Using the Digital Semiconductor 21140A with Boot ROM, Serial ROM, and External Register:

An Application Note

Order Number: EC–QPQWA–TE

This application note provides information necessary to implement connections between the Digital Semiconductor 21140A Fast Ethernet LAN controller and boot ROM, serial ROM, and external register. It also describes the serial ROM programming format.

Revision/Update Information: This is a new document.

Digital Equipment Corporation
Maynard, Massachusetts

March 1996

While Digital believes the information included in this publication is correct as of the date of publication, it is subject to change without notice.

Digital Equipment Corporation makes no representations that the use of its products in the manner described in this publication will not infringe on existing or future patent rights, nor do the descriptions contained in this publication imply the granting of licenses to make, use, or sell equipment or software in accordance with the description.

This document was prepared using VAX DOCUMENT Version 2.1.

Contents

Figures

Tables

1 Overview

The information contained in this application note describes how to connect the Digital Semiconductor 21140A Fast Ethernet LAN controller (21140A) to its boot ROM, serial ROM, and external register peripheral devices. Note that connection to either a boot ROM or to an external register is not a requirement for correct operation of the controller. Any combination of these connections may be used.

The programming information supplied in this application note applies to device drivers supplied by Digital Semiconductor. Users may use other formats supported by their own device drivers.

For detailed technical product requirements, the product developer should refer to the *Digital Semiconductor 21140A Fast Ethernet LAN Controller Data Sheet* and the *Digital Semiconductor 21140A Fast Ethernet LAN Controller Hardware Reference Manual*.

2 Functional Overview

The 21140A allows connection to an upgradable boot ROM (flash or EEPROM) of 64KB, 128KB, or 256KB. The boot ROM typically contains code that can be executed for device-specific initialization and, possibly, a system boot function.

The 21140A also supports connection to the serial ROM for read and write operations. The serial ROM contains the IEEE address and other optional system parameters. The interface to serial ROM is fully software driven.

Connection to a general-purpose external register can be done for read and write operations. This connection allows a general-purpose bidirectional port for various applications.

The 21140A provides an interface for the connections described in this application note. The access control to the different devices is managed by software using CSR9 and CSR10.

Table 1 lists the function of each interface signal.

Table 1 Boot ROM, Serial ROM, and External Register Interface Pins

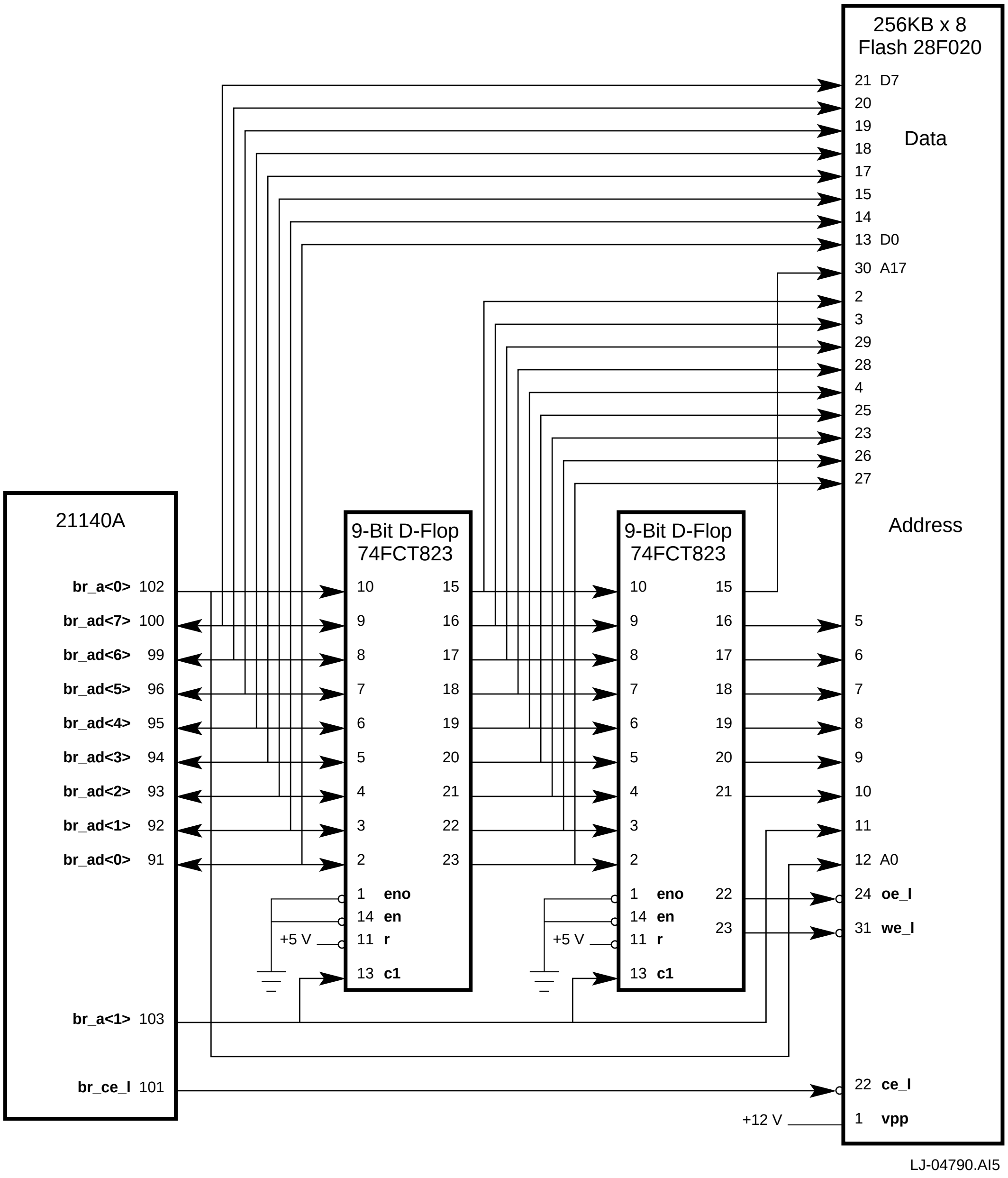
	Pin		Function	
Signal	Number	Boot ROM	Serial ROM	External Register
br_ad<7:6, 5:0>	100:99, 96:91	Address and data lines, **we_l**, **oe_l**	Not used	Data bits
br_a<1>	103	Address bit 1, latch control for external latches	Not used	Not used
br_a<0>	102	Address bits 0, 16, and 17	Not used	Read and write control
br_ce_l	101	Chip enable control	Not used	Chip enable or read and write control
sr_do	76	Not used	Data out	Not used
sr_di	77	Not used	Data in	Not used
sr_ck	78	Not used	Clock	Not used
sr_cs	79	Not used	Chip select	Not used

3 Connection to Boot ROM

Figure 1 shows a connection of a 256KB flash boot ROM. The required components for this configuration are

- Two 9-bit-high, edge-triggered latches (74FCT823)
- Flash ROM chip (28F020)

Figure 1 Boot ROM (256KB) Connection



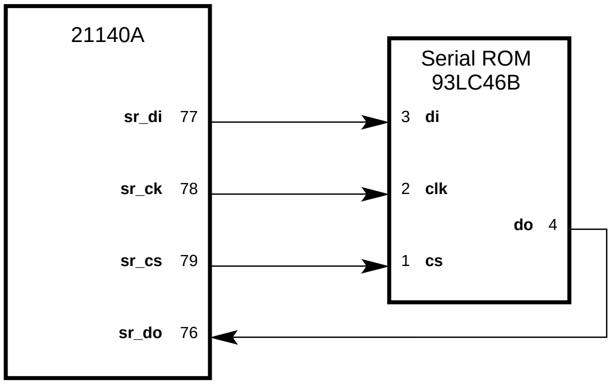
4 Connection to Serial ROM

The following sections describe connnections to the serial ROM.

4.1 Single Digital Semiconductor 21140A Connection

Figure 2 shows a connection between a single Digital Semiconductor 21140A and a MicroWire 1024-bit serial EEPROM. No additional components are needed for this connection. MicroWire serial EEPROM connections are provided for up to 4Kb.

Figure 2 Serial ROM (1024-Bit) Connection



LJ-04791.AI5

5 External Register Connection

This section describes two configuration types for using the general-purpose, 8-bit external register.

- A minimum configuration without boot ROM, using the external register port in one direction only
- A maximum configuration with boot ROM, using the external register as a bidirectional port

5.1 Configuration of External Register Without Boot ROM

This configuration assumes that boot ROM is not used and the general-purpose external register is used for read-only or write-only operations.

Figure 3 shows a 21140A external register write operation to the 74FCT273.

Figure 3 External Register Connection—Write Only (No Boot ROM)

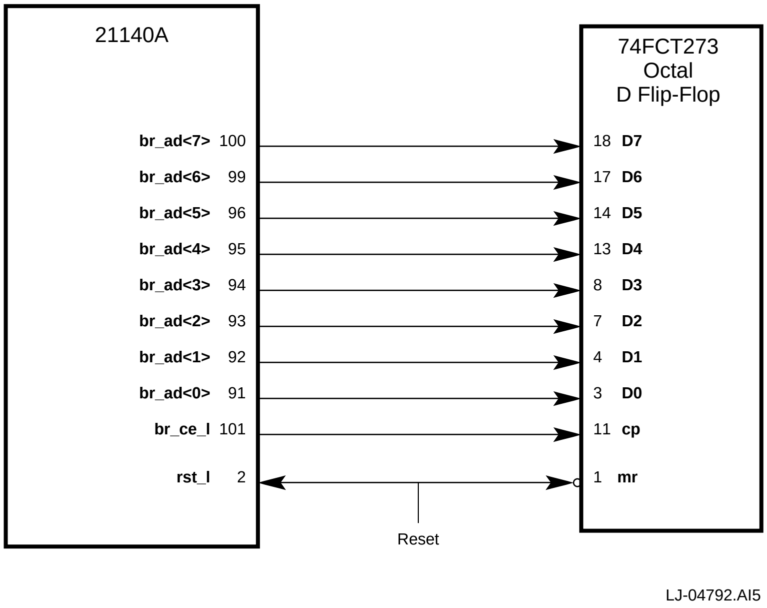
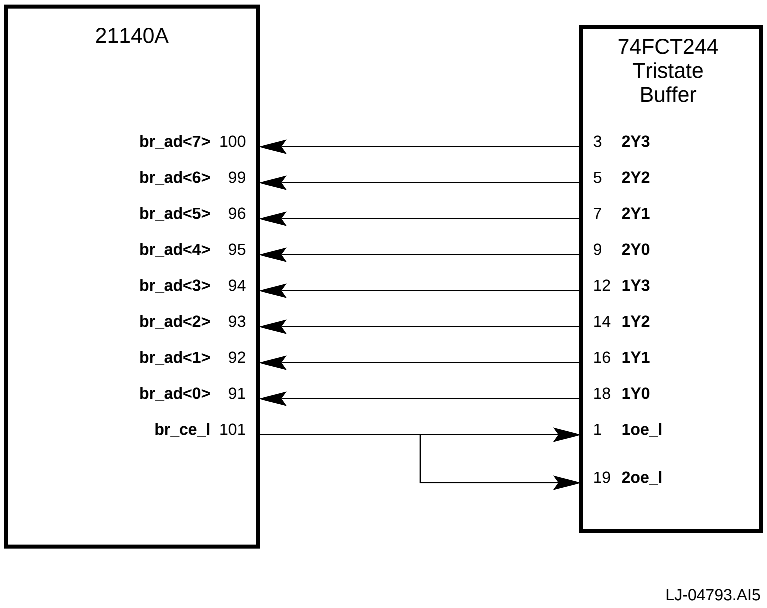


Figure 4 shows a configuration that uses the external register for read operations only. Data read by the 21140A should be driven constantly on the 74FCT244 inputs.

Figure 4 External Register Connection—Read Only (No Boot ROM)



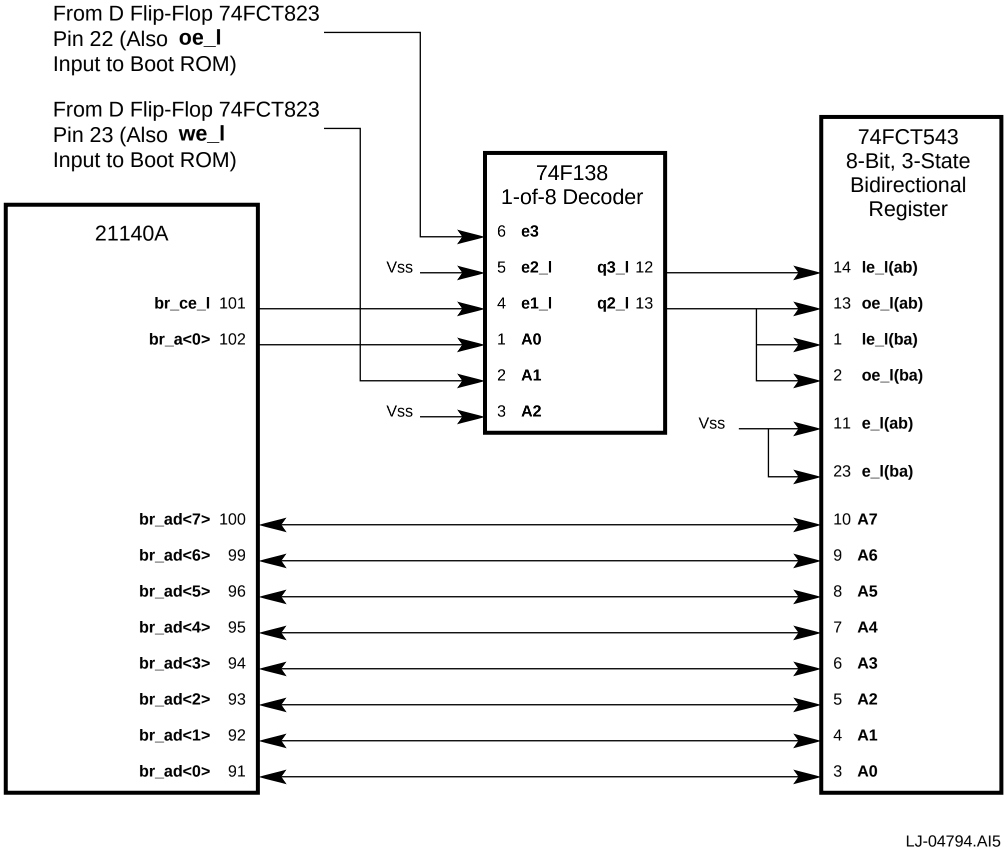
5.2 Configuration of External Register with Boot ROM

This connection assumes that both the external register and the boot ROM are used by the 21140A. This connection also allows read and write accesses to the external register, making it a bidirectional general-purpose port. Note that Figure 1 shows the boot ROM connection.

Figure 5 describes the connection of the external register used for read and write operations with the boot ROM included on the adapter. The required components for this configuration are

- 1-of-8 decoder
- Octal latched transceiver (3-state)

Figure 5 External Register Connection—Read and Write with Boot ROM



6 Serial ROM Programming

The definition for serial ROM programming that is described in this section supports:

- Multiple chips on a single board sharing a single serial ROM
- Multiple PHY chips connected to the same adapter

The serial ROM programming information is applicable for device drivers supplied by Digital Semiconductor. Bit fields labeled as reserved throughout the remainder of this application note must contain all 0s. Users may apply other formats supported by their own device drivers.

Note

To optimize the ROM space usage, byte fields are used. Because the serial ROM supports only word accesses, Digital Semiconductor recommends that you first download the entire ROM into a memory shadow table.

This section permits board manufacturers to use parts of the serial ROM for private data. Discuss this usage with an authorized Digital Semiconductor representative to avoid conflicts with future versions of the serial ROM format.

Figure 6 shows the structure of the serial ROM, and Table 2 describes the byte fields.

Figure 6 Serial ROM Structure

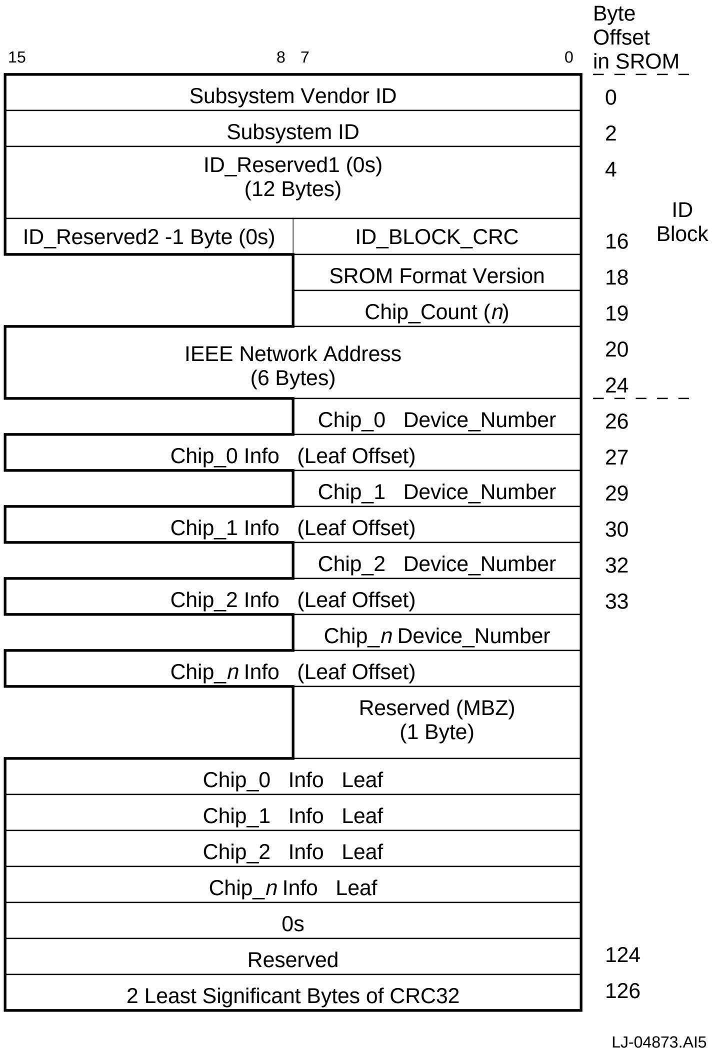


Table 2 Serial ROM Field Description

Field	Size (Bytes)	Definition
Subsystem Vendor ID	2	This field is used to uniquely identify the 21140A based on the 21X4X family of controllers.
Subsystem ID	2	This field is used to uniquely identify the subsystem ID.
ID_Reserved1	12	Reserved, must be zero (MBZ).
ID_BLOCK_ CRC	1	Contains the CRC8 value of the ID block that is calculated on word 0, word 1, ... word 8 inclusive (the ID_Reserved2 value is also included). Appendix B describes how this algorithm is calculated.
ID_Reserved2	1	Reserved, must be zero (MBZ).

(continued on next page)

Table 2 (Cont.) Serial ROM Field Description

Field	Size (Bytes)	Definition
SROM Format Version	1	SROM format version. Current version is 0x03.
Chip_Count (*n*)	1	Number of chips sharing this ROM. A single port board will have a value of 1 in this field.
IEEE Network Address	6	This is the IEEE address of the chip in a single-chip board. In a multiple-chip board, this is the base IEEE address. Every chip (0..*n*) adds its index (*n*) to this base IEEE address.
Chip_*n* Device_ Number	1	There is one such field per chip sharing the SROM. In a multiple chip board, this field contains the Device_Number value by which the *n*th chip's configuration space can be accessed on this board's secondary PCI bus. This value depends on the hardware routing of the board. The Device_Number is the *chip select* line routed from this chip to the PCI-to-PCI bridge chip on board. In a single-chip board, this field has no meaning and should be ignored by the driver.
Chip_*n* Info	2	Byte offset (from beginning of SROM) where chip_*n* information block is located. There is one such field per chip sharing the SROM. The information block is chip specific. That is, the block varies between chips. Refer to the format of the chip information leaf for details. **Note:** If multiple chips have identical information blocks, a single leaf can be shared and all leaf pointers can be set to point to it. This is correct only if the user cannot select between multiple media ports for each chip. For example: A 4-TP port card can share one info block for all 4 chips.
Reserved	1	MBZ. Note that the location of this field depends on the number of chips supported by this card.
Reserved	2	This field is reserved for the use by the chip manufacturer. Standard drivers do not use this field. This field is always located in the 2 bytes that immediately precede the SROM_CRC field. If the manufacturer's data exceeds 2 bytes, this field can be used as a pointer to the actual data.

(continued on next page)

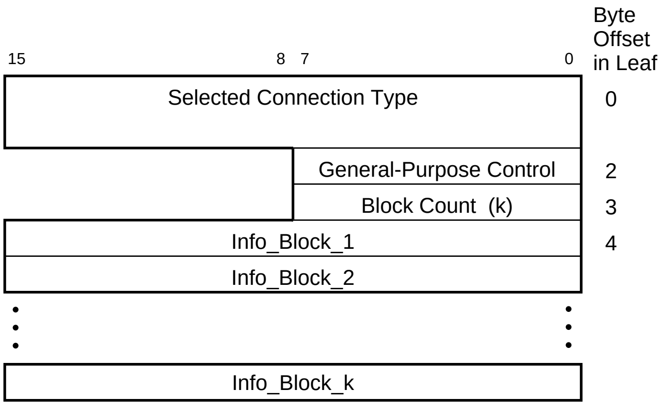
Table 2 (Cont.) Serial ROM Field Description

Field	Size (Bytes)	Definition
2 LSB of CRC32	2	Calculated on all the words of the SROM from word[0] to the word before the CRC (word[SROM_word_size –2]). The CRC word is derived by calculating the CRC32 of all the SROM until the last word (not including it) and taking the 2 least significant bytes of the result. That is, if the CRC is 4 bytes long with byte 0 being the least significant byte, then SROM_BYTE[BYTE_LEN –2] holds CRC<0> (least significant byte) and SROM_BYTE[BYTE_LEN –1] holds CRC<1>. The bytes are written in little endian. Appendix A defines the serial ROM CRC calculation algorithm.

6.1 Info Leaf Format

Figure 7 shows the info leaf format, and Table 3 describes the byte fields.

Figure 7 Info Leaf Format



LJ-04874.AI5

Table 3 Info Leaf Description

Field	Size (Bytes)	Meaning
Selected Connection Type	2	Usually, the connection type used by the chip is selected by the user in the drivers' configuration files. However, this field has been provided to allow setup utilities that are unable to modify the configuration files and save this information in the SROM instead. Normally, when the media selection information is stored in the driver's configuration files, this field is set to one of the following values depending on the board's capabilities: • 0x0800—Power-up AutoSense and dynamic AutoSense (if supported by board) • 0x8800—Power-up AutoSense only The possible values for the setup utilities using SROM are: 0x0000—TP (10BASE-T) 0x0100—TP with autonegotiation 0x0204—TP full-duplex 0x0001—BNC (10BASE2) 0x0003—SYM_SCR (100BASE-TX) 0x0205—SYM_SCR (full-duplex) 0x0006—100BASE-T4 0x0007—100BASE-FX (fiber) 0x0208—100BASE-FXFD (fiber full-duplex) 0x0009—MII TP (10BASE-T) 0x020A—MII TP (full-duplex) 0x000D—MII (100BASE-TX) 0x020E—MII (100BASE-TX full-duplex) 0x000F—MII (100BASE-T4) 0x0010—MII (100BASE-FX 100Mb/s fiber) 0x0211—MII (100BASE-FX 100Mb/s fiber full-duplex) 0x0800—Power-up AutoSense, dynamic AutoSense (if possible) 0x8800—Power-up AutoSense only 0xFFFF—No selected media interface If this field is not used, it must be set to 0xFFFF. Any other value is invalid and may cause unpredictable results.
General-Purpose Control	1	This field contains the value of the general-purpose mask register of adapter_*n*, regardless of the media involved. This value is adapter specific. It determines the direction of the general-purpose port bits (defining bits that are input and bits that are output).
Block Count (k)	1	The number of info blocks present for this adapter.

(continued on next page)

Table 3 (Cont.) Info Leaf Description

Field	Size (Bytes)	Meaning
Info_Block_k	Media dependent	Describes one supported medium/PHY chip. There is one such field per supported non-MII medium and one for every MII PHY chip. See details in Section 6.2. The order of the info blocks define their precedence during autosensing. That is, the first entry is the medium PHY chip with the lowest precedence and will be checked last. The final entry in the list is the medium PHY chip with the highest precedence and will be checked first.

6.2 Info Block Format

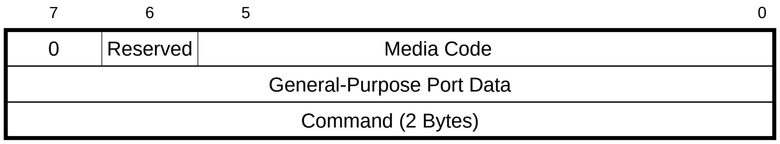
The info block format can be in one of two formats:

- **Compact format** (Version 1.04 for non-MII media only). This format can be identified by a 0 in info_block byte 0, bit 7 (Figure 8).
- **Extended format**. This format can be identified by a 1 in info_block byte 0, bit 7 (Figure 9).

6.2.1 Compact Format

Figure 8 shows the compact format bit field, and Table 4 describes the bit field.

Figure 8 Compact Format



LJ-04875.AI5

Table 4 Compact Format Description

Field	Size (Bits)	Function
Format Indicator	1	The value in this field must be 0 to select the compact format.
Reserved	1	Reserved.
Media Code	6	This field indicates the adapter supported medium code to the driver. The supported adapter medium codes include the following: 00—TP (10Mb/s) 01—BNC (10Mb/s) 03—SYM_SCR (100BASE-TX) 04—TP full-duplex 05—SYM_SCR full-duplex (100BASE-TX) 06—100BASE-T4 07—100BASE-FX (fiber) 08—100BASE-FXFD (fiber full-duplex)
General-Purpose Port Data	8	When this medium is selected, 8 data bits are written to the general-purpose data register of adapter_*n* (21140A). The value of this parameter is board and adapter specific. The data is defined by the board's manufacturer, and its purpose is to initialize and enable the selected medium's hardware.

(continued on next page)

Table 4 (Cont.) Compact Format Description

Field	Size	Function
	(Bits)	
Command	16	When this medium is selected, this field (bits 15:0) generates the CSR6 mode bits of adapter_*n* (21140A) and is defined as follows: **Bit 15, Active_Invalid**—When set, indicates that the media sense bit number is not valid and that there is no media activity indication in the general-purpose register (GPR). Dynamic autosensing is only attempted between media when this bit is reset (indicating that there is a valid media sense bit to test). **Bit 14, Default_Media**—When set, indicates that the default medium is selected if no active link is found during the AutoSense process (power-up and dynamic). This bit is valid only if **active_invalid** (bit 15) is reset for this medium. This bit is not valid for full-duplex media entries, it is set for one medium only. **Bits 13:8, MBZ**—Must be zero. **Bit 7, Polarity**—This bit indicates the polarity of the media activity indication bit in the general-purpose register. When this bit is reset and **active_invalid** is set, the media activity bit in the GPR reads 1 when the medium is active. When this bit is set and **active_invalid** is reset, the media activity bit in the GPR reads 0 when the medium is active. Table 5 describes the state definitions for the GPR media activity bit. **Bit 6, CSR6, Scrambler Mode**—MII/SYM port transmits and receives scrambled symbols. **Bits 5:4, CSR6, PCS Function**—When set, the MII/SYM port operates in symbol mode. **Bits 3:1, Media Sense Bit Number**—The driver senses and obtains the media bit number from the general-purpose port register. **Bit 0, CSR6 Port Select**—When reset, the SRL port is selected. When set, the MII/SYM port is selected.

Table 5 describes the state definitions of the GPR media activity bit. The **active_invalid** bit must be reset for these state definitions.

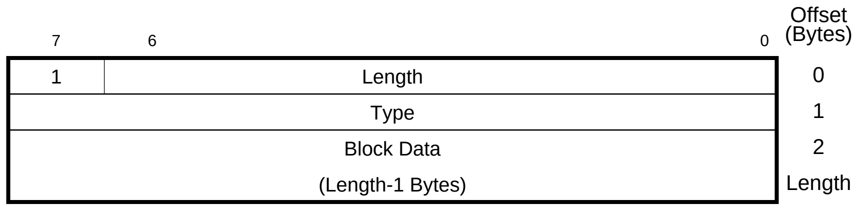
Table 5 GPR State Description

Media Activity Bit	Polarity Bit	Medium State
0	0	Not active
0	1	Active
1	0	Active
1	1	Not active

6.2.2 Extended Format

Figure 9 shows the extended format bit field, and Table 6 describes the bit field.

Figure 9 Extended Format



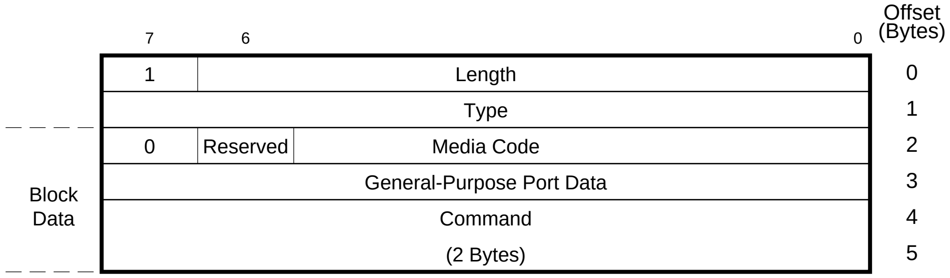
LJ-04876.AI5

Table 6 Extended Format Description

Field	Size (Bits)	Function
Format Indicator	1	The value in this field must be 1 to select the extended format.
Length	7	The value in this field is the size, in bytes, of this info block. This byte size includes the type field and the block data. It does not include the length field itself.
Type	8	There are two extended block types: • 00—Non-MII media block (Section 6.2.2.1). • 01—MII PHY chip block (Section 6.2.2.2).
Block Data	8 (Length-1)	The value in this field is determined by the block type (Section 6.2.2.1 and Section 6.2.2.2).

6.2.2.1 Non-MII Media—Block Type 00 Figure 10 shows the non-MII media block format, and Table 7 describes the byte fields.

Figure 10 Non-MII Media Block Format



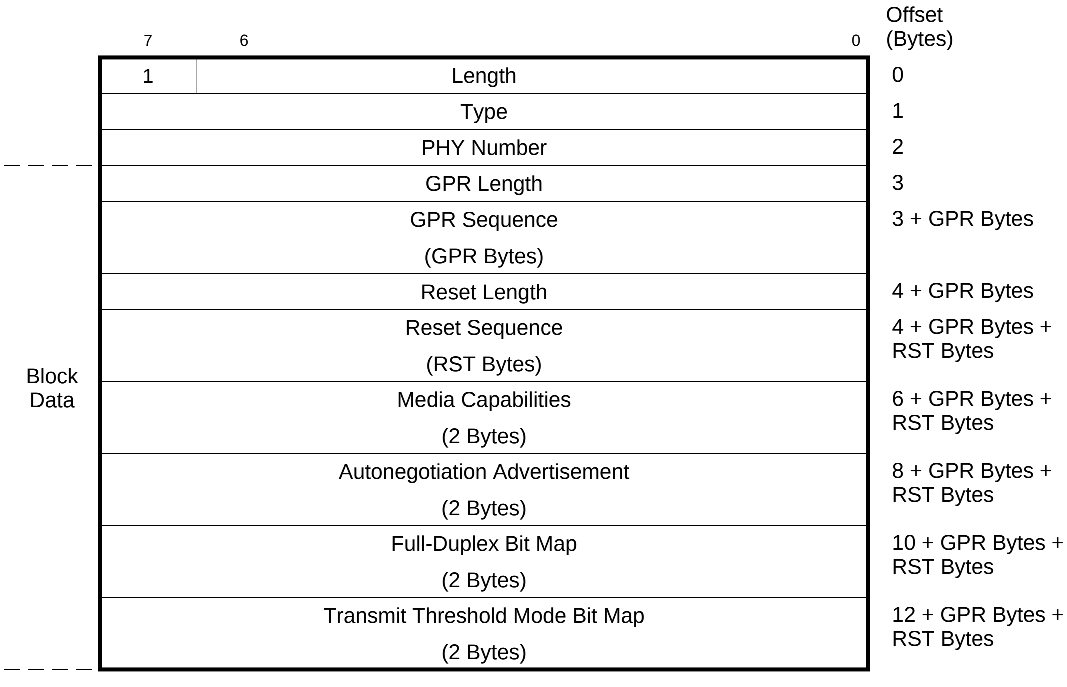
LJ-04877.AI5

Table 7 Non-MII Media Format Description

Field	Size (Bits)	Function
Format Indicator	1	The value in this field must be 1 to select the extended format.
Length	7	The value in this field is always 0x05 for block type 00.
Type	8	This field displays block type 0x00.
Block Data	32	This field is identical to the compact format (Table 4).

6.2.2.2 MII PHY Chip—Block Type 01 Figure 11 shows the MII PHY chip block format, and Table 8 describes the byte fields.

Figure 11 MII PHY Chip Block Format



7	6 … 0	Offset (Bytes)
1	Length	0
Type		1
PHY Number		2
GPR Length		3
GPR Sequence (GPR Bytes)		3 + GPR Bytes
Reset Length		4 + GPR Bytes
Reset Sequence (RST Bytes)		4 + GPR Bytes + RST Bytes
Media Capabilities (2 Bytes)		6 + GPR Bytes + RST Bytes
Autonegotiation Advertisement (2 Bytes)		8 + GPR Bytes + RST Bytes
Full-Duplex Bit Map (2 Bytes)		10 + GPR Bytes + RST Bytes
Transmit Threshold Mode Bit Map (2 Bytes)		12 + GPR Bytes + RST Bytes

Table 8 MII PHY Chip Format Description

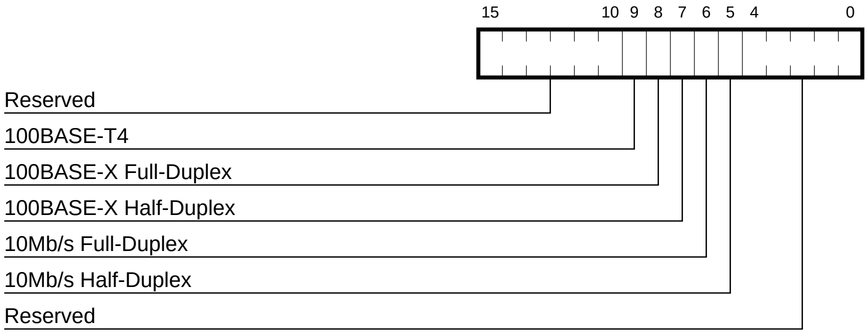
Field	Size (Bits)	Function
Format Indicator	1	The value in this field must be 1 to select the extended format.
Length	7	The value in this field is always 12 + GPR length + reset length for block type 01.
Type	8	This field displays block type 01.
PHY Number	8	This value represents the index of the PHY chip on the board. The PHY value is determined by the chip address: the lowest chip address is 0, the next chip address is 1, and so on. If there is an external MII connector on the board, it must be described in the last block (and is assigned the highest PHY number), despite the MII specification determination that its address must be zero.
GPR Length	8	Contains the number of bytes in the GPR sequence field. A GPR length of 0 indicates that no value needs to be written to the GPR to select and activate this PHY chip.

(continued on next page)

Table 8 (Cont.) MII PHY Chip Format Description

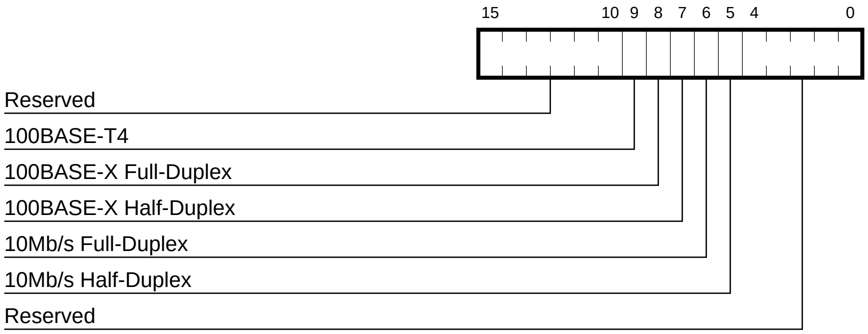
Field	Size (Bits)	Function
GPR Sequence	See GPR Length	Provides sequence of data bytes written to the GPR for PHY chip operation. These bytes are written each time the adapter switches media to one supported by this PHY chip. The bytes are written in the order displayed in this field (one byte at a time).
Reset Length	8	Contains the number of bytes in the reset sequence field. A reset length of 0 indicates that the PHY chip is not reset via the GPR.
Reset Sequence	See Reset Length	Provides the sequence of data bytes to the GPR to reset this PHY chip. The bytes are written in the order displayed in this field. The reset sequence is executed the first time this PHY chip is selected prior to GPR sequence execution.
Media Capabilities	16	This field provides a bit map (Figure 12) that describes the media supported for this specific PHY chip. Each bit in the map represents a different medium. If a bit is set, this indicates that the medium is supported. If the bit is reset, this indicates that the medium is not supported. This permits board designers to select and support a subset of the total capabilities initially supported by the PHY chip itself.
Autonegotiation Advertisement	16	This field permits board designers to determine the capabilities that the PHY should advertise during the autonegotiation process. This can be a subset of the supported capabilities but should never include media that is not supported in the media capabilities field. Figure 13 shows the bit map for the autonegotiation advertisement register as defined in the MII specification.
Full-Duplex Bit Map	16	This field indicates the value that is written to the full-duplex bit in CSR6 for each medium (according to the bit map of the media capabilities field).
Transmit Threshold Mode Bit Map	16	This field indicates the value that is written to the transmit threshold mode bit in CSR6 for this medium (according to the bit map of the media capabilities field).

Figure 12 Media Capabilities Bit Map



LJ-04879.AI4

Figure 13 Autonegotiation Advertisement Bit Map



LJ-04879.AI4

A

Serial ROM CRC Calculation Algorithm

This appendix provides the algorithm to calculate the serial ROM CRC.

```
unsigned short CalcSromCrc(unsigned char *SromData);

#define  DATA_LEN        126    // 1024 bits SROM

struct {
        unsigned char  SromData[DATA_LEN];
        unsigned short SromCRC;
        } Srom;
main()
{
   Srom.SromCRC = CalcSromCrc(&Srom.SromData);
}

unsigned short CalcSromCrc(unsigned char *SromData)
{
#define POLY 0x04C11DB6L

   unsigned long crc = 0xFFFFFFFF;
   unsigned long FlippedCRC = 0;

   unsigned char CurrentByte;
   unsigned Index;
   unsigned Bit;
   unsigned Msb;
   int i;

   for (Index = 0; Index < DATA_LEN; Index++)
   {
      CurrentByte = SromData[Index];

      for (Bit = 0; Bit < 8; Bit++)
      {
         Msb = (crc >> 31) & 1;
         crc <<= 1;

         if (Msb ^ (CurrentByte & 1))
         {
            crc ^= POLY;
            crc |= 0x00000001;
         }

         CurrentByte >>= 1;
      }
   }
   for (i = 0; i < 32; i++)
   {
      FlippedCRC <<= 1;
      Bit = crc & 1;
      crc >>= 1;
      FlippedCRC += Bit;
  }
   crc = FlippedCRC ^ 0xFFFFFFFF;
   return (crc & 0xFFFF);
}
```

B

ID Block CRC Calculation Algorithm

This algorithm calculates the CRC, which sums the serial ROM header. The serial ROM header contains 9 words and is read when the chip is reset. If the CRC result of these 9 words equals 0, it means that the data has been read correctly.

The CRC contains 8 bits and its polynomial is $X8 + X2 + X1 + 1$. Note that unlike a normal CRC, this CRC is calculated on the data stream from the most significant bit to the least significant bit. It is done this way because the serial ROM data flows in this manner.

The predefined serial ROM header is as follows:

Word Number	Definition
0	Subsystem vendor ID.
1	Subsystem ID.
2	CIS pointer, low word.
3	CIS pointer, high word.
4	Reserved, value equals 0.
5	Reserved, value equals 0.
6	Reserved, value equals 0.
7	Reserved, value equals 0.
8	High byte reserved, value equals 0. Low byte equals CRC.

```
main()
{
#define POLY 0x6

   unsigned short DAT[9];

   int i,Word,n;
   char Bit;
   unsigned char BitVal;
   unsigned char crc;

   n=0;
   crc = -1;
   for (Word=0; Word<9; Word++)
   {
      for (Bit=15; Bit>=0; Bit--)
      {
         if ((Word == 8) && (Bit == 7))
         {
            /*
            ** Insert the correct CRC result into input data stream in place.
            */
            DAT[8] = (DAT[8] & 0xff00) | (unsigned short)crc;
            break;
         }

         n++;

         BitVal = ((DAT[Word] >> Bit) & 1) ^ ((crc >> 7) & 1);
         crc = crc << 1;
         if (BitVal == 1)
         {
            crc ^= POLY;
            crc |= 0x01;
         }
      }
   }
}
```

C

Technical Support and Ordering Information

C.1 Obtaining Technical Support

If you need technical support or help deciding which literature best meets your needs, call the Digital Semiconductor Information Line:

United States and Canada	**1–800–332–2717**
Outside North America	**+1–508–628–4760**

C.2 Ordering Digital Semiconductor Products

To order the Digital Semiconductor 21140A Fast Ethernet LAN Controller and for more information about an Evaluation Board, contact your local distributor.

The following table lists some of the Digital Semiconductor products available from Digital. To obtain a Digital Semiconductor Product Catalog, contact the Digital Semiconductor Information Line.

Product	Order Number
Digital Semiconductor 21140A Ethernet LAN Controller	21140–AC
Digital Semiconductor 21140A Evaluation Board Kit	21A40-TX
Digital Semiconductor 21041 PCI Ethernet LAN Controller	21041–AB
Digital Semiconductor 21041 Evaluation Board Kit	21A41–01

C.2 Ordering Digital Semiconductor Products

Ordering Digital Semiconductor Literature

The following table lists some of the available Digital Semiconductor literature. For a complete list, contact the Digital Semiconductor Information Line.

Title	Order Number
Digital Semiconductor 21140A Fast Ethernet LAN Controller Product Brief	EC–QN7MB–TE
Digital Semiconductor 21140A Fast Ethernet LAN Controller Data Sheet	EC–QN7PC–TE
Digital Semiconductor 21140A Fast Ethernet LAN Controller Hardware Reference Manual	EC–QN7NC–TE

Ordering Third-Party Literature

You can order the following third-party literature directly from the vendor:

Title	Vendor
PCI System Design Guide	PCI Special Interest Group 1–800–433–5177 (U.S.) 1–503–797–4207 (International) 1–503–234–6762 (FAX)
PCI-to-PCI Bridge Architecture Specification Revision 1.0	PCI Special Interest Group 1–800–433–5177 (U.S.) 1–503–797–4207 (International) 1–503–234–6762 (FAX)
PCI Local Bus Specification, Revisions 2.0 and 2.1 PCI BIOS Specification, Revision 2.1	PCI Special Interest Group 1–800–433–5177 (U.S.) 1–503–797–4207 (International) 1–503–234–6762 (FAX)