Code compression has been shown to be an effective technique to reduce code size in memory constrained embedded systems. It has also been used as a way to increase cache hit ratio, thus reducing power consumption and improving performance. This paper proposes an approach to mix static/dynamic instruction profiling in dictionary construction, so as to best exploit trade-offs in compression ratio/performance. Compressed instructions are stored as variable-size indices into fixed-size codewords, eliminating compressed code misalignments. Experimental results, using the Leon (SPARCv8) processor and a program mix from MiBench and Mediabench, show that our approach halves the number of cache accesses and power consumption while produces compression ratios as low as 56%.