Bit	Description	Default
0	Enable “system initialization completion” to interrupt	1
1	Enable “internal system error” to interrupt	
2	Enable “port partition event” to interrupt	
3	Enable “internal ARL” to interrupt	
4	Reserved	
5		
6		
7		

SPEED register (register 18)

The SPEED register specifies or indicates the speed rate of each port. Table-7.18 describes all the bits of this register. These two modes are also applied to the SPEED (register-18), LINK (register-19), DPLX (register-27), PAUSE (register-26) register.

(1) Automatic PHY management mode (default setting): These four registers controlled by ACD82224's PHY management Hardware update their status. To enable this mode if bit-16 of register-25 is cleared, and the corresponding bit (port) in nPM register (register-29) is cleared.

(2) CPU mode: CPU sets these registers through UART interface. To enable this mode if bit-16 of register-25 is set, or the corresponding bit (port) in nPM register (register-29) is set.

Table-7.18: SPEED Register

Bit	Description	Default
[0:23]	0 – Port X at 10Mbps 1 – Port X at 100Mbps	0

LINK register (register 19)

The LINK register specifies or indicates the link status of each port. Table-7.19 describes all the bits of this register.

Table-7.19: LINK Register

Bit	Description	Default
[0:23]	0 – Port X link not established 1 – Port X link established	0

nFWD register (register 20)

The nFWD register defines the forwarding mode of each port. Under *forwarding* mode, a port can forward all frames. Under *block-and-listen* mode, a port will not forward regular frames, except BPDU frames. If the spanning tree algorithm discovers redundant links, the control CPU will allow only one link remaining in *forwarding* mode and force the other links into *block-and-listen* mode. Setting the associated bit in this register will put the port into *block-and-listen* mode. Table-7.20 describes all the bits of this register.

Table-7.20: nFWD Register

Bit	Description	Default
[0:23]	0 – Port X in forwarding state. 1 – Port X in block-and-listen state.	0

nBP register (register 21)

The nBP register defines backpressure flow control capability for each port. Table-7.21 describes all the bits of this register.

Table-7.21: nBP Register

Bit	Description	Default
[0:23]	0 – Port X back-pressure scheme enabled 1 – Port X back pressure scheme disabled	0

nPORT register (register 22)

The nPORT register is used to isolate ports from the network. Setting the associated bit in this register will stop a port from either receiving or transmitting any frame. Table-7.22 describes all the bits of this register. But, the source MAC address is still learned by ARL unless ARL nLearnReg is set for the specific port.

Table-7.22: nPORT Register

Bit	Description	Default
[0:23]	0 – Port X enabled 1 – Port X disabled	0

PVID register (register 23)

The PVID registers assign VLAN IDs for each port. There are 24 PVID registers, one for each port. A PVID consists of 4 bits, each corresponding bit mapping to one of the 4 VLANs. A port can belong to more than one VLAN at the same time. Table-7.23 describes all the bits of this register.

Table-7.23: PVID Register

Bit	Description	Default Port 1~23 (index = 1~23)	Default Port 0 (index = 0)
0	0 – port not in VLAN-I. 1 – port in VLAN-I.	1	1
1	0 – port not in VLAN-II. 1 – port in VLAN-II.	0	1
2	0 – port not in VLAN-III. 1 – port in VLAN-III.	0	1
3	0 – port not in VLAN-IV. 1 – port in VLAN-IV.	0	1

VPID register (register 24)

The VPID registers specify the dumping port for each VLAN. There are 4 VPID 5-bit registers, one for each VLAN. A valid VPID's ID is “0” through “23". Table-7.24 describes all the bits of this register. If the multiple VLANs are set, the dumping port for the associated VLAN has to be assigned correctly even bit-12 of register-25 is set (unknown DA forwarded to all ports).

Table-7.24: VPID Registers *(4 registers)*

VPID register	Index	Bit	Description	Default
VPID[0]	0	4:0	Dumping port ID for VLAN-1	“00000”
VPID[1]	1	4:0	Dumping port ID for VLAN-2	Port-0
VPID[2]	2	4:0	Dumping port ID for VLAN-3	
VPID[3	3	4:0	Dumping port ID for VLAN-4	

POSCFG register (register 25)

The POSCFG register specifies a certain configuration setting for the switch system. The default values of this register can be changed through pull-up/pull-down of specific pins, as described in the “Configuration Interface” section of the “Interface Description” chapter. Table-7.25 describes all the bits of this register.

Table-7.25: POSCFG Register

Bit	Description	Default	Shared Pin
3:0	ZBT SRAM Read Timing Adjustment *(16 levels within a 10 ns clock cycle, each delay unit adds approximately 0.5-0.7 ns, "inversion" adds 5 ns or one half of clock cycle to the delay)* 0001 – no delay 0000 – inversion plus no delay 0011 – 1 units delay 0010 – inversion plus 1 units delay 0101 – 2 units delay 0100 – inversion plus 2 units delay 0111 – 3 units delay 0110 – inversion plus 3 units delay 1001 – 4 units delay 1000 – inversion plus 4 units delay 1011 – 5 units delay 1010 – inversion plus 5 units delay 1101 – 6 units delay 1100 – inversion plus 6 units delay 1111 – 7 units delay 1110 – inversion plus 7 units delay	0000	P21TXD1 P21TXD0 P20TXD1 P20TXD0 (From Bit3 to Bit0)
7:4	ZBT SRAM Clock Timing Adjustment *(16 levels within a 10 ns clock cycle, each delay unit adds approximately 0.5-0.7 ns, "inversion" adds 5 ns or one half of clock cycle to the delay)* 0101 – no delay 0100 – inversion plus no delay 0111 – 1 units delay 0110 – inversion plus 1 units delay 0001 – 2 units delay 0000 – inversion plus 2 units delay 0011 – 3 units delay 0010 – inversion plus 3 units delay 1101 – 4 units delay 1100 – inversion plus 4 units delay 1111 – 5 units delay 1110 – inversion plus 5 units delay 1001 – 6 units delay 1000 – inversion plus 6 units delay 1011 – 7 units delay 1010 – inversion plus 7 units delay	0000	P18TXD1 P18TXD0 P17TXD1 P17TXD0
9:8	SRAM size selection: 00 – 64K words 01 – 128K words 10 – 256k words 11 – 512K words	01	P21TXEN P20TXEN
10	0 – MDC latched by rising edge; 1 – MDC latched by falling edge;	0	P17TXEN
11	0 – Long Event defined as frame longer than 1518 byte. 1 – Long Event defined as frame longer than 1530 byte.	1	P18TXEN
12	0 – Frames with unknown DA forwarded to the dumping port. 1 – Frames with unknown DA forwarded to all ports.	0	LEDLCK
13	0 – Internal ARL selected (2K MAC address entry). 1 - External ARL selected (11K MAC address entry).	0	LEDVLD0
14	0 – PHY Ids start from 0, range from 1 to 23. 1 – PHY Ids start from 1, range from 0 to 24	0	LEDCLD1
15	0 – Re-transmit after excessive collision. 1 – Drop after excessive collision.	0	nLED3
16	0 – Automatic PHY Management enabled 1 – Automatic PHY Management disabled: CPU need to update SPEED, LINK, DPLX and PAUSE registers	0	nLED2
17	0 – Flow Control on broadcast queue utilization enabled 1 – Flow control on broadcast queue utilization disabled: broadcast frames dropped if the queue is full	0	nLED1
18	0 – System errors will trigger software reset 1 – System errors will trigger hardware reset	0	nLED0
19	0 – System will start by itself upon hardware reset 1 – System will not start until bit-5/6 of register-16 is set	0	P15TXEN
21:20	2-bit device ID for UART communication. The device responses only to UART commands with matching ID	0	P14TXEN P12TXEN
22	0 – RMII TX's data is driven on falling edge 1 – RMII TX's data is driven on rising edge	1	P23TXEN
23	0 – RMII RX's data is latched on rising edge 1 – RMII RX's data is latched on falling edge	1	P11TXEN

PAUSE register (register 26)

The PAUSE register defines the pause-frame based flow control capability of each port. Table-7.26 describes all the bits of this register.

Table-7.26: PAUSE Register

Bit	Description	Default
[0:23]	0 – Port X Pause-Frame disabled 1 – Port X Pause-Frame enabled	0

DPLX register (register 27)

The DPLX register specifies or indicates the half/full-duplex mode of each port. Table-7.27 describes all the bits of this register.

Table-7.27: DPLX Register

Bit	Description	Default
[0:23]	0 – Port X under half duplex mode 1 – Port X under full duplex mode	0

nPM register (register 29)

The nPM register indicates the automatic PHY management capability of each port. If a bit is set in this register, the corresponding SPEED, LINK, DPLX, and PAUSE registers of the port will not be updated by Automatic PHY Management. Table-7.29 describes all the bits of this register.

Table-7.29: nPM Register

Bit	Description	Default
[0:23]	0 – Port X's status update enabled 1 – Port X's status update disabled	0

ERRMSK register (register 30)

The ERRMSK register defines certain errors as *system errors*. It is reserved for factory use only. Table-7.30 lists all the error masks specified by this register.

Table-7.30: ERRMSK register

Bit	Description	Default	Shared Pin
0	Reserved	1	P00TXD0
1	Reserved	1	P00TXD1
2	Reserved	1	P02TXD0
3	Reserved	1	P02TXD1
4	Reserved	1	P03TXD0
5	Reserved	1	P03TXD1
6	Reserved	1	P05TXD0
7	Reserved	1	P05TXD1

CLKADJ register (register 31)

The CLKADJ register defines the delay time of the ARLCLK relative to the transition edge of the data signals. The ARLCLK provides reference timing for supporting chips, such as the

ACD80800 and the ACD80900, which need to snoop the data bus for certain activities. Table-7.31 describes all the bits of this register.

Table-7.31: CLKADJ Register

Bit	Description	Default	Shared Pin
3:0	ARL Clock Timing Adjustment: *(16 levels within a 10 ns clock cycle, each delay unit adds approximately 0.5-0.7 ns, "inversion" adds 5 ns or one half of clock cycle to the delay)* 0001 – no delay 0000 – inversion plus no delay 0011 – 1 units delay 0010 – inversion plus 1 units delay 0101 – 2 units delay 0100 – inversion plus 2 units delay 0111 – 3 units delay 0110 – inversion plus 3 units delay 1001 – 4 units delay 1000 – inversion plus 4 units delay 1011 – 5 units delay 1010 – inversion plus 5 units delay 1101 – 6 units delay 1100 – inversion plus 6 units delay 1111 – 7 units delay 1110 – inversion plus 7 units delay	0011	P23TXD3 P23TXD2 P23TXD1 P23TXD0 (From Bit3 to Bit0)
5:4	Write Data Window Width Adjustment *(Approximately 0.7 ns per increment)* 00 – Default 2.5 ns wide 01 – 1 units increment 10 – 2 units increment 11 – 3 units increment	00	P06TXEN P05TXEN
7:6	Write Data Window Location Adjustment *(Approximately 1.25ns per increment)* 00 – no delay 01 – 1 unit delay 10 – 2 units delay 11 – 3 units delay	00	P09TXEN P08TXEN

PHYREG register (register 32-63)

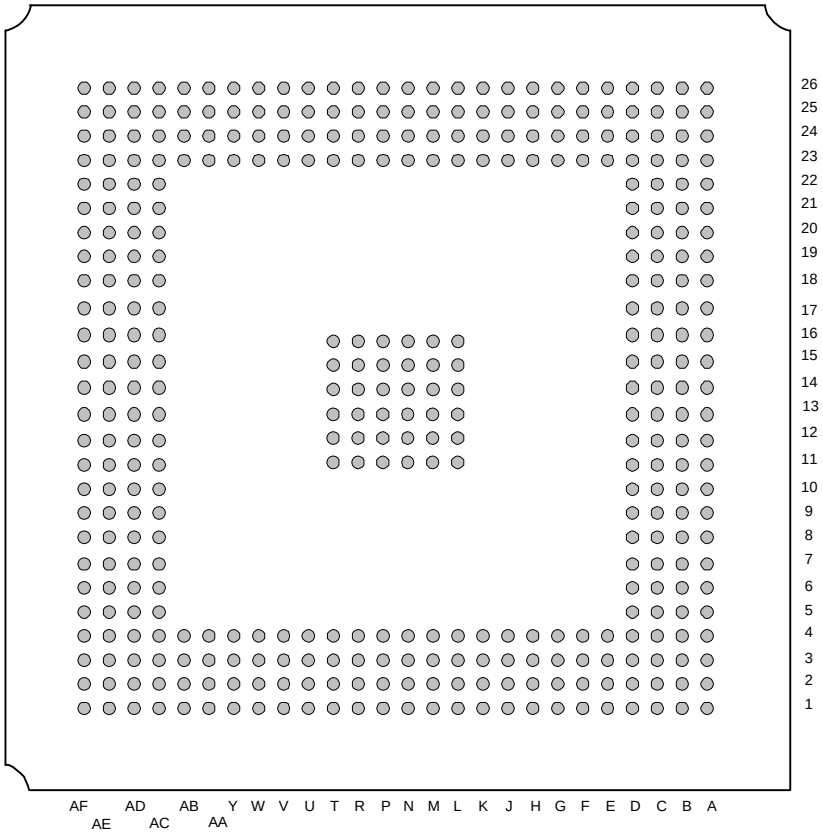
The PHYREG refers to the registers residing on the PHY devices. The ACD82224 provides a mirror access path for the control of CPU to access the registers on the PHYs. For detailed information about the PHY registers, please refer to the PHY data sheet. The register index is used by the ACD82224 to specify the PHY's internal registers.
For example, register-36 with index-0 would refer to the control register (register-0) in the device of PHY's ID is 4.

Table-7.24: PHY Registers

PHYREG 32~63	Index	Description	Default
Register 32	0~31	PHY Management registers with PHY's ID is 0	PHY Default Value
Register 33	0~31	PHY Management registers with PHY's ID is 1	
Register 34	0~31	PHY Management registers with PHY's ID is 2	
Register 35	0~31	PHY Management registers with PHY's ID is 3	
..	..	..	
..	..	..	
Register 63	0~31	PHY Management registers with PHY's ID is 31	

8. PIN DESCRIPTIONS

Figure-8.1: Pin Diagram/Bottom View



RMII Clock Interface

Pin Name	Location	I/O	Description
RMIICLK0 RMIICLK1 RMIICLK2 RMIICLK3 RMIICLK4 RMIICLK5	AC07 AF16 AE24 V24 J25 C23	O	Reduced MII clock (50 MHz)

RMII Interface (Port 0 ~ Port 22)

Pin Name	Location	I/O	Description
P00CRS_DV	AE10	I	Carrier Sense/Receive data valid
P00RXD0	AF10	I	Receive data bit 0
P00RXD1	AC10	I	Receive data bit 1
P00TXD0	AC09	I/O*	Transmit data bit 0 ERRMSK REG Bit-0: Reserved (default high)
P00TXD1	AD08	I/O*	Transmit data bit 1 ERRMSK REG Bit-1: Reserved (default high)
P00TXEN	AD07	I/O*	Transmit enable ARL's POSCFG REG Bit-1: NOCPU mode (default low). Suggest pull-high with 4.7K resister to set NOCPU enabled.
P01CRS_DV	AE12	I	Carrier Sense/Receive data valid
P01RXD0	AF12	I	Receive data bit 0
P01RXD1	AD11	I	Receive data bit 1
P01TXD0	AD10	O	Transmit data bit 0
P01TXD1	AF11	O	Transmit data bit 1
P01TXEN	AE11	I/O*	Transmit enable ARL's POSCFG REG Bit-2: Reserved (default low).
P02CRS_DV	AD12	I	Carrier Sense/Receive data valid
P02RXD0	AE14	I	Receive data bit 0
P02RXD1	AC14	I	Receive data bit 1
P02TXD0	AC12	I/O*	Transmit data bit 0 ERRMSK REG Bit-2: Reserved (default high)
P02TXD1	AF13	I/O*	Transmit data bit 1 ERRMSK REG Bit-3: Reserved (default high)
P02TXEN	AE13	O	Transmit enable
P03CRS_DV	AD14	I	Carrier Sense/Receive data valid
P03RXD0	AE16	I	Receive data bit 0
P03RXD1	AD15	I	Receive data bit 1
P03TXD0	AD13	I/O*	Transmit data bit 0 ERRMSK REG Bit-4: Reserved (default high)
P03TXD1	AE15	I/O*	Transmit data bit 1 ERRMSK REG Bit-5: in: Reserved (default high)
P03TXEN	AF14	I/O*	Transmit enable SYSCFG REG Bit-15: Port23 MII/RMII Selection. (default high)
P04CRS_DV	AC17	I	Carrier Sense/Receive data valid
P04RXD0	AE18	I	Receive data bit 0
P04RXD1	AD17	I	Receive data bit 1
P04TXD0	AD16	O	Transmit data bit 0
P04TXD1	AF17	O	Transmit data bit 1
P04TXEN	AC15	O	Transmit enable
P05CRS_DV	AD18	I	Carrier Sense/Receive data valid
P05RXD0	AE20	I	Receive data bit 0
P05RXD1	AC19	I	Receive data bit 1
P05TXD0	AE19	I/O*	Transmit data bit 0

			ERRMSK REG Bit-6 (default high)
P05TXD1	AF19	I/O*	Transmit data bit 1 ERRMSK REG Bit-7 (default high)
P05TXEN	AF18	I/O*	Transmit enable (default Low) CLKADJ REG Bit-4: Write Data Window Width Adjustment bit-0
P06CRS_DV	AF21	I	Carrier Sense/Receive data valid
P06RXD0	AD20	I	Receive data bit 0
P06RXD1	AE22	I	Receive data bit 1
P06TXD0	AD19	O	Transmit data bit 0
P06TXD1	AE21	O	Transmit data bit 1
P06TXEN	AF20	I/O*	Transmit enable (default low) CLKADJ REG Bit-5: Write Data Window Width Adjustment bit-1
P07CRS_DV	AC22	I	Carrier Sense/Receive data valid
P07RXD0	AF23	I	Receive data bit 0
P07RXD1	AD22	I	Receive data bit 1
P07TXD0	AD21	O	Transmit data bit 0
P07TXD1	AE23	O	Transmit data bit 1
P07TXEN	AF22	O	Transmit enable
P08CRS_DV	AD25	I	Carrier Sense/Receive data valid
P08RXD0	AD26	I	Receive data bit 0
P08RXD1	AC25	I	Receive data bit 1
P08TXD0	AF24	O	Transmit data bit 0
P08TXD1	AE26	O	Transmit data bit 1
P08TXEN	AD23	I/O*	Transmit enable CLKADJ REG Bit-6: Write Data Window Location Adjustment. bit-0 (default low)
P09CRS_DV	AB23	I	Carrier Sense/Receive data valid
P09RXD0	AB24	I	Receive data bit 0
P09RXD1	Y23	I	Receive data bit 1
P09TXD0	AC26	O	Transmit data bit 0
P09TXD1	AB25	O	Transmit data bit 1
P09TXEN	AC24	I/O*	Transmit enable CLKADJ REG Bit-7: Write Data Window Location Adjustment. bit-1 (default low)
P10CRS_DV	Y26	I	Carrier Sense/Receive data valid
P10RXD0	Y24	I	Receive data bit 0
P10RXD1	W25	I	Receive data bit 1
P10TXD0	AA26	O	Transmit data bit 0
P10TXD1	Y25	O	Transmit data bit 1
P10TXEN	AA24	O	Transmit enable
P11CRS_DV	V25	I	Carrier Sense/Receive data valid
P11RXD0	V26	I	Receive data bit 0
P11RXD1	U25	I	Receive data bit 1
P11TXD0	W26	O	Transmit data bit 0
P11TXD1	W24	O	Transmit data bit 1
P11TXEN	V23	I/O*	Transmit enable POS REG Bit-23: RMII RX's data is latched on falling edge/rising edge(default high), Suggest pull-low with 4.7K resister.
P12CRS_DV	U24	I	Carrier Sense/Receive data valid
P12RXD0	R25	I	Receive data bit 0
P12RXD1	R26	I	Receive data bit 1
P12TXD0	U23	O	Transmit data bit 0
P12TXD1	T25	O	Transmit data bit 1
P12TXEN	U26	I/O*	Transmit enable (default low)

			POS REG Bit-20: 2-bit device ID bit-0 for UART communication. ACD82224 responses only to UART commands with matching ID.
P13CRS_DV	R24	I	Carrier Sense/Receive data valid
P13RXD0	N23	I	Receive data bit 0
P13RXD1	N26	I	Receive data bit 1
P13TXD0	R23	O	Transmit data bit 0
P13TXD1	P26	O	Transmit data bit 1
P13TXEN	P25	O	Transmit enable
P14CRS_DV	M26	I	Carrier Sense/Receive data valid
P14RXD0	L25	I	Receive data bit 0
P14RXD1	M24	I	Receive data bit 1
P14TXD0	M25	O	Transmit data bit 0
P14TXD1	N24	O	Transmit data bit 1
P14TXEN	P24	I/O*	Transmit enable (default low) POS REG Bit-21: 2-bit device ID bit-1for UART communication. The device responses only to UART commands with matching ID.
P15CRS_DV	L24	I	Carrier Sense/Receive data valid
P15RXD0	K26	I	Receive data bit 0
P15RXD1	K23	I	Receive data bit 1
P15TXD0	M23	O	Transmit data bit 0
P15TXD1	K25	O	Transmit data bit 1
P15TXEN	L26	I/O*	Transmit Enable (default low) POS REG Bit-19: system will start by itself upon hardware reset System will not start until bit-5/6 of register-16 is set .
P16CRS_DV	G25	I	Carrier Sense/Receive Data Valid
P16RXD0	H23	I	Receive data bit 0
P16RXD1	G26	I	Receive data bit 1
P16TXD0	H26	O	Transmit data bit 0
P16TXD1	J24	O	Transmit data bit 1
P16TXEN	K24	O	Transmit enable
P17CRS_DV	F26	I	Carrier Sense/Receive data valid
P17RXD0	G24	I	Receive data bit 0
P17RXD1	E25	I	Receive data bit 1
P17TXD0	F25	I/O*	Transmit data bit 0 (default low) POS REG Bit-4 ZBT SRAM Clock Timing Adjustment Bit-0
P17TXD1	G23	I/O*	Transmit data bit 1 (default low) POS REG Bit-5: ZBT SRAM Clock Timing Adjustment Bit-1
P17TXEN	H24	I/O*	Transmit enable (default low) POS REG Bit-10: Reserved
P18CRS_DV	E23	I	Carrier Sense/Receive data valid
P18RXD0	D26	I	Receive data bit 0
P18RXD1	C25	I	Receive data bit 1
P18TXD0	F24	I/O*	Transmit data bit 0 (default low) POS REG Bit-6: ZBT SRAM Clock Timing Adjustment Bit-2
P18TXD1	D25	I/O*	Transmit data bit 1 (default low) POS REG Bit-7: ZBT SRAM Clock Timing Adjustment Bit-3
P18TXEN	E26	I/O*	Transmit enable POS REG Bit-11: Long Event Defined As Frame Longer Than 1518/1530 Byte (default high)
P19CRS_DV	B24	I	Carrier Sense/Receive data valid
P19RXD0	A24	I	Receive data bit 0
P19RXD1	B23	I	Receive data bit 1
P19TXD0	C26	O	Transmit data bit 0
P19TXD1	A25	O	Transmit data bit 1

P19TXEN	D24	O	Transmit enable
P20CRS_DV	A22	I	Carrier Sense/Receive data valid
P20RXD0	B21	I	Receive data bit 0
P20RXD1	C21	I	Receive data bit 1
P20TXD0	B22	I/O*	Transmit data bit 0 (default low) POS REG Bit-0 ZBT SRAM Read Timing Adjustment bit-0
P20TXD1	D22	I/O*	Transmit data bit 1 (default low) POS REG Bit-1: ZBT SRAM Read Timing Adjustment bit-1
P20TXEN	A23	I/O*	Transmit enable (default high) POS REG Bit-8: SRAM size selection bit-0
P21CRS_DV	C20	I	Carrier Sense/Receive data valid
P21RXD0	D18	I	Receive data bit 0
P21RXD1	A19	I	Receive data bit 1
P21TXD0	B20	I/O*	Transmit data bit 0 (default low) POS REG Bit-2 ZBT SRAM Read Timing Adjustment bit-2
P21TXD1	A20	I/O*	Transmit data bit 1 (default low) POS REG Bit-3: ZBT SRAM Read Timing Adjustment bit-3
P21TXEN	A21	I/O*	Transmit enable (default low) POS REG Bit-9: SRAM size selection bit-1
P22CRS_DV	B17	I	Carrier Sense/Receive data valid
P22RXD0	C18	I	Receive data bit 0
P22RXD1	A17	I	Receive data bit 1
P22TXD0	B18	O	Transmit data bit 0
P22TXD1	A18	O	Transmit data bit 1
P22TXEN	C19	O	Transmit enable

MII Interface (Port 23)

Pin Name	Location	I/O	Description
P23CRS	D08	I	Carrier sense (Shared with RMII P23CRS_DV)
P23RXDV	B15	I	Receive data valid
P23RXCLK	A15	I	Receive clock (25/2.5 MHz)
P23RXERR	C16	I	Receive error
P23RXD0	A16	I	Receive data bit 0(Shared with RMII P23RXD0)
P23RXD1	C17	I	Receive data bit 1(Shared with RMII P23RXD1)
P23RXD2	B16	I	Receive data bit 2
P23RXD3	D17	I	Receive data bit 3
P23COL	C10	I	Collision indication
P23TXEN	D13	I/O*	Transmit data valid (Shared with RMII P23TXEN) POS REG Bit-22: RMII TX's data is driven on falling edge/rising edge (default high). Suggest pull-low with 4.7K resister.
P23TXCLK	D15	I	Transmit clock (25/2.5 MHz)
P23TXD0	C13	I/O*	Transmit data bit 0 (Shared with RMII P23TXD0) CLKADJ REG Bit-0: ARL Clock Timing Adjustment bit-0 (default low).
P23TXD1	D12	I/O*	Transmit data bit 1(Shared with RMII P23TXD1) CLKADJ REG Bit-1: ARL Clock Timing Adjustment bit-1 (default low).
P23TXD2	C11	I/O*	Transmit data bit 2 CLKADJ REG Bit-2: ARL Clock Timing Adjustment bit-2 (default high).
P23TXD3	D10	I/O*	Transmit data bit 3 CLKADJ REG Bit-3: ARL Clock Timing Adjustment bit-3 (default high).

PHY Management Interface Signals

Pin Name	Location	I/O	Description
MDC	AD04	O	PHY management clock (1.25MHz)

MDIO	AC03	I/O	PHY management data

CPU Interface Signals

Name	Type		Description
UARTDI	U03	I	CPU data input
UARTDO	T03	O	CPU data output
SWIRQ	C06	O	CPU interrupt request

ZBT SRAM Interface

Pin Name	Location	I/O	Description
DATA00	F01	I/O	Memory data bus
DATA01	F02		
DATA02	G01		
DATA03	G02		
DATA04	H01		
DATA05	H02		
DATA06	J01		
DATA07	J02		
DATA08	K01		
DATA09	K02		
DATA10	L01		
DATA11	L02		
DATA12	M01		
DATA13	M02		
DATA14	N01		
DATA15	N02		
DATA16	P01		
DATA17	P02		
DATA18	R01		
DATA19	R02		
DATA20	T01		
DATA21	T02		
DATA22	U01		
DATA23	U02		
DATA24	V01		
DATA25	V02		
DATA26	W01		
DATA27	W02		
DATA28	Y01		
DATA29	Y02		
DATA30	AA01		
DATA31	AA02		
DATA32	AB01		
DATA33	AB02		
DATA34	AC01		
DATA35	AC02		
DATA36	AD01		
DATA37	AD02		
DATA38	AF02		
DATA39	AF03		
DATA40	AE03		
DATA41	AF04		
DATA42	AE04		
DATA43	AF05		
DATA44	AE05		
DATA45	AF06		
DATA46	AE06		
DATA47			

	AF07		
BE00	AF08	I/O	Byte enable
BE01	AE08	I/O	Byte enable
BE02	AF09	I/O	Byte enable
EOF	AE07	I/O	End of frame

ADDR00 ADDR01 ADDR02 ADDR03 ADDR04 ADDR05 ADDR06 ADDR07 ADDR08 ADDR09 ADDR10 ADDR11 ADDR12 ADDR13 ADDR14 ADDR15 ADDR16 ADDR17 ADDR18	B03 B01 C02 C01 D02 D01 E01 E02 B07 A07 A03 B04 A04 B05 A05 B06 A06 B08 A08	O	Memory address bus
nWE	C04	O	Write enable: 0=Write 1=Read
nCS	D03	O	Chip enable, low active

ARL & MIB Interfaces Signals

Pin Name	Location	I/O	Description
SWDIR00 SWDIR01	K03 L03	O	Data Direction Indicator : 00 = idle, 01 = receive, 10 = transmit, 11 = control
SWSYNC	E03	O	Port synchronization: indicating when Port-0 is driving DAT[47:0]
SWSTAT00 SWSTAT01 SWSTAT02 SWSTAT03	A14 A13 B13 A12	O	Data state indicator SWSTAT[3:0]: 0000 – Idle 0001 – First word (DA) 0010 – Second word (SA) 0011 – Third through last word 0100 – Filter Event 0101 – Drop Event 0110 – Jabber 0111 – False Carrier (receive) - Deferred Transmission (transmit) 1000 – Alignment error (receive) - Single Collision (transmit) 1001 – Flow Control (receive) - Multiple Collision (transmit) 1010 – Short Event (receive) - Excessive Collision (transmit) 1011 – Runt (receive) - Late Collision (transmit) 1100 – Symbol Error 1101 – FCS Error 1110 – Long Event 1111 – Reserved
SWRXCLK	A09	O	Receive Clock, used by ACD80800 and ACD80900
SWTXCLK	A11	O	Transmit Clock, used by ACD80900 only
ARLDI00 ARLDI01	G04 F03	I	ARL Data Input used by ACD80800 only: Returns 12-bit ARL look-up result to the switch:

ARLDI02 ARLDI03	H03 K04		• Bit[11:7] - Source Port ID (0 - 23) • Bit[6:5] – Look-up Result: 00 – reserved 01 – matched 10 – not matched 11 – forced discard • Bit[4:0] - Destination Port ID (0 - 23)
ARLDIV	J04	I	ARL Data Input Valid: Assertion to indicate start of a new result on ARLDI[3:0]. Used by ACD80900 only.

LED Interface Signals

Pin Name	Location	I/O	Description
LEDVLD0	W03	I/O*	LED Signal Valid #0 (default low) POS REG Bit-13: Internal ARL Selected (2K MAC Address Entry)/ External ARL Selected (11K MAC Address Entry).
LEDVLD1	Y04	I/O*	LED Signal Valid #1 (default low) POS REG Bit-14: PHY IDs Start From 0 or 1
LEDCLK	V03	I/O*	2.5 MHz LED Clock POS REG Bit-12: Frames With Unknown DA Forwarded To The Dumping Port OR all ports (default low).
nLED0	R04	I/O*	When LEDVLD1 Is High, It's Frame Error Indicator. POS REG Bit-18: System errors will trigger software/hardware reset. (default low)
nLED1	R03	I/O*	When LEDVLD1 Is High, It's full duplex Indication. When LEDVLD1 Is High, It's Collision Indication. POS REG Bit-17: Flow Control on broadcast queue utilization enable/disable. (default low)
nLED2	N03	I/O*	When LEDVLD0 Is High, It's Speed (1:10Mbps, 0: 100Mbps). When LEDVLD1 Is High, It's Receiving Activity. POS REG Bit-16: Automatic PHY Management Enabled/Disabled. (default low)
nLED3	P04	I/O*	When LEDVLD0 Is High, It's Link Status. When LEDVLD0 Is High, It's Transmit Activity. POS REG Bit-15: Re-Transmit After Excessive Collision/Drop after Excessive Collision. (default low)

System Control interface Signals

Pin Name	Location	I/O	Description
CLK100	AB04	I	100 MHz Main Clock input
WCHDOG	Y03	O	Watch dog life pulse. 2.5Mhz continuous clock.
SYSERROR	M03	O	System error indication high active. When there is any error occurs in SYSERR REG, it's asserted to HIGH.
nRESET	AB03	I	Hardware reset, low active
TESTEN	D07	I	Factory used only. Connect to Ground.
CLKSEL	D20	I	Factory used only. Connect to Ground.
EN16P	AC05	I	Factory used only. Connect to Ground.
ZBTCLK	P03	O	Factory used only. Not Connect

Table-8.1a: Pin List Sorted by Location (1 of 3)

Location	Pin Name	I/O	POS	Location	Pin Name	I/O	POS
A01	VSS			C14	VDDQ		
A02	VSS			C15	VDD		
A03	ADDR10	O		C16	P23RXER	I	
A04	ADDR12	O		C17	P23RXD1	I	
A05	ADDR14	O		C18	P22RXD0	I	
A06	ADDR16	O		C19	P22TXEN	O	
A07	ADDR09	O		C20	P21CRS_DV	I	
A08	ADDR18	O		C21	P20RXD1	I	
A09	SWRXCLK	O		C22	VDDQ		
A10	PID2	O		C23	MIICLK5	O	
A11	SWTXCLK	O		C24	VSS		
A12	STAT3	O		C25	P18RXD1	I	
A13	STAT1	O		C26	P19TXD0	O	
A14	STAT0	O		D01	ADDR05	O	
A15	P23RXCLK	I		D02	ADDR04	O	
A16	P23RXD0	I		D03	nCS	O	
A17	P22RXD1	I		D04	VSS		
A18	P22TXD1	O		D05	VDDQ		
A19	P21RXD1	I		D06	VDDQ		
A20	P21TXD1	I/O	3	D07	TESTEN	I	
A21	P21TXEN	I/O	9	D08	P23CRS	I	
A22	P20CRS_DV	I		D09	VSS		
A23	P20TXEN	I/O	8	D10	P23TXD3	I/O	28
A24	P19RXD0	I		D11	VDDQ		
A25	P19TXD1	O		D12	P23TXD1	I/O	26
A26	VSS			D13	P23TXEN	I/O	22
B01	ADDR01	O		D14	VSS		
B02	VSS			D15	P23TXCLK	O	
B03	ADDR00	O		D16	VDDQ		
B04	ADDR11	O		D17	P23RXD3	I	
B05	ADDR13	O		D18	P21RXD0	I	
B06	ADDR15	O		D19	VSS		
B07	ADDR08	O		D20	CLKSEL	I	
B08	ADDR17	O		D21	VDDQ		
B09	PID4	O		D22	P20TXD1	I/O	1
B10	PID3	O		D23	VSS		
B11	PID1	O		D24	P19TXEN	O	
B12	PID0	O		D25	P18TXD1	I/O	7
B13	STAT2	O		D26	P18RXD0	I	
B14	VDDQ			E01	ADDR06	O	
B15	P23RXDV	I		E02	ADDR07	O	
B16	P23RXD2	I		E03	SWSYNC	O	
B17	P22CRS_DV	I		E04	VDDQ		
B18	P22TXD0	O		E23	P18CRS_DV	I	
B19	VDD			E24	VDD		
B20	P21TXD0	I/O	2	E25	P17RXD1	I	
B21	P20RXD0	I		E26	P18TXEN	I/O	11
B22	P20TXD0	I/O	0	F01	DATA00	I/O	
B23	P19RXD1	I		F02	DATA01	I/O	
B24	P19CRS_DV	I		F03	ARLDI1	I	
B25	VSS			F04	VDDQ		
B26	VSS			F23	VDDQ		
C01	ADDR03	O		F24	P18TXD0	I/O	6
C02	ADDR02	O		F25	P17TXD0	I/O	4
C03	VSS			F26	P17CRS_DV	I	
C04	new	O		G01	DATA02	I/O	
C05	VDD			G02	DATA03	I/O	
C06	SWIRQ	O		G03	VDD		
C07	VDDQ			G04	ARLDI0	I	
C08	VDDQ			G23	P17TXD1	I/O	5
C09	VDD			G24	P17RXD0	I	
C10	P23COL	I		G25	P16CRS_DV	I	
C11	P23TXD2	I/O	27	G26	P16RXD1	I	
C12	VDDQ			H01	DATA04	I/O	
C13	P23TXD0	I/O	25	H02	DATA05	I/O	

Table-8.1b: Pin List Sorted by Location (2 of 3)

Location	Pin Name	I/O	POS	Location	Pin Name	I/O	POS
H03	ARLDI2	I		P02	DATA17	I/O	
H04	VSS			P03	ZBTCLK	O	
H23	P16RXD0	I		P04	nLED3	I/O	15
H24	P17TXEN	I/O	10	P11	VSS		
H25	VDDQ			P12	VSS		
H26	P16TXD0	O		P13	VSS		
J01	DATA06	I/O		P14	VSS		
J02	DATA07	I/O		P15	VSS		
J03	VDDQ			P16	VSS		
J04	ARLDIV	I		P23	VSS		
J23	VSS			P24	P14TXEN	I/O	21
J24	P16TXD1	O		P25	P13TXEN	O	
J25	MIICLK4	O		P26	P13TXD1	O	
J26	VDD			R01	DATA18	I/O	
K01	DATA08	I/O		R02	DATA19	I/O	
K02	DATA09	I/O		R03	nLED1	I/O	17
K03	SWDIR0	O		R04	nLED0	I/O	18
K04	ARLDI3	I		R11	VSS		
K23	P15RXD1	I		R12	VSS		
K24	P16TXEN	O		R13	VSS		
K25	P15TXD1	O		R14	VSS		
K26	P15RXD0	I		R15	VSS		
L01	DATA10	I/O		R16	VSS		
L02	DATA11	I/O		R23	P13TXD0	O	
L03	SWDIR1	O		R24	P13CRS_DV	I	
L04	VDDQ			R25	P12RXD0	I	
L11	VSS			R26	P12RXD1	I	
L12	VSS			T01	DATA20	I/O	
L13	VSS			T02	DATA21	I/O	
L14	VSS			T03	UARTDO	O	
L15	VSS			T04	VDDQ		
L16	VSS			T11	VSS		
L23	VDDQ			T12	VSS		
L24	P15CRS_DV	I		T13	VSS		
L25	P14RXD0	I		T14	VSS		
L26	P15TXEN	I/O	19	T15	VSS		
M01	DATA12	I/O		T16	VSS		
M02	DATA13	I/O		T23	VDDQ		
M03	SYSERR	O		T24	VDDQ		
M04	VDD			T25	P12TXD1	O	
M11	VSS			T26	VDD		
M12	VSS			U01	DATA22	I/O	
M13	VSS			U02	DATA23	I/O	
M14	VSS			U03	UARTDI	I	
M15	VSS			U04	VDD		
M16	VSS			U23	P12TXD0	O	
M23	P15TXD0	O		U24	P12CRS_DV	I	
M24	P14RXD1	I		U25	P11RXD1	I	
M25	P14TXD0	O		U26	P12TXEN	I/O	20
M26	P14CRS_DV	I		V01	DATA24	I/O	
N01	DATA14	I/O		V02	DATA25	I/O	
N02	DATA15	I/O		V03	nLEDCLK	I/O	12
N03	nLED2	I/O	16	V04	VSS		
N04	VSS			V23	P11TXEN	I/O	23
N11	VSS			V24	MIICLK3	O	
N12	VSS			V25	P11CRS_DV	I	
N13	VSS			V26	P11RXD0	I	
N14	VSS			W01	DATA26	I/O	
N15	VSS			W02	DATA27	I/O	
N16	VSS			W03	LEDVLD0	I/O	13
N23	P13RXD0	I		W04	VDDQ		
N24	P14TXD1	O		W23	VSS		
N25	VDD			W24	P11TXD1	O	
N26	P13RXD1	I		W25	P10RXD1	I	
P01	DATA16	I/O		W26	P11TXD0	O	

Table-8.1c: Pin List Sorted by Location (3 of 3)

Location	Pin Name	I/O	POS	Location	Pin Name	I/O	POS
Y01	DATA28	I/O		AD16	P04TXD0	O	
Y02	DATA29	I/O		AD17	P04RXD1	I	
Y03	WCHDOG	O		AD18	P05CRS_DV	I	
Y04	LEDVLD1	I/O	14	AD19	P06TXD0	O	
Y23	P09RXD1	I		AD20	P06RXD0	I	
Y24	P10RXD0	I		AD21	P07TXD0	O	
Y25	P10TXD1	O		AD22	P07RXD1	I	
Y26	P10CRS_DV	I		AD23	P08TXEN	I/O	31
AA01	DATA30	I/O		AD24	VSS		
AA02	DATA31	I/O		AD25	P08CRS_DV	I	
AA03	VDD			AD26	P08RXD0	I	
AA04	VDDQ			AE01	VSS		
AA23	VDDQ			AE02	VSS		
AA24	P10TXEN	O		AE03	DATA40	I/O	
AA25	VDD			AE04	DATA42	I/O	
AA26	P10TXD0	O		AE05	DATA44	I/O	
AB01	DATA32	I/O		AE06	DATA46	I/O	
AB02	DATA33	I/O		AE07	EOF	I/O	
AB03	nRESET	I		AE08	BE1	I/O	
AB04	CLK100	I		AE09	VDDQ		
AB23	P09CRS_DV	I		AE10	P00CRS_DV	I	
AB24	P09RXD0	I		AE11	P01TXEN	O	
AB25	P09TXD1	O		AE12	P01CRS_DV	I	
AB26	VDDQ			AE13	P02TXEN	I/O	42
AC01	DATA34	I/O		AE14	P02RXD0	I	
AC02	DATA35	I/O		AE15	P03TXD1	I/O	38
AC03	MDIO	I/O		AE16	P03RXD0	I	
AC04	VSS			AE17	VDDQ		
AC05	EN16P	I		AE18	P04RXD0	I	
AC06	VDDQ			AE19	P05TXD0	I/O	39
AC07	MIICLK0	O		AE20	P05RXD0	I	
AC08	VSS			AE21	P06TXD1	O	
AC09	P00TXD0	I/O	33	AE22	P06RXD1	I	
AC10	P00RXD1	I		AE23	P07TXD1	O	
AC11	VDDQ			AE24	MIICLK2	O	
AC12	P02TXD0	I/O	35	AE25	VSS		
AC13	VSS			AE26	P08TXD1	O	
AC14	P02RXD1	I		AF01	VSS		
AC15	P04TXEN	O		AF02	DATA38	I/O	
AC16	VDDQ			AF03	DATA39	I/O	
AC17	P04CRS_DV	I		AF04	DATA41	I/O	
AC18	VSS			AF05	DATA43	I/O	
AC19	P05RXD1	I		AF06	DATA45	I/O	
AC20	VDD			AF07	DATA47	I/O	
AC21	VDDQ			AF08	BE0	I/O	
AC22	P07CRS_DV	I		AF09	BE2	I/O	
AC23	VSS			AF10	P00RXD0	I	
AC24	P09TXEN	I/O	32	AF11	P01TXD1	O	
AC25	P08RXD1	I		AF12	P01RXD0	I	
AC26	P09TXD0	O		AF13	P02TXD1	I/O	36
AD01	DATA36	I/O		AF14	P03TXEN	I/O	24
AD02	DATA37	I/O		AF15	VDD		
AD03	VSS			AF16	MIICLK1	O	
AD04	MDC	O		AF17	P04TXD1	O	
AD05	VDD			AF18	P05TXEN	I/O	29
AD06	VDDQ			AF19	P05TXD1	I/O	40
AD07	P00TXEN	I/O	41	AF20	P06TXEN	I/O	30
AD08	P00TXD1	I/O	34	AF21	P06CRS_DV	I	
AD09	VDD			AF22	P07TXEN	O	
AD10	P01TXD0	O		AF23	P07RXD0	I	
AD11	P01RXD1	I		AF24	P08TXD0	O	
AD12	P02CRS_DV	I		AF25	VSS		
AD13	P03TXD0	I/O	37	AF26	VSS		
AD14	P03CRS_DV	I					
AD15	P03RXD1	I					

Table-8.2a: Pin List Sorted by Name (1 of 3)

Pin Name	I/O	POS	Location	Pin Name	I/O	POS	Location
ADDR00	O		B03	DATA36	I/O		AD01
ADDR01	O		B01	DATA37	I/O		AD02
ADDR02	O		C02	DATA38	I/O		AF02
ADDR03	O		C01	DATA39	I/O		AF03
ADDR04	O		D02	DATA40	I/O		AE03
ADDR05	O		D01	DATA41	I/O		AF04
ADDR06	O		E01	DATA42	I/O		AE04
ADDR07	O		E02	DATA43	I/O		AF05
ADDR08	O		B07	DATA44	I/O		AE05
ADDR09	O		A07	DATA45	I/O		AF06
ADDR10	O		A03	DATA46	I/O		AE06
ADDR11	O		B04	DATA47	I/O		AF07
ADDR12	O		A04	EN16P	I		AC05
ADDR13	O		B05	EOF	I/O		AE07
ADDR14	O		A05	LEDVLD0	I/O	13	W03
ADDR15	O		B06	LEDVLD1	I/O	14	Y04
ADDR16	O		A06	MDC	O		AD04
ADDR17	O		B08	MDIO	I/O		AC03
ADDR18	O		A08	MIICLK0	O		AC07
ARLDI0	I		G04	MIICLK1	O		AF16
ARLDI1	I		F03	MIICLK2	O		AE24
ARLDI2	I		H03	MIICLK3	O		V24
ARLDI3	I		K04	MIICLK4	O		J25
ARLDIV	I		J04	MIICLK5	O		C23
CLKSEL	I		D20	nCS	O		D03
BE0	I/O		AF08	nLED0	I/O	18	R04
BE1	I/O		AE08	nLED1	I/O	17	R03
BE2	I/O		AF09	nLED2	I/O	16	N03
CLK100	I		AB04	nLED3	I/O	15	P04
DATA00	I/O		F01	nLEDCLK	I/O	12	V03
DATA01	I/O		F02	nRESET	I		AB03
DATA02	I/O		G01	nWE	O		C04
DATA03	I/O		G02	P00CRS_DV	I		AE10
DATA04	I/O		H01	P00RXD0	I		AF10
DATA05	I/O		H02	P00RXD1	I		AC10
DATA06	I/O		J01	P00TXD0	I/O	33	AC09
DATA07	I/O		J02	P00TXD1	I/O	44	AD08
DATA08	I/O		K01	P00TXEN	I/O	41	AD07
DATA09	I/O		K02	P01CRS_DV	I		AE12
DATA10	I/O		L01	P01RXD0	I		AF12
DATA11	I/O		L02	P01RXD1	I		AD11
DATA12	I/O		M01	P01TXD0	O		AD10
DATA13	I/O		M02	P01TXD1	O		AF11
DATA14	I/O		N01	P01TXEN	O		AE11
DATA15	I/O		N02	P02CRS_DV	I		AD12
DATA16	I/O		P01	P02RXD0	I		AE14
DATA17	I/O		P02	P02RXD1	I		AC14
DATA18	I/O		R01	P02TXD0	I/O	35	AC12
DATA19	I/O		R02	P02TXD1	I/O	36	AF13
DATA20	I/O		T01	P02TXEN	I/O	42	AE13
DATA21	I/O		T02	P03CRS_DV	I		AD14
DATA22	I/O		U01	P03RXD0	I		AE16
DATA23	I/O		U02	P03RXD1	I		AD15
DATA24	I/O		V01	P03TXD0	I/O	37	AD13
DATA25	I/O		V02	P03TXD1	I/O	38	AE15
DATA26	I/O		W01	P03TXEN	I/O	24	AF14
DATA27	I/O		W02	P04CRS_DV	I		AC17
DATA28	I/O		Y01	P04RXD0	I		AE18
DATA29	I/O		Y02	P04RXD1	I		AD17
DATA30	I/O		AA01	P04TXD0	O		AD16
DATA31	I/O		AA02	P04TXD1	O		AF17
DATA32	I/O		AB01	P04TXEN	O		AC15
DATA33	I/O		AB02	P05CRS_DV	I		AD18
DATA34	I/O		AC01	P05RXD0	I		AE20
DATA35	I/O		AC02	P05RXD1	I		AC19

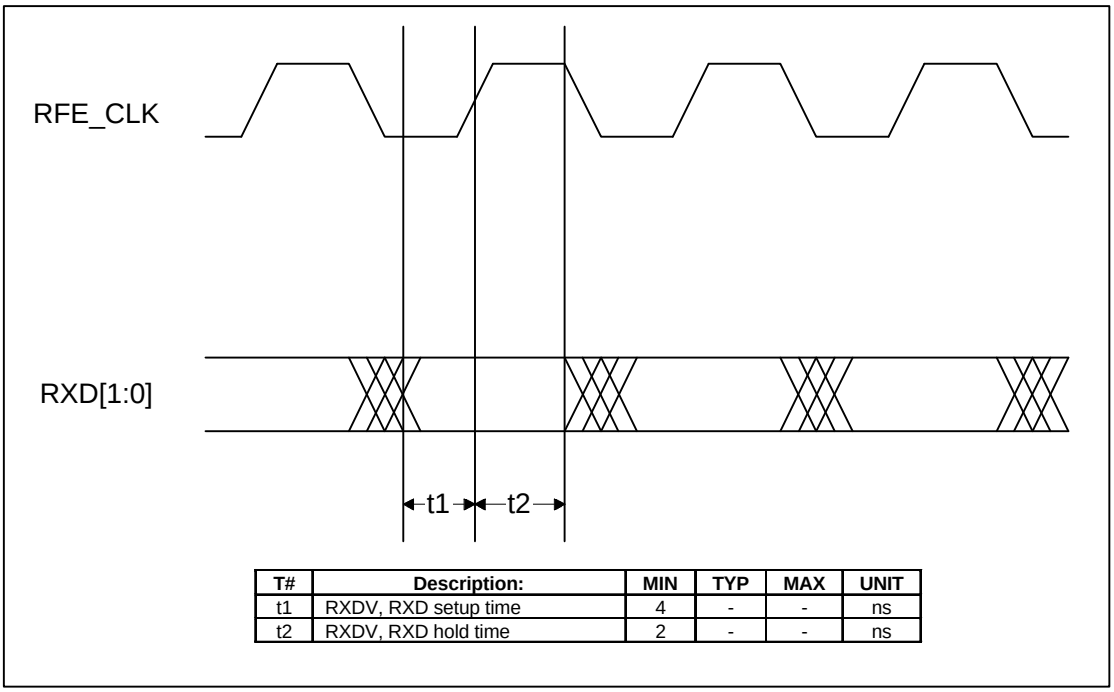
Table-8.2b: Pin List Sorted by Name (2 of 3)

Pin Name	I/O	POS	Location	Pin Name	I/O	POS	Location
P05TXD0	I/O	39	AE19	P16RXD1	I		G26
P05TXD1	I/O	40	AF19	P16TXD0	O		H26
P05TXEN	I/O	29	AF18	P16TXD1	O		J24
P06CRS_DV	I		AF21	P16TXEN	O		K24
P06RXD0	I		AD20	P17CRS_DV	I		F26
P06RXD1	I		AE22	P17RXD0	I		G24
P06TXD0	O		AD19	P17RXD1	I		E25
P06TXD1	O		AE21	P17TXD0	I/O	4	F25
P06TXEN	I/O	30	AF20	P17TXD1	I/O	5	G23
P07CRS_DV	I		AC22	P17TXEN	I/O	10	H24
P07RXD0	I		AF23	P18CRS_DV	I		E23
P07RXD1	I		AD22	P18RXD0	I		D26
P07TXD0	O		AD21	P18RXD1	I		C25
P07TXD1	O		AE23	P18TXD0	I/O	6	F24
P07TXEN	O		AF22	P18TXD1	I/O	7	D25
P08CRS_DV	I		AD25	P18TXEN	I/O	11	E26
P08RXD0	I		AD26	P19CRS_DV	I		B24
P08RXD1	I		AC25	P19RXD0	I		A24
P08TXD0	O		AF24	P19RXD1	I		B23
P08TXD1	O		AE26	P19TXD0	O		C26
P08TXEN	I/O	31	AD23	P19TXD1	O		A25
P09CRS_DV	I		AB23	P19TXEN	O		D24
P09RXD0	I		AB24	P20CRS_DV	I		A22
P09RXD1	I		Y23	P20RXD0	I		B21
P09TXD0	O		AC26	P20RXD1	I		C21
P09TXD1	O		AB25	P20TXD0	I/O	0	B22
P09TXEN	I/O	32	AC24	P20TXD1	I/O	1	D22
P10CRS_DV	I		Y26	P20TXEN	I/O	8	A23
P10RXD0	I		Y24	P21CRS_DV	I		C20
P10RXD1	I		W25	P21RXD0	I		D18
P10TXD0	O		AA26	P21RXD1	I		A19
P10TXD1	O		Y25	P21TXD0	I/O	2	B20
P10TXEN	O		AA24	P21TXD1	I/O	3	A20
P11CRS_DV	I		V25	P21TXEN	I/O	9	A21
P11RXD0	I		V26	P22CRS_DV	I		B17
P11RXD1	I		U25	P22RXD0	I		C18
P11TXD0	O		W26	P22RXD1	I		A17
P11TXD1	O		W24	P22TXD0	O		B18
P11TXEN	I/O	23	V23	P22TXD1	O		A18
P12CRS_DV	I		U24	P22TXEN	O		C19
P12RXD0	I		R25	P23COL	I		C10
P12RXD1	I		R26	P23CRS-DV/CRS	I		D08
P12TXD0	O		U23	P23RXCLK	I		A15
P12TXD1	O		T25	P23RXD0	I		A16
P12TXEN	I/O	20	U26	P23RXD1	I		C17
P13CRS_DV	I		R24	P23RXD2	I		B16
P13RXD0	I		N23	P23RXD3	I		D17
P13RXD1	I		N26	P23RXDV	I		B15
P13TXD0	O		R23	P23RXER	I		C16
P13TXD1	O		P26	P23TXCLK	O		D15
P13TXEN	O		P25	P23TXD0	I/O	25	C13
P14CRS_DV	I		M26	P23TXD1	I/O	26	D12
P14RXD0	I		L25	P23TXD2	I/O	27	C11
P14RXD1	I		M24	P23TXD3	I/O	28	D10
P14TXD0	O		M25	P23TXEN	I/O	22	D13
P14TXD1	O		N24	PID0	O		B12
P14TXEN	I/O	21	P24	PID1	O		B11
P15CRS_DV	I		L24	PID2	O		A10
P15RXD0	I		K26	PID3	O		B10
P15RXD1	I		K23	PID4	O		B09
P15TXD0	O		M23	STAT0	O		A14
P15TXD1	O		K25	STAT1	O		A13
P15TXEN	I/O	19	L26	STAT2	O		B13
P16CRS_DV	I		G25	STAT3	O		A12
P16RXD0	I		H23	SWDIR0	O		K03

Table-8.2c: Pin List Sorted by Name (3 of 3)

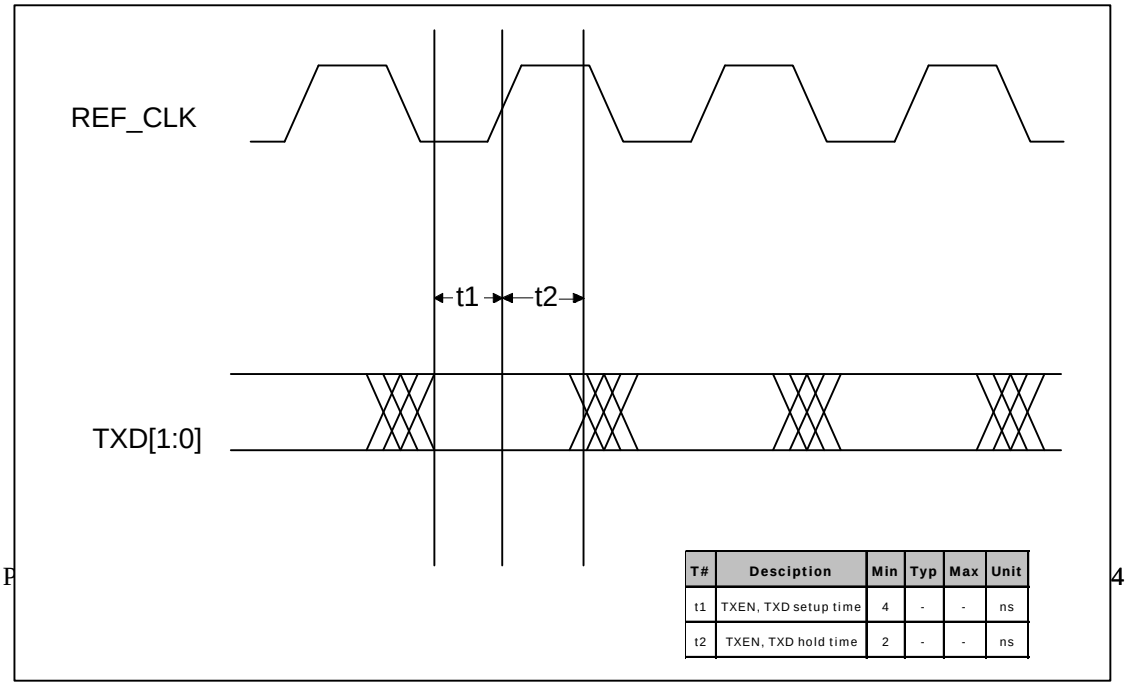
Pin Name	I/O	POS	Location	Pin Name	I/O	POS	Location
SWIRQ	O		C06	VSS			AC23
SWRXCLK	O		A09	VSS			AD03
SWSYNC	O		E03	VSS			AD24
SWTXCLK	O		A11	VSS			AE01
SYSERR	O		M03	VSS			AE02
TESTEN	I		D07	VSS			AE25
UARTDI	I		U03	VSS			AF01
UARTDO	O		T03	VSS			AF25
VDD			AA03	VSS			AF26
VDD			AA25	VSS			B02
VDD			AC20	VSS			B25
VDD			AD05	VSS			B26
VDD			AD09	VSS			C03
VDD			AF15	VSS			C24
VDD			B19	VSS			D04
VDD			C05	VSS			D09
VDD			C09	VSS			D14
VDD			C15	VSS			D19
VDD			E24	VSS			D23
VDD			G03	VSS			H04
VDD			J26	VSS			J23
VDD			M04	VSS			L11
VDD			N25	VSS			L12
VDD			T26	VSS			L13
VDD			U04	VSS			L14
VDDQ			AA04	VSS			L15
VDDQ			AA23	VSS			L16
VDDQ			AB26	VSS			M11
VDDQ			AC06	VSS			M12
VDDQ			AC11	VSS			M13
VDDQ			AC16	VSS			M14
VDDQ			AC21	VSS			M15
VDDQ			AD06	VSS			M16
VDDQ			AE09	VSS			N04
VDDQ			AE17	VSS			N11
VDDQ			B14	VSS			N12
VDDQ			C07	VSS			N13
VDDQ			C08	VSS			N14
VDDQ			C12	VSS			N15
VDDQ			C14	VSS			N16
VDDQ			C22	VSS			P11
VDDQ			D05	VSS			P12
VDDQ			D06	VSS			P13
VDDQ			D11	VSS			P14
VDDQ			D16	VSS			P15
VDDQ			D21	VSS			P16
VDDQ			E04	VSS			P23
VDDQ			F04	VSS			R11
VDDQ			F23	VSS			R12
VDDQ			H25	VSS			R13
VDDQ			J03	VSS			R14
VDDQ			L04	VSS			R15
VDDQ			L23	VSS			R16
ZBTCLK	o		P03	VSS			T11
VDDQ			T04	VSS			T12
VDDQ			T23	VSS			T13
VDDQ			T24	VSS			T14
VDDQ			W04	VSS			T15
VSS			A01	VSS			T16
VSS			A02	VSS			V04
VSS			A26	VSS			W23
VSS			AC04	WCHDOG	O		Y03
VSS			AC08				
VSS			AC13				

9. TIMING DESCRIPTION



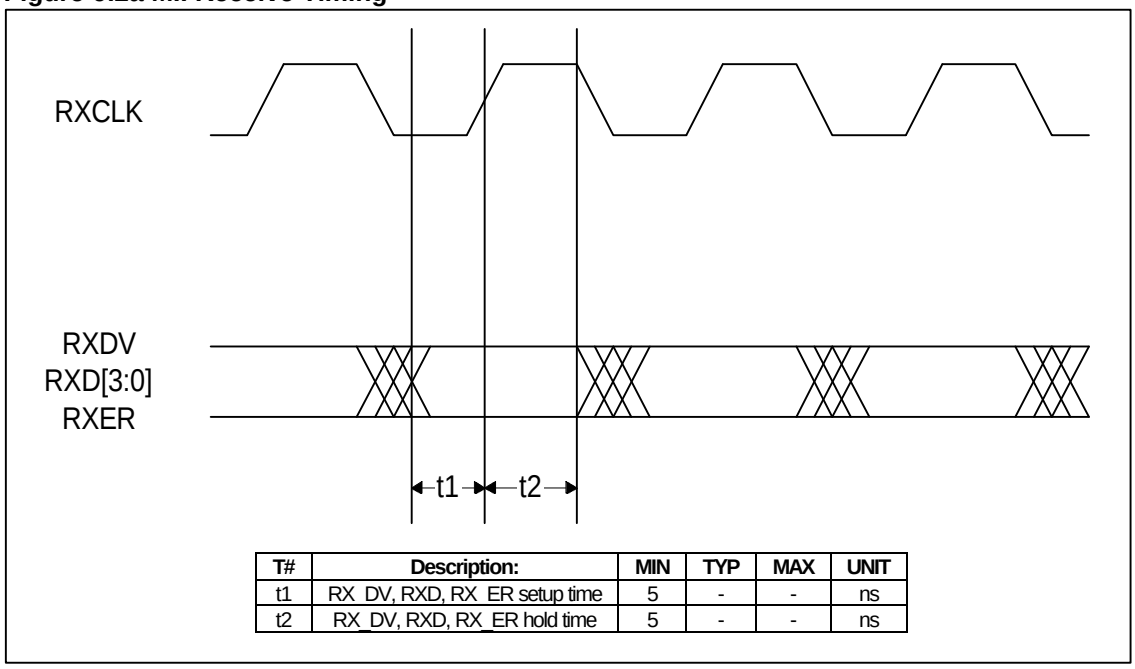
T#	Description:	MIN	TYP	MAX	UNIT
t1	RXDV, RXD setup time	4	-	-	ns
t2	RXDV, RXD hold time	2	-	-	ns

Figure-9.1a RMII Receive Timing
Figure-9.1b: RMII Transmit Timing



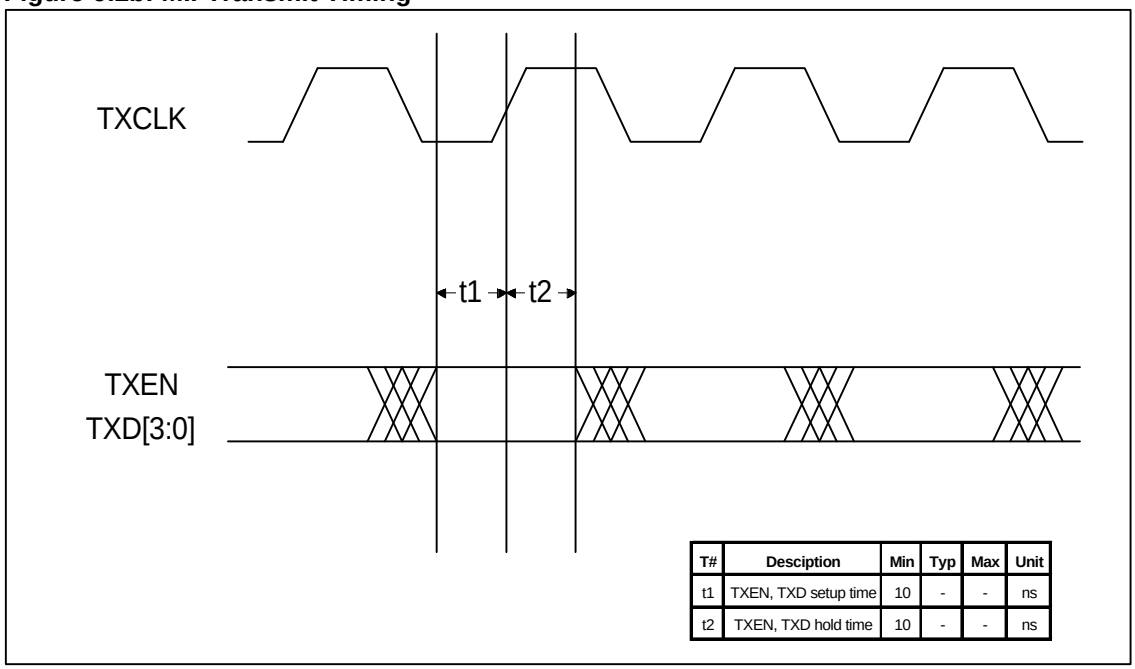
T#	Desciption	Min	Typ	Max	Unit
t1	TXEN, TXD setup time	4	-	-	ns
t2	TXEN, TXD hold time	2	-	-	ns

Figure-9.2a MII Receive Timing



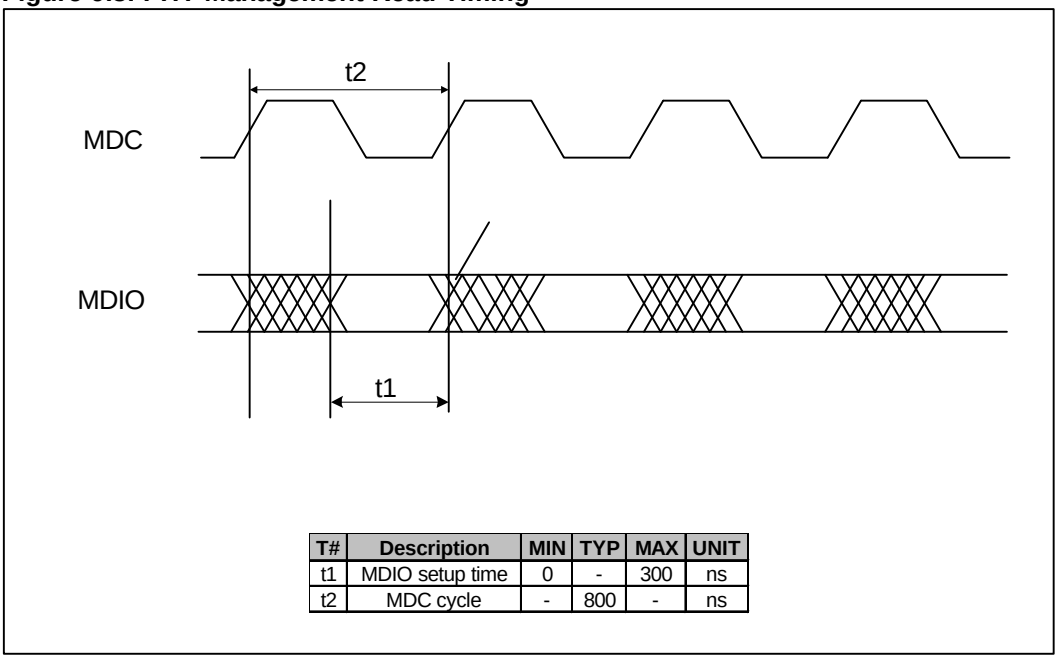
T#	Description:	MIN	TYP	MAX	UNIT
t1	RX_DV, RXD, RX_ER setup time	5	-	-	ns
t2	RX_DV, RXD, RX_ER hold time	5	-	-	ns

Figure-9.2b: MII Transmit Timing



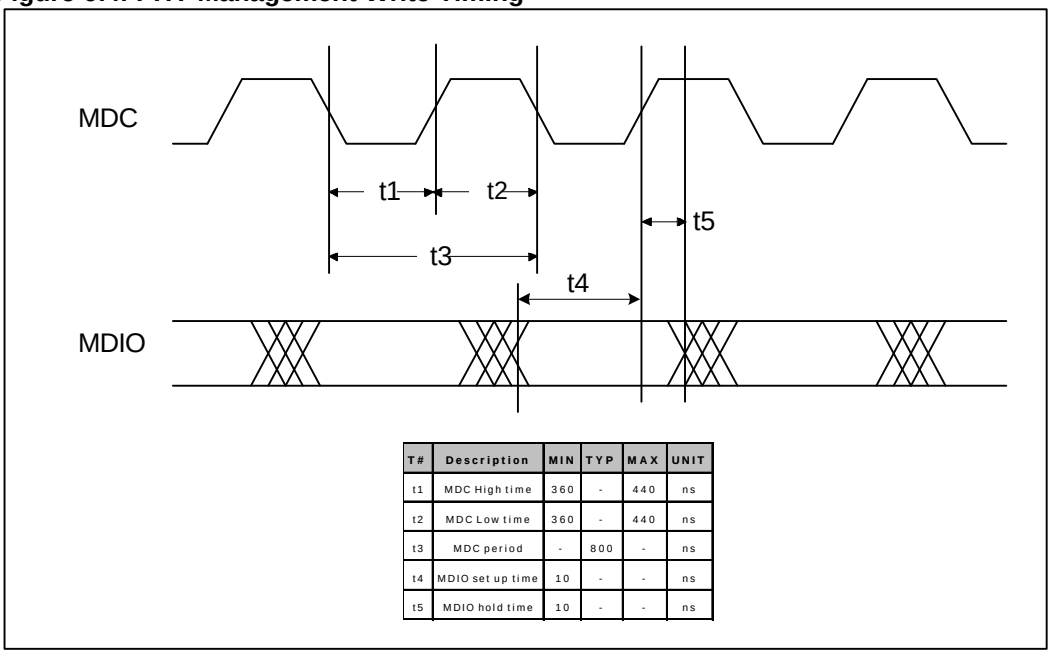
T#	Desciption	Min	Typ	Max	Unit
t1	TXEN, TXD setup time	10	-	-	ns
t2	TXEN, TXD hold time	10	-	-	ns

Figure-9.3: PHY Management Read Timing



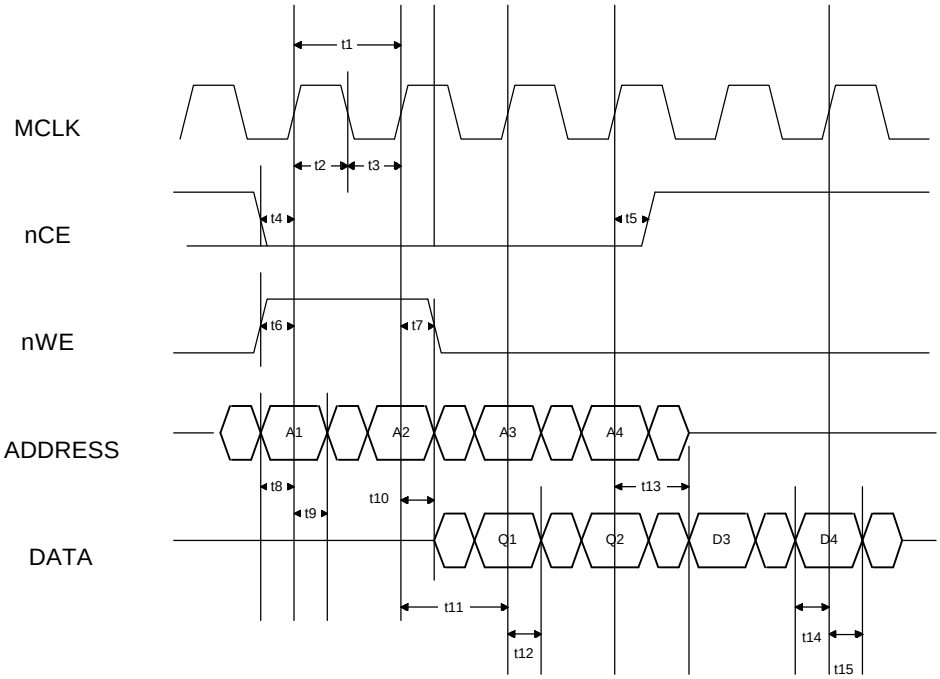
T#	Description	MIN	TYP	MAX	UNIT
t1	MDIO setup time	0	-	300	ns
t2	MDC cycle	-	800	-	ns

Figure-9.4: PHY Management Write Timing



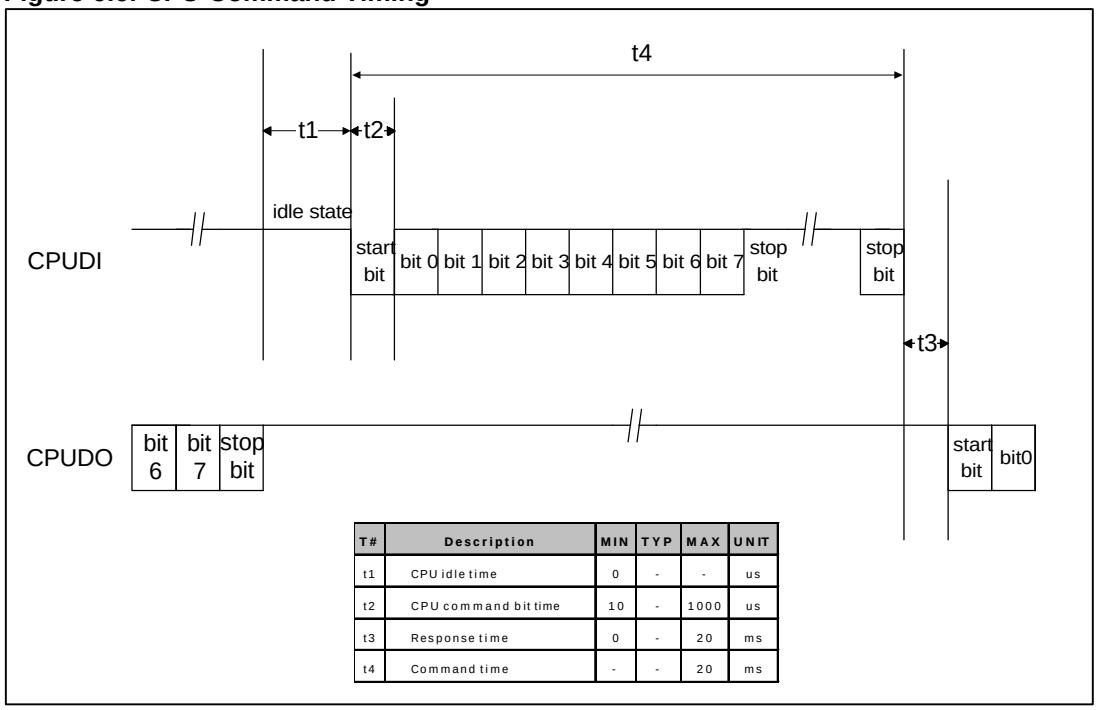
T#	Description	MIN	TYP	MAX	UNIT
t1	MDC High time	360	-	440	ns
t2	MDC Low time	360	-	440	ns
t3	MDC period	-	800	-	ns
t4	MDIO set up time	10	-	-	ns
t5	MDIO hold time	10	-	-	ns

Figure-9.5: SRAM (ZBT) Read/Write Timing



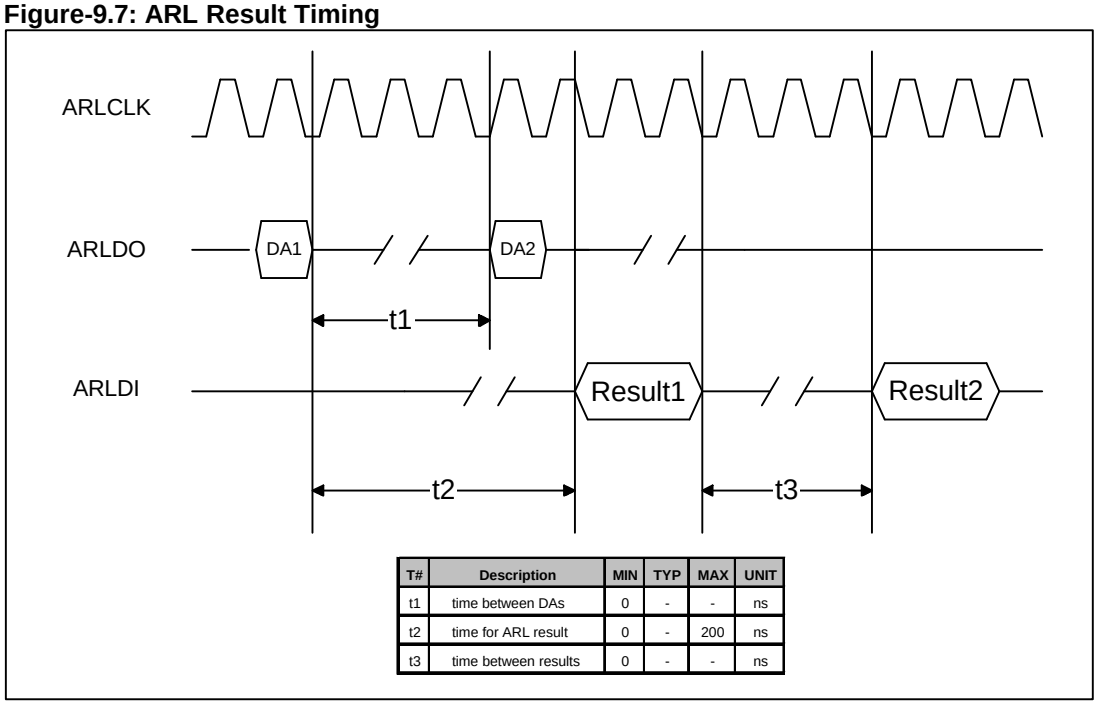
T#	Description:	MIN	MAX	UNIT
t1	Clock cycle time	10	-	ns
t2	Clock HIGH time	4	-	ns
t3	Clock LOW time	4	-	ns
t4	Chip Enable setup time	3	-	ns
t5	Chip Enable hold time	2	-	ns
t6	Read/Write setup time	3	-	ns
t7	Read/Write hold time	2	-	ns
t8	Address setup time	3	-	ns
t9	Address hold time	2	-	ns
t10	Clock to output data in Low-Z	1.5	-	ns
t11	Clock to output data valid	-	5	ns
t12	Clock to output data invalid	1.5	-	ns
t13	Clock to output data in High-Z	-	4	ns
t14	Input data setup time	3	-	ns
t15	Input data hold time	1	-	ns

Figure-9.6: CPU Command Timing



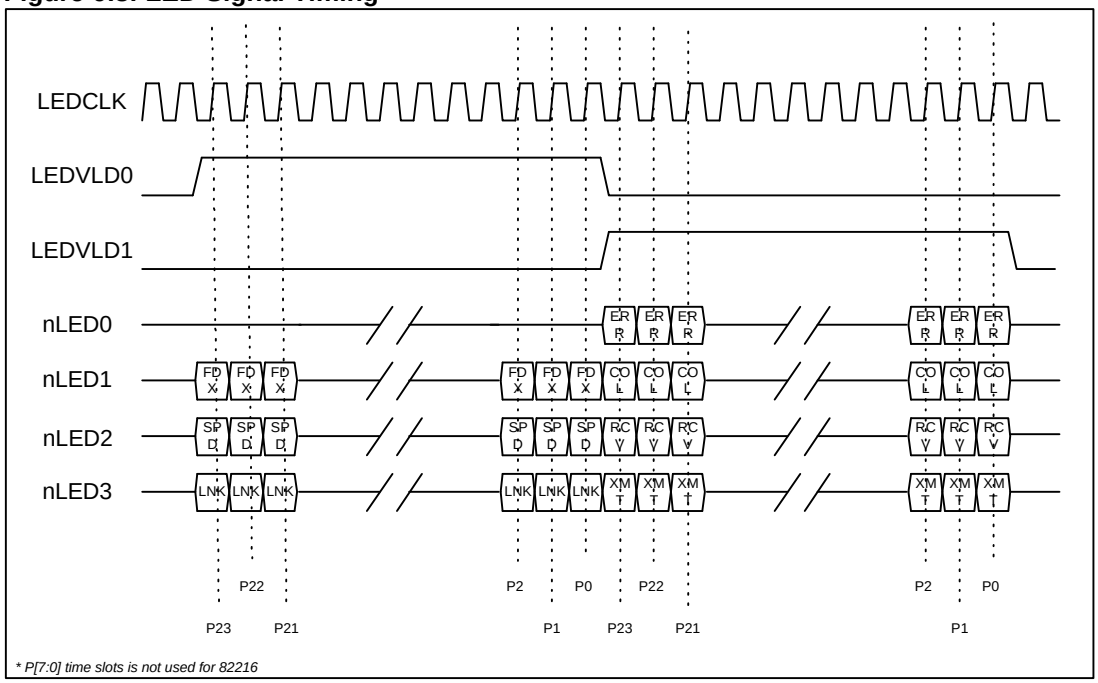
T#	Description	MIN	TYP	MAX	UNIT
t1	CPU idle time	0	-	-	us
t2	CPU command bit time	10	-	1000	us
t3	Response time	0	-	20	ms
t4	Command time	-	-	20	ms

Figure-9.7: ARL Result Timing



T#	Description	MIN	TYP	MAX	UNIT
t1	time between DAs	0	-	-	ns
t2	time for ARL result	0	-	200	ns
t3	time between results	0	-	-	ns

Figure-9.8: LED Signal Timing



** P[7:0] time slots is not used for 82216*

10. ELECTRICAL SPECIFICATION

10. ELECTRICAL SPECIFICATION

Figure-10.1: Absolute Maximum Ratings

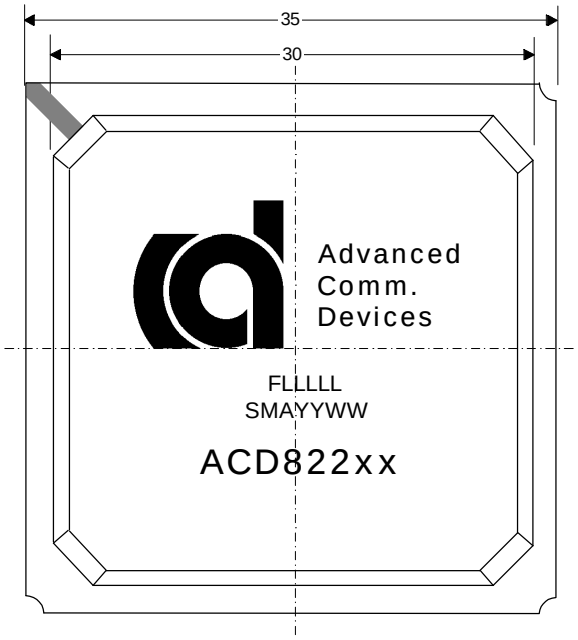
Item	Symbol	Rating
DC supply voltage for Core	VDD	3.0 V
DC supply voltage for I/O	VDDQ	4.0 V
Input signal voltage	V_{in}	3.6 V
Signal current	$I_{i/o}$	± 2.5mA
DC output voltage	Vo	2.8V

Figure-10.2 Recommended Operation Conditions

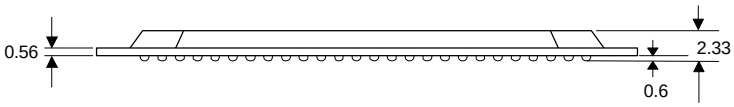
Item	Symbol	Rating
DC supply voltage for Core	VDD	2.5V
DC supply voltage for I/O	VDDQ	3.3V
Operating temperature	T_a	0 – 70 °C
Maximum power consumption	N/A	TBD

11. PACKAGING

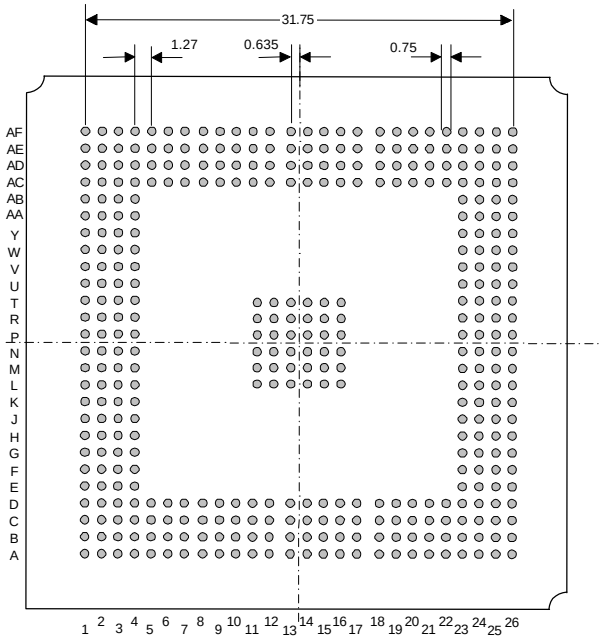
Top View



Side View



Bottom View



Appendix-A1

Address Resolution Logic

Built-in ARL with 2048 MAC Addresses

1. SUMMARY

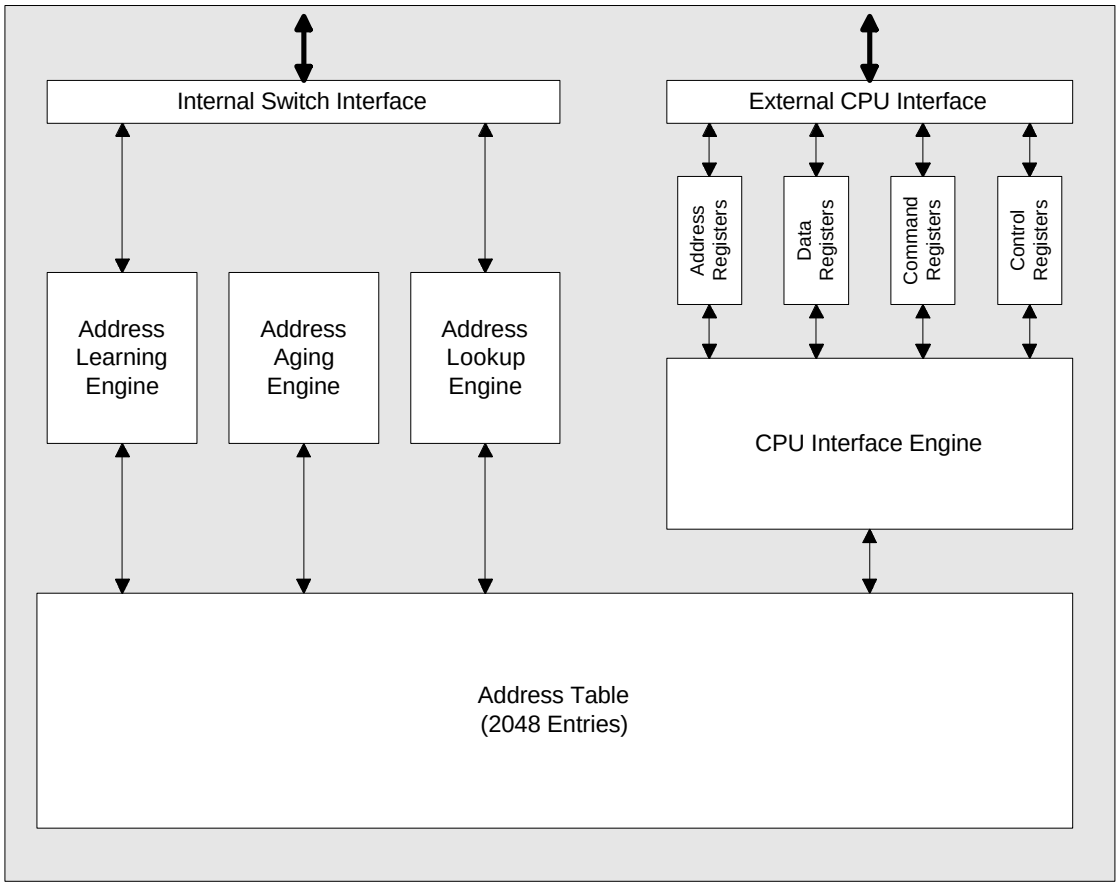
The internal Address Resolution Logic (ARL) of the switch controllers automatically builds up an address table and maps up to 2,048 MAC addresses for the associated ports. CPU intervention is not required in an UN-managed system.

For a managed system, the management CPU can configure the operation mode of the ARL, learn all the addresses in the address table, add new addresses into the lookup table, control security or filtering feature of each address entry etc.

The ARL high performance design guarantee very low latency and will never slow down the frame switching operation. It helps the switch controllers maintain wire speed forwarding rate under any type of traffic load.

The 2K internal addresses space can be expanded to 11K entries by using the external ARL, ACD80800.

Figure-1: Built-in ARL Block Diagram



2. FEATURES

- Supports up to 2,048 internal MAC address lookup
- Provides UART type of interface for management CPU
- Wire speed address lookup time.
- Wire speed address learning time.
- Address can be automatically learned from switch without external CPU intervention
- Address can be manually added by the CPU through CPU interface
- Each MAC address can be secured by the CPU from being changed or aged out
- Each MAC address can be marked by the CPU from receiving any frame
- Each newly learned MAC address is notified to the CPU
- Each aged out MAC address is notified to the CPU
- Automatic address aging control, with configurable aging period

3. FUNCTIONAL DESCRIPTION

The internal ARL provides Address Resolution service for the switch controllers. *Figure-1* is a block diagram of the ARL.

Traffic Snooping

All Ethernet frames received by the switch controller have to be stored into memory buffer. As the frame data are written into memory. The status of the data shown on the data bus is displayed by the switch controller through the SWSTAT[3:0] bus. The ARL interface with the Switch Controller contains the signals of the data bus and the state bus. By snoop the data bus and the state bus of the switch controller, the internal ARL can detect destination MAC address and source MAC address embedded inside each frame.

Address Learning

Each source MAC address extracted from the data bus, along with the ingress port ID, is passed to the Address Learning Engine of the ARL.

1. The Address Learning Engine first determines whether the frame is a valid frame.
2. For a valid frame, it will first try to find the source address from the current address table.
3. If that address is not listed, OR the port ID associated with the listed MAC address does not match the ingress port ID, it will be learned into the address table as a new address.
4. After an address is learned by the address learning engine, the CPU can be notified to read this newly learned address so that it can add it into the CPU's address table.
5. If the Address Table is full, Address Learning Engine will not learn any the new MAC address unless there is new entry available (i.e. Address aging-out).

Address Aging

After each source address is learned into the address table, it has to be refreshed at least once within each address aging period. Refresh means it is caught again from the switch interface. If it has not occurred for a pre-set aging period, the A*ddress Aging Engine* will remove the address from the address table. After an address is removed by the address aging engine, the CPU can be notified through interrupt request that it needs to read this aged out address so that it can remove this address from the CPU's address table.

The default aging time is 300 seconds. That means A*ddress Aging Engine* checks the timer every 300 seconds from power up. If the new address is learned just after the aging-out checking process finished. The worst case aging time can be about 600 seconds. You can program the timer register through CPU interface (UART). The register is resided in Register18 and 19 of ARL.

Address Lookup

Each destination address is passed to the Address Lookup Engine of the ARL. The Address Lookup Engine checks if the destination address matches with any existing address in the address table. If it does, the ARL returns the associated Port ID to switch controller. Otherwise, a "no match" result is passed to switch controller.

CPU Interface

The CPU can access the registers of the ARL by sending commands to the UART data input line. Each command is consists of action (read or write), register type, register index, and data. Each result of command execution is returned to the CPU through the UART data output line.

Registers

The ARL provides a number of registers for the control CPU. Through these registers, the CPU can read all address entries of the address table, delete particular addresses from the table, add particular addresses into the table, secure an address from being changed, set filtering on some addresses, change the hashing algorithm etc. Through interrupt request signals, the CPU can be notified whenever it needs to retrieve data for a newly-learned address or an aged-out address so that the CPU can build an exact same address table learned by the ARL.

CPU Interface Engine

The command sent by the control CPU is executed by the CPU Interface Engine. For example, the CPU may send a command to learn the first newly learned address. The CPU Interface Engine is responsible to find the newly learned address from the address table, and passes it to the CPU. The CPU may request to learn next newly learned address. And the CPU Interface Engine starts to search for next newly learned address from the address table.

Address Table

The address table can hold up to 2,048 MAC addresses, together with the associated port ID, security flag, filtering flag, new flag, aging information etc. The address table resides in the embedded SRAM inside the built-in ARL.

4. INTERFACE DESCRIPTION

CPU Interface

The CPU can communicate with the ARL through the UART interface of the switch controller. The management CPU can send commands to the ARL by writing into associated registers, and retrieve result from the ARL by reading out of the corresponding registers. The registers are described in the section on “Register Description.” The CPU interface signals are described by *table-1*:

Table-1: CPU Interface

Name	I/O	Description
UARTDI	I	UART input data line.
UARTDO	O	UART output data line.

UARTDI is used by the control CPU to send commands into the ARL. The baud rate will be automatically detected by the ARL. The result is returned through the UARTDO line with the detected baud rate. The format of the command packet is shown as follows:

A command sent by the CPU through the CPUDI line consists of 7 octets. Command frames transmitted on CPUDI have the format shown below:

ARL CPUDI Format

Operation	Command	Address	Data	Checksum
Write	0100XX11	A[7:0]	D[31:0]	C[7:0]
Read	0100XX01	A[7:0]	D[31:0]	C[7:0]

The byte order of data in all fields follows the big-endian convention, i.e. most significant octet first. The bit order is the least significant order first.

The Command octet specifies the type of the operation. The Bit-7, bit-6, and bit-5 of the command octet are specified the Device Type.
(1) Switch Controller, the device type is 001.
(2) ARL Controller, the device type is 010.
(3) Management Controller, the device type is 100.

The Bit-2, and bit-3 of the command octet are used to specify the device ID of the chip which is shared with ACD82224 device ID (ACD82224 bit 20 and bit 21 of Register 25).

The address octet specifies the number of the register.

For write operation, the Data field is a 4-octet value to specify what to write into the register.
For read operation, the Data field is a 4-octet 0 as padded data. If the data of register is less than 32-bit, it is align to bit-0 of Data field.

The checksum value is an 8-bit value of exclusive-OR of all octets in the frame, starting from the Command octet.

For each valid command received, the ARL will always send a response. Response from the ARL is sent through the CPUDO line. Response frames sent by the ACD82224 have the following format:

ARL UARTDO (Response) Format

Response	Command	Address	Data	Checksum
Write	0100XX11	A[7:0]	D[31:0]	C[7:0]

Read	0100XX01	A[7:0]	D[31:0]	C[7:0]

For response to a read operation, the Data field is a 4-octet value to indicate the content of the register. For response to a write operation, the Data field is 32 bits of 0. If the data of register is less than 24-bit, it is align to bit-0 of Data field.

The checksum value is an 8-bit value of exclusive-OR of all octets in the response frame, starting from the Command octet.

The ARL will always check the command header to see if both the device type and the device ID matches with its setting. If not, it ignores the command and does not generate any response to this command.

5. REGISTER DESCRIPTION

The built-in ARL provides a number of registers for the CPU to access the address table. Commands are sent to ARL by writing into the associated registers. Before the CPU can pass a command to ARL, it must check the Result register *(Register-11)* for execution status of the previous command. The CPU may need to retrieve the previous result before sending new command. Then the CPU will write the new command parameters into the Data Registers, and the command type into the Command Register. The ARL will then reset the Result Register to 0. The Result register will indicate the completion of the command at the end of the execution. Before the completion of the execution, any command written into the command register is ignored by the ARL.

The registers accessible to the CPU are described by *table-2*:

Table-2: Register Description

Reg.	Register Name	Type	Size	Description
0	DataReg0	R/W	8 Bit	Byte 0 of data
1	DataReg1	R/W	8 Bit	Byte 1 of data
2	DataReg2	R/W	8 Bit	Byte 2 of data
3	DataReg3	R/W	8 Bit	Byte 3 of data
4	DataReg4	R/W	8 Bit	Byte 4 of data
5	DataReg5	R/W	8 Bit	Byte 5 of data
6	DataReg6	R/W	8 Bit	Byte 6 of data
7	DataReg7	R/W	8 Bit	Byte 7 of data
8	AddrReg0	R/W	8 Bit	LSB of address value
9	AddrReg1	R/W	8 Bit	MSB of address value
10	CmdReg	R/W	8 Bit	Command register
11	RsltReg	R/W	5 Bit	Result register
12	CfgReg	R/W	8 Bit	Configuration register
13	IntSrcReg	R/W	8 Bit	Interrupt source register
14	IntMskReg	R/W	8 Bit	Interrupt mask register
15	nLearnReg0	R/W	8 Bit	Address learning disable register for port 0 – 7
16	nLearnReg1	R/W	8 Bit	Address learning disable register for port 8 – 15
17	nLearnReg2	R/W	8 Bit	Address learning disable register for port 16 – 23
18	AgeTimeReg0	R/W	8 Bit	LSB of aging period register
19	AgeTimeReg1	R/W	8 Bit	MSB of aging period register
20	PosCfg	R/W	3 Bit	Power On Strobe configuration register

DataReg0 ~ DataReg7 (Register 0 ~ Register 7)

The *DataReg[0:7]* are registers used to pass the command parameters to the ARL, and the execution results to the CPU. **ARL only stores 47-bit of MAC address, the first bit of the first Byte (MSB) is not stored in ARL table (this bit to indicate broadcast/multicast frame). The data in Data register 0 is shift left one bit compared to the MAC MSB.** For example, if the MAC address is "08-00-12-34-56-78", DataReg-0 is stored the value of "04" instead of "08".

AddrReg0 and AddrReg1 (Register 8 and Register 9)

The *AddrReg[0:1]* are used to specify the address associated with the command.

CmdReg (Register 10)

The *CmdReg* is used to pass the type of command to the ARL. The command types are listed in *table-3*. The details of each command are described in the chapter of "Command Description."

Table-3: Command List

Command	Description
0x09	Add the specified MAC address into the address table
0x0A	Set a lock for the specified MAC address
0x0B	Set a filtering flag for the specified MAC address
0x0C	Delete the specified MAC address from the address table
0x0D	Assign a port ID to the specified MAC address
0x10	Read the first entry of the address table
0x11	Read next entry of address book
0x20	Read first valid entry
0x21	Read next valid entry
0x30	Read first new entry
0x31	Read next new entry
0x40	Read first aged entry
0x41	Read next aged entry
0x50	Read first locked entry
0x51	Read next locked entry
0x60	Read first filtered entry
0x61	Read next filtered entry
0x80	Read first entry with specified PID
0x81	Read next entry with specified PID
0xFF	ARL reset

RstReg (Register 11)

The *RstReg* is used to indicate the status of command execution. The result code is listed as follows:

Bit	Description	Default
3:0	4-bit error code 0000 – No error 0001 – Cannot find the specified entry Other – Errors	0000
4	Command completed. 0 – Execution has been started but not yet completed 1 – Execution has been completed, Must check Bit[3:0], any error occurring.	0

CfgReg (Register 12)

The *CfgReg* is used to configure the ARL functions. The bit definition of CfgReg is described as:

Bit	Description	Default
0	Disable address aging	0
1	Disable address lookup	0
2	NA	0
3	NA	0
7:4	Hashing algorithm selection	0000

IntSrcReg (Register 13)

The *IntSrcReg* is used to indicate what can cause interrupt request to CPU. The source of interrupt is listed as:

Bit	Description	Default
0	Aged address exists	0
1	New address exists	0
2	Reserved	0
3	Reserved	0
4	Bucket overflowed	0
5	Command is done	1
6	System initialization is completed	1
7	Self test failure	0

IntMskReg (Register 14)

The *IntMskReg* is used to enable an interrupt source to generate an interrupt request. The bit definition is the same as IntSrcReg. A 1 in a bit enables the corresponding interrupt source to generate an interrupt request once it is set.

Bit	Description	Default

0	Aged address exists	1
1	New address exists	1
2	Reserved	1
3	Reserved	1
4	Bucket overflowed	1
5	Command is done	1
6	System initialization is completed	1
7	Self test failure	1

nLearnReg0 ~ nLearnReg2 (Register 15 ~ Register 17)

The *nLearnReg[2:0]* are used to disable address learning activity from a particular port. If the bit corresponding to a port is set, the ARL will not try to learn new addresses from that port.

The nLearnReg0/1/2 are bit-to-port mapping registers.
The bit[0:7] of nLearnReg0 is represented by port[0:7].
The bit[0:7] of nLearnReg1 is represented by port[8:15].
The bit[0:7] of nLearnReg2 is represented by port[16:23].

AgeTimeReg0 and AgeTimeReg1 (Register 18 and Register 19)

The *AgeTimeReg[1:0]* are used to specify the period of address aging control. The aging period can be from 0 to 65535 units, with each unit counted as 2.684 second. The default age time is 300 seconds. **To make the new setting age period effective, CPU must send "ARL Reset" (0xff, see ARL Table-3) command to ARL after configuring the new AgeTimeReg[1:0] and set the bit-0 of Register-20 to one to wake up ARL engine**

PosCfgReg (Register 20)

The *PosCfgReg* is a configuration register whose default value is determined by the pull-up or pull-down status of the associated hardware pin. The bits of PosCfgReg0 is listed as follows:

Bit	Description	Default	Shared Pin
0	Reserved	0	NA
1	NOCPU 0 – Wait for CPU 1 –ARL initializes by itself	0	P00TXEN
2	Reserved	0	P02TXEN

Note: If *NOCPU* is set to 0, the ARL will not start the initialization process until Bit-1 of PosCfgReg is set to 1.

6. COMMAND DESCRIPTION

Command 09H

Description: Add the specified MAC address into the address table.

Parameter: Store the MAC address into DataReg5 – DataReg0, with DataReg5 contains the MSB of the MAC address and DataReg0 contains the LSB. Store the associated port number into DataReg6.

Result: the MAC address will be stored into the address table if there is space available. The result is indicated by the Result register.

Command 0AH

Description: Set the Lock bit for the specified MAC address.

Parameter: Store the MAC address into DataReg5 – DataReg0, with DataReg5 contains the MSB of the MAC address and DataReg0 contains the LSB.

Result: the state machine will seek for an entry with matched MAC address, and set the Lock bit of the entry. The result is indicated by the Result register.

Command 0BH

Description: Set the Filter flag for the specified MAC address.

Parameter: Store the MAC address into DataReg5 – DataReg0, with DataReg5 contains the MSB of the MAC address and DataReg0 contains the LSB.

Result: the state machine will seek for an entry with matched MAC address, and set the Filter bit of the entry. The result is indicated by the Result register.

Command 0CH

Description: Delete the specified MAC address from the address table.

Parameter: Store the MAC address into DataReg5 – DataReg0, with DataReg5 contains the MSB of the MAC address and DataReg0 contains the LSB.

Result: the MAC address will be removed from the address table. The result is indicated by the Result register.

Command 0DH

Description: Assign the associated port number to the specified MAC address.

Parameter: Store the MAC address into DataReg5 – DataReg0, with DataReg5 contains the MSB of the MAC address and DataReg0 contains the LSB. Store the port number into DataReg6.

Result: the port ID field of the entry containing the specified MAC address will be changed accordingly. The result is indicated by the Result register.

Command 10H

Description: Read the first entry of the address table.

Parameter: None

Result: The result is indicated by the Result register. If the command is completed with no error, the content of the first entry of the address book will be stored into the Data registers. The MAC

address will be stored into DataReg5 – DataReg0, with DataReg5 contains the MSB of the MAC address and DataReg0 contains the LSB. The port number is stored in DataReg6, and the Flag* bits are stored in DataReg7.The Read Pointer will be set to point to second entry of the address book.

Note – the Flag bits are defined as:

b7	b6	b5	b4	b3	b2	b1	b0
Rsvd	Rsvd	Filter	Lock	New	Old	Age	Valid

Where

- Filter – 1 indicates the frame heading to this address should be dropped.
- Lock – 1 indicates the entry should never be changed or aged out.
- New – 1 indicates the entry is a newly learned address.
- Old – 1 indicates the address has been aged out.
- Age – 1 indicates the address has not been visited for current age cycle.
- Valid – 1 indicates the entry is a valid one.
- Rsvd – Reserved bits.

Command 11H

Description: Read next entry of address book.

Parameter: None

Result: The result is indicated by the Result register. If the command is completed with no error, the content of the address book entry pointed by Read Pointer will be stored into the Data registers. The MAC address will be stored into DataReg5 – DataReg0, with DataReg5 contains the MSB of the MAC address and DataReg0 contains the LSB. The port number is stored in DataReg6, and the Flag bits are stored in DataReg7. The Read Pointer will be increased by one.

Command 20H

Description: Read first valid entry.

Parameter: None

Result: The result is indicated by the Result register. If the command is completed with no error, the content of first valid entry of the address book will be stored into the Data registers. The MAC address will be stored into DataReg5 – DataReg0, with DataReg5 contains the MSB of the MAC address and DataReg0 contains the LSB. The port number is stored in DataReg6, and the Flag bits are stored in DataReg7. The Read Pointer is set to point to this entry.

Command 21H

Description: Read next valid entry.

Parameter: None

Result: The result is indicated by the Result register. If the command is completed with no error, the content of next valid entry from the Read Pointer of the address book will be stored into the Data registers. The MAC address will be stored into DataReg5 – DataReg0, with DataReg5

contains the MSB of the MAC address and DataReg0 contains the LSB. The port number is stored in DataReg6, and the Flag bits are stored in DataReg7. The Read Pointer is set to point to this entry.

Command 30H

Description: Read first new entry.

Parameter: None

Result: The result is indicated by the Result register. If the command is completed with no error, the content of first new entry of the address book will be stored into the Data registers. The MAC address will be stored into DataReg5 – DataReg0, with DataReg5 contains the MSB of the MAC address and DataReg0 contains the LSB. The port number is stored in DataReg6, and the Flag bits are stored in DataReg7. The Read Pointer is set to point to this entry.

Command 31H

Description: Read next new entry.

Parameter: None

Result: The result is indicated by the Result register. If the command is completed with no error, the content of next new entry from the Read Pointer of the address book will be stored into the Data registers. The MAC address will be stored into DataReg5 – DataReg0, with DataReg5 contains the MSB of the MAC address and DataReg0 contains the LSB. The port number is stored in DataReg6, and the Flag bits are stored in DataReg7. The Read Pointer is set to point to this entry.

Command 40H

Description: Read first aged entry.

Parameter: None

Result: The result is indicated by the Result register. If the command is completed with no error, the content of first aged entry of the address book will be stored into the Data registers. The MAC address will be stored into DataReg5 – DataReg0, with DataReg5 contains the MSB of the MAC address and DataReg0 contains the LSB. The port number is stored in DataReg6, and the Flag bits are stored in DataReg7. The Read Pointer is set to point to this entry.

Command 41H

Description: Read next aged entry.

Parameter: None

Result: The result is indicated by the Result register. If the command is completed with no error, the content of next aged entry from the Read Pointer of the address book will be stored into the Data registers. The MAC address will be stored into DataReg5 – DataReg0, with DataReg5 contains the MSB of the MAC address and DataReg0 contains the LSB. The port number is stored in DataReg6, and the Flag bits are stored in DataReg7. The Read Pointer is set to point to this entry.

Command 50H

Description: Read first locked entry.

Parameter: None

Result: The result is indicated by the Result register. If the command is completed with no error, the content of first locked entry of the address book will be stored into the Data registers. The MAC address will be stored into DataReg5 – DataReg0, with DataReg5 contains the MSB of the MAC address and DataReg0 contains the LSB. The port number is stored in DataReg6, and the Flag bits are stored in DataReg7. The Read Pointer is set to point to this entry.

Command 51H

Description: Read next locked entry.

Parameter: None

Result: The result is indicated by the Result register. If the command is completed with no error, the content of next locked entry from the Read Pointer of the address book will be stored into the Data registers. The MAC address will be stored into DataReg5 – DataReg0, with DataReg5 contains the MSB of the MAC address and DataReg0 contains the LSB. The port number is stored in DataReg6, and the Flag bits are stored in DataReg7. The Read Pointer is set to point to this entry.

Command 60H

Description: Read first filtered entry.

Parameter: None

Result: The result is indicated by the Result register. If the command is completed with no error, the content of first filtered entry of the address book will be stored into the Data registers. The MAC address will be stored into DataReg5 – DataReg0, with DataReg5 contains the MSB of the MAC address and DataReg0 contains the LSB. The port number is stored in DataReg6, and the Flag bits are stored in DataReg7. The Read Pointer is set to point to this entry.

Command 61H

Description: Read next valid entry.

Parameter: None

Result: The result is indicated by the Result register. If the command is completed with no error, the content of next filtered entry from the Read Pointer of the address book will be stored into the Data registers. The MAC address will be stored into DataReg5 – DataReg0, with DataReg5 contains the MSB of the MAC address and DataReg0 contains the LSB. The port number is stored in DataReg6, and the Flag bits are stored in DataReg7. The Read Pointer is set to point to this entry.

Command 80H

Description: Read first entry with specified port number.

Parameter: Store port number into DataReg6.

Result: The result is indicated by the Result register. If the command is completed with no error, the content of first entry of the address book with the said port number will be stored into the Data registers. The MAC address will be stored into DataReg5 – DataReg0, with DataReg5 contains the MSB of the MAC address and DataReg0 contains the LSB. The port number is stored in DataReg6, and the Flag bits are stored in DataReg7. The Read Pointer is set to point to this entry.

Command 81H

Description: Read next valid entry.

Parameter: Store port number into DataReg6.

Result: The result is indicated by the Result register. If the command is completed with no error, the content of next entry from the Read Pointer of the address book with the said port number will be stored into the Data registers. The MAC address will be stored into DataReg5 – DataReg0, with DataReg5 contains the MSB of the MAC address and DataReg0 contains the LSB. The port number is stored in DataReg6, and the Flag bits are stored in DataReg7. The Read Pointer is set to point to this entry.

Command FFH

Description: ARL reset.

Parameter: None

Result: This command will reset the ARL. All entries of the address book will be cleared and set the bit-0 of Register-20 to one to wake up ARL engine