A Design Method of Finding Optimal Sampling Pulses and Transistor Sizes in a Sampling Circuit for Liquid Crystal Displays

Shingo Takahashi†, Shuji Tsukiyama†, Masanori Hashimoto††, and Isao Shirakawa†††

† Dept. of Electrical, Electronic, and Communication Eng., Chuo University,
†† Graduate School of Information Science and Technology, Osaka University ,
††† Graduate School of Applied Informatics, University of Hyogo

† 1-13-27 Kasuga, Bunkyo-ku, Tokyo, 112-0003, Japan
mail : sint@tsuki.elect.chuo-u.ac.jp, tsuki@elect.chuo-u.ac.jp

Abstract : In the design of column driver circuit of liquid crystal displays, a sampling circuit must be designed so that the pixel voltage of a pixel is as close to an input video voltage as possible in various conditions. In this paper, we propose a design method of finding optimal combinations of a sampling switch size and a sampling pulse waveform, which attains this objective. Moreover, in order to optimize both buffer and sampling circuits, we propose a figure of suitability of each combination to the design of buffer circuit generating sampling pulses. We show an experimental result which indicates that the proposed method produced almost equal quality design in eleven times faster than the optimizer in SPICE.

I. Introduction

Recently, with the rapid advance of LCD (Liquid Crystal Display) technologies, the column and row drivers of an active matrix LCD have to be integrated more and more finely on the same substrate as the picture plane[1-2], especially in small portable displays. In such integration, there occur considerable fluctuations in circuit parameters, which make the design of drivers complicated, mainly from the aspect of the functional capability as well as the reduction of the area and power consumption[3-5].

In the column driver circuit of LCD, the ratio of the pixel voltage V_{px} to the video voltage V_{vd} of a pixel, which is abbreviated to RPV in this paper, determines the quality of display, and hence must be maintained as close to 100% as possible in various conditions such as feeding types of video voltage and fluctuations of circuit parameters. In order to achieve this requirement, the design of sampling switch and sampling pulse waveform are most important, since they capture given video voltages. The circuit consisting of a sampling switch and a pixel is a kind of a sample-and-hold circuit[6] and has been considered in the various applications[7-12] and also from the viewpoint of modeling and analysis methods[13]. However, as far as we know, there is no specific report treating a sampling switch from the viewpoint of the quality of display.

In the application to LCDs, smaller sampling circuits are preferred so as to integrate them on the same substrate as the picture plane, and hence a single CMOS switch is used. For the CMOS switch, we must use a full transistor model (a SPICE model) in order to calculate RPV precisely. Due to this requirement together with the various fluctuations, we need long design time if a conventional circuit optimizer such as the one provided in SPICE is used. Moreover, in the design of the column driver circuit, we must consider not only a sampling circuit but also a buffer circuit which generates the sampling pulses input to the sampling circuit. Generally speaking, if we use a smaller sampling switch, we must use a sharp and long sampling pulse with steeper transition times, which makes the design of buffer circuit hard. Therefore, we must resolve a trade-off between a sampling switch size and a sampling pulse waveform. However, it is not easy to resolve this trade-off by an optimizer in SPICE.

Thus, we propose in this paper a design method of finding an optimal combination of a sampling switch size and a sampling pulse waveform, such that the range of RPV varied by the various conditions is minimized and the difference from 100% is also minimized. In order to take the design of buffer circuit into consideration, the method generates a number of optimal combinations of a sampling switch size and a sampling pulse waveform, and for each such combination, the method gives a figure of suitability to the buffer design. By designing buffer circuits in the ascending order of the figure, we can get optimal buffer and sampling circuits without searching all possibilities. We also show an experimental result in order to see the performances of the proposed method and figure.

II. Preliminaries

Fig. 1 shows a column driver circuit of an active matrix LCD[1,2], in which the grey shade of each pixel is controlled individually by a designated pixel voltage. A pair of nMOS TFT and pMOS TFT connected in parallel constitutes a

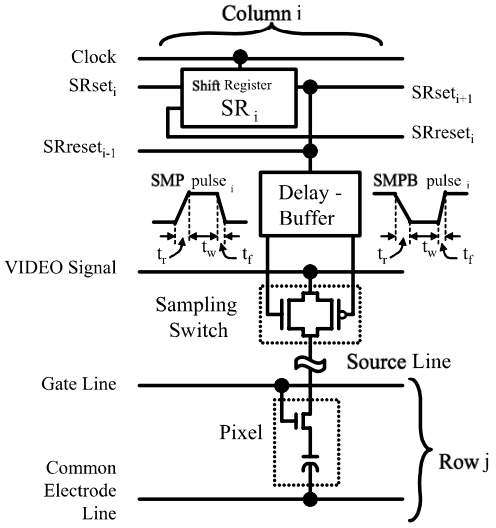

Fig. 1 : A column driver circuit in active matrix LCD

sampling switch, which is to sample the video signal and then to transmit the signal to the source line.

Let W_n and W_p be the widths of nMOS and pMOS TFTs in a sampling switch, respectively, and let SMP and SMPB be sampling pulses input to the gates of nMOS and pMOS TFTs, respectively. For simplicity, let both of SMP and SMPB be of piece-wise linear waveforms, reverse to each other, with the high and low voltage levels V_{high} and V_{low}. Hence, the rising time t_r and the falling time t_f of SMP are the falling and rising times of SMPB, respectively, and the intermediate time t_w between rising and falling of SMP is equal to that of SMPB.

Even if the same video voltage V_{vd} is fed into a sampling switch, RPV may change according to the common electrode voltage V_{com}, and V_{com} alternates between high and low voltages, frame by frame. Hence, we must consider two feeding types of video voltage; one is the case of high V_{com} and the other is that of low V_{com}.

As mentioned above, we must take a variety of fluctuations in the design parameters into account. We assume that V_{high} and V_{low} fluctuate within $V_{high}\pm\upsilon_{high}$ and $V_{low}\pm\upsilon_{low}$, respectively. SPICE parameters of nMOS and pMOS TFTs fluctuate among *best*, *typical*, and *worst* cases, and we consider the following five combinations of fluctuations: nBpB, nTpT, nWpW, nBpW, and nWpB, where n and p denote nMOS and pMOS, respectively, and B, T, and W represent best, typical, and worst cases, respectively. Moreover, SMPB delays from SMP by δ_{SMP}, and time T indicating the interval during which the video signal keeps a voltage for a pixel, fluctuates between T and $T-\tau_T$. Note that T is the half of the period of the system clock.

Due to these fluctuations, it is hard to keep RPV 100% for any V_{vd}. As for V_{vd}, we consider three cases; high, middle, and low V_{vd} voltages. Therefore, total number of situations to be simulated is $270 = 2 \times 3 \times 3 \times 3 \times 5$ cases, where these numbers denote 2 types of feeding, 3 video voltages, 3 high voltages ($V_{high}-\upsilon_{high}$, V_{high}, and $V_{high}+\upsilon_{high}$), 3 low voltages ($V_{low}-\upsilon_{low}$, V_{low}, and $V_{low}+\upsilon_{low}$), and 5 combinations of TFT fluctuations. Henceforth, among all RPVs calculated for these 270 cases, let RPV_{max} and RPV_{min} be the maximum and minimum values, respectively.

Since the delay-buffer circuit generating SMP and SMPB has similar fluctuations occurred in the sampling circuit, t_r, t_f, t_w, and δ_{SMP} also fluctuate. Therefore, in order to consider their fluctuations, we introduce upper bounds $t_{r\text{-}max}$, $t_{f\text{-}max}$, and $\delta_{SMP\text{-}max}$ to t_r, t_f, and δ_{SMP}, respectively, and lower bound $t_{w\text{-}min}$ to t_w. Because, if a sampling circuit is optimally designed for SMP and SMPB with $t_{r\text{-}max}$, $t_{w\text{-}min}$, $t_{f\text{-}max}$, and $\delta_{SMP\text{-}max}$, then the sampling circuit yields acceptable RPV_{max} and RPV_{min} for SMP and SMPB with $t_r \le t_{r\text{-}max}$, $t_{w\text{-}min} \le t_w$, $t_f \le t_{f\text{-}max}$, and $\delta_{SMP} \le \delta_{SMP\text{-}max}$.

Usually, the upper bound $\delta_{SMP\text{-}max}$ is specified by a user, and so is the minimum value $t_{s2v\text{-}min}$ of time interval t_{s2v} between the instance at which SMP falls 50% down and the instance at which the video signal begins to change to the next pixel video voltage. With the use of these values, we determine the upper bound $t_{f\text{-}max}$ of t_f as follows:

$$t_{f\text{-}max} = 2\cdot(t_{s2v\text{-}min} - \delta_{SMP\text{-}max}) \quad (1)$$

This $t_{f\text{-}max}$ is the longest value which guarantees SMP falls down before the video signal changes to the next pixel's voltage. Since the value of the right hand side of Eq. (1) is small, we set $t_{f\text{-}max}$ as a fixed value.

On the other hand, the upper bound $t_{r\text{-}max}$ of t_f is determined by the following inequality.

$$t_{r\text{-}max} \le T-\tau_T - \delta_{SMP\text{-}max} - t_{f\text{-}max} \quad (= t_{r\text{-}U}) \quad (2)$$

Note here that $t_{r\text{-}max}$ can be any value not greater than the right hand side of this inequality, which is denoted by $t_{r\text{-}U}$ in the following.

Eq. (2) guarantees that SMP swings fully from V_{low} to V_{high} in the case of single sampling, and two consecutive SMPs do not change simultaneously in the case of double sampling, where the *simple sampling* is the ordinary sampling method such that the sampling switch for the *i*th pixel opens only during the time interval of T when the video signal holds $V_{vd}(i)$ for the *i*th pixel; whereas the *double sampling* opens the sampling switch for the *i*th pixel during the time interval when the video signal holds $V_{vd}(i-1)$ for the $(i-1)$th pixel and $V_{vd}(i)$ for the *i*th pixel[1]. Double sampling is used in order to make V_{px} be close to V_{vd} as quickly as possible.

In order that two consecutive SMPs do not overlap, the length $t_r + t_w + t_f$ of SMP must satisfy the following inequality,

$$t_r + t_w + t_f \le k\cdot T - \delta_{SMP\text{-}max} \quad (3)$$

where $k = 1$ or 2 if simple sampling or double sampling is used, respectively. We can see that the longer the length $t_r + t_w + t_f$ of SMP is, the smaller W_n we can use. Hence, we set the length of SMP as long as possible, and determine the lower bound $t_{w\text{-}min}$ as follows.

$$t_{w\text{-}min} = k\cdot T - \tau_T - \delta_{SMP\text{-}max} - t_{r\text{-}max} - t_{f\text{-}max} \quad (4)$$

Thus, the problem that we consider in this paper is stated as follows. Find optimum values of W_n, W_p, and $t_{r\text{-}max}$ such that RPV_{max} and RPV_{min} are as close to 100% as possible, and not only the sampling switch but also the buffer circuit can be minimal. Note here that $t_{w\text{-}min}$ is a constraint in the design of delay circuit, and $t_{r\text{-}max}$, $t_{f\text{-}max}$, $\delta_{SMP\text{-}max}$ and Eq. (3) are constraints in the design of buffer circuit.

III. Design Method

When a sampling switch turns off, V_{px} may change by the channel charge injection[8-12,14,15], and so may RPV. Although this voltage change ΔV_{px} varies with V_{vd}, ΔV_{px} can be zero for a single V_{vd} value by setting $W_p = a\cdot W_n + b$. Hence, for a pixel voltage $V_{px\text{-}mdl}$ which changes the transmitted luminance most steeply, constant values of a and b are decided so that $\Delta V_{px} = 0$. By this relation, W_p is determined from W_n.

For given SMP and SMPB with $t_r = t_{r\text{-}max}$, $t_w = t_{w\text{-}min}$, $t_f = t_{f\text{-}max}$, and $\delta_{SMP} = \delta_{SMP\text{-}max}$, RPV_{min} changes as W_n changes, and its shape is unimodal with respect to W_n. The reasons are as follows: If W_n is small, it is hard to raise RPV to 100% during the sampling switch is open, so that RPV_{min} is low. As W_n gradually increases, RPV can be raised easily, and hence RPV_{min} approaches 100%. However, if W_n increases further, RPV_{min} deteriorates, because pixel voltage change ΔV_{px} caused by the charge injection becomes negative and its absolute value $|\Delta V_{px}|$ becomes large.

On the other hand, RPV_{max}, which is greater than 100%,

increases monotonically as W_n increases, because the reason why RPV_{max} exceeds 100% is the charge injection, and ΔV_{px} increases in proportion to W_n. Note that since W_n and W_p satisfy $W_p = a \cdot W_n + b$, $\Delta V_{px} = 0$ when $V_{vd} = V_{px\text{-}mdl}$, but when $V_{vd} \neq V_{px\text{-}mdl}$, ΔV_{px} becomes positive and negative depending on V_{vd}, and $|\Delta V_{px}|$ increases as W_n increases.

From these facts, we can see that the following $f(W_n)$ is also unimodal with respect to W_n.

$$f(W_n) = (RPV_{max} - 100)^2 + (100 - RPV_{min})^2 \quad (5)$$

Therefore, we can find the optimum W_n such that $f(W_n)$ is minimum with the use of the golden section search[16]. Henceforth, this optimum W_n is denoted by $W_n(t_{r\text{-}max})$.

If we check and simulate all circuit situations (270 cases) to find RPV_{max} and RPV_{min}, then it is time consuming, since RPV must be calculated by using SPICE in each iteration of the golden section search. Hence, we select the following cases from among all situations.

Case-Easy (A case where RPV easily exceeds 100%):

Case-Difficult (A case where RPV is hard to reach 100%):

These cases can be characterized as follows: Case-Easy is a case where V_{px} reaches easily to V_{vd}, and ΔV_{px} is positive. The case where high and low voltages of SMP are $V_{high} + \upsilon_{high}$ and $V_{low} - \upsilon_{low}$, respectively, and SPICE parameter is nWpB is an example of Case-Easy. On the other hand, Case-Difficult is a case where V_{px} is hard to reach V_{vd}, or ΔV_{px} is negative, and the case where high and low voltages of SMP are $V_{high} - \upsilon_{high}$ and $V_{low} + \upsilon_{low}$, respectively, and SPICE parameter is nBpW is an example of Case-Difficult.

We select 10 Case-Easy cases and 10 Case-Difficult cases by simulating all circuit situations for a large W_n value and a small W_n value, and use these 20 cases to determine RPV_{max} and RPV_{min}. From our experiments, these 20 cases are sufficient to find RPV_{max} and RPV_{min} values correctly, but this number can be reduced so as to shorten the simulation time.

Although a pair $(t_{r\text{-}max}, W_n(t_{r\text{-}max}))$ gives a desirable range $[RPV_{max}, RPV_{min}]$ of RPV, it may be hard to design a buffer circuit generating SMP and SMPB with $t_r \leq t_{r\text{-}max}$, $t_{w\text{-}min} \leq t_w$, $t_f \leq t_{f\text{-}max}$, and $\delta_{SMP} \leq \delta_{SMP\text{-}max}$ under various fluctuations. Therefore, we generate such pairs for several possible $t_{r\text{-}max}$, and proposes a design figure $A(t_{r\text{-}max})$ for each pair. This figure is an estimated area of the final inverters in the buffer circuit which can generate SMP and SMPB with $t_r \leq t_{r\text{-}max}$ and $t_f \leq t_{f\text{-}max}$, and shows a suitability of pair $(t_{r\text{-}max}, W_n(t_{r\text{-}max}))$ to the design of buffer circuit. Therefore, by designing buffer circuits in the ascending order of $A(t_{r\text{-}max})$, we can find an optimal sampling circuit without searching all possibilities.

Consider the final inverter of a buffer circuit generating SMP, and the load C_n of the inverter can be calculated by

$$C_n = m \cdot (W_n \cdot L_n / t_{ox}) \cdot \varepsilon_{ox} \cdot \varepsilon_0 \quad (6)$$

where m is the number of sampling switches driven by the buffer circuit simultaneously, L_n is the gate length of nMOS in the sampling switch, t_{ox} is the thickness of gate oxide film, and ε_0 and ε_{ox} are the permittivity in vacuum and dielectric constant of SiO_2, respectively

When a step input enters a CMOS inverter, the output transition time t_f can be estimated by the following expression:

$$t_f = \frac{2 C_{load}}{\frac{W_{bufn}}{L_{bufn}} \mu_n C_0 (V_{dd} - V_{ss} - V_{thn})} \times \left\{ \frac{V_{thn} - 0.1(V_{dd} - V_{ss})}{V_{dd} - V_{ss} - V_{thn}} + \frac{1}{2} \ln \frac{19 (V_{dd} - V_{ss}) - 20 V_{thn}}{V_{dd} - V_{ss}} \right\} \quad (7)$$

where C_{load} is load capacitance, μ_n is electron mobility, C_0, W_{bufn}, L_{bufn}, and V_{thn} are the oxide capacitance per unit area, the gate width, the gate length, and the threshold voltage of nMOS in the inverter, respectively, and V_{dd} and V_{ss} are high and low supply voltages, respectively[17]. If we assume $V_{thn} = 0.2(V_{dd} - V_{ss})$, then Eq. (7) becomes the following equation.

$$t_f = 3.85 \cdot C_{load} \cdot L_{bufn} / \{(V_{dd} - V_{ss}) \cdot \mu_n \cdot C_0 \cdot W_{bufn}\} \quad (8)$$

Substitute Eq. (6) into Eq. (8) by setting $C_{load} = C_n$, and we have the following equation,

$$W_{bufn} = K_{nf} \cdot W_n / t_f \quad (9)$$

where K_{nf} is a constant obtained by merging all constants in Eqs. (6) and (8). With the use of Eq. (9), for a given pair $(t_{r\text{-}max}, W_n(t_{r\text{-}max}))$, we can obtain the necessary gate width W_{bufn} in the final inverter generating SMP by the following equation.

$$W_{bufn} = K_{nf} \cdot W_n(t_{r\text{-}max}) / t_{f\text{-}max} \quad (10)$$

From a similar discussion, we can estimate a necessary gate width W_{bufp} of pMOS in the final inverter generating SMP, and necessary gate widths W_{bufBn} and W_{bufBp} of nMOS and pMOS in the final inverter generating SMPB. Note that since these W_{bufn}, W_{bufp}, W_{bufBn}, and W_{bufBp} are obtained by assuming a step input, they are under estimations.

With the use of these values, design figure $A(t_{r\text{-}max})$ for each pair $(t_{r\text{-}max}, W_n(t_{r\text{-}max}))$ is defined as follows, where L_{bufp} is the gate length of pMOS in the buffer circuit.

$$A(t_{r\text{-}max}) = L_{bufn} \cdot (W_{bufn} + W_{bufBn}) + L_{bufp} \cdot (W_{bufp} + W_{bufBp}) \quad (11)$$

IV. Experimental Result

We applied the proposed method to a column driver circuit in [1], and compared with the results obtained by the optimizer in SmartSPICE[18]. We divided the value $t_{r\text{-}U}$ of the right hand side of Eq. (2) by 8, and generated 8 pairs $(t_{r\text{-}max}, W_n(t_{r\text{-}max}))$. These results are shown as *OURS* in the following.

In order to use the optimizer in SmartSPICE, we provided a netlist containing 270 separated circuits each of which corresponds to a different situation, and input eight $t_{r\text{-}max}$ values which are the same as our method. Then, the gate widths W_n and W_p were optimized so as to minimize the sum of the differences of RPV_{max} and RPV_{min} from 100%. The results of this optimization are denoted as *divided*.

Fig. 2 shows areas $W_n \cdot L_n + W_p \cdot L_p$ of sampling switches, where the values are normalized by the smallest area in the results of *divided*. We can see from the figure that larger sampling switch is needed as $t_{r\text{-}max}$ increases, and our method produced smaller sampling switches than SPICE optimizer.

Table 1 shows values of RPV_{max}, RPV_{min}, and the objective function $f(W_n)$ normalized by the smallest value obtained by *divided*. From these results, we can see that the proposed method generates almost the same results as the SmartSPICE optimizer. Moreover, we see that if $t_{r\text{-}max}$ is small, the range $[RPV_{max}, RPV_{min}]$ of RPV becomes small, that is, the

sampling circuit becomes robust against various fluctuations. However, it is not easy to design a buffer circuit generating SMP and SMPB with small $t_{r\text{-}max}$ in various fluctuations.

Fig. 3 shows the design figures $A(t_{r\text{-}max})$ and the actual areas of buffer circuits with the use of the values normalized by the minimum value. Here, the actual area is the area of the buffer circuit designed by the optimizer in SmartSPICE under the condition that the high power voltage is $V_{high}-\upsilon_{high}$, the low power voltage is $V_{low}+\upsilon_{low}$, and transistor parameters are nWpW, which is one of worst conditions for the operation of buffer circuit. From the figure, we see that the proposed design figure can be used as an estimation of the actual buffer area.

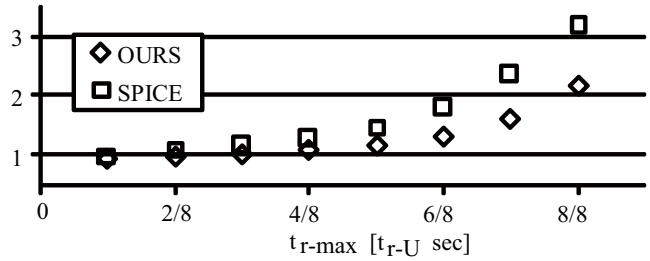

Fig. 2 : Normalized areas of sampling switches.
$t_{r\text{-}U}$ is the right-hand side value of Eq. (2).

Table 1 : Ranges of RPV and values of $f(W_n)$.

$t_{r\text{-}max}$ [$t_{r\text{-}U}$ sec]		1/8	2/8	3/8	4/8	5/8	6/8	7/8	8/8
OURS	RPV_{max}	100.7	100.7	100.8	100.9	101.0	101.1	101.5	102.1
	RPV_{min}	98.0	98.0	97.9	97.9	97.7	97.6	97.2	96.6
	$f(W_n)$	1.08	1.14	1.24	1.36	1.58	1.82	2.49	4.12
divided	RPV_{max}	100.7	100.7	100.7	100.8	100.9	101.1	101.4	101.8
	RPV_{min}	98.1	98.1	98.1	98.1	97.9	97.7	97.3	96.7
	$f(W_n)$	1.00	1.02	1.05	1.13	1.29	1.60	2.29	3.50

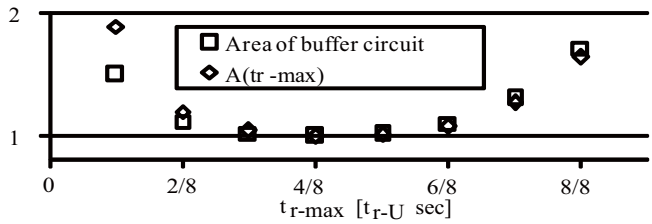

Fig. 3 : Normalized values of figure $A(t_{r\text{-}max})$ and buffer areas.

CPU times needed to get results of *OURS* and *divided* were 3.3 hours and 47 hours, respectively, by a Pentim4(3.2GHz) processor. Therefore, from the results shown above, we can see that the proposed method can find the optimal combinations of sampling switch sizes and sampling pulse waveforms in 11 times faster than the optimizer in smartSPICE.

V. Conclusions

In this paper, we proposed the method of finding optimal combinations of the widths of sampling transistors and the sampling pulse waveforms, each of which minimizes the variation of RPV from 100% caused by various fluctuations. For each combination, we also proposed the design figure which is used to determine the order of designing buffer circuit. An experimental result showed the effectiveness of the proposed method and the figure. From the results, we could also see that our method found almost equal quality design in eleven times faster than the optimizer in SPICE.

Devising an effective and efficient design method for buffer circuit which takes fluctuations into account is an important future work.

References

[1] S.Takahashi, S.Tsukiyama, M.Hashimoto, and I.Shirakawa, "A sampling switch design procedure for active matrix liquid crystal displays," *IEICE Trans. on Fundamentals*, vol.E89-A, no.12, pp.3538-3545, 2006.

[2] E.Lueder, *Liquid Crystal Displays: Addressing Schemes and Electro-Optical Effects*, John Wiley & Sons, 2001.

[3] J.Ohwada, M.Takabatake, Y.A.Ono, A.Mimura, K.Ono, N.Konishi, "Peripheral circuit integrated polySi TFT LCD with gray scale representation," *IEEE Trans. Electron Devices*, vol.36, no.9, pp.1923-1928, 1989.

[4] J.H.Atherton, "Integrating electronics into active matrix liquid crystal displays," *Proc. IEEE Conf. AMLCD*, pp.58-65, 1995.

[5] A.G.Lewis, D.D.Lee, and R.H.Bruce, "Polysilicon TFT circuit design and performance," *IEEE J. Solid-State Circuits*, vol.27, no.12, pp.1833-1842, 1992.

[6] B.Razavi, *Design of Analog CMOS Integrated Circuits*, McGraw-Hill College, 1999.

[7] A.K.Ong, V.I.Prodanov, and M.Tarsia, "A method for reducing the variation in 'On' resistance of a MOS sampling switch," *IEEE Conf. ISCAS*, vol.5, pp.437-440, 2000.

[8] J.H.Shieh, M.Patil, and A.J.Sheu, "Measurement and analysis of charge injection in MOS analog switches," *IEEE J. Solid-State Circuits*, vol. SC-22, no. 2, pp. 277-281, 1987.

[9] G.Wegmann, E.A.Vittoz, and A.Rahali, "Charge injection in analog MOS switches," *IEEE J. Solid-State Circuits*, vol. SC-22, no. 6, pp.1091-1097, 1987.

[10] C.Eichenberger, and W.Guggenbuhl, "On charge injection in analog MOS switches and dummy switch compensation techniques", *IEEE Trans. Circuit and Systems*, vol.37, no.2, pp.256-264, 1990.

[11] K.Leelavattananon, J.B.Hughes, and C.Toumazou, "Very low charge injection switched-current memory cell", *IEEE Conf. ISCAS*, vol.1, pp.531-534, 1998.

[12] W.Xu, and E.G.Friedman, "A CMOS Miller hold capacitance sample-and-hold circuit to reduce charge sharing effect and clock feedthrough", *IEEE Conf. ASIC/SOC*, pp.92-96, 2002.

[13] F.Yuan, and A.Opal, "Computer methods for switched circuits", *IEEE Trans. Circuits and Systems*, vol.50, no.8, pp.1013-1024, 2003.

[14] J.H.Shieh, M.Patil, and A.J.Sheu, "Measurement and analysis of charge injection in MOS analog switches," *IEEE J. Solid-State Circuits*, vol. SC-22, no. 2, pp. 277-281, 1987.

[15] G.Wegmann, E.A.Vittoz, and A.Rahali, "Charge injection in analog MOS switches," *IEEE J. Solid-State Circuits*, vol. SC-22, no. 6, pp.1091-1097, 1987.

[16] W.H.Press, S.A.Teukolsky, W.T.Vetterling, and, B.P.Flannery, *Numerical Recipes in C : The Art of Scientific Computing*, Press Syndicate of the University of Cambridge, 1988.

[17] T.Enomoto, *CMOS Integrated circuit*, Baifukan, 1996. (in Japanese)

[18] SmartSPICE, http://www.silvaco.com