Minimize metastability in 50MHz state machines AN032

Engineers are excited to discover the PLUS405–55, a PLD state machine IC rated for a maximum operating frequency of 55MHz. It has a flexible architecture offering 65 product terms, and a programmable OR array driving 16 J-K flip-flops, 8 of which are buried (see Figure 1). This design allows the 64 product terms to realize 64 state transitions in a general state machine implementations. (State machines based on a counter will be implemented much more efficiently.) In order to estimate if a particular state machine will fit in the PLUS405, one need only count the state transitions and assure that there are fewer than 65! There are the remaining issues of number of states, inputs and outputs. The PLUS405 has 8 buried registers, allowing representation of 256 unique states. A dual complement array is available for the “ELSE” condition of state equations, and along with dual clocking capabilities allows two independent state machines to be synthesized on one IC.

Ease of design is further enhanced by SNAP, the PC-based PLD development tool. SNAP supports Boolean and State Equation entry of the design, simulation, and downloading of the programming information to a programmer. SNAP allows an abstract approach to design with PLDs, as the target device is not specified by the engineer until he is done fully integrating and simulating his efforts. After device selection, SNAP can back-annotate the design files with target silicon characteristics, allowing simulation of the actual device.

The engineer sets out to solve all his high speed state machine design problems armed with this new silicon and software, only to discover all this performance has its price. Studying the data sheet on the PLUS405–55 shows the following performance:

f_{MAX1}	55.6MHz minimum
Input Setup time	10ns minimum
Input Hold time	0ns minimum

The cycle time at 55MHz is roughly 18.2 nsec. The window during which data must be stable to guarantee no metastability is 10 nsec long. The difference between the setup and hold time, and the cycle time, is the allowed time interval for changes to occur. this example leaves 8.2ns for any changes.

From a system standpoint, this means the design engineer must be extremely careful in implementing his system, or he will violate the setup and hold specifications of the PLUS405–55. This can lead to metastable conditions in the state machine with several negative effects:

1. Jumps to undefined states. (May hang up system!)
2. Lengthened clock to Q times (slows down!).
3. Jumps to states out of proper sequence.

All of the above problems will yield a system that is unreliable, unpredictable and expensive in terms of servicing elusive bugs in the field.

The preceding analysis said nothing regarding asynchronicity. It is feasible to design the above system in a fully synchronous manner and have acceptable results. What about the system where known asynchronous inputs will be used in the state machine? Examples of common asynchronous signals are refresh request in DRAM controller applications and interrupts in a real-time control system. One approach to managing asynchronous inputs is to precede the state machine with a D-type flip-flop. This can serve as a synchronizing stage . . . or can it??? A simple analysis will explore the feasibility of using a simple synchronizing flip-flop.

A common Dual-D flip-flop frequently selected for this application in TTL high-performance systems is the FAST 74F74. The asynchronous data is fed into the D input of the flip-flop, and the Q output is fed into the logic input of the PLUS405 state machine. A common clock is used for both parts (see Figure 2). Based on current published data sheets, the 74F74 has a clock-to-Q time of 9.2ns maximum. The worst case setup time on the PLUS405–55 is 10ns. The minimum cycle time of the combined system is (9.2 + 10) ns, yielding a maximum clock frequency of 52MHz. Let’s assume for this example a desired system clock is 50MHz.

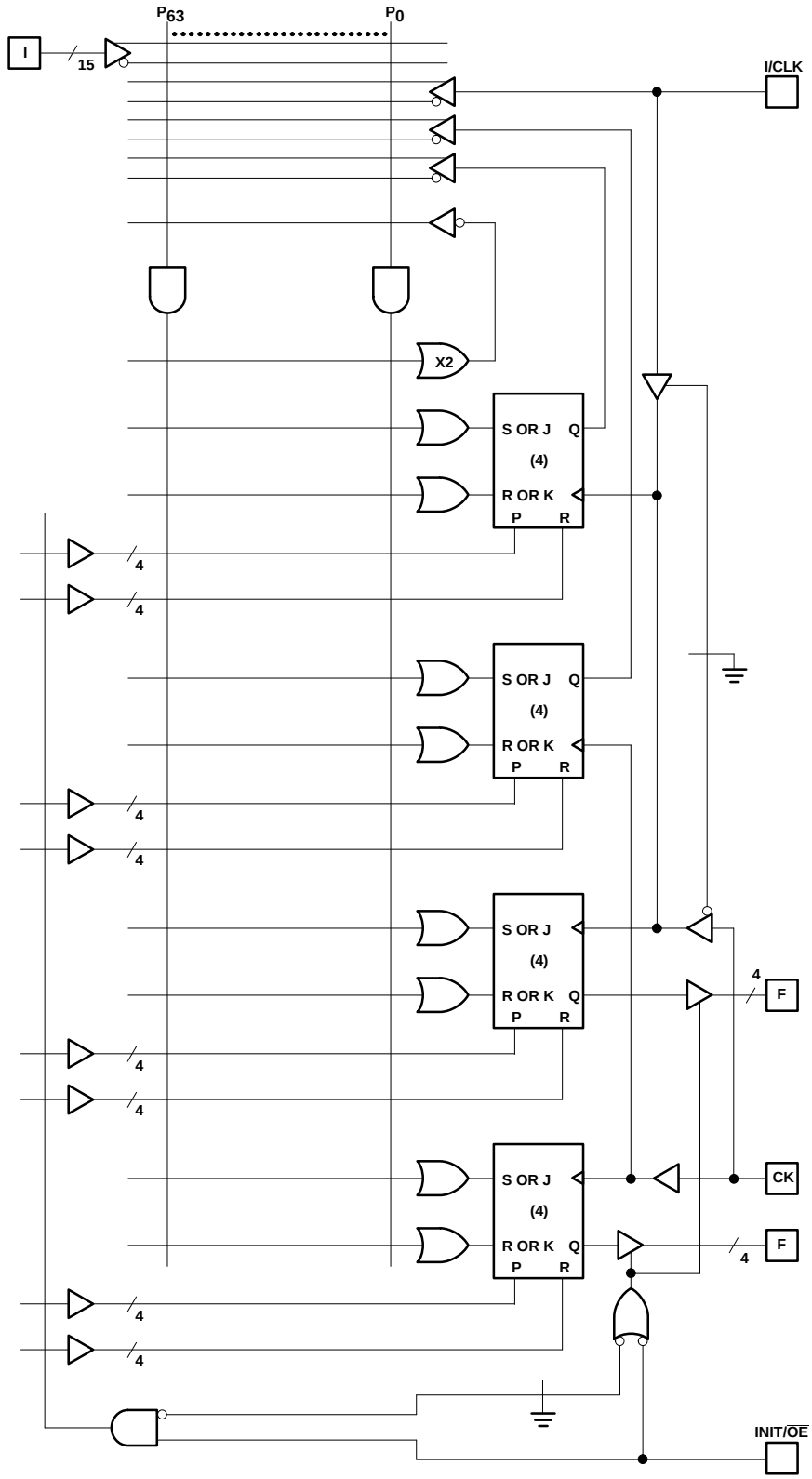

Figure 1. PLUS405–55 Functional Diagram

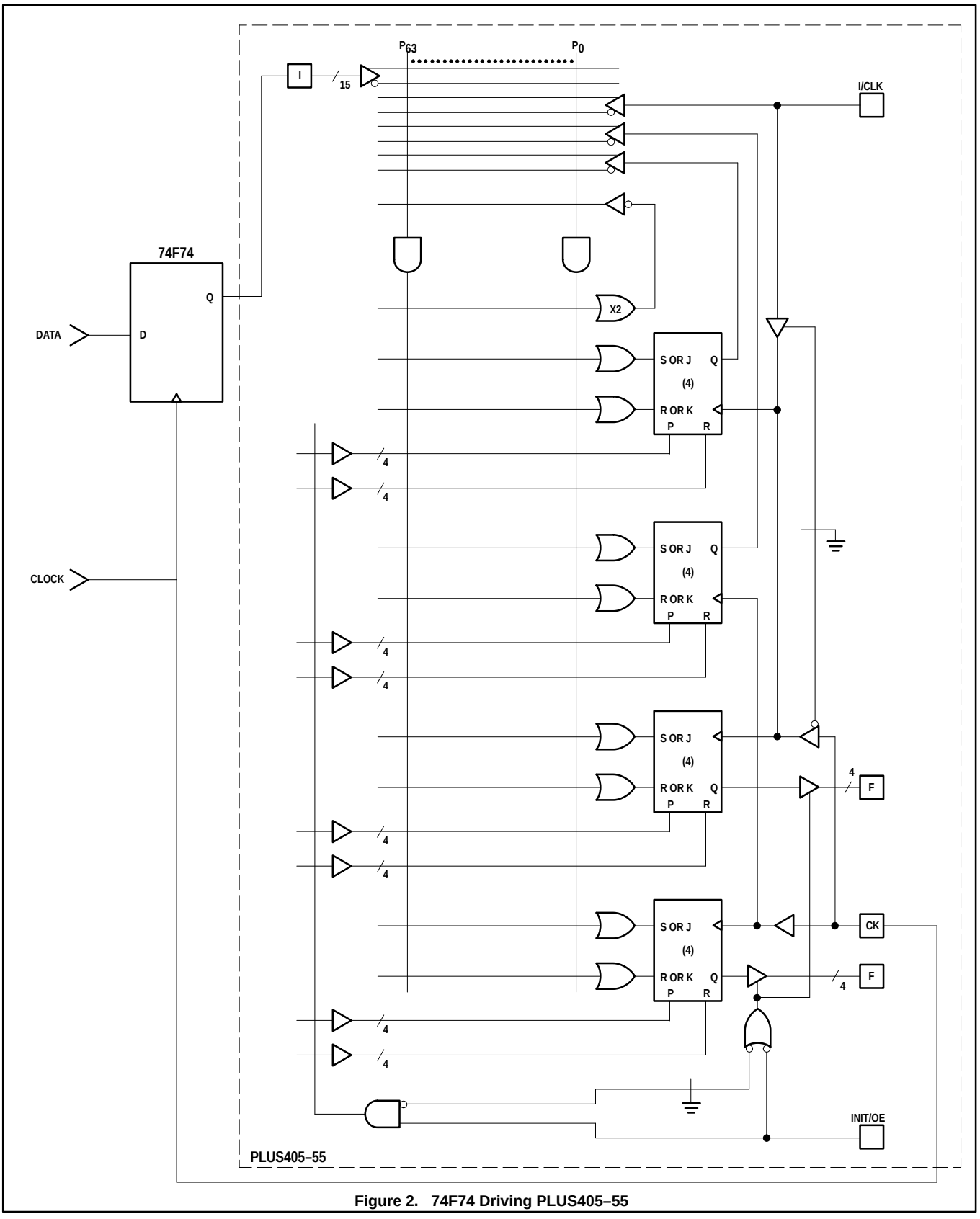

Figure 2. 74F74 Driving PLUS405–55

Minimize metastability in 50MHz state machines AN032

The important issue to examine in the timing diagram is the time that elapses between the end of the 74F74 clock-to-Q interval and the beginning of the PLUS405's setup time (see Figure 3). This is 20ns minus 10ns minus 9.2ns, which equals .8ns! At 50MHz, this combination is just able to work reliably on a worst case basis, assuming no instances of metastability. If metastable operation is encountered, the 800 picosecond window is the only time left in the clock cycle to resolve the situation. The next issue to examine is the mean time between failures (MTBF) for this system. From the work of Mr. Chaney, an equation which models metastable behavior is:

EQUATION 1. $MTBF = \exp(\tau r/\tau) / (T0 * f * a)$

{Explanation of above symbols}

MTBF is mean time between failures, in seconds.

τ**r** is the elapsed time before sampling the process

or

the time allotted for metastability to resolve.

τ is the "Metastability Time Constant".

T0 in seconds, the zero intercept of aperture time versus propagation delay. T0 indicates the propensity of a device to enter the metastable state.

f is the clock speed, in Hertz.

a is the transition rate of data being sampled (i.e., edges per second) in Hertz.

Assume for this discussion that the asynchronous input data is roughly 2MHz, meaning the edges that can cause metastability occur at a 4MHz rate. The system clock is assumed to be 50MHz, and the elapsed time before sampling is 10ns. (The sample time is calculated from the difference between the cycle time (1/50MHz = 20ns) and the setup time of the PLUS405–55 (10ns). The other parameters can be determined from measurements of an 'F74, or can be found in Mr. Chaney's paper. τ was found to be .4ns and T0 .2 milliseconds. Armed with a calculator and Equation #1, the MTBF for this particular scenario is calculated:

MTBF = exp(10/.4) / (.2e–3* 50e6 * 4e6) = 1.8 seconds

Clearly, this level of failure in any system is unacceptable. A better solution for this class of problem must be found!

Philips Semiconductors has recently introduced a new family of parts designed with metastability performance in mind. The first four members of this family are the 74F5074, 74F50109, 74F50728 and 74F50729. These are excellent general purpose flip-flops, but special attention has been paid to short setup and hold times, and fast clock-to-Q times. The output stage has also been designed with a balanced drive characteristic, leading to tight matching between rise and fall propagation delays, and matching of skews between other outputs. this makes them useful in clock driver applications also. Let's repeat the former calculation using the measured τ and T0 values for the 74F5074 used as a synchronizer (see Figure 4) ahead of the PLUS405–55.

τ = .135ns
T0 = 9.8 E 6 seconds

MTBF = exp(10/.135) / (9.8e6 * 50e6 * 4e6) = 75.46 e9 seconds

NOTE: For the reader's reference, a century is 3.154 e9 seconds.

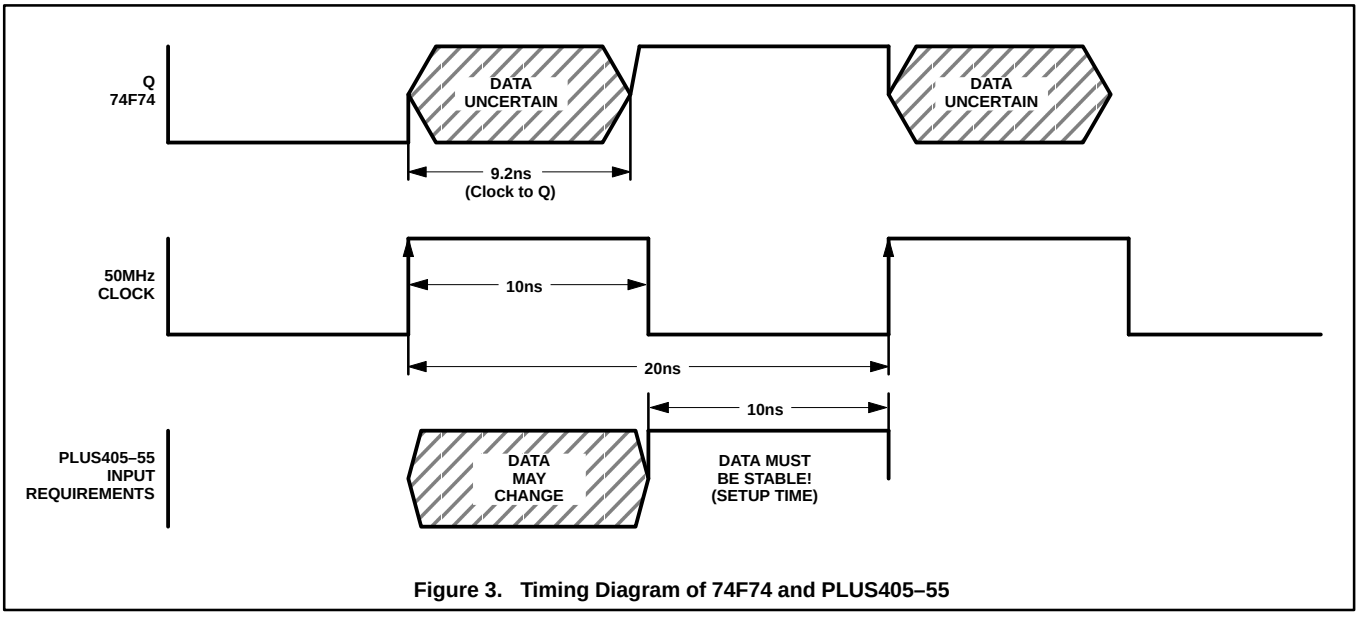

Figure 3. Timing Diagram of 74F74 and PLUS405–55

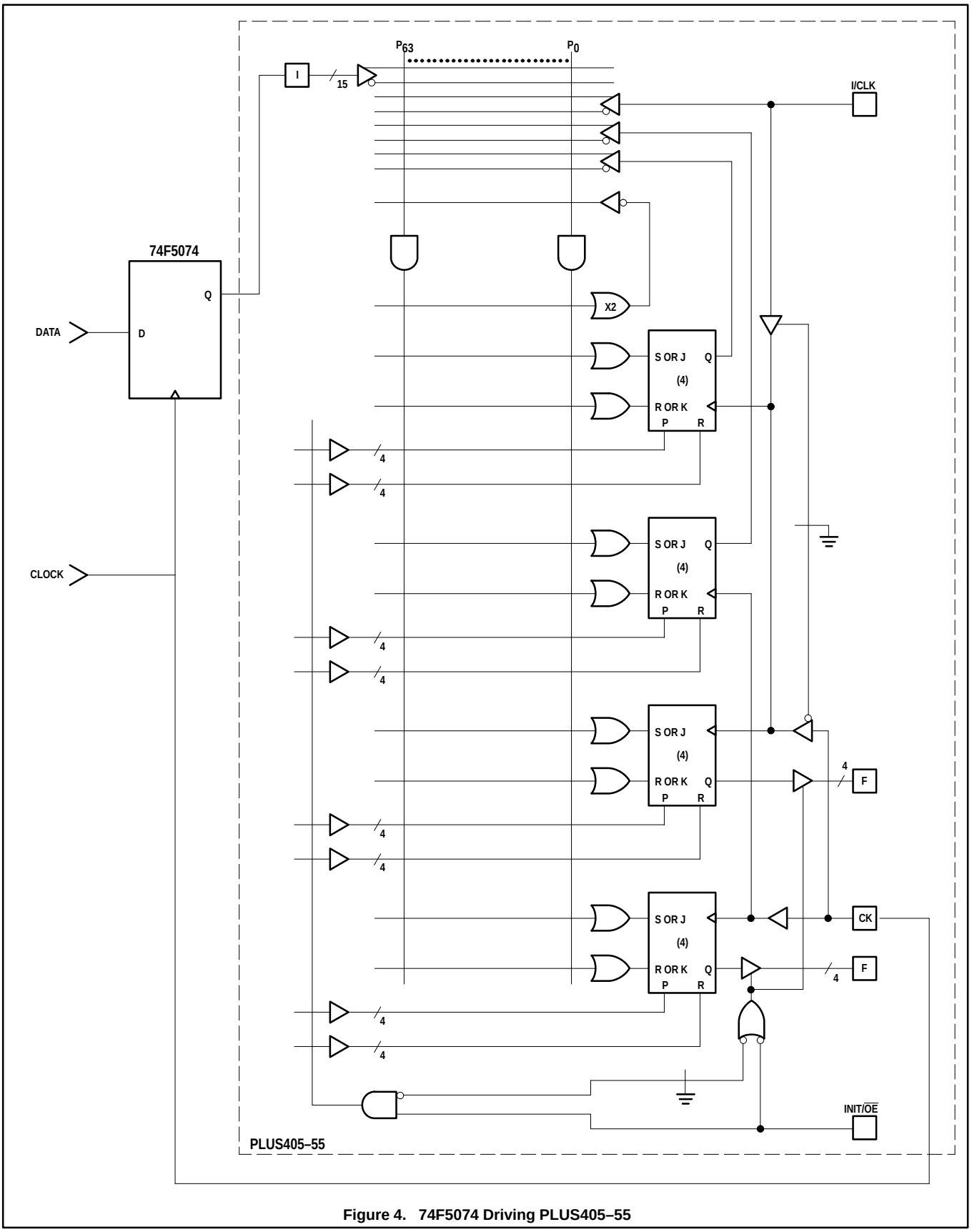

Figure 4. 74F5074 Driving PLUS405–55

Minimize metastability in 50MHz state machines AN032

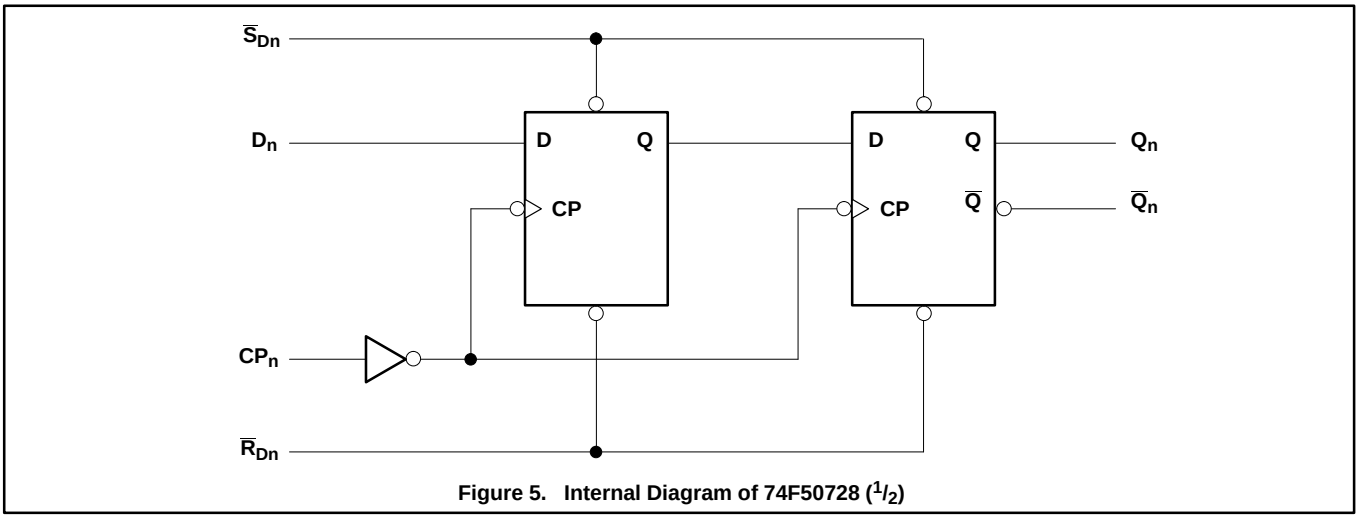

Figure 5. Internal Diagram of 74F50728 ($^1/_2$)

A system that was unreliable is now found to be quite acceptable by using the 74F5074. The major drawback to the synchronizing flip-flop solution is the added delay on the asynchronous signal before it enters the state machine. In the case of the 74F5074, this will amount to one clock cycle delay. For designs that demand the maximum in freedom from metastability, Philips Semiconductors has developed a product with cascaded D flip-flops for synchronizing applications. The 74F50728 (see Figure 5) will therefore introduce a two clock cycle delay into the system. It is pin compatible with the 74F5074 and 74F74 to allow retrofits on existing systems.

Calculation for the MTBF of a system using the 74F50728 is similar to the technique used earlier. In this case though, at least one entire clock cycle is used to resolve any metastability.

EQUATION 2. MTBF = exp(τr/τ) / (T0 * f * a)

{Explanation of above symbols}

All symbols are the same as EQUATION 1 with the exception of τr.

τr is the elapsed time before sampling the process or the time allotted for metastability to resolve. In the case of 74F50728, one entire clock cycle.

The flip-flops embedded in the ’F50728 are essentially the same as the flip-flops used for the ’F5074, therefore the same “Metastability Time Constant” τ, and T0, can be used in the calculation.

MTBF = exp(20/.135)/ (9.8e6 * 50e6 * 4e6) = 1.12 e43 seconds!

Now that the designer is comfortable with handling metastability, it is feasible to begin approaching the design of the system by stating a goal for MTBF and adjusting the state machine’s clock to meet the desired failure level.

Let’s assume out system is to have an MTBF of 5 years from metastability induced anomalies. The calculations would proceed as follows, assuming the same 2MHz data rate from our previous example:

MTBF = 5 years * (31.54 e6 seconds/year) = 157.7 e6 seconds

Setting up the equation to find the roots yields:

EQUATION 3. T(setup)/τ – 1/(f * τ) + 1n(T0*a*MTBF*f) = 0

(T(setup) is the setup time on the PLUS405)

Equation 3 is not solved using algebra, but simple numerical methods will allow easy solution, especially since we have a good initial guess for the value of f. (50 to 55MHz!) An HP 32S calculator was used to find the root of this equation by the following program:

				COMMENTS
	PRGM			start program entry
	GTO . .			go to top of memory
B01	LBL/RTN	{LBL}	B	label program as B
B02	INPUT	A		a, Data rate, Edges/Sec.
B03	INPUT	F		Clock frequency, Hertz
B04	INPUT	J		τ, seconds
B05	INPUT	M		MTBF, seconds
B06	INPUT	T		T0, seconds
B07	INPUT	U		T(setup), seconds
B08	RCL	A		begin calculation of 1n argument
B09	RCL x	T		
B10	RCL x	M		
B11	RCL x	F		
B12	LN			
B13	RCL	F		
B14	RCL x	J		
B15	1/x			
B16	+/–			change sign
B17	+			add
B18	RCL	U		
B19	RCL ÷	J		divide
B20	+			
B21	LBL/RTN	{RTN}		end, return from routine

To execute this program we must use the SOLVE capability on the calculator.

SOLVE {FN}	FN=		Prompt for label of function
B			
50 E 6	STO	F	load initial guess 50MHz
SOLVE {SOLVE}	SOLVE		prompt for unknown variable
F			frequency in this case!
A?	4.0 E 6		set edge rate
R/S			run
J?	135 E –12		set τ
R/S			
M?	157.7 E 6		set MTBF
R/S			
T?	9.8 E 6		set T0
R/S			
U?	10. E –9		setup time
R/S			

Minimize metastability in 50MHz state machines AN032

At this point the calculator will set off to find the root based on the initial guess and the desired conditions entered. The system clock speed determined from this technique is 52.16MHz.

Designers who are forced to deal with an uncertain system for the first time are uncomfortable with the idea that is is possible for the system to fail. Lower speed systems have been traditionally designed using worst case data sheet numbers to guarantee that the system will always work. As system clock speeds cross over 50MHz, meeting the setup and hold times becomes very difficult for TTL-based designs. The allowed time to resolve metastability gets shorter and the data stream edges become much more frequent, increasing the incidence of metastability. Faster systems demand that a design methodology based on statistics be used and the burden is now on the Engineer to manage likelihood of failure to acceptable levels. Persons defining high performance systems will need to specify goals for MTBF due to metastability. Usage of parts that have been characterized for metastability behavior will become mandatory in future systems. New parts, such as the 74F5074 and 74F50728 from Philips Semiconductors, which have published metastable traits and are pin compatible with other industry standard ICs, can make solving these problems as easy as plugging in a new part!

Dike, Charles, *"AN219, A Metastability Primer"*, Philips Semiconductors

Chaney, Thomas J., *"Measured Flip-Flop Responses to Marginal Triggering"* [*IEEE Transactions on Computing*, Vol. C-32, No. 12, December 1983, pp. 1207-1209]

Wakerly, John, *"Designers Guide to Synchronizers and Metastability, Part 1 and 2"* September 1987, *VLSI Design*