MOTOROLA
Semiconductor Products Inc.

AN-820
Application Note



# HARDWARE CONSIDERATIONS FOR DIRECT MEMORY ACCESS USING THE MC6809 MICROPROCESSOR UNIT AND MC6844 DMA CONTROLLER

Prepared by:
David Smith
MOS Microprocessor
Systems and Applications

## Introduction

This application note discusses hardware considerations which must be applied to any Direct Memory Access (DMA) design used with the MC6809 Microprocessor Unit and the MC6844 Direct Memory Access Controller (DMAC). Additionally, any circuit using the DMAC requires some form of "dead-cycle" protection during the time in which the bus control is being transferred from processor to DMAC and back. It is assumed that the user has an intimate knowledge of M6800 Family processors and peripherals; and, in general, the concept of DMA. Figure 1 contains a block diagram showing an application for the MC6844 DMAC used with the MC6809 MPU. For additional DMA or microprocessor information, refer to the respective data sheets and the MC6809 Users Manual.

## MC6809 Requirements

DMA design, using the MC6809, is made easier by the inclusion of the DMA/Bus request ($\overline{\text{DMA/BREQ}}$) feature on-chip. When the $\overline{\text{DMA/BREQ}}$ line is asserted, the MC6809 relinquishes control of the bus by: (1) setting its address, data, and control lines to the high-impedance state; and (2) transferring control to the DMAC by asserting a Bus Grant signal (BA = 1 and BS = 1). This DMA/Bus request feature has timing constraints which must be considered by the designer in order to stay within published specifications.

As shown in Figure 2, the $\overline{\text{DMA/BREQ}}$ signal must occur at least $t_{PCSD}$ before the falling edge of E. This guarantees that the transfer begins on the next falling edge of the E clock. In addition, $\overline{\text{DMA/BREQ}}$ must not be allowed to change during the last 120 ns of this E cycle. If this timing is violated, the MC6809 could enter an undefined state. A simple sync circuit can be made, using a D-type flip-flop as shown in Figure 3. This circuit causes all bus transfer requests to be synchronized with Quadrature Clock Q. Although the MC6844 is shown as an example throughout the application note, the same conditions must be followed when using the MC6809, regardless of the DMAC used.

## DMA Dead-State Protection

Most DMA circuits have inherent properties (such as noise susceptibility) which causes difficulty in the exchange of bus control. The high-impedance approach (using three-state buses) has particular constraints in the area of address timing. Refer to Figure 4. When the MPU places its lines into the high-impedance state (relinquishes the bus) and before the DMAC assumes control of the bus, there is a period of time in which the bus lines are not driven. During this "exchange of bus control" time, the bus lines are extremely susceptible to noise. This condition could cause unwanted reads and writes during the bus control transfer. To alleviate this condition, a means of protecting memories and peripherals during transfer must be provided.

One method to provide this protection is to generate a signal called $\overline{\text{DMAVMA}}$, as shown in Figure 5. This signal line (Direct Memory Access Valid Memory Address) is then used as one of the inputs for the memory decoding scheme. Thus, when the $\overline{\text{DMAVMA}}$ line goes high, the memory cannot be enabled. The $\overline{\text{DMAVMA}}$ signal is the result of Bus Grant or DMA Grant (DGRNT, BA = 1 BS = 1) being applied to an Exclusive OR; one direct input and one delayed input. As long as DGRNT does not change, $\overline{\text{DMAVMA}}$ is low and the system decoder is functional. When the MPU relinquishes the bus (places its lines in the high-impedance state) BA and BS both go high, causing DGRNT to change to a high. The DGRNT high causes the $\overline{\text{DMAVMA}}$ line to go high until the next E-clock negative edge clocks the 74LS74 (both Exclusive OR inputs become equal): causing the

$\overline{\text{DMAVMA}}$ line to again go low. Thus, during the time shown as "DEAD" in Figure 4, the system cannot be enabled and the memories and peripherals are protected during the exchange of bus control. The above description pertained to a change from MPU to DMA. However, in Figure 4 the $\overline{\text{DMAVMA}}$ goes high for a corresponding time when the change is from DMA to MPU with one exception: the DGRNT signal change is from high-to-low; but, the effect on the Exclusive OR is the same.

One consideration is that the actual time when $\overline{\text{DMAVMA}}$ is high is not sufficient to provide full protection during the entire bus transition time. This is because the logical function Bus Grant (BA•BS; DGRNT in Figure 4) occurs simultaneously with the high-impedance transition of the address and control bus (see (1) of Figure 4). However, a period of 20 to 30 ns could elapse before $\overline{\text{DMAVMA}}$ goes high, resulting from propogation delay through the logic elements. During this 20-30 ns time, the memory is still functional but addresses are invalid; therefore, spurious read/write complications may be possible with some memories. One solution to this problem would be to enable the memory decoding with the E clock. One benefit of this is that most decoding must be synchronized with the falling edge of E anyway, since all data transfers occur then. One negative aspect to this is that the maximum memory access time is shortened to 450 ns (E high time).

Another solution (more oriented toward lengthening access time) would be to protect the memory, starting at the falling edge of E and continuing only throughout the 20-30 ns propogation delay period that the bus is unprotected. Since this unprotected state cannot be detected before the Bus Grant is sent, this period should be protected during every cycle. Then if a request occurs, and that cycle is a high-impedance (dead) state, the $\overline{\text{DMAVMA}}$ signal will cause protection to extend to the end of that cycle. A convenient period of time to use for that additional "pre-transition" protection is the first full quarter-cycle of E. The length of that signal (Figure 4) is long enough to cover the address change time (200 ns maximum) and still be short compared to the remaining 750 ns of cycle time. (Since the addresses are not guaranteed until 200 ns after falling E, and a 30 ns delay in bus transition is inherent, only 20 ns typically is lost in memory access time with this signal.) This first-quarter cycle signal, called FQ (Figures 4 and 6), enters the de-enable path through the OR gate along with $\overline{\text{DMAVMA}}$. It will keep $\overline{\text{DMAVMA}}'$ high during the first quarter of all cycles.

If the designer already uses fast memories, the decoding can be enabled with the E clock. This method does cut access time to 450 ns for all parts, but if this is a more cost effective solution, it will protect the bus during the entire first half of each cycle. This should cover the exchange of control during all DMA dead-states (this method could be used with some peripherals due to address setup times).

These interface procedures should be used with all DMA systems, and in particular with the MC6844 and MC6809. DMA ***does*** require exact timing and bus protection, and these methods will fulfill those requirements.

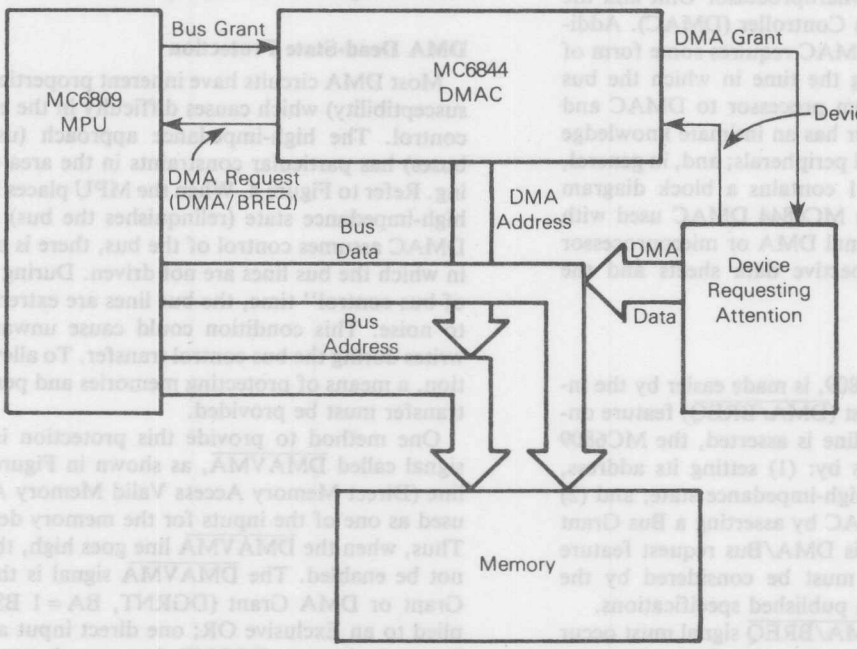



**FIGURE 1. Block Diagram Showing DMAC Used in System**

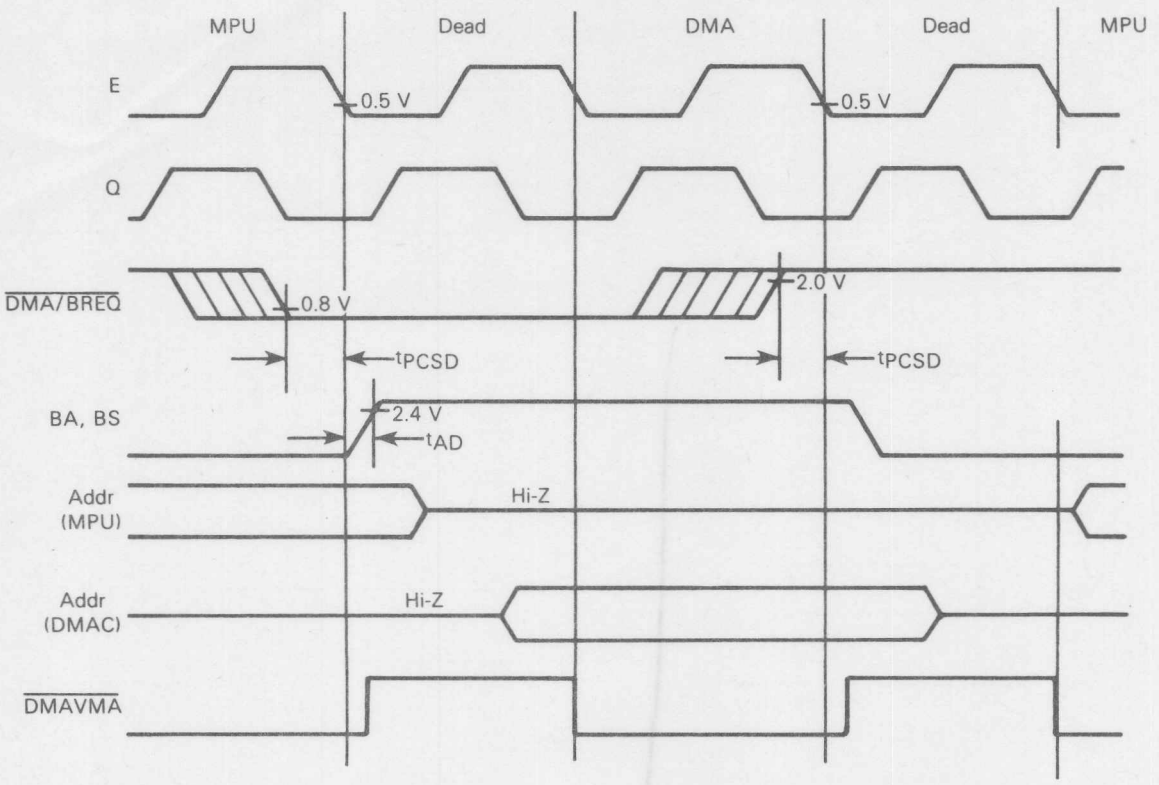



**FIGURE 2. MC6809 Bus Timing During MPU-DMA-MPU Bus Control Transfer**

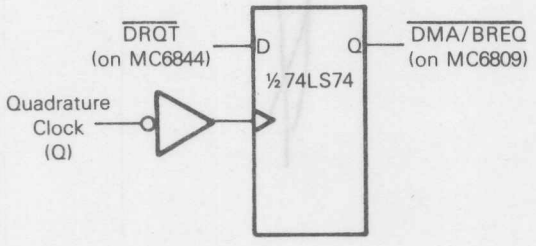
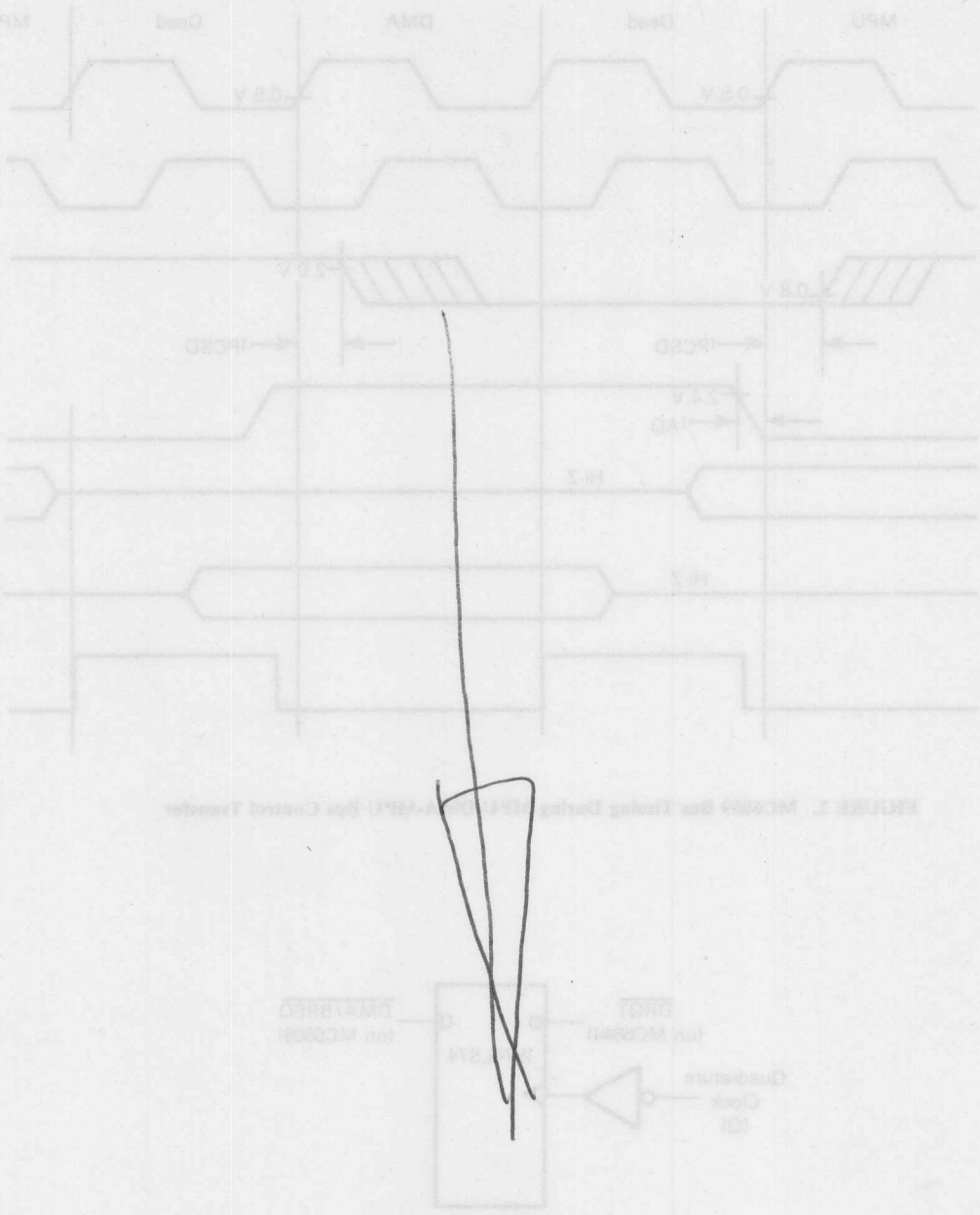



**FIGURE 3. Synchronizing DMA/BREQ with Q Clock**





**MOTOROLA** *Semiconductor Products Inc.*

Printed in Switzerland

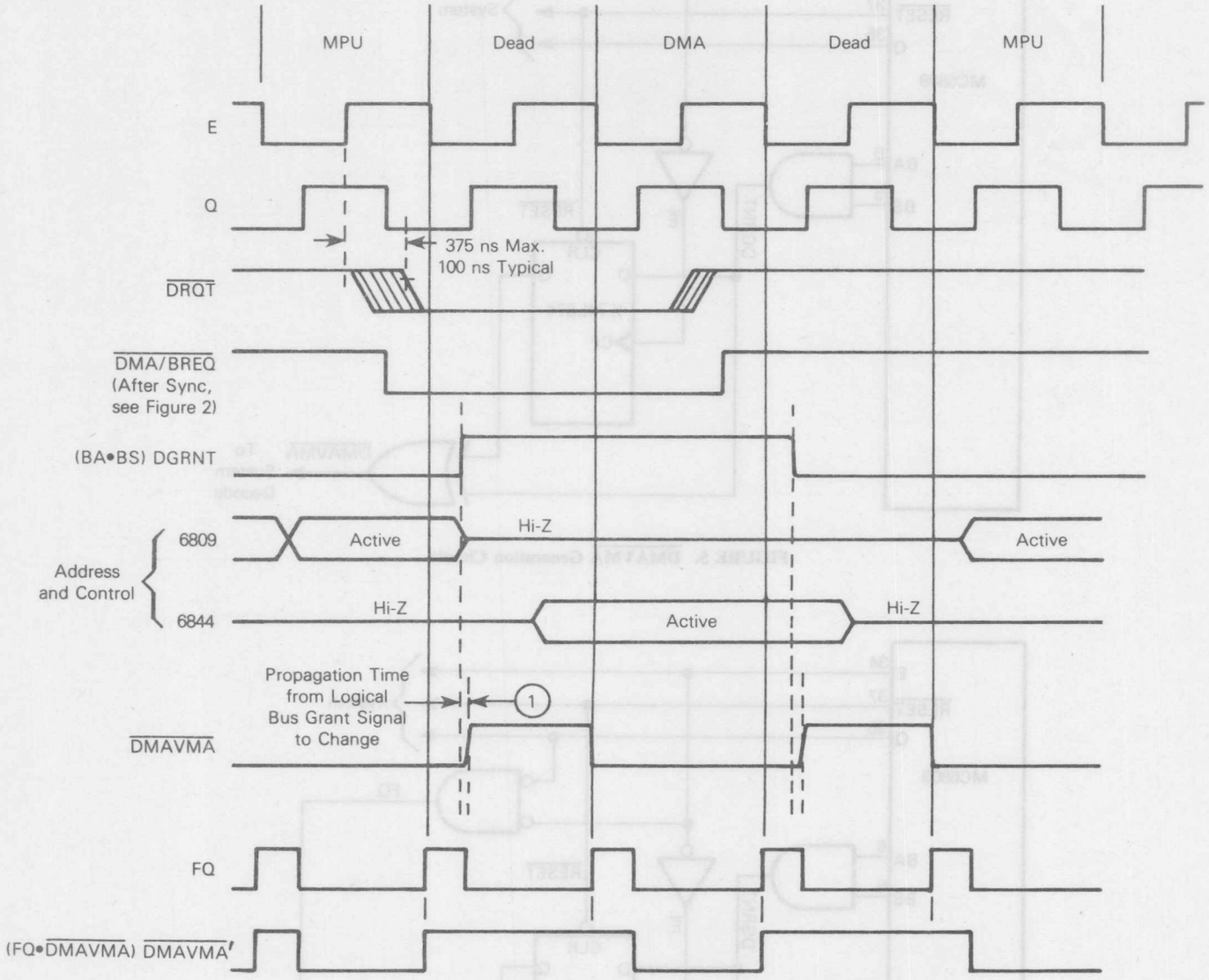



**FIGURE 4. System Timing**

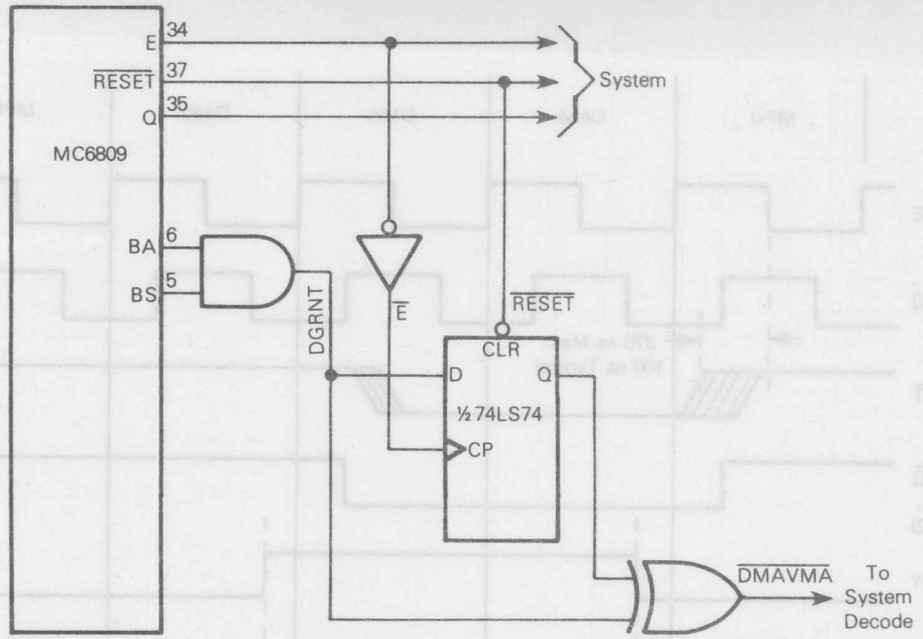



**FIGURE 5. $\overline{\text{DMAVMA}}$ Generation Circuit**

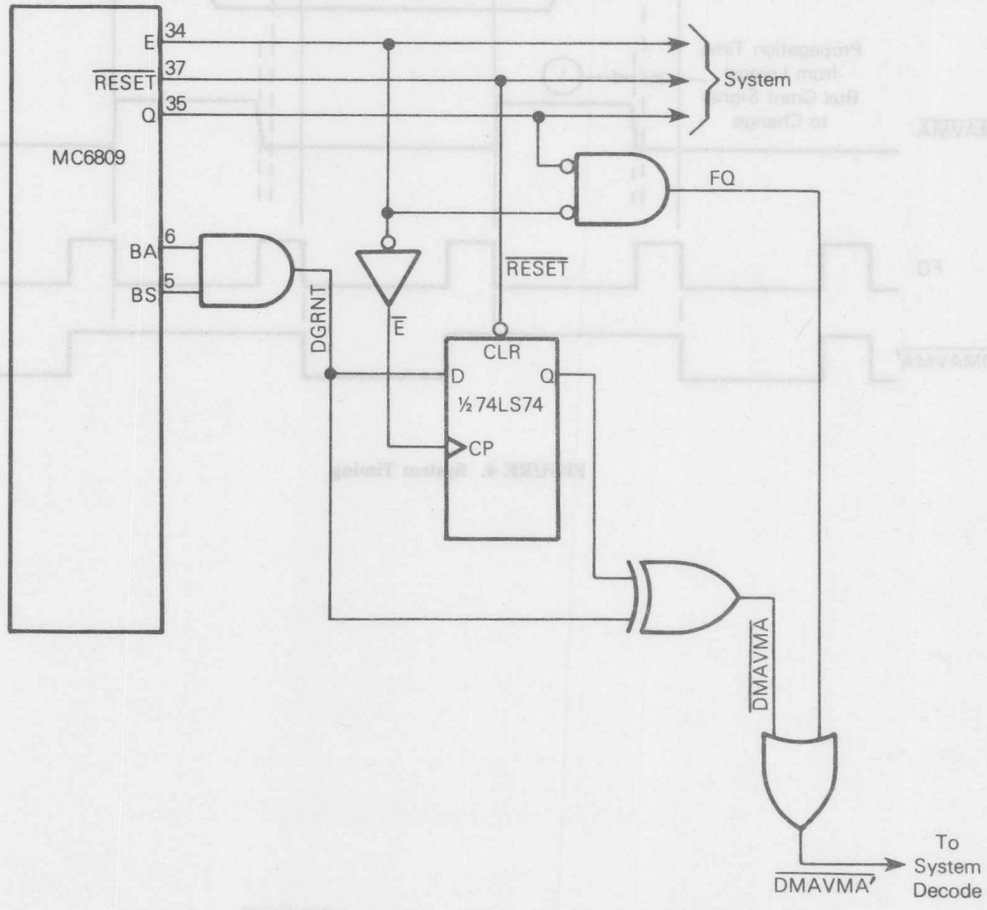



**FIGURE 6. $\overline{\text{DMAVMA}}'$ Generation With First-Quarter E-Cycle Protection**