Design Checkout of Logic Networks

ITARU ARAMAKI* AND JUN SHIBATA*

1. *Introduction*

This paper deals with an application program named GPLS (General Purpose Logic Simulator) which has been developed to check the validity of logical designs while they are still on the drawing board.

GPLS can execute logic simulation of (1) synchronous logic circuits, (2) asynchronous logic circuits, and (3) mixtures of both types of circuits when basic logical design is over (AND, OR level), or when the logic circuit is expressed in terms of actual logic elements such as Texas Instruments' J-K flip-flops in dual-in-line packages, or if the designer finds it useful, when the circuit is a mixture of various stages of logical design, as is often found in designers' memos.

The input data for GPLS are the interconnections of logic elements and input signals applied to the circuit. From these data GPLS prints out the static time chart of logical 0 or logical 1 at the arbitrarily specified terminals of arbitrarily specified logic elements. The times at which logical signals are to be printed out can also be predetermined. Signal voltage levels are treated as either high (logical 1) or low (logical 0).

GPLS is unique in that it uses the concept of simulation "steps" and "phases" in order to handle synchronous and asynchronous logic circuits as well as mixtures of both.

The fundamental unit of time in GPLS is called a "step". Each logical state within a step which corresponds to a state of the register is called a "phase".

Thus, there may be a number of phases in one step when the logic circuit contains loops without delays, or asynchronous elements.

GPLS is best suited for the design checkout of medium to small scale industrial digital machines such as traffic control terminals and central equipment.

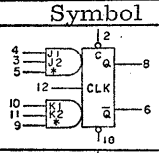
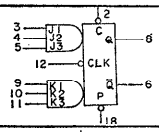
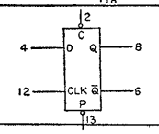
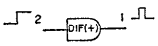
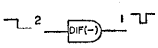
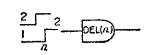
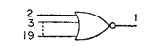
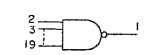
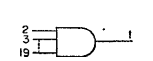
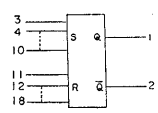
Many design checks have proved that GPLS is a very useful and convenient tool for preconstruction debugging of logic networks.

The General Purpose Logic Simulator (GPLS) is so named because it can perform logic simulation of any circuit constructed from the logic elements listed in Table 1.

This paper first appeared in Japanese in Joho-Shori (the Journal of the Information Processing Society of Japan), Vol. 11, No. 4 (1970), pp. 198-208.

* Sumitomo Electric Industries, Ltd.

Table 1. Logic elements registered in GPLS.

No.	Code	Function	Symbol	Remarks
12	SN7470N	J-K FLIPFLOP		Manufactured by Texas Instruments
13	SN7472N	J-K MASTER-SLAVE FLIPFLOP		Manufactured by Texas Instruments
15	SN7474N	D FLIPFLOP		Manufactured by Texas Instruments
24	DIF(+)	Assumes logical 1 when input changes from logical 0 to logical 1		Output terminal is terminal 1 Input terminal is terminal 2
25	DIF(-)	Assumes logical 0 when input changes from logical 1 to logical 0		
26	DEL(n)	Delays signal by units of simulator clock time		
27	NOR	$Z=\overline{A+B+.....+S}$		Terminal 1 is output terminal. Terminals 2 through 19 are input terminals. Z denotes output while A, B, . . . , S denote inputs at terminals 2, 3, . . . , 19, respectively.
28	NAND	$Z=\overline{A\cdot B\cdot C.....S}$		
29	OR	$Z=A+B+.....+S$		
30	AND	$Z=A\cdot B\cdot C.....S$		
31	NOT	$Z=\overline{A}$		Terminal 1 is output terminal Terminal 2 is input terminal
32	RSFF	R-S FLIPFLOP		Inputs at terminals 3, 4, . . . , 10 are ORed to form SET input. Inputs at terminals 11, 12, . . . , 18 are ORed to form RESET input.

Notes: 1. Positive logic is employed throughout.
2. Terminals not used may be neglected.

2. *Algorithm for Logic Simulation*

A general form of sequential machines treated by GPLS is shown in Fig. 1.

In Fig. 1 the combinational network is a network of logic elements other than memory elements which is capable of certain logical operation.

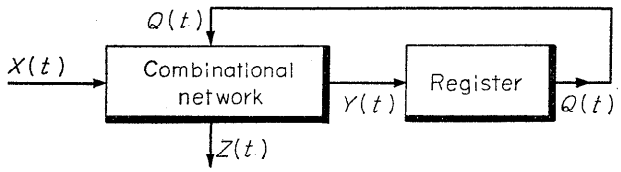


Fig. 1. General form of sequential machines.

The register is made up of flipflops and delay elements and constitutes a memory block.

$X(t)$, $Y(t)$, etc. designates the input (or output) of the two blocks, which are regarded as vectors such as

$$X(t)=[X_1(t),\ X_2(t),\ \cdots,\ X_k(t)]$$

if there are k inputs (or outputs).

Logic level numbers are assigned to facilitate logic operation. The register output $Q(t)$ and primary input $X(t)$ are assigned a logic level of zero. Subsequent logic level i must be known prior to determining the output of all gates of logic level $i+1$.

The logic circuit shown in Fig. 1 is not necessarily synchronous. Moreover the feedback loop from $Y(t)$ to $Q(t)$ does not always contain delay elements.

For this reason, the concept of steps and phases was introduced.

The step is the fundamental unit of time in GPLS, and the primary inputs $X(t)$ are synchronized to and timed by the step.

The phase is a logical state of the network corresponding to a state of the memory elements. In cases where asynchronous elements are included in the network or the loop from $Y(t)$ to $Q(t)$ does not include delay elements as in Fig. 2, 4, many phases within a step may be required to determine the output of $Y(t)$ and $Q(t)$ for one set of the input $X(t)$.

Let n and m denote the number of the step and phase respectively. Then the following recursive equations hold at step n and phase m

$$\left.\begin{aligned} &Y(n,\ m)=C[X(n),\ Q(n,\ m)]\\ &Q(n,\ m)=R[Y(n,\ m-1)]\\ &\qquad (m=2,\ 3,\ 4,\ \cdots\cdots)\end{aligned}\right\} \qquad (1)$$

$$\left.\begin{aligned} &Y(n,\ 1)=C[X(n),\ Q(n,\ 1)]\\ &Q(n,\ 1)=R[Y(n-1)]\\ &\qquad (m=1)\end{aligned}\right\} \qquad (2)$$

Where $Y(n,\ m)$ and $Q(n,\ m)$ denote the respective logical values of $Y(t)$ and $Q(t)$ at phase m of step n.

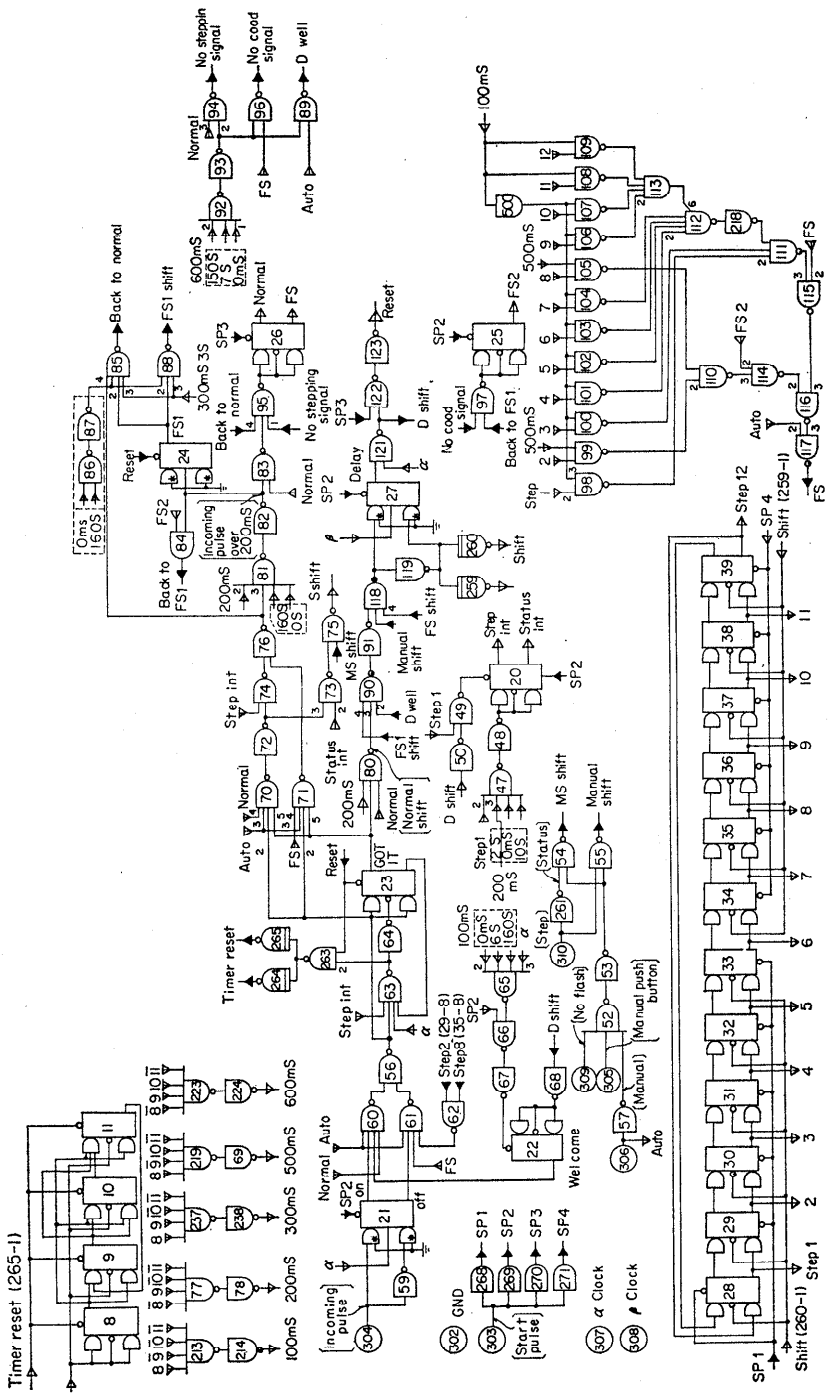


Fig. 2. Control section of the SSC-4 area traffic control terminal.

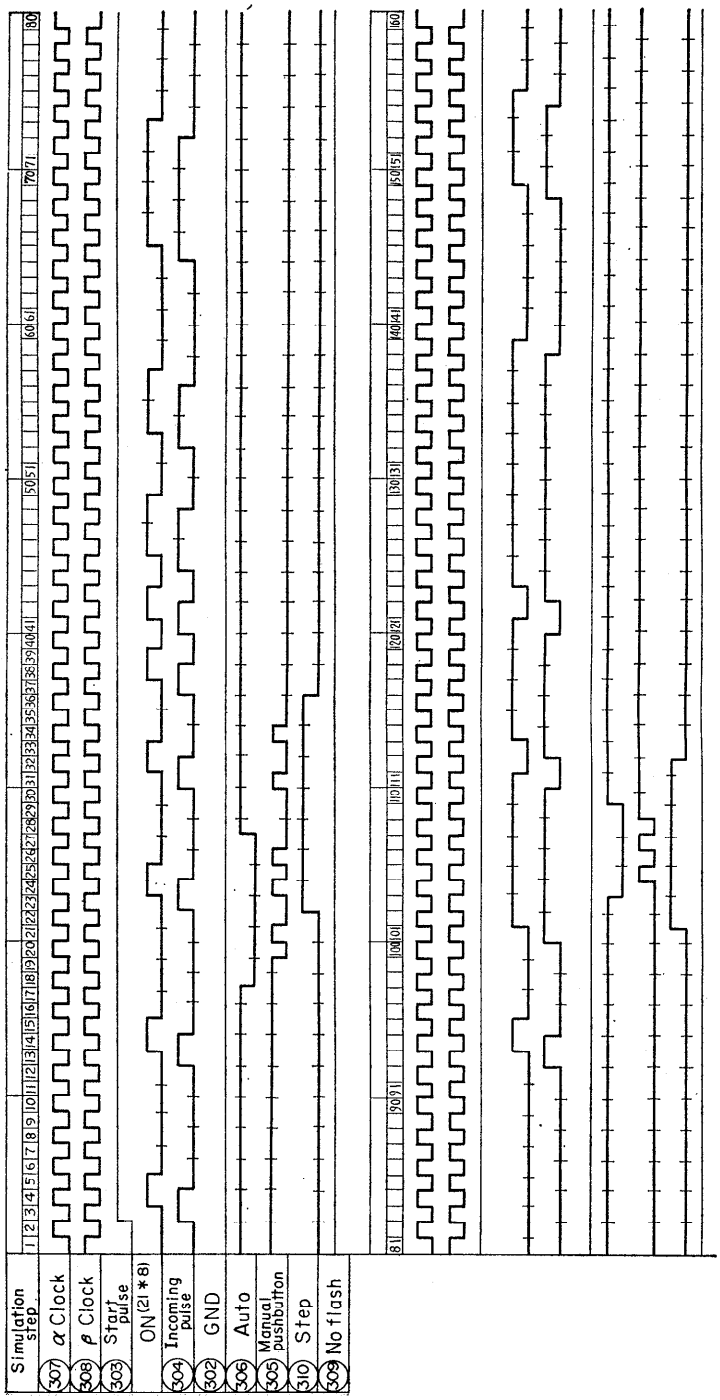


Fig. 3. Input signals to the control section of SSC-4.

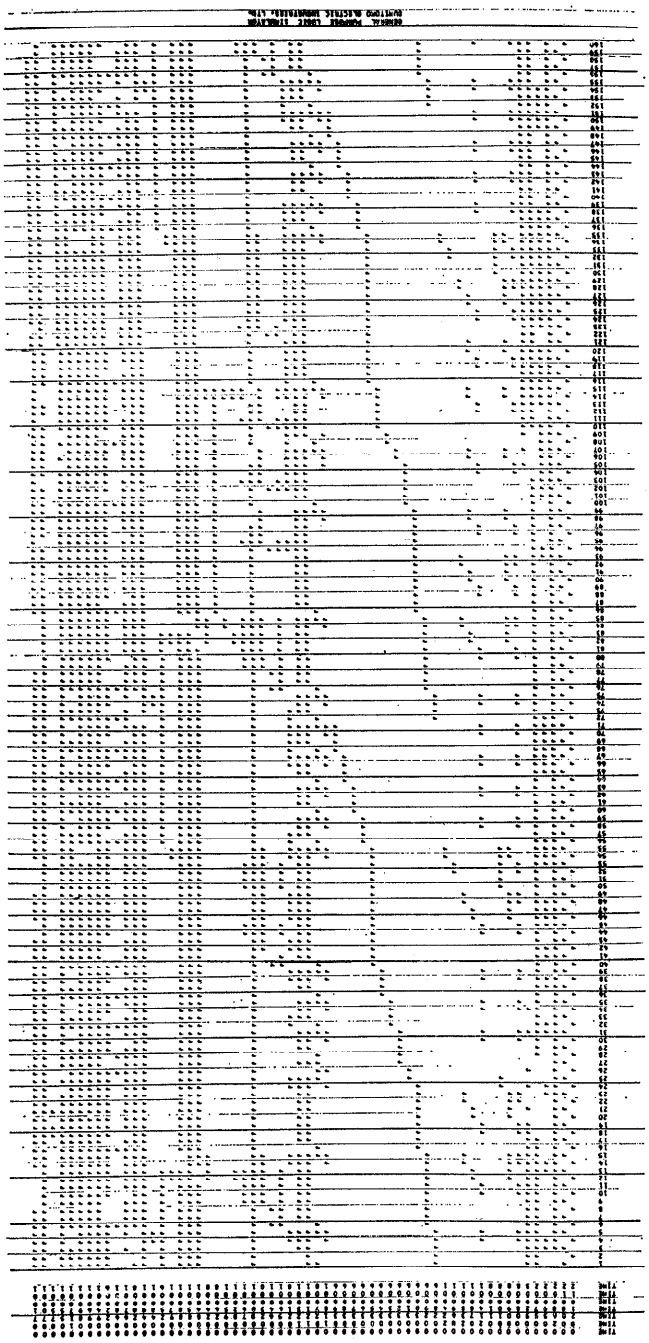


Fig. 4. The result of the simulation run of SSC-4.

C and R denote logical functions which describe the combinational circuit and the memory circuit respectively.

$Y(n-1)$ is the logical value of $Y(t)$ finally determined at step $n-1$. That is

$$Y(n-1) \equiv Y(n-1,\ m)$$

for an m such that

$$Y(n-1,\ m) = Y(n-1,\ m+1)$$

If no such m exists, the circuit is unstable and the design has to be corrected.

3. *An Example of Logical Design Checkout by GPLS*

The area traffic control terminal equipment shown in Fig. 2 responded to the primary input signals shown in Fig. 3 as indicated in Fig. 4.

4. *Conclusion*

A logic simulator named GPLS has been developed in order to check out the logical design of medium and small scale industrial digital machines. GPLS has the following features:

(1) A total of twelve different kinds of logic elements can be treated, including differentiator, delay element (whose delay time is variable), and Texas Instruments IC's in dual-in-line packages.

(2) Both synchronous and asynchronous logic networks at various stages of logical design can be simulated.

(3) Logic feedback loops can be handled.

(4) Circuit errors such as races and hazards can be detected.

(5) Designers with no knowledge of computers or computer programming can use GPLS without difficulty.