

Fig. 2. Clarifying the definition of  $\text{POP}_{k+1}(x, y)$ .

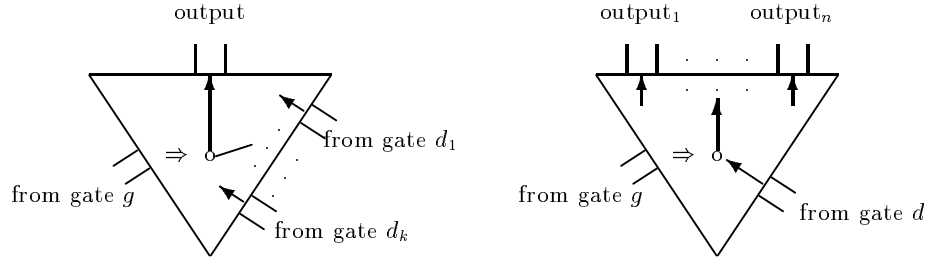
#### 4 Deterministic Circuits

This section presents the simulation of *CROW*-PRAMs by deterministic auxiliary push-down automata. Although not stated explicitly in [6], this fact has already been shown by Dymond and Ruzzo. The basic idea is to describe the work of the owner write PRAM by a recursion of bounded width and depth  $\log f$  which leads to a time bound of  $f^{O(1)}$ . The space requirement is determined by the word length of the PRAM which is  $\log g$ . Observe the importance of the boundedness of the recursion width which is made possible by the owner write restriction of the simulated PRAM.

We will now exhibit this relation by investigating *multiplex select* gates which were introduced by Reinhardt [17]. They were inspired by the *select* gates of Niepel and Rossmanith [16,19]. While characterizations by circuits of depth  $O(\log f)$  for  $f^{O(1)}$  time-bounded nondeterministic auxiliary push-down automata and for  $O(\log f)$  time-bounded concurrent write and exclusive write PRAMs are well-known [11,16,20,23], this will lead to the first characterization of this type in the case of determinism and owner write.

A multiplex select gate has two kinds of input signals: one bundle of  $O(\log(s))$  steering signals and up to  $s$  bundles of  $O(\log(s))$  data signals. Here,  $s$  is the size of the circuit containing the gate. The multiplex select gate, interpreting the steering bits as the binary encoding of a number  $j$ , will switch the data bits of

bundle  $j$  to its bundle of output signals. We will say that the gate *selected* its  $j$ th input. A gate like that might be drawn as sketched in Fig. 3.



**Fig. 3.** Select and deselect gates

A multiplex select gate can be described by the encoding of  $(g, d_1, \dots, d_n)$  where  $g$  is the description of the steering input bundle,  $n$  is the number of data input bundles and  $d_j$  is the description of the  $j$ 'th data input bundle. A description of an input bundle may start with fixed signals for the bits with higher values and may continue with the number of an output bundle of another gate for the bits with lower value. For example, the description 01100*i* says that the first 5 signals of the bundle are 01100 followed by the signals from the output bundle number  $i$  (this is the output of gate number  $i$ , if we have only this type of gates).

We will not go into further details of defining the uniformity of a family of circuits with multiplex select gates. This is handled as in the boolean case (see, e.g., [20]). Let  $MDepthSize(d, s)$  be the class of languages which can be recognized by a uniform family of multiplex select circuits with depth  $d$  and size  $s$ .

**Theorem 2.** *Let  $f$  and  $g$  be constructible functions fulfilling  $f(n) \geq n$ ,  $g(n) \geq n$ ,  $\log(f) = O(g)$ , and  $\log(g) = O(f)$ . Then we have*

$$CROW-TIPR(\log f, g^{O(1)}) \subseteq MDepthSize(\log f, g^{O(1)}).$$

**Proof:** W.l.o.g. we may assume that every processor has at most  $c$  local memory cells and that it always uses the last local cell to store the result of a reading from a global memory cell. Such a configuration of a processor  $p$  at time  $t$  including its local memory cells can be described by a number  $m$  encoding

$$state(p, t)local(p, t, 1) \dots local(p, t, c)$$

with  $O(\log g)$  bits forming a bundle of signals. This bundle is the steering input for a multiplex select gate which has to calculate the following configuration, which is determined by the number  $m$  except the contents of the local cell  $c$  which is eventually loaded from a global cell  $i$  indexed with a local cell. Since