References

1 SERAJI, H., and TAROKH, M.: 'Design of proportional-plus-derivative output feedback for pole assignment', *Proc. IEEE*, 1977, 128, pp. 729-732
2 PARASKEVOPOULOS, P. N., and KING, R. E.: 'A Kronecker product approach to pole assignment by output feedback', *Int. J. Control*, 1976, 24, pp. 325-334
3 BELLMAN, R.: 'Introduction to matrix analysis' (McGraw Hill, New York, 1960)
4 PARASKEVOPOULOS, P. N., and TZAFESTAS, S. G.: 'New results in feedback modal controller design', *Int. J. Control*, 1975, 21, pp. 911-928
5 MUNRO, N., and VARDOULAKIS, A.: 'Pole-shifting using output feedback', *ibid.*, 1973, 18, pp. 1267-1273
6 SERAJI, H.: 'On pole-shifting using output feedback', *ibid.*, 1974, 20, pp. 721-726

0013-5194/78/0614-0034 $1.50/0

THREE-VALUED C.M.O.S. CYCLING GATES

Indexing terms: Ternary logic, Integrated logic circuits, Logic gates, Field effect integrated circuits

The design of three-valued cycling gates with c.m.o.s. integrated circuits is presented. Circuits for the cycling and the inverse cycling gates prove to be simpler than those previously reported.

The use of the complementary metal-oxide semiconductor family of integrated circuits (c.m.o.s.) in the realisation of three-valued cycling gates has been reported by several authors.[1-3] Kaniel[1] has used a 'shifter' circuit and an inverting element to build a cyclic counting device having a relatively large number of transistors. In Reference 2 a systematic building block technique based on some ternary primitives has been used, which introduces some redundant elements in the cycling gate. Huertas *et al.*[3] presented a circuit that includes signal-level translators which make the realisation of a ternary cycling gate quite complex. In this letter a simpler design of three-valued cycling and inverse cycling gates is described. The three logic levels (0, 1 and 2) are represented here by −4V for 0, zero V for 1 and +4V for 2.

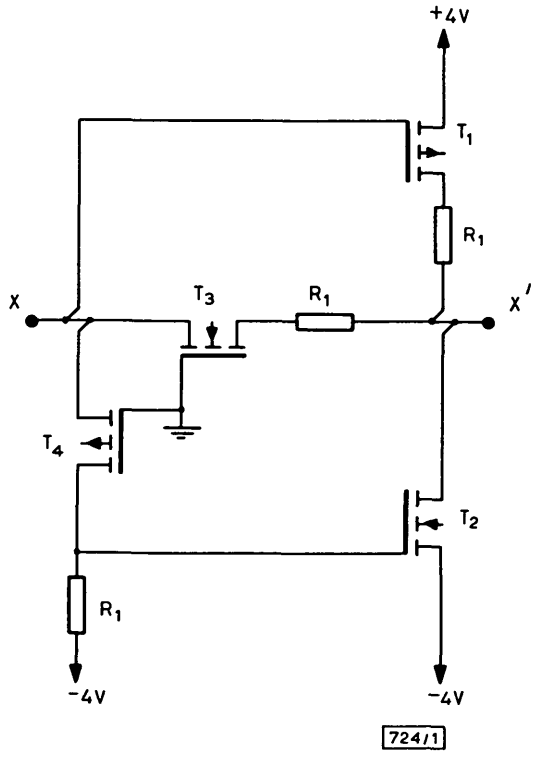

Fig. 1 *Three-valued c.m.o.s. cycling gate*

The three-valued c.m.o.s. cycling-gate circuit is composed of two *p*-channel m.o.s. transistors, two *n*-channel m.o.s. transistors and three resistors connected as shown in Fig. 1. It realises the function of the cycling operator defined by

$$X' = (X + 1) \bmod 3$$

The operation of the circuit may be described briefly as follows: If the input X takes the logic value 2 transistors T_1 and T_3 will be off, while transistors T_2 and T_4 will be on, causing the output X' to take the logic value 0. If X is at level 1 transistors T_2, T_3 and T_4 will be off, while transistor T_1 will be on, and the output X' will be at level 2. If input X is at level 0, the output X' will take the logic value 1 because transistors T_2 and T_4 will be off, while transistors T_1 and T_3 will be on. Fig. 2 shows the three-valued c.m.o.s. inverse cycling-gate circuit consisting of the same number of elements as the above circuit. It is also designed following the same idea. It realises the function defined by

$$X'' = (X - 1) \bmod 3$$

If input X is at level 2 the output X'' will be at level 1 because T_2 and T_3 will be on while the other transistors will be off. If X is at level 1 the output X'' will be at level 0 because only transistor T_2 will be on. If X is at level 0 only transistors T_1 and T_4 will be on causing X'' to be at level 2.

These two circuits can implement any three-valued combinational or sequential-logic function. With the inclusion of the simple ternary inverter[2] and either the circuit of Fig. 1 or Fig. 2 a functionally complete ternary system can be set up.[4]

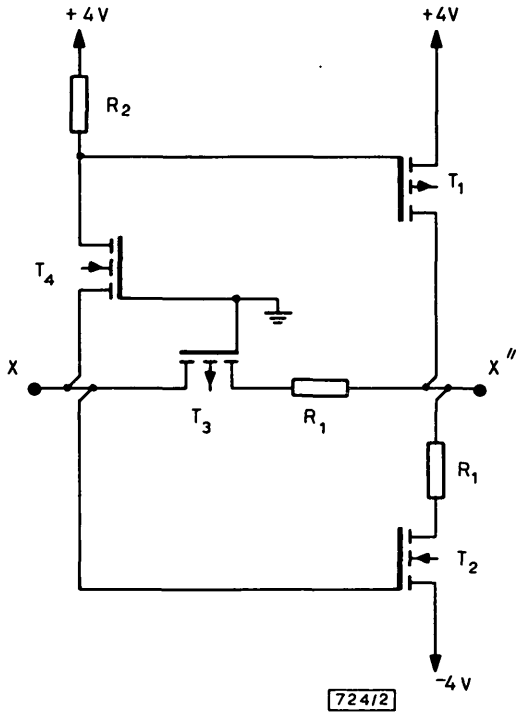

Fig. 2 *Three-valued c.m.o.s. inverse cycling gate*
$R_1 = 12\ k\Omega$
$R_2 = 100\ k\Omega$

Finally it has to be noted that all *p*-channel substrates, in both circuits, are connected to the positive power supply (+4V) and all *n*-channel substrates are connected to the negative power supply (−4V). These two circuits have been realised with the commercially available MC 14007 c.m.o.s. integrated circuits.

Conclusion: The use of c.m.o.s. integrated circuits in the realisation of a three-valued cycling and inverse cycling gate has been presented. The design of these two circuits is simpler than any that has been previously reported. It is hoped that

evidence of the simplicity of ternary c.m.o.s. devices may contribute to the acceptance of three-valued digital systems by industry.

H. T. MOUFTAH
Bell-Northern Research
P.O. Box 3511, Station C
Ottawa, Ontario, Canada

K. C. SMITH
Dept. Of Electrical Eng.,
University of Toronto
Toronto, Ontario, Canada

29th November 1977

References

1 KANIEL, A.: 'Trilogic, a three-level logic system provides greater memory density', *EDN*, 1973, pp. 80–83
2 MOUFTAH, H. T., and JORDAN, I. B.: 'Integrated circuits for ternary logic'. Proceedings of the international symposium on multivalued logic, May 1974, pp. 285–302
3 HUERTAS, J. L., ACHA, J. I., and CARMONA, J. M.: 'Implementation of some ternary operations with CMOS integrated circuits', *Electron. Lett.*, 1976, **12**, pp. 385–386
4 POST, E. L.: 'Introduction to a general theory of elementary propositions', *Am. J. Math.*, 1921, **43**, pp. 163–185

0013-5194/78/0724-0036 $1.50/0

ENVELOPE-CONSTRAINED TIME-DOMAIN DECONVOLUTION FOR TRANSVERSAL-FILTER EQUALISERS

Indexing terms: Circuit analysis computing, Equalisers, Filters, Time-domain analysis

A deconvolution algorithm is presented for calculating the N tap weights of a transversal-filter equaliser. The procedure attempts to control the size of individual, time-domain response errors rather than simply minimise total mean-square error of the response. A least-square error solution for the tap weights is iteratively modified, using only elementary algebraic matrix operations, in an effort to satisfy tolerance constraints on the envelope of the equalised impulse response. The algorithm is suitable for digital computation.

In a recent letter,[1] the author proposed a matrix solution for time-domain deconvolution to determine the N tap weights of a transversal-filter equaliser. A least-square-error solution was selected because the discretisation scheme used to obtain an approximate solution to the convolutional integral equation yielded a system of incompatible linear equations. In some instances, unfortunately, such a solution can produce undesirably large side lobes or, perhaps, have mean-square-error that is unusually sensitive to specific choices of equalised impulse response.[2,3] The present communication considers a closely related equalisation problem wherein the equalised impulse response is required to lie between prescribed upper and lower bounds—that is its envelope is constrained. If a solution is feasible then side lobes can be reduced while, in noncritical regions, the response is allowed to assume values within some reasonable tolerance. For practical equalisation problems, accepting somewhat greater mean-square error in return for more precise control of the response envelope may be a useful alternative to increasing filter complexity.

Let $h(t)$ and $g(t)$ be the respective impulse responses of a linear system before and after equalisation by a filter whose impulse response is $f(t)$. Assuming $h(t)$ and $g(t)$ are zero for times $t<0$ and that both responses are specified for $t\geqslant 0$, then *deconvolution* is the process of solving the integral equation

$$\int_0^t f(x)\,h(t-x)dx = g(t) \tag{1}$$

for the unknown filter impulse response $f(t)$. Suppose that, rather than being specified, the impulse response $g(t)$ of the equalised system is only constrained to be bounded above by $a(t)$ and below by $b(t)$, that is

$$b(t)\leqslant g(t)\leqslant a(t) \tag{2}$$

and, further, that the most desirable equalised impulse response $d(t)$ is given by the mean of these two bounds, namely

$$d(t)=\{a(t)+b(t)\}/2 \tag{3}$$

then, *envelope-constrained deconvolution* seeks a solution for $f(t)$ in eqn. 1 which yields a $g(t)$ that closely approximates $d(t)$ and that also satisfies eqn. 2. If it were possible to solve eqn. 1 *exactly* then, of course, envelope-constrained deconvolution would be unnecessary. This particular deconvolution procedure is useful precisely because the normally used least-square-error *approximate* solution for $f(t)$ in eqn. 1 can produce some rather large deviations from $g(t)$ in spite of the fact that the total mean-square error is minimised.[2,3]

The equaliser is assumed to be a conventional transversal filter (tapped-analog delay line[4]) with N equally-spaced adjustable taps. It is necessary, therefore, to replace $f(t)$ in eqn. 1 with a finite sequence of equally-spaced sampled values. This sequence can be represented in matrix form by the row vector $f'=[f(1), f(2), f(3), \ldots, f(N)]$ where the prime symbol denotes transpose. Thus, f, as defined, is an N element column vector (namely the unknown 'tap-weight vector'). In a similar way, the equalised impulse response $g(t)$ and the system impulse response $h(t)$ are represented, respectively, by the sequences of equally-spaced sampled values written as $g'=[g(1), g(2), g(3), \ldots, g(M)]$ and $h'=[h(1), h(2), h(3), \ldots, h(M-N+1)]$ where $M>N$. With these definitions, a discrete-time matrix representation of eqn. 1 is,[1]

$$Hf=g \tag{4}$$

where the elements of the $M\times N$ transmission matrix H are $h_{mn}=h(m-n+1)$ for $1\leqslant m-n+1\leqslant M-N+1$ and equal zero otherwise. In writing eqn. 4 it is assumed, without loss of generality, that the time-interval between samples is unity. Note that eqn. 4 represents a system of M linear equations in N unknowns for which $M>N$. This result implies that a unique solution for f generally does not exist; thus, it is appropriate to seek solutions that satisfy eqn. 4 approximately. In the following development, the envelope constraints $a(t)$ and $b(t)$ as well as the most desirable response $d(t)$ shall be represented, in a similar way, by the M element column vectors a, b, and d, respectively.

For a chosen g, the tap-weight vector $\hat{f}$ yielding the least-square-error approximation to g in eqn. 4 is[2]

$$\hat{f}=Sg \tag{5}$$

where the $N\times M$ matrix $S=(H'H)^{-1}H'$. Note that $H'H$ is Toeplitz and, therefore, can be inverted easily.[5,6] The mean-square error produced by $\hat{f}$ is $(H\hat{f}-g)'(H\hat{f}-g)/M$. Eqn. 5 can be interpreted as the 'least-square-mapping' from changes in the constrained-response vector g to changes in the tap-weight vector f, that is

$$\delta f=S\delta g \tag{6}$$

In fact, each element s_{nm} of S is a sensitivity coefficient or proportionality constant which linearly relates the change in the nth tap weight to the change in the mth constraint. Note that the elements of S only depend upon the impulse response of the linear system to be equalised and not the specific choice of g.

Eqns. 2, 4, 5 and, especially, 6 form the basis for the iterative tap-weight adjustment procedure to follow. Briefly,