

# AN827

## The Technique of Direct Programming by Using a Two-Modulus Prescaler

Prepared by: PLL Applications

*The theory in this application note is still applicable, but some of the products referenced may be discontinued.*

### INTRODUCTION

The MC12009, MC12011, or MC12013 can be used as part of a variable modulus (divisor) prescaling subsystem used in certain Digital Phase-Locked Loops (PLL).

More often than not, the feedback loop of any PLL contains a counter-divider. Many methods are available for building a divider, but not all are simple, economical, or convenient in a particular application.

The technique and system described here offer a new approach to the construction of a phase-locked loop divider. In addition to using either the MC12009, MC12011, or the MC12013 variable modulus prescaler, this system requires an MC12014 Counter Control Logic Function, together with suitable programmable counters (e.g., MC4016s or SN74LS716s). Data sheets for these additional devices should be consulted for their particular functional descriptions.

### DESIGN CONSIDERATIONS

The disadvantage of using a fixed modulus ( $\div P$ ) for frequency division in high-frequency phase-locked loops (PLL) is that it requires dividing the desired reference frequency by  $P$  also (desired reference frequency equals channel spacing).

The MC12009/11/13 are especially designed for use with a technique called "variable modulus prescaling". This technique allows a simple MECL two-modulus prescaler to be controlled by a relatively slow MTTL programmer counter. The use of this technique permits direct high-frequency prescaling without any sacrifice in resolution since it is no longer necessary to divide the reference frequency by the modulus of the high-frequency prescaler.

The theory of "variable modulus prescaling" may be explained by considering the system shown in Figure 1. For the loop shown:

$$f_{out} = N \cdot P \cdot f_{ref} \quad (1)$$

where  $P$  is fixed and  $N$  is variable. For a change of 1 in  $N$ , the output frequency changes by  $P \cdot f_{ref}$ . If  $f_{ref}$  equals the desired channel spacing, then only every  $P$  channel may be programmed using this method. A problem remains: how to program intermediate channels.

One solution to this problem is shown in Figure 2. A  $\div P$  is placed in series with the desired channel spacing (fre-

quency) to give a new reference frequency: channel spacing/ $P$ .

Another solution is found by considering the defining Equation 1 for  $f_{out}$  of Figure 1. From the equation it may be seen that only every  $P$  channel can be programmed simply, because  $N$  is always an integer. To obtain intermediate channels,  $P$  must be multiplied by an integer plus a fraction. This fraction would be of the form:  $A/P$ . If  $N$  is defined to be an integer number,  $N_p$ , plus a fraction,  $A/P$ ,  $N$  may be expressed as:

$$N = N_p + A/P.$$

Substituting this expression for  $N$  in Equation 1 gives:

$$f_{out} = (N_p + A/P) \cdot P \cdot f_{ref} \quad (2)$$

$$\text{or} \quad f_{out} = (N_p P + A) \cdot f_{ref} \quad (3)$$

$$f_{out} = N_p \cdot P \cdot f_{ref} + A \cdot f_{ref}. \quad (4)$$

Equation 4 shows that all channels can be obtained directly if  $N$  can take on fractional values. Since it is difficult to multiply by a fractional number, Equation 4 must be synthesized by some other means.

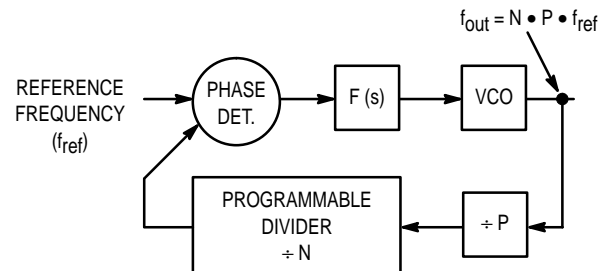


Figure 1. Frequency Synthesis by Prescaling

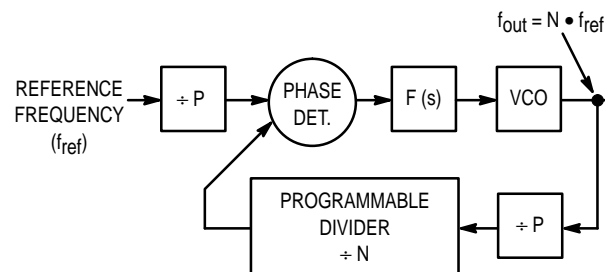


Figure 2. Frequency Synthesis by Prescaling

Taking Equation 3 and adding  $\pm AP$  to the coefficient of the  $f_{ref}$ , the equation becomes:

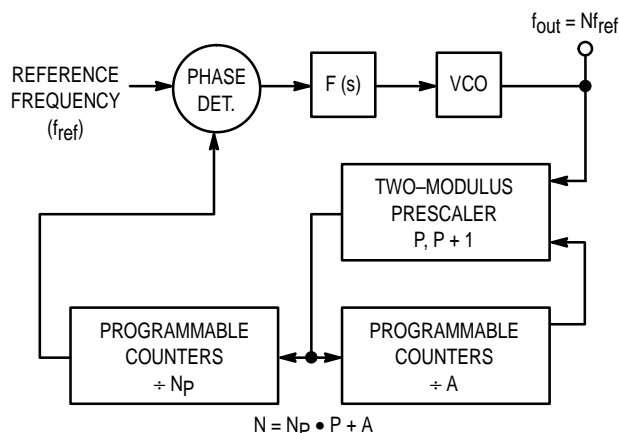
$$f_{out} = (Np \cdot P + A + A \cdot P - A \cdot P) f_{ref}. \quad (5)$$

Collecting terms and factoring gives:

$$f_{out} = [(Np - A) P + A (P + 1)] f_{ref} \quad (6)$$

From Equation 6 it becomes apparent that the fractional part of  $N$  can be synthesized by using a two-modulus counter ( $P$  and  $P + 1$ ) and dividing by the upper modulus,  $A$  times, and the lower modulus ( $Np - A$ ) times.

Equation 6 suggests the circuit configuration in Figure 3. The  $A$  counter shown must be the type that counts from the programmed state ( $A$ ) to the enable state, and remains in this state until divide by  $Np$  is completed in the programmable counter.



**Figure 3. Frequency Synthesis by Two-Modulus Prescaling**

In operation, the prescaler divides by  $P + 1$ ,  $A$  times. For every  $P + 1$  pulse into the prescaler, both the  $A$  counter and  $Np$  counter are decremented by 1. The prescaler divides by  $P + 1$  until the  $A$  counter reaches the zero state. At the end of  $(P + 1) \cdot A$  pulses, the state of the  $Np$  counter equals  $(Np - A)$ . The modulus of the prescaler then changes to  $P$ . The variable modulus counter divides by  $P$  until the remaining count,  $(Np - A)$  in the  $Np$  counter, is decremented to zero. Finally, when this is completed, the  $A$  and  $Np$  counters are reset and the cycle repeats.

To further understand this prescaling technique, consider the case with  $P = 10$ . Equation 6 becomes:

$$f_{out} = (A + 10 Np) \cdot f_{ref} \quad (7)$$

If  $Np$  consists of 2 decades of counters then:

$$Np = 10 Np_1 + Np_0$$

( $Np_1$  is the most significant digit),

and Equation 7 becomes:

$$f_{out} = (100 Np_1 + 10 Np_0 + A) f_{ref}.$$

To do variable modulus prescaling using the variable modulus prescalers (MC12009/11/13) and programmable divide by  $N$  counters (MC4016, MC4018) one additional part is required: the MC12014 (Counter Control Logic).

In variable modulus prescaling the MC12014 serves a dual purpose: it detects the terminal (zero) count of the  $A$  counter,

to switch the modulus of the MC12013; and it extends the maximum operating frequency of the programmable counters to above 25 MHz. (See the MC12014 data sheet for a detailed description of the Counter Control Logic).

Figure 4 shows the method of interconnecting the MC12013, MC12014, and MC4016 (or MC4018) for variable modulus prescaling. To understand the operation of the circuit shown in Figure 4, consider division by 43. Division by 43 is done by programming  $Np_1 = 0$ ,  $Np_0 = 4$ , and  $A = 3$ .

Waveforms for various points in the circuit are shown in Figure 5 for this division. From the waveforms it may be seen that the two-modulus prescaler starts in the divide by 11 mode, and the first input pulse causes point  $A$  to go high. This positive transition decrements the  $Np$  counter to 3, and counter  $A$  to 2.

After 11 pulses, point  $A$  again goes high; the  $Np$  counter decrements to 2 and the  $A$  counter to 1. The "2" contained in the  $Np$  counter enables the inputs to the frequency extender portion of the MC12014. After 11 more pulses point  $A$  goes high again.

With this position transition at  $A$ , the output ( $f_{out}$ ) of the MC12014 goes low, the  $Np$  counter goes to 1, and the  $A$  counter goes to 0. The zero state of the  $A$  counter is detected by the MC12014, causing point  $B$  to go to 1 and changing the modulus of the MC12013 to 10 at the start of the cycle.

When  $f_{out}$  goes low, the programmable counters are reset to the programmed number. After 11 pulses (the enable went high after the start of the cycle and therefore doesn't change the modulus until the next cycle), point  $A$  makes another positive transition. This positive transition causes  $f_{out}$  to return high, release the preset on the counter, and generates a pulse to clear the latch (return point  $B$  to 0).

After 10 pulses the cycle begins again (point  $B$  was high prior to point  $A$  going high). The number of input pulses that have occurred during this entire operation is:  $11 + 11 + 11 + 10 = 43$ . Figures 6 and 7 show the waveforms for divide by 42 and divide by 44 respectively.

The variable modulus prescaling technique may be used in any application as long as the number in the  $Np$  counter is greater than or equal to the number in the  $A$  counter. Failure to observe this rule will result in erroneous results. (For example, for the system shown in Figure 4, if the number 45 is programmed, the circuit actually will divide by 44. This is not a serious restriction since  $Np$  is greater than  $A$  in most applications.)

It is important to note that the  $A$  counter has been composed of only one counter for discussion only; where required, the  $A$  counter may be made as large as needed by cascading several programmable counters. Figure 8 shows the method of interconnecting counters. Operation is previously described. The number of stages in the  $A$  Counter should not exceed the number of stages for the  $Np$  counters. As many counters as desired may be cascaded, as long as fan-in and fan-out rules for each part are observed.

The theory of "variable modulus prescaling" developed above, examined a case in which the upper modulus of the two-modulus prescaler was 1 greater than the lower modulus. However, the technique described is by no means limited to this one special case. There are applications in which it is desirable to use moduli other than  $P/(P + 1)$ .

It can be shown that for a general case in which the moduli of the two-modulus prescaler are P and P + M, Equation 6 becomes:

$$f_{out} = [(N_p - A) P + A (P + M)] \cdot f_{ref}$$

or

$$f_{out} = [N_p \cdot P + M \cdot A] \cdot f_{ref}. \quad (8)$$

From Equation 8 it may be seen that the upper modulus of the two-modulus prescaler has no effect on the  $N_p$  counter, and that the number programmed in the A counter is simply multiplied by M.

There is no one procedure which will always yield the best counter configuration for all possible applications. Each designer will develop his own special design for the counter portion of his PLL system.

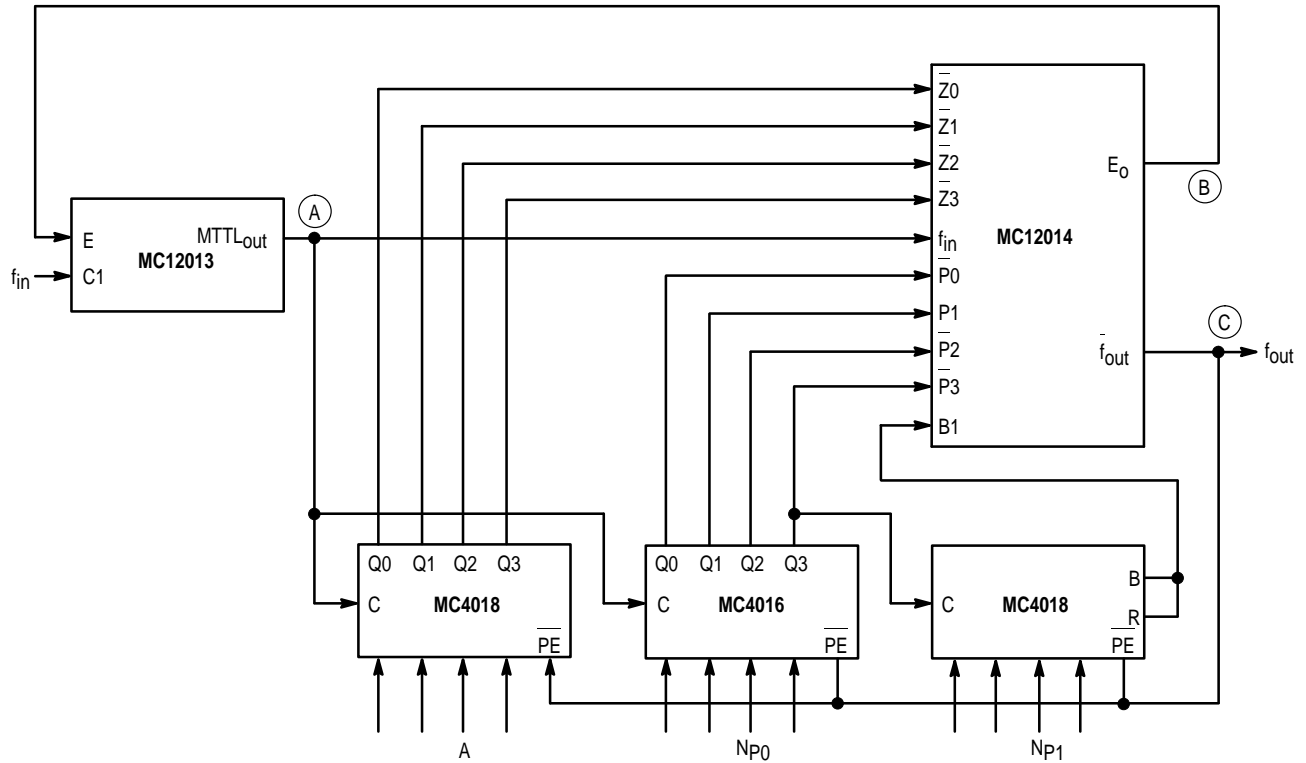
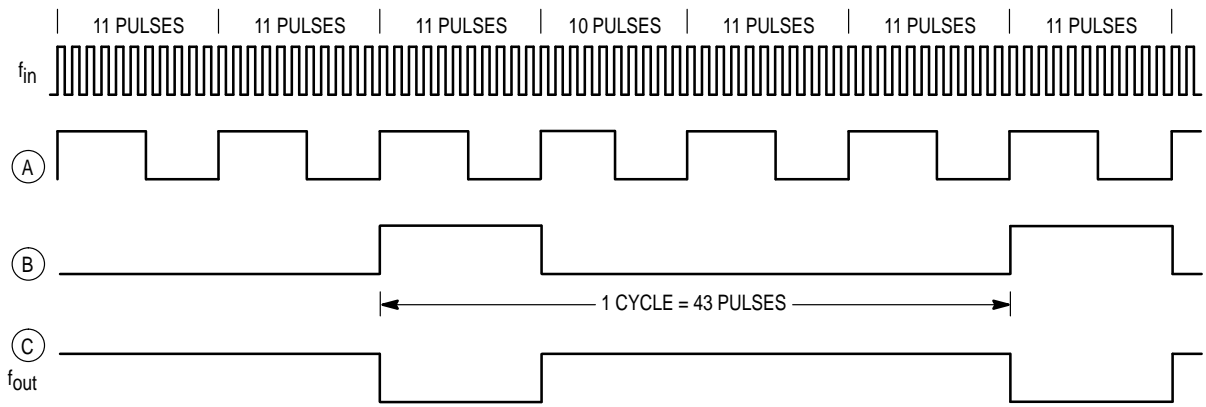
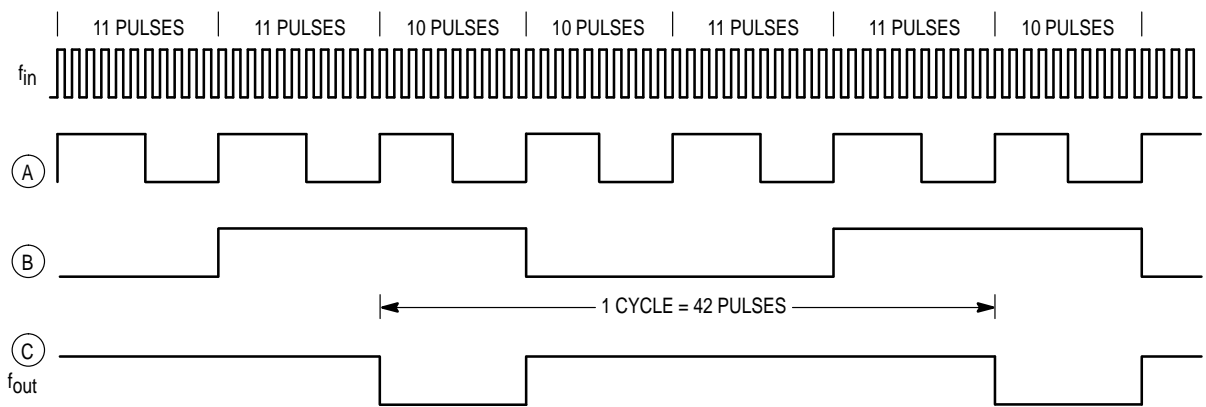


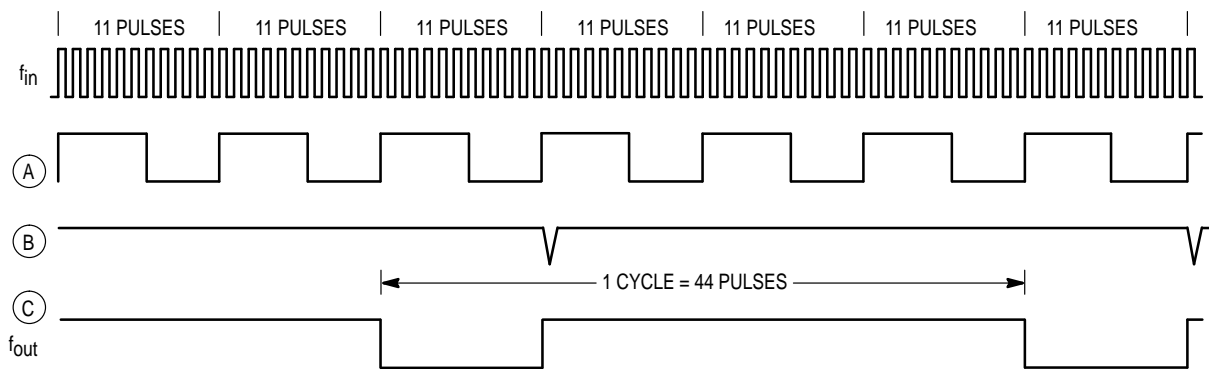
Figure 4. Direct Programming Utilizing Two-Modulus Prescaler



**Figure 5. Waveforms for Divide by 43**



**Figure 6. Waveforms for Divide by 42**



**Figure 7. Waveforms for Divide by 44**

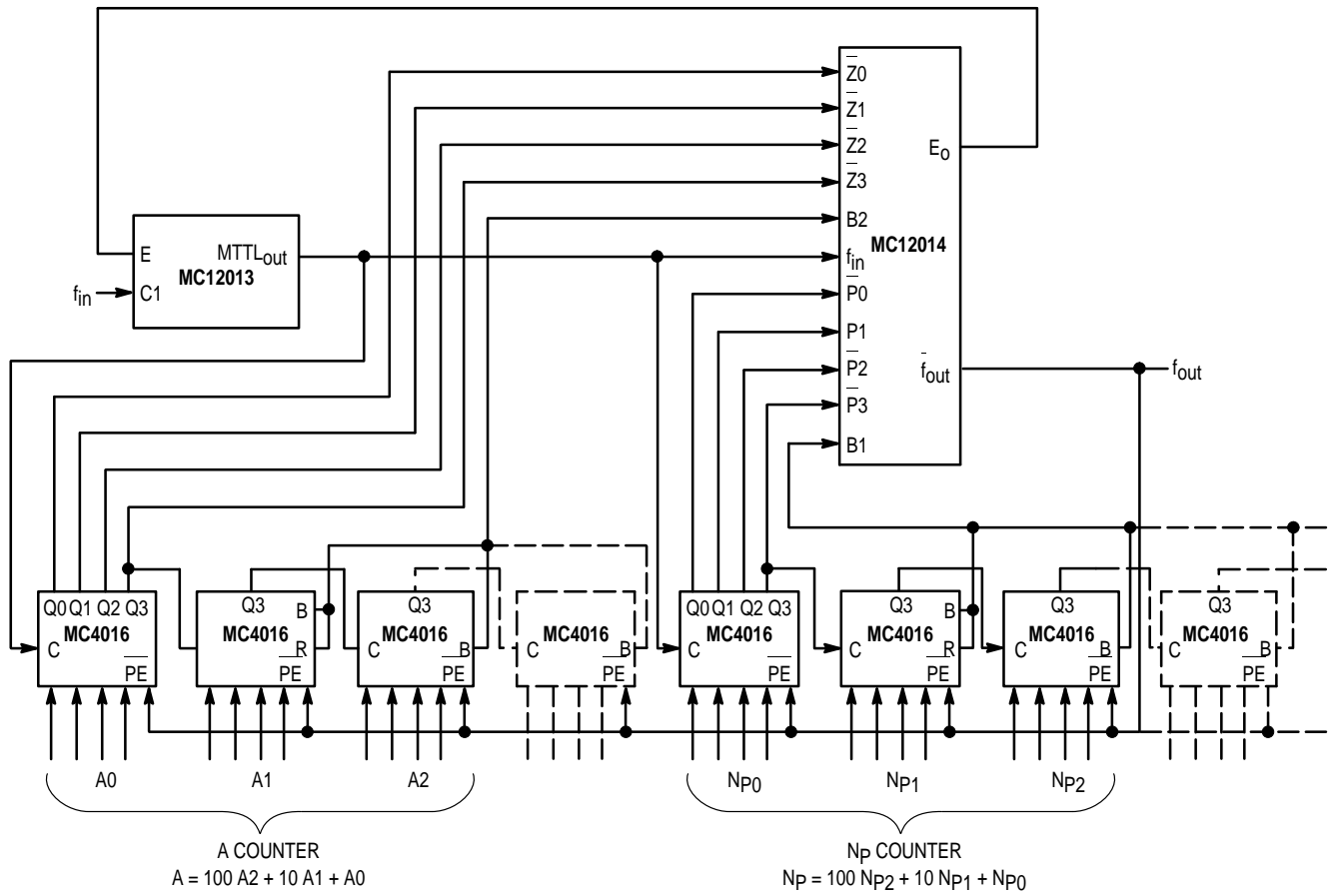


Figure 8. Method of Interconnecting Counters

