the power consumption. From a hardware perspective, [7]–[22] proposed solutions to improve SHA-256. For instance, the authors of [7] employed the carry-save adder to improve the computation time of the critical path, which increased the maximum frequency and processing rate, while [8]–[12] used pipeline technology to improve the SHA-256 throughput. A cache memory technique was presented in [13] to reuse data, minimize the critical paths, and reduce the number of memory accesses for SHA-256 processing. The authors of [14] adopted the unfolding technique to reduce the computing latency for SHA-256. The authors of [15] proposed using a 7-3-2 array compressor to reduce the critical path delay for SHA-256. The carry-save adders technique is used in [16] to reduce the latency of additions in the SHA-256 algorithm. The authors of [17] used a combination of techniques such as carry-save-adders and pipelines to increase the performance of SHA-256. Pipeline and unrolled techniques are presented in [18] and [19] to increase the throughput of SHA-256. The authors of [20]–[22] presented a SHA-256 implementation on an FPGA for performance evaluation, with no technique optimization. Despite providing improvements in terms of hardware cost and power consumption, the hardware circuits developed in [7]–[22] have low processing rates because they require several (up to 64) clock cycles to compute a single 256-bit hash value.

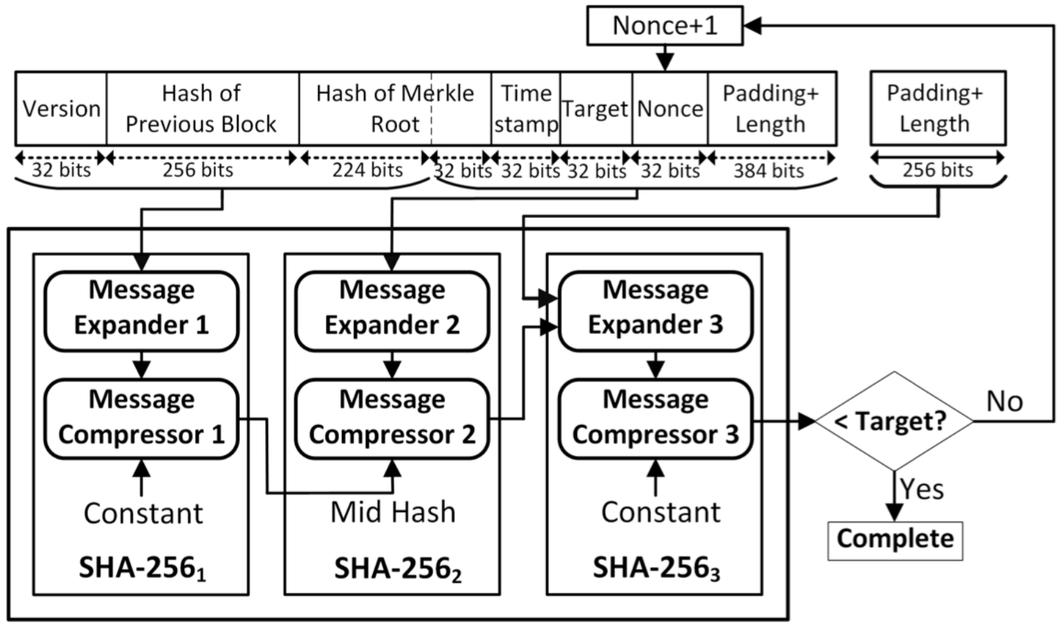

FIGURE 1. Overview architecture of double SHA-256 in Bitcoin Mining.

The remainder of this paper is organized as follows. Section II presents a preliminary study. Section III describes our proposed CME double SHA-256 architecture, and the CME algorithms and hardware circuits are explained in detail. Section IV reports our evaluation in terms of theory, ASIC, and FPGA experiments. Finally, Section V concludes the paper.

II. PRELIMINARIES

A. DOUBLE SHA-256 ARCHITECTURE FOR