5.1 Multiplexers

In this section we present designs of $(n : 1)$-multiplexers. Multiplexers are often also called *selectors*.

We first review the definition of a MUX-gate (also known as a $(2 : 1)$-multiplexer).

Definition 5.1 *A* MUX*-gate is a combinational gate that has three inputs* $D[0], D[1], S$ *and one output* Y. *The functionality is defined by*

$$Y = \begin{cases} D[0] & \text{if } S = 0 \\ D[1] & \text{if } S = 1. \end{cases}$$

Note that we could have used the shorter expression $Y = D[S]$ to define the functionality of a MUX-gate.

An (n:1)-MUX is a combinational circuit defined as follows:

Input: $D[n - 1 : 0]$ and $S[k - 1 : 0]$ where $k = \lceil \log_2 n \rceil$.

Output: $Y \in \{0, 1\}$.

Functionality:
$$Y = D[\langle \vec{S} \rangle].$$

We often refer to $\vec{D}$ as the *data input* and to $\vec{S}$ as the *select input*. The select input $\vec{S}$ encodes the index of the bit of the data input $\vec{D}$ that should be output. To simplify the discussion, we will assume in this section that n is a power of 2, namely, $n = 2^k$.

Example 5.1 *Let* $n = 4$, $D[3 : 0] = 0101$, *and* $S[1 : 0] = 11$. *The output* Y *should be* 0.

$$D[n-1:0] \qquad S[k-1:0]$$

DECODER(k)

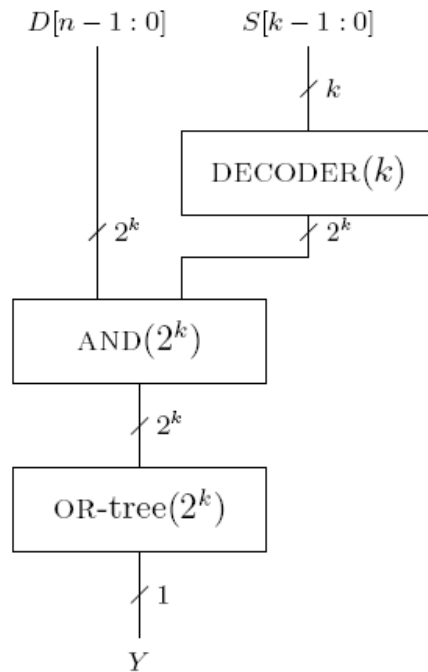2^k $\qquad$ 2^k

AND(2^k)

2^k

OR-tree(2^k)

1

Y

Figure 5.1: An (n:1)-MUX based on a decoder $(n = 2^k)$.

Question 5.1 *The following question deals with the implementation of (n:1)-MUX suggested in Figure 5.1.*

1. *Prove the correctness of the design.*

2. *Analyze the cost and delay of the design.*

3. *Prove that the cost and delay of the design are asymptotically optimal.*

Good Luck !!