

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
25 September 2003 (25.09.2003)

PCT

(10) International Publication Number
WO 03/079550 A2

- (51) International Patent Classification⁷: **H03K 19/177**
- (21) International Application Number: PCT/IB03/00967
- (22) International Filing Date: 17 March 2003 (17.03.2003)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
02076049.2 18 March 2002 (18.03.2002) EP
- (71) Applicant (for all designated States except US): **KONINKLIJKE PHILIPS ELECTRONICS N.V.** [NL/NL];
Groenewoudseweg 1, NL-5621 BA Eindhoven (NL).

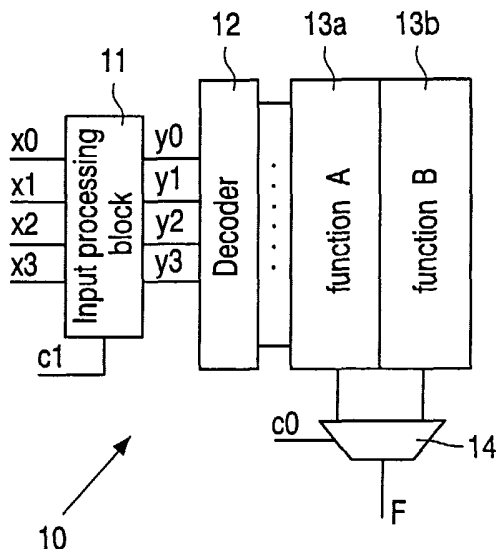
- (81) Designated States (national): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NI, NO, NZ, OM, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.
- (84) Designated States (regional): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

- (72) Inventor; and
- (75) Inventor/Applicant (for US only): **LEIJTEN-NOWAK, Katarzyna** [PL/NL]; c/o Prof. Holstlaan 6, NL-5656 AA Eindhoven (NL).
- (74) Agent: **DE JONG, Durk, J.**; Internationaal Octrooibureau B.V., Prof. Holstlaan 6, NL-5656 AA Eindhoven (NL).

Published:
— without international search report and to be republished upon receipt of that report

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: IMPLEMENTATION OF WIDE MULTIPLEXERS IN RECONFIGURABLE LOGIC



(57) Abstract: A reconfigurable processing device comprising one or more reconfigurable processing units is disclosed. At least a processing unit includes a computational unit having a preprocessing module for receiving n input signals, and s1 selection signals, and providing k output signals wherein k < n+s1. The computational unit further comprises a m-output look-up table being addressed by the k output signals of the preprocessing module and an output multiplexer for selecting one of the m output signals of the look-up table under control of s2 further selection signals. This allows for the implementation of relatively large multiplexers also in architectures using multi-bit output LUTs. In addition a reconfigurable processing unit is described having an input multiplexer for selecting input signal from a communication network, which input multiplexer is configurable statically or dynamically.

WO 03/079550 A2

Implementation of wide multiplexers in reconfigurable logic

The present invention relates to a reconfigurable processing device.

Reconfigurable logic (RL) is used to implement functions which are unknown at the design time. To enable this reconfigurable logic architectures are made generic in the sense that various logic functions can be mapped onto them. Although this offers flexibility to a user, at
5 the same time it leads to a large area overhead compared with the logic which would be required for a standard ASIC implementation, for example. Because of this, a basic computational element of traditional reconfigurable logic devices (such as field programmable gate arrays (FPGAs)), the so called logic cell, contains only a limited amount of logic resources, e.g. look-up tables. Moreover, the resources of a logic cell are chosen in
10 such a way that they reflect the common requirements of different applications. However, if coarser or more specialised functions are to be mapped, usually either much more logic cells are required or such functions cannot be mapped at all.

Reconfigurable logic devices comprise data flow controlling elements such as multiplexers, demultiplexers, gates etc and dataflow manipulating elements, such as logical
15 gates, adders and lookup tables. The latter allow implementation of various functions, which can be redefined easily by loading them with a different content. Reconfigurable logic devices further comprise configuration memory units (configuration memory) for storing settings of data flow controlling elements, such as multiplexers, demultiplexers, switches etc. In this way connections between different parts of the reconfigurable processing unit can be
20 rapidly redefined. Control signals for the memory cells are generated by an address decoder in response to an address offered at the input thereof.

In particular multiplexers of different sizes and with different operand widths are heavily used in random logic as well as data-path applications. Despite that, in a large majority of traditional reconfigurable logic architectures, there is support for the
25 implementation of small multiplexers only. This limitation has two reasons:

1. A specific type of a logic cell which hampers a multiplexer implementation.
2. A limited number of logic cell inputs which causes that a logic cell cannot obtain a required number of signals from the routing resources.

For applications, in which wide multiplexers are heavily used (e.g. DSP data-paths, cryptography, networking), this is an important limitation. Although, this is particularly an issue for architectures with multi-bit output LUTs (moLUTs), any current FPGAs, both fine- and coarse-grained, face this problem in some way.

5

It is an object of the invention to provide a reconfigurable processing device which allows for the implementation of relatively large multiplexers also in architectures using multi-bit output LUTs. In order to achieve this object the reconfigurable processing device is defined by Claim 1.

10

In the reconfigurable processing device according to the invention, the preprocessing module reduces the number of signals, i.e. number n of input signals, and the number $s1$ of selection signals, to a smaller set of k output signals. The combination of the preprocessing module and the look-up table now can handle a relatively large number of input signals as compared to the look-up table alone.

15

Preferably the look-up table is implemented according to the definition of claim 2. This has the advantage that the decoder can be used both for writing and reading the table. The storage unit could also be used for temporarily storing data which is calculated in the reconfigurable processing device.

20

A practical embodiment is described in claim 3. In this embodiment the preprocessing unit can have a relatively small number of gates.

The embodiment of claim 4 introduces additional flexibility to the reconfigurable device, in that it renders it possible to configure the computational unit either as a multiplexer, or as a general look-up table.

25

Claim 5 provides an efficient way to implement an even wider multiplexer.

Claim 6 claims a computational unit provided with an input multiplexer for selecting signals available at a communication network. In particular for this purpose wide input multiplexers are important to enable a high degree of flexibility in coupling different reconfigurable processing units of a reconfigurable processing device to each other. The input multiplexer can be configured either statically or dynamically. In the statical case the selection made by the input multiplexer is determined by the values stored in the configuration memory. In the dynamical case the selection made by the input multiplexer is determined during runtime by signals available at the communication network. In case of a

30

computational unit with a plurality of input pins, each of the pins or a part thereof may be coupled to a multiplexer in this way.

5 These and other aspects of the invention are described in more detail with respect to the drawing. Therein:

Figure 1 schematically shows a reconfigurable processing device,

Figure 2 shows in more detail the coupling between a reconfigurable processing unit and a communication network,

10 Figure 3 shows a conventional element used in a reconfigurable processing device,

Figure 4A, 4B and 4C shows three conventional approaches for implementing multiplexers in prior art reconfigurable devices,

Figure 5 shows an embodiment according to the invention,

15 Figure 6 shows an implementation of a preprocessing unit,

Figure 7A and 7B show two alternative embodiments for the preprocessing unit,

Figure 8 shows a further embodiment of the invention,

20 Figure 9 shows a computational unit having an input multiplexer which is configurable statically or dynamically.

Figure 1 shows a reconfigurable processing device 100 comprising one or more reconfigurable processing units 1. The units 1 may communicate to each other via a communication network comprising horizontal buses 90_H and vertical buses 90_V . The units 1 are connected to configuration bus CB, comprising an address bus and a data bus and control signals for allowing configuration data to be loaded in a configuration memory which controls the functioning of the units 1. In the embodiment shown the reconfigurable processing device 100 comprises a first 60 and a second global decoder 70 which respectively activate a first control line, e.g. 61a and a second control line e.g. 71 b. The reconfigurable processing unit, here 1ab selected by the control lines is then reconfigured with the data from the configuration bus CB. In another embodiment the reconfigurable processing device 100 may only have local address decoders. In again another embodiment the reconfigurable processing device has no decoder at all, for example in an embodiment

wherein the configuration storage units are arranged in a chain. In that case configuration takes place by subsequently shifting the configuration data in the chain.

For many purposes it is necessary to select input signals for a reconfigurable processing unit 1 from a plurality of signals available in the communication network. Figure 5 2 shows an example wherein the multiplexer 13 selects an input signal from the signals available at a horizontal bus 90_H of the communication network.

Reconfigurable logic architectures with multi-bit output LUTs (moLUTs), have shown to offer multi-functionality at a reduced implementation cost. The most common way of implementing moLUTs is similar to traditional SRAMs, this is by using one decoder 10 for addressing several (LUT) memory columns. A 4-input LUT is found to be the most area-efficient for random logic implementation. For that reason, a 4-input LUT is typically used in the implementation of the moLUT-based devices. The LUT type determines the total number of logic cell input pins. Thus, a device with a 4-input moLUT will have four pins. If an additional 2:1 multiplexer is placed at the LUT outputs, it will result in maximally five pins 15 (see Fig. 3). This is not enough even to implement a 4:1 multiplexer, which requires six inputs (four primary and two selection inputs). Thus, typically only very small multiplexers can be implemented in such devices.

Figure 4A, 4B, 4C show some prior art implementations of a multiplexer. Figure 4A shows a first approach, in which the multiplexer is implemented by dedicated 20 circuitry. In such architectures the multiplexer function is implemented by programming connections in a fixed topology of logic gates as well as by the selection of right operands for them (e.g. constants).

The second approach, shown in Figure 4B, is characteristic for the multiplexer-based devices as designed by Actel. In such devices, a logic function is 25 implemented by programming multiplexer inputs in a way required by the mapped function. The multiplexer-based reconfigurable logic devices are of a fine granularity and contain a small set of 2:1 multiplexers (usually three). Thus, maximally a 4:1 multiplexer can be implemented in such a structure.

The third type of implementation as shown in Figure 4C assumes the use of 30 look-up tables (LUTs), which is typical for most of the current FPGAs, e.g. from Atmel and Xilinx.

According to this approach, look-up tables in a logic cell implement single multiplexers of the limited size (typically a 4:1 multiplexer), while wider multiplexers are created by using additional 2:1 multiplexers present at the look-up table outputs.

Figure 5 shows a computational unit in an embodiment of a reconfigurable processing device according to the invention. The computational unit 10 comprises a preprocessing module 11 for receiving $n=4$ input signals x_0, x_1, x_2, x_3 , and $s_1=1$ selection signals, signal c_1 and providing $k = 4$ output signals y_0, y_1, y_2, y_3 , wherein $k < n+s_1$. It further comprises a $m=2$ -output look-up table 12, 13a, 13b, being addressed by the k output signals of the preprocessing module, and an output multiplexer 14 for selecting one of the $m=2$ output signals of the look-up table 12, 13a, 13b under control of $s_2=1$ further selection signals as the output signal F .

The basic idea behind the method proposed here is based on the decomposition of the multiplexer function in such a way, that it can be mapped onto the multi-bit output LUT which has been enhanced with a small amount of extra logic. By way of example, this is shown for the implementation of a 4:1 multiplexer on the multi-bit output LUT with 4-inputs and 2-outputs (4/2-LUT).

A 4:1 multiplexer can be described by a logic function F of six variables: x_0, x_1, x_2, x_3 , being inputs of the multiplexer and c_0, c_1 being control (selection) signals, as shown by the following equation:

$$(1) \quad F(x_0, x_1, x_2, x_3, c_0, c_1) = \overline{c_0} \cdot \overline{c_1} \cdot x_0 + c_0 \cdot \overline{c_1} \cdot x_1 + \overline{c_0} \cdot c_1 \cdot x_2 + c_0 \cdot c_1 \cdot x_3$$

This equation can be further modified to the form:

$$(2) \quad F(x_0, x_1, x_2, x_3, c_0, c_1) = \overline{c_0} \cdot A + c_0 \cdot B, \text{ wherein:}$$

$$(3a) \quad A = \overline{c_1} \cdot x_0 + c_1 \cdot x_2, \text{ and}$$

$$(3b) \quad B = \overline{c_1} \cdot x_1 + c_1 \cdot x_3$$

Eqn 2 describes a 2:1 multiplexer with inputs A, B , and c_0 being a control signal. This multiplexer can be mapped onto the 2:1 multiplexer which is present at the outputs of the 4/2-LUT. Such a mapping is possible only if both functions A and B can be encoded in the memories (memory columns) of the moLUT. Functions A and B require in total five different logic variables while the given 4/2-LUT has only four inputs. However, all partial products

of the A and B functions share the same logic variable c_1 . If it is assumed that these partial products $\overline{c_1} \cdot x_0, \overline{c_1} \cdot x_1, c_1 \cdot x_2, c_1 \cdot x_3$ are generated outside the LUT, then the partial product results y_0, y_1, y_2, y_3 can be treated as inputs of this LUT. Thus, a 4/2-LUT implements functions A and B of the form as in Eqn. 4

5 (4a) $A = y_0 + y_2$, and

(4b) $B = y_1 + y_3$

The only modification required with respect to the standard 4/2-LUT implementation is a relatively small preprocessing unit 11.

The above described principle is not limited to the implementation of 4:2 multiplexers. Analogously any selection function F:

(5) $F(x_0, \dots, x_k, c_0, \dots, c_L) = \overline{c_0} \dots \overline{c_L} \cdot x_0 + \dots + c_0 \cdot c_L \cdot x_k$ for selecting an output signal from input signals x_0, \dots, x_k may be rewritten in the form

(6) $F(x_0, \dots, x_k, c_0, \dots, c_L) = \overline{c_0} \cdot A + c_0 \cdot B$, where A is a function of the variables x_0, x_2, \dots, x_{2i} and c_1, \dots, c_L , and B is a function of the variables $x_1, x_3, \dots, x_{2i+1}$, and c_1, \dots, c_L . The function A on its turn can be rewritten as the logical OR of the variables y_0, y_2, y_{2i} , wherein y_i is a function which only depends on the variable x_i and the selection variables c_1, \dots, c_L .

Likewise the function B can be rewritten as the logical OR of the variables y_1, y_3, y_{2i+1} , wherein y_i is a function which only depends on the variable x_i and the selection variables c_1, \dots, c_L . Again, the functions y_i can be calculated with a preprocessing unit, while the resulting function F is calculated from the values of y_i and the remaining selection variables c_1, \dots, c_L .

The preprocessing unit for generating the values y_i is characterized in that the number n of input signals x_i is equal to the number k of output signals y_i and that the preprocessing unit has a first input for receiving a first selection signal c_1 , as well as further inputs for receiving the further selection signals c_2, \dots, c_L . The preprocessing unit generates in response to each of the input signals x_i an output signal y_i . For one half of the output signals y_i the value is the logical AND function of the corresponding input signal x_i and the first selection signal c_1 and for another half of the output signals y_i the value is the logical AND function of the corresponding input signal x_i and the inverse of the first selection signal c_1 .

By way of example Figure 6 shows a preprocessing unit 11 in a practical embodiment of the invention. The preprocessing unit comprises 4 logical AND gates 11a, 11b, 11c, 11d, and an inverter 11e to calculate the values y_0, \dots, y_3 . The values y_0 and y_1 are realized by the logical AND function of their corresponding input signals x_0, x_1 and the

inverse of the first selection signal c_1 . The values y_2 and y_3 are realized by the logical AND function of their corresponding input signals x_2 , x_3 and first selection signal c_1 itself.

Preferably the preprocessing module comprises a mode select input for selecting a further operational mode, in which further operational mode the output signals y_i are identical to their corresponding input signals x_i . If a logic function of only four inputs is to be implemented in the modified 4/2-LUT, than the input processing block can be either bypassed Figure 7A or it can be used in such a way that the primary logic inputs are passed through it without any conversion. In Figure 7A parts being referred to by reference numbers having a quote (') correspond to parts having the same reference number in Figure 6. In Figure 7B parts referred to by reference numbers having a double quote correspond to parts having a single quote in Figure 7A.

In the embodiment shown in Figure 7A each of the AND gates 11a', ... 11d' is coupled to an output via a respective auxiliary multiplexer 15a', ... 15d'. The multiplexers are coupled via a control input 17' of the preprocessing unit 11' to an output of the configuration memory 30'.

In the embodiment of Figure 7B the signal of the inverter 11e'' is bypassed in the further operational mode. To that end the auxiliary multiplexers 16a'' and 16c'' are coupled to control input 17'' which is coupled to configuration memory 30''. In the further operational mode the auxiliary multiplexers select the logical value 1 as their output signals. The result is that each of the output signals y_i is equal to x_i . An additional advantage is that the auxiliary multiplexers 16a'' and 16c'' do not delay the signals $y_0, .. y_3$.

Figure 8 shows a further embodiment of a reconfigurable processing device according to the invention claim 1. It comprises an enlarged computational unit having a first 10a and a second computational unit 10b. Each of the computational units 10a, 10b comprises a preprocessing module, a m-output look-up table, and an output multiplexer. In the embodiment shown the computational units 10a, 10b are identical to the one shown in Figure 5. The enlarged computational unit shown in Figure 8 further comprises a further multiplexer 18 for selecting an output signal of either the first 10a or the second unit 10b as its output signal F in response to a further selection signal c_2 .

Figure 9 shows a reconfigurable processing device wherein the computational unit 10 is arranged as an input multiplexer for a reconfigurable logic unit 1. The computational unit 10 selects an input signal for the reconfigurable logic unit 1 from signals available at a communication network $90_H, 90_V$. Although in the embodiment of Figure 9 the reconfigurable logic unit 1 has only one input, it may have a plurality of inputs. Each, or a

part of those inputs may be coupled to a computational unit 10 which is arranged as an input multiplexer.

In the embodiment shown in Figure 9 the selection signals for the computational unit 10 are provided by auxiliary computational units 20, 21, 22. Each auxiliary computational unit 20, 21, 22 selects either a signal from a configuration memory M2, M3 and M4 respectively or a signal from the communication network 90_v as its input signal. This selection is made in response to an auxiliary selection signal provided by the configuration memory M1.

It is remarked that the scope of protection of the invention is not restricted to the embodiments described herein. Neither is the scope of protection of the invention restricted by the reference numerals in the claims. The word 'comprising' does not exclude other parts than those mentioned in a claim. The word 'a(n)' preceding an element does not exclude a plurality of those elements. Means forming part of the invention may both be implemented in the form of dedicated hardware or in the form of a programmed general purpose processor. The invention resides in each new feature or combination of features.

CLAIMS:

1. Reconfigurable processing device (100) comprising one or more reconfigurable processing units (1) including a computational unit (10) having
 - a preprocessing module (11) for receiving n input signals (x_0, \dots, x_3), and s_1 selection signals (c_1), and providing k output signals (y_0, \dots, y_1) wherein $k < n + s_1$,
 - 5 - an m -output look-up table (12, 13a, 13b) being addressed by the k output signals of the preprocessing module (11),
 - an output multiplexer (14) for selecting one of the m output signals of the look-up table (12, 13a, 13b) under control of s_2 further selection signals (c_0).
- 10 2. Reconfigurable processing device according to claim 1, characterized in that the look-up table includes a $k:2^k$ decoder (12) for decoding k address signals (y_0, \dots, y_3) into 2^k control signals and a storage unit (13a, 13b) comprising m columns of 2^k storage elements each.
- 15 3. Reconfigurable processing device according to claim 1, characterized in that $n = k$ and that the preprocessing unit (11) has a first input for receiving a first selection signal (c_1), wherein the preprocessing unit (11) in an operational mode generates in response to each of the input signals (x_i) an output signal (y_i) wherein for one half of the output signals (y_2, y_3) the value is the logical AND function of the corresponding input signal (x_2, x_3 resp.) and the first selection signal (c_1) and for another half of the output signals (y_0, y_1) the value is
20 the logical AND function of the corresponding input signal (x_0, x_1 resp.) and the inverse of the first selection signal (c_1).
4. Reconfigurable processing device according to claim 3, characterized in that
25 the preprocessing unit (11'; 11'') comprises a mode select input (17', 17'') for selecting a further operational mode, in which further operational mode the output signals (y_i) are identical to their corresponding input signals (x_i).

5. Reconfigurable processing device according to claim 1, characterized in that it comprises an enlarged computational unit having a first and a second computational unit (10a, 10b) comprising each a preprocessing module (11), a m-output look-up table (12, 13a, 13b), and an output multiplexer (14), the enlarged computational unit further comprising a further multiplexer (18) for selecting an output signal of either the first or the second unit (10a, 10b) as its output signal (F) in response to a further selection signal (c₂).

6. Reconfigurable processing device comprising one or more computational units (1) with at least one input terminal which is coupled via an input multiplexer (10) to a communication network (90_V, 90_H 90_H), characterized in that selection signals for the input multiplexer of the computational unit (1) are provided by auxiliary selection units (20, 21, 22), each auxiliary selection unit selecting either a signal from a configuration memory (M2, M2, M3) or from the communication network (90_V) as its input signal, in response to an auxiliary selection signal (M1) provided by the configuration memory.

1/4

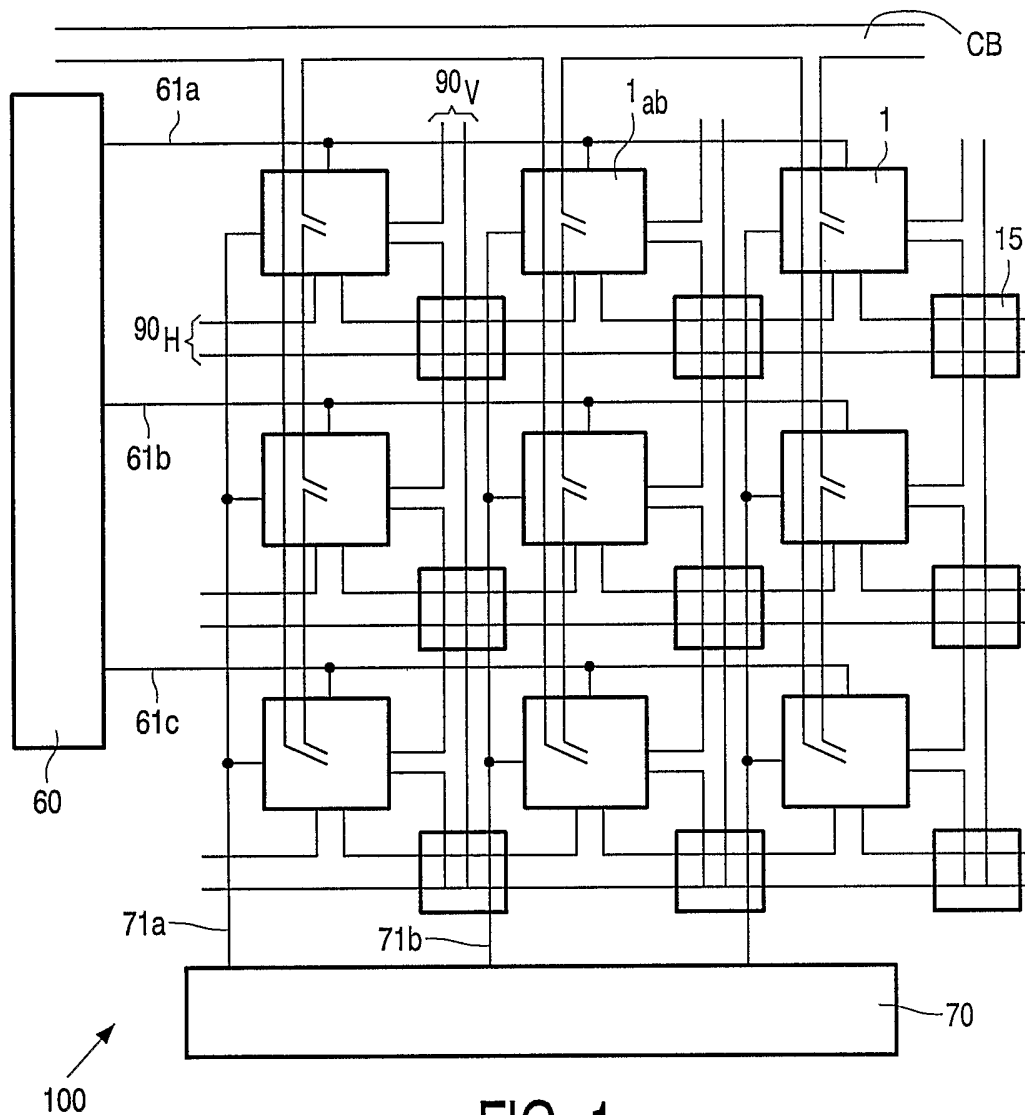


FIG. 1

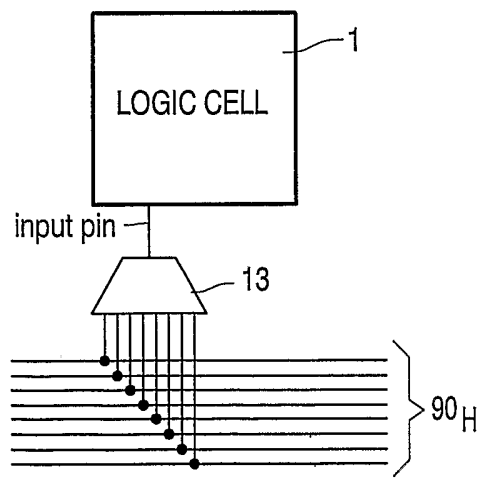


FIG. 2

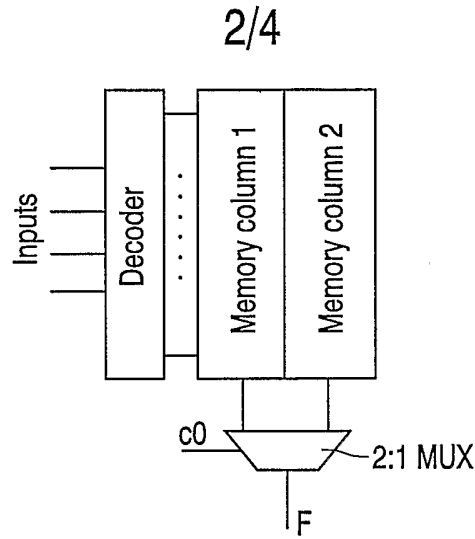


FIG. 3

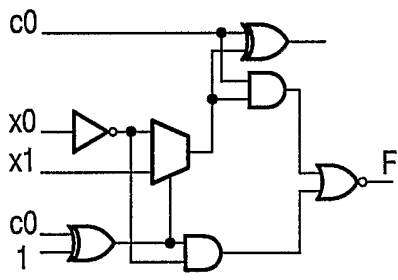


FIG. 4A

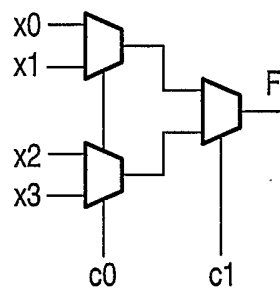


FIG. 4B

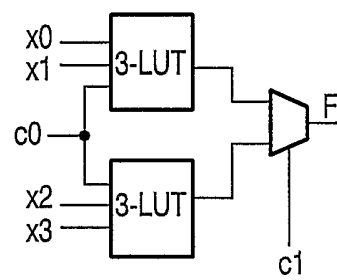


FIG. 4C

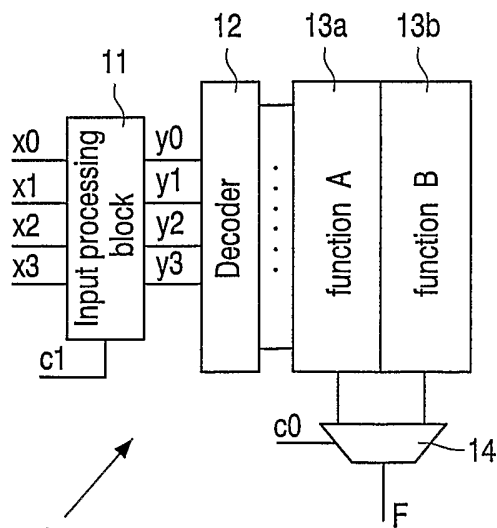


FIG. 5

3/4

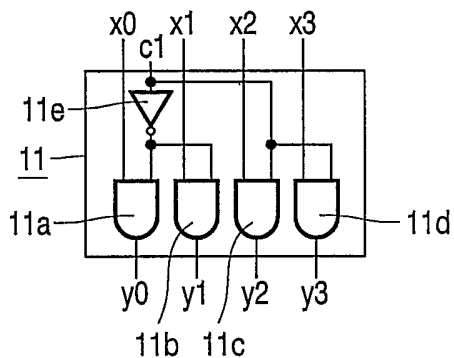


FIG. 6

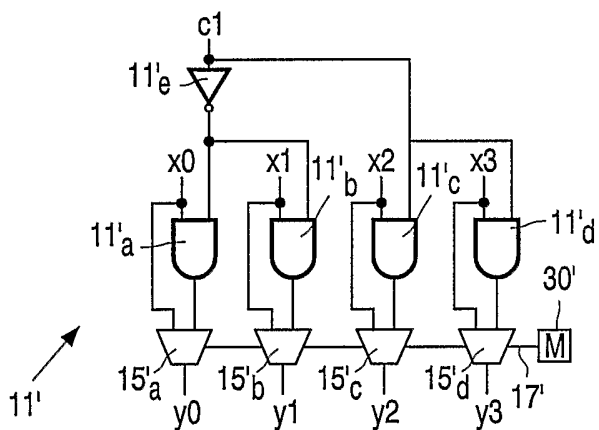


FIG. 7A

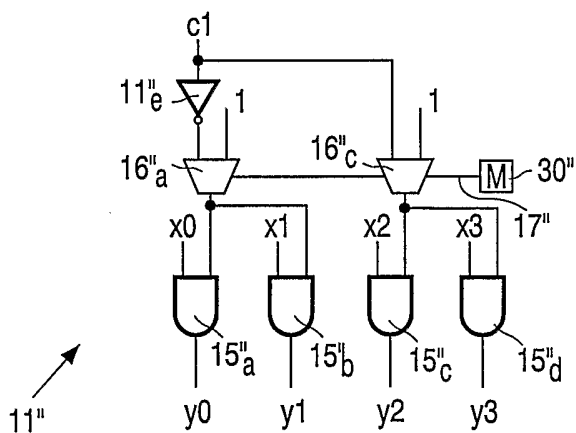


FIG. 7B

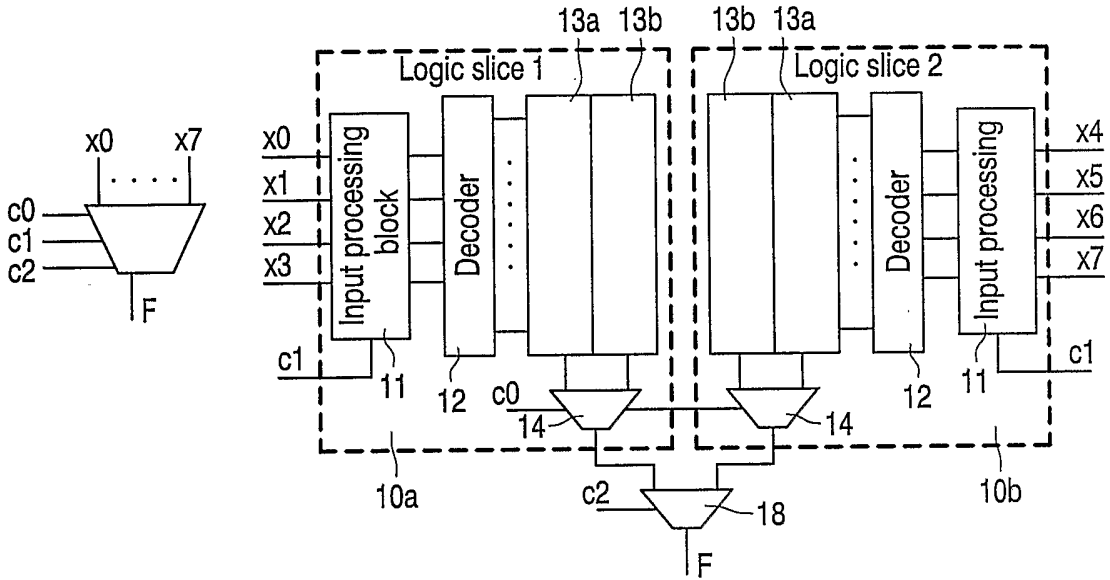


FIG. 8

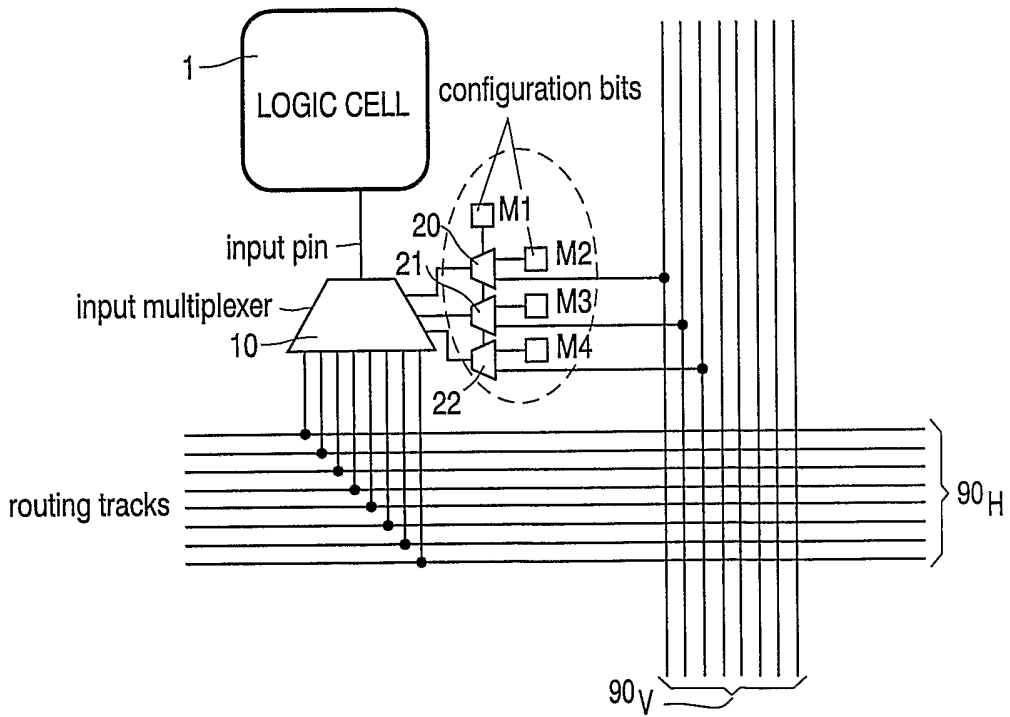


FIG. 9