Harris Semiconductor

APPNOTE

No. AN120.1 March 1997

Harris Digital

Interfacing the 80C286-16 with the 80287-10

Author: Ted Schaufelberger

Introduction

An important requirement in many systems is the ability to off-load numeric data processing. In an 80C286 system, this can be accomplished with an 80287 numeric co-processor. However, as processor speeds increase, it may become necessary to interface a high speed 80C286 processor with a lower speed 80287. This Document will briefly describe the interface between a 16MHz 80C286 (80C286-16) and a 10MHz 80287 (80287-10).

Interfacing the 80C286 with an 80287 can be broken down into three main areas:

(1) Bus control lines and data lines which coordinate and implement the flow of data between the two processors (i.e. the data lines, chip select lines, and read/write lines).

(2) The clock line(s), which drive the two processors.

(3) The four status lines through which the 80C286 and 80287 directly communicate status information to one another - comprised of the $\overline{\text{BUSY}}$, $\overline{\text{ERROR}}$, Peripheral Request (PEREQ), and Peripheral Acknowledge ($\overline{\text{PEACK}}$) lines.

Bus Control Lines

The various bus control and data lines in most systems would be coordinated by either a bus controller (such as the 82C288), or a bus controller subsection of an 80C286 oriented chip set. All requisite bus control timing between a 16MHz 80C286, and a 10MHz 80287 would then be handled by these devices (typically with one wait-state inserted to allow for the slower 80287-10).

Clock Lines

A system using a 16MHz 80C286 with a 10MHz 80287 requires separate clock lines for the two processors. The 32MHz system clock used by the 80C286-16 is too fast for the 80287 ±10, necessitating a dedicated clock driver for the 80287. This clock driver should supply a 10MHz clock to the 80287 with a 1/3 duty cycle to allow the 80287-10 to run at it's full 10MHz capability. One solution for providing this clock is the 82C84A-1, which meets this specification with either a 30MHz crystal at it's crystal inputs, or a 30MHz external frequency input to it's EFI pin. In either case, a 10MHz 1/3 duty cycle clock is output to the 80287. Note that when using a dedicated clock driver such as this, the CKM pin of the 80287 must be pulled up.

Status Lines

The 80C286 and 80287 communicate status information with one another through four signals; the $\overline{\text{BUSY}}$ line, the $\overline{\text{ERROR}}$ line, the peripheral request line (PEREQ), and the $\overline{\text{PEACK}}$ line.

The $\overline{\text{BUSY}}$ and $\overline{\text{ERROR}}$ lines can be connected from the 80287 to a 80C286-oriented chipset, or from the 80287 directly to a 80C286. In the case of the chipset interface, the signal timing between the 80287 and 80C286 is coordinated by the chipset. In the case of the direct 80287 to 80C286 interface, the signal timing is handled by the 80C286, and, since the signal flow direction is from the 80287 to the 80C286 (i.e. from the slower device to the faster device), no additional hardware is required to achieve proper timing.

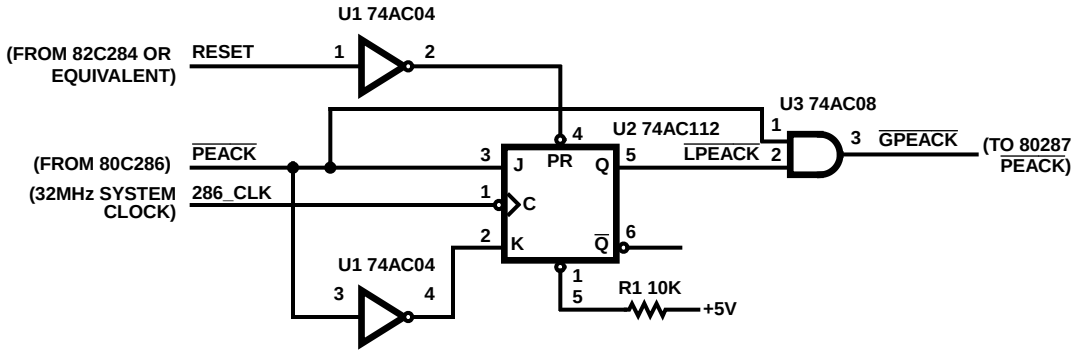

FIGURE 1. PEACK STRETCH CIRCUIT

The peripheral request (PEREQ) line should be connected directly from the 80287 to the 80C286, and again, since the signal flow direction is from the 80287 to the 80C286, no additional hardware is required.

The peripheral acknowledge ($\overline{\text{PEACK}}$) line is normally connected directly from the 80C286 to the 80287. In this case the signal flow direction is from the 80C286 to the 80287 (i.e. faster device to slower device), and the $\overline{\text{PEACK}}$ active time is not guaranteed to meet the requirements of the slower 80287-10. Worst case timing for the 80C286-16 reveals that $\overline{\text{PEACK}}$ output could be as short as 45.5ns (i.e. $\overline{\text{PEACK}}$ (min) = 45.5ns). The 80287-10 input requirement is $\overline{\text{PEACK}}$ (min) = 60ns.

The proper $\overline{\text{PEACK}}$ timing can be achieved using the circuit shown in Figure 1 comprised of a 74AC04, 74AC08, and a 74AC112. Referring to the timing diagram shown in Figure 2, it can be seen that this circuit effectively "stretches" the 80C286's $\overline{\text{PEACK}}$ output (in the form of $\overline{\text{GPEACK}}$) to 72.7ns, which satisfies the 80287-10 requirement.

The operation of the circuit shown in Figure 1 is as follows:

(1) The RESET signal (which is also applied to the 80C286) is used to initialize the 'AC112 to a known inactive state (Q = 1).

(2) When the 80C286 asserts the $\overline{\text{PEACK}}$ signal, the gated version of this signal ($\overline{\text{GPEACK}}$) is asserted with minimal delay (7.9ns through the 'AC08).

(3) On the falling edge of the 80C286 CLK at the beginning of Phase 2 of the T_S cycle, the low state of $\overline{\text{PEACK}}$ is clocked into the 'AC112. This effectively holds $\overline{\text{GPEACK}}$ low for an additional clock cycle longer than standard $\overline{\text{PEACK}}$ timing.

(4) On the falling edge of the 80C286 CLK at the beginning of phase 2 of the first T_C cycle, the high state of $\overline{\text{PEACK}}$ is clocked into the 'AC112, which then causes $\overline{\text{GPEACK}}$ to go inactive.

The net effect of this circuit operation is to extend the 80C286's Peripheral Acknowledge signal to the 80287-10 sufficiently to meet it's requirements.

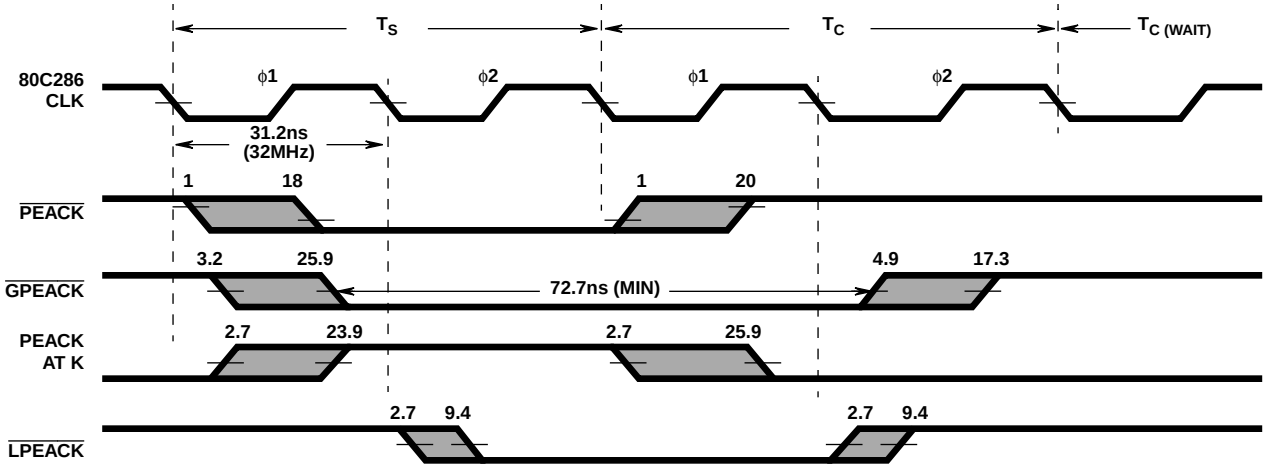

FIGURE 2. PEACK CYCLE TIMING