

Random I/O (RIO) codes, recently introduced by Sharon and Alrod, is a coding scheme to improve the random input/output performance of flash memories. Multilevel flash memories require, on the average, more than a single read threshold in order to read a single logical page. This number is important to be optimized since it sets the read latency of flash memories. An  $(n, M, t)$  RIO code assumes that  $t$  pages are stored in  $n$  cells with  $t + 1$  levels. The first page is read by applying a read threshold between levels  $t$  and  $t + 1$ . Similarly, the second page is read by applying a read threshold between levels  $t - 1$  and  $t$ , and so on. As a consequence, if a cell reads as bit 1 for a page (say page  $m$ ), the cell also reads as bit 1 for all the successive pages ( $m + 1, m + 2, \dots$ ). Therefore, Sharon and Alrod showed that the design of RIO codes is equivalent to the design of WOM codes. The latter family of codes attracted substantial attention in recent years in order to improve the lifetime of flash memories by allowing writing multiple messages to the memory without the need for an erase operation. In this paper, we notice two important distinctions between RIO codes and WOM codes. While in WOM codes, the messages are received one after the other and thus are not known all in advance, in RIO codes the information of all logical pages can be known in advance when programming the cells. Even though this knowledge does not improve the maximum sumrate of RIO codes, it allows the design of efficient high-rate codes with a moderate block length, which do not exist for WOM codes. We also study another family of RIO codes, called here partial RIO codes, that allow to find even more efficient codes while allowing to sense more than a single threshold to read some pages.