Using Motorola's Fast Static RAM CAMs with the MPC860T's Media Independent Interface

Roy Bieda, FAE, Freescale Semiconductors
Jim Soroka, FAE, Arrow Electronics

1. Introduction

Not too long ago we were happy to have a 10 Mbps LAN connection to our desktop computer. However, given the growing popularity of the Internet and other services we find ourselves needing even more bandwidth. 100 Mbps Ethernet promises to deliver the bandwidth we need without the need to rewire or rewrite our existing protocol stacks.

Freescale's new MPC860T Fast Ethernet Communications Controller is an enhanced version of the MPC860 PowerQUICC™ family. In addition to all existing MPC860MH capabilities, the MPC860T includes an additional 10/100BaseT Ethernet channel capable of handling 10/100Mbps operation. This makes the 860T a perfect candidate for LAN/WAN routers/bridges.

The transition from 10Base to 100Base Ethernet does present some implementation differences. Given the relatively low data rate of 10Base Ethernet, address filtering in bridge applications was often performed by the host CPU in software. The higher data rate of 100Base Ethernet doesn't allow the host CPU in the bridge enough time to accept or reject frames. There is a need to implement a hardware method to accept or reject frames. This need can be filled by the addition of a Content Addressable Memory (CAM) to the 860T's Ethernet Media Access Controller (MAC).

The operation of a router or bridge would be to simply present to the CAM a 48-bit MAC address obtained from an incoming frame. The CAMs' job is to search its contents and indicate to the MAC if the incoming address was present or not within the CAM, indicating acceptance or rejection of the frame.

Although CAMs have been available for years, they were often too expensive to justify their use. The MCM69C232 CAM addresses this problem with a different approach to CAM technology. By combining logic with fast and dense SRAM memory cores Freescale is able to produce cost-effective, fast, and deep CAMs. The MCM69C232 is 4k x 64 CAM and the MCM69C432 is a 16k x 64 CAM.

2. Purpose

The purpose of this note is to illustrate the connection of a CAM in an 860T system. The CAM would exist between the Physical Interface Device (the PHY) and the MAC. The 860T's MAC has an industry standard interface to the PHY known as a Media Independent Interface. What is needed is a single logic device that can interface the CAM to the MII / PHY connection and signal the 860T to accept or reject an incoming frame. This logic is easily implemented in a FPGA. We will discuss the operation of this FPGA in this note.

3. Media Independent Interface Description

The Media Independent Interface is a standard to facilitate the transfer of data between the 860T and the physical layer interface. The MII interface is a four bit (nibble) wide data path interface that runs at 25 MHz for a 100 Mbps networks and at 2.5 MHz for 10 Mbps networks.

The MII receive clock is generated by the PHY and is sent to the 860T on the RX_CLK pin. The data from the PHY on RXD[3:0] is presented synchronous to the rising edge of RX_CLK. The receive process is started when RX_DV is asserted and remains asserted for the entire receive frame length.

In order to support external CAM address filtering on the MII interface it will be necessary to gate the RX_DV signal generated by the PHY. A signal, RX_DV_860T, will be generated by the FPGA for the 860T. The FPGA, upon a reject (no address match) from the CAM, will negate RX_DV_860T. The 860T will interpret this as a runt (<64 bytes) frame and subsequently discard the frame. The CAMs' match time is far faster than the time it takes to receive 64 bytes.

Relevant Signals:

Pin Name	Pin Function	Type	No. of Pins	Comments
RX_CLK	Receive Clock	Input	1	Recovered receive clock from PHY
RXD[3:0]	Receive Data	Input	4	Received nibbles
RX_DV	Receive Data Valid	Input	1	RXD[3:0] has valid data
RX_ER	Receive Error	Input	1	PHY has detected an error in the current frame

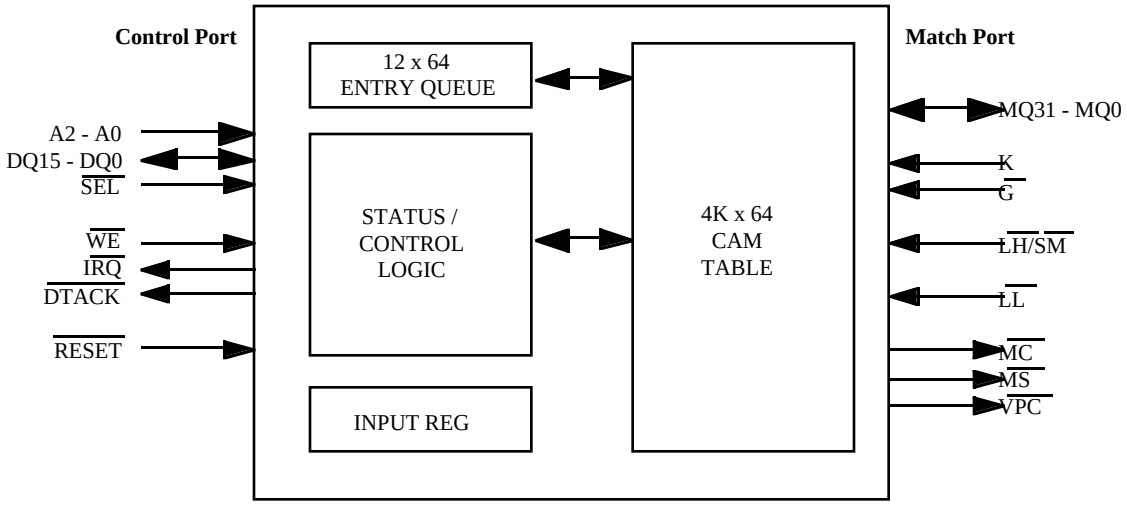
4. CAM Description

In its basic operating mode, the MCM69C232 reads a data input word and compares it to all the entries in its CAM table. Whether a match is found or not, the /MC pin is asserted after the comparisons have been made. If a match is found, the /MS pin is asserted, and the data associated with the matching entry is output on the MQ bus. If no match is found, the MQ bus remains in a high impedance state to facilitate depth expansion via the cascading of multiple CAMs.

The CAM is prepared for match operations by writing to data and instruction registers via the control port. Since we are only interested in matching 48 bits it will be necessary to set the global mask to ignore 16 bits of the 64 bit match field.

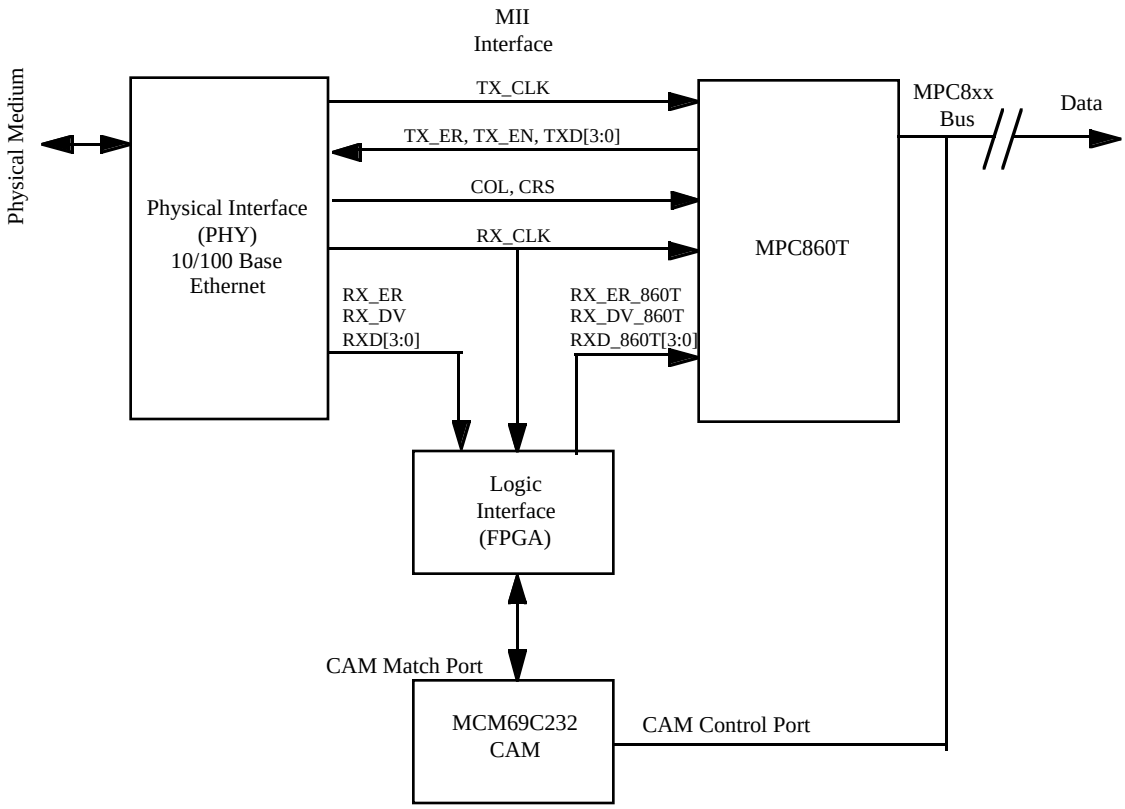
Relevant Signals:

Pin Name	Pin Function	Type	No. of Pins	Comments
/LH//SM	Latch High / Start Match	Input	1	Initiates match sequence on match data present on MQ31-MQ0
/LL	Latch Low	Input	1	Latches low order bits if match width is >32 bits
/MC	Match Complete	Output	1	Open Drain, Pull-ups required
/MS	Match Successful	Output	1	Open Drain, Pull-ups required
MQ31 - MQ0	32-bit common I/O CAM data	Input / Output	32	Used for input of match compare data values



MCM69C232 CAM BLOCK DIAGRAM

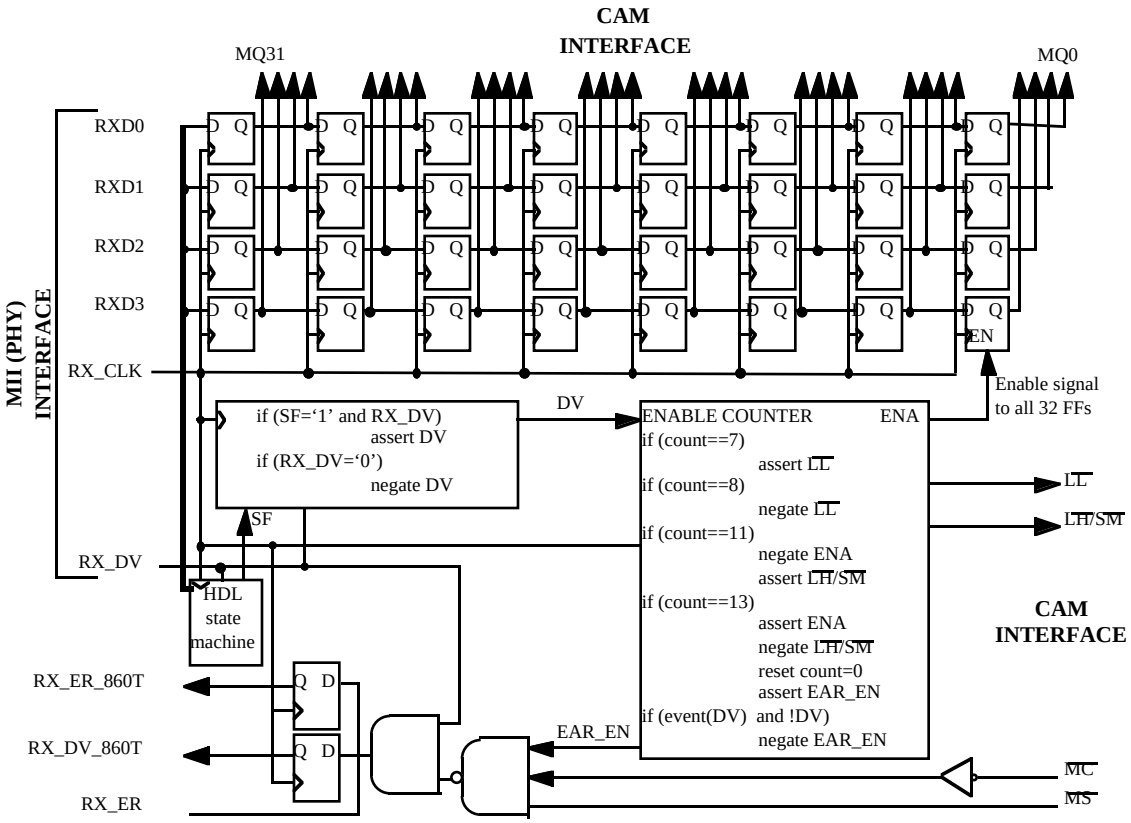
6. System Block Diagram



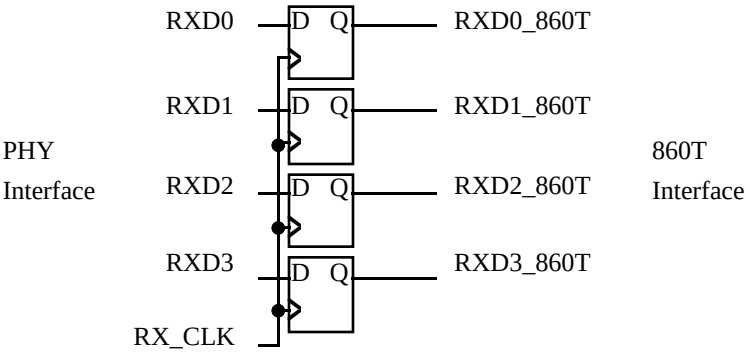
The FPGA uses several signals of both the CAM, 860T, and the PHY. The interface between the MPC860T and the CAMs' control port is described in the following application note: MPC860SAR Microprocessor ATM CAM Interface Example, available on the internet at: http://www.mot.com/netcomm/aesop/mpc8XX/860/860SARCAM.pdf.

However, there is one major difference from the MPC860SAR Microprocessor ATM CAM Interface Example application and this paper, it is possible to have concurrent match and control port operations. It may be desirable to use the CAMs' DTACK signal to terminate a control port access. The 860s' chip select option register should be changed to indicate an externally generated cycle termination. The CAMs' DTACK signal will need to be passed through the FPGA to provide the 860 with a one clock TA (transfer acknowledge) that meets the 860s' setup and hold times.

7. Logic Interface Diagram



The process of gating the RX_DV signal to the 860T (RX_DV_860T) introduces a one clock delay. The following circuit is used to ensure the 860T receives RXD[3:0] nibbles in phase with the one RX_CLK delayed RX_DV.



RX_DV	EAR_EN	/MC	/MS	RX_DV_860T	Comments
0	X	X	X	0	No valid receive data.
1	0	X	X	1	Address not yet latched.
1	1	1	X	1	Address compare in process.
1	1	0	0	1	Match successful. No Reject.
1	1	0	1	0	Match fail. Reject.

8. Operation of the Logic Interface

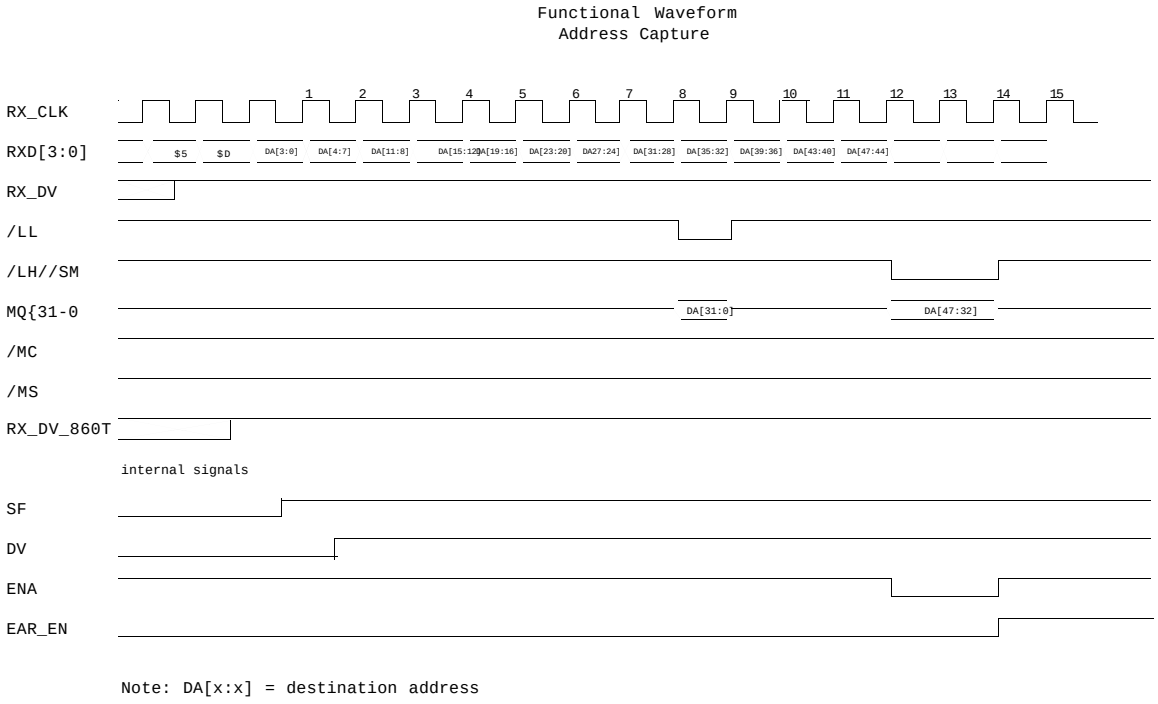
Receive data from the PHY is continuously clocked into the FPGA via RXD[3:0] and the RX_CLK signals. RX_DV is used to indicate that valid receive data is being presented on the RXD[3:0] signals. RX_DV remains asserted until after the last nibble of the CRC is driven on RXD[3:0]. RX_DV, when negated, resets the FPGA to its initial state.

It will be necessary to detect a start of frame (SF) in order to detect and capture the destination address within the FPGA and present it to the CAM. Ethernet specifies that a preamble bit pattern of one or more 0101 ($5) nibbles followed by a 1101 ($D) nibble signifies the start of frame (destination address). A small piece of hardware description language (HDL) can be used to detect the start of a frame and generate the SF signal using only a three flipflop state machine:

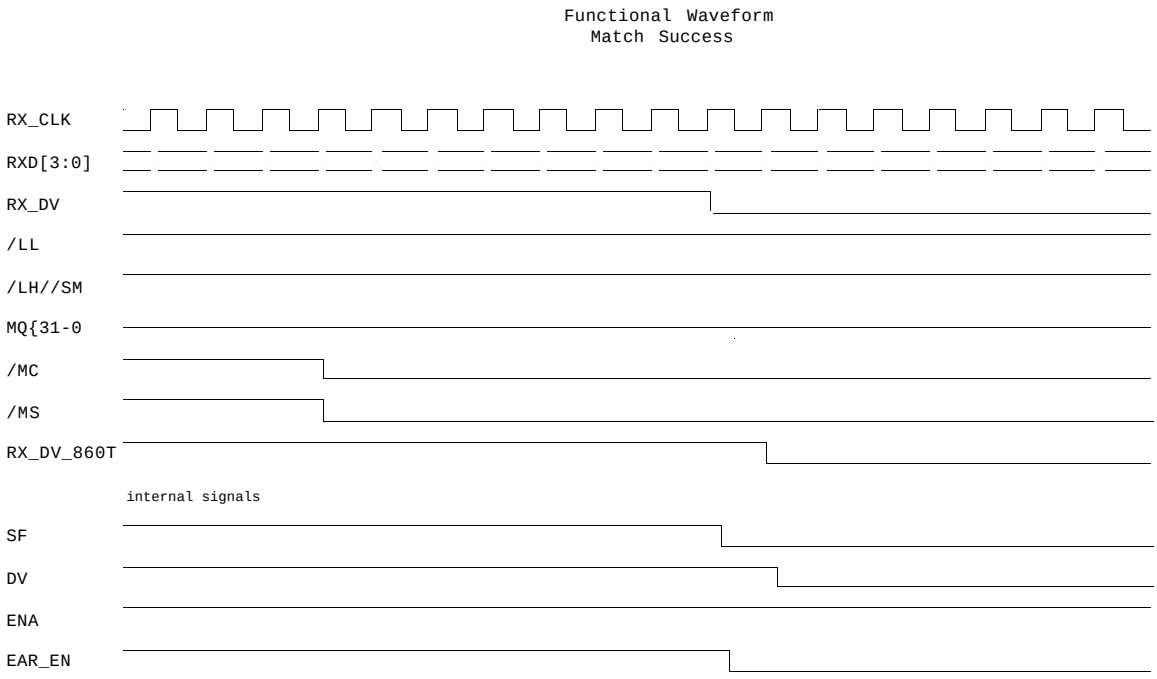
```
entity SAMPLE is
      port(
RX_CLK:     in std_logic;
RX_DV:      in std_logic;
RXD_bus:    in std_logic_vector(3 downto 0);
SF          out std_logic;
);
end SAMPLE

architecture archSAMPLE of SAMPLE is
      type OPERATINGSTATE is (Idle, Got5, GotD);
      signal OPState:   OPERATINGSTATE;
begin
      process (SCLK)
      begin
            if SCLK'Event and SCLK = '1' then
             case OPState is
                  when Idle =>
                   if RX_DV = '1' and Data = "0101" then
                    OPState <= Got5;
                   endif;

                  when Got5 =>
                   if RX_DV = '0' then
                        OPState <= Idle;
                        SF <= '0';
                   elsif Data = "1101" then
                    OPState <= GotD;
                    SF <= '1';
                   elsif Data /= "0101" then
                    OPState <= Idle;
                   end if;

                  when GotD =>
                   if RX_DV = '0' then
                        OPState <= Idle;
                        SF <= '0';
                   end if;
             end case;
            end if;
      end process;
end archSAMPLE;
```

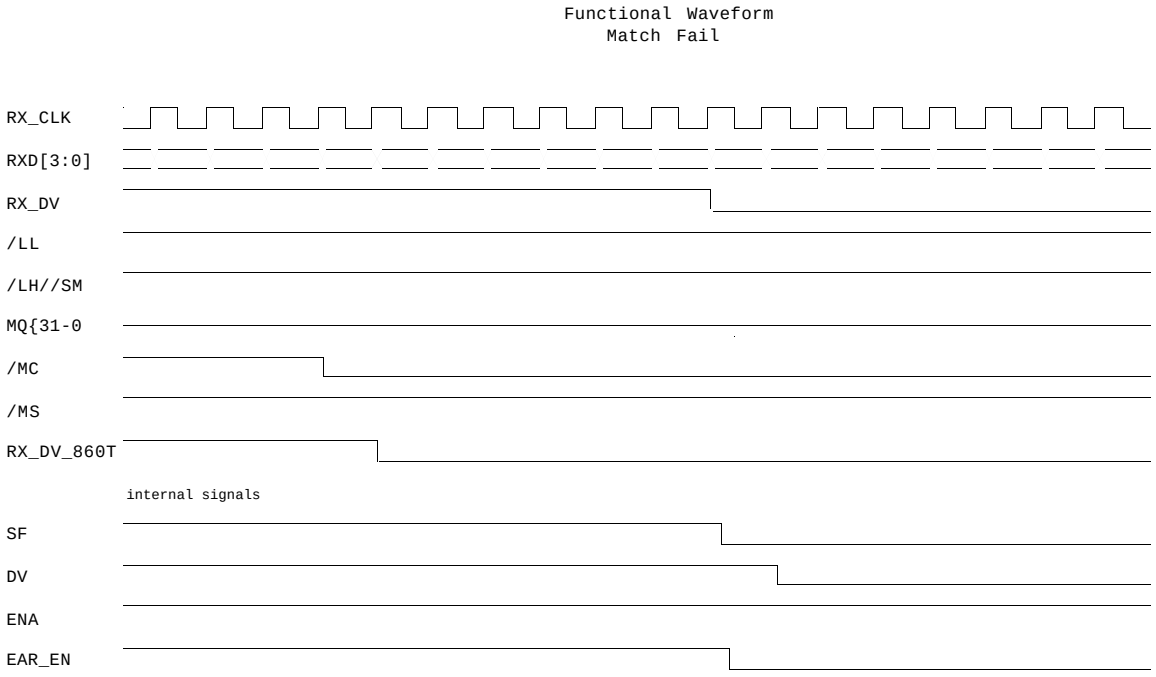
An initial rising edge of the SF signal indicates that a start of valid data is present on the RXD[3:0] pins. The destination address, 48 bits wide, is then shifted into the FPGA, four bits per clock, and then presented to the CAM 32 bits at a time via the /LL and /LH//SM signals. The /LH//SM signal then starts the match process of the CAM. The CAM will then respond with MC and if the destination address was not in the CAM, the MS will not assert. The FPGA then signals rejection of the frame to the 860T via its RX_DV_860T signal.

9. Simulation Diagram for Match Successful and Unsuccessful



Note: DA[x:x] = destination address





10. Summary

With the addition of the MPC860T to Motorola's family of embedded communications controllers, coupled with Motorola's advent of the cost effective CAM, creates new opportunities to build faster and more cost effective network devices. The addition of a small amount of logic makes it is easy to interface Motorola's MCM69C232 CAM to a 860T/PHY Ethernet interface. The logic can be easily implemented in a FPGA such as

Motorola's MPA1016, the smallest and lowest cost member of Motorola's fine grain FPGA family.

11. References

1. MCM69C232 4Kx64 CAM, Freescale Inc., 1997.
2. MPC860T Fast Ethernet Controller, Freescale Inc., 1997.

How to Reach Us:

Home Page:
www.freescale.com

E-mail:
support@freescale.com

USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
+1-800-521-6274 or +1-480-768-2130
support@freescale.com

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
support@freescale.com

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com