to the right, and the rightmost column has reappeared at the left (a cyclic permutation of the columns).

This establishes the symmetry among the seven bits. Iterating the above procedure five more times, we can make a total of seven different **H** matrices for the same original code, each of which assigns each bit to a different role.

We may also construct the super-redundant seven-row parity-check matrix for the code,

$$\mathbf{H}''' = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}. \tag{1.54}$$

This matrix is 'redundant' in the sense that the space spanned by its rows is only three-dimensional, not seven.

This matrix is also a *cyclic* matrix. Every row is a cyclic permutation of the top row.

Cyclic codes: if there is an ordering of the bits $t_1 \ldots t_N$ such that a linear code has a *cyclic* parity-check matrix, then the code is called a *cyclic code*.

The codewords of such a code also have cyclic properties: any cyclic permutation of a codeword is a codeword.

For example, the Hamming $(7, 4)$ code, with its bits ordered as above, consists of all seven cyclic shifts of the codewords `1110100` and `1011000`, and the codewords `0000000` and `1111111`.

Cyclic codes are a cornerstone of the algebraic approach to error-correcting codes. We won't use them again in this book, however, as they have been superceded by sparse-graph codes (Part VI).

Solution to exercise 1.7 (p.13). There are fifteen non-zero noise vectors which give the all-zero syndrome; these are precisely the fifteen non-zero codewords of the Hamming code. Notice that because the Hamming code is *linear*, the sum of any two codewords is a codeword.

Graphs corresponding to codes

Solution to exercise 1.9 (p.14). When answering this question, you will probably find that it is easier to invent new codes than to find optimal decoders for them. There are many ways to design codes, and what follows is just one possible train of thought. We make a linear block code that is similar to the $(7, 4)$ Hamming code, but bigger.

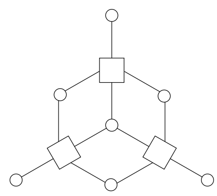

Figure 1.20. The graph of the $(7, 4)$ Hamming code. The 7 circles are the bit nodes and the 3 squares are the parity-check nodes.

Many codes can be conveniently expressed in terms of graphs. In figure 1.13, we introduced a pictorial representation of the $(7, 4)$ Hamming code. If we replace that figure's big circles, each of which shows that the parity of four particular bits is even, by a 'parity-check node' that is connected to the four bits, then we obtain the representation of the $(7, 4)$ Hamming code by a *bipartite graph* as shown in figure 1.20. The 7 circles are the 7 transmitted bits. The 3 squares are the parity-check nodes (not to be confused with the 3 parity-check *bits*, which are the three most peripheral circles). The graph is a 'bipartite' graph because its nodes fall into two classes – bits and checks