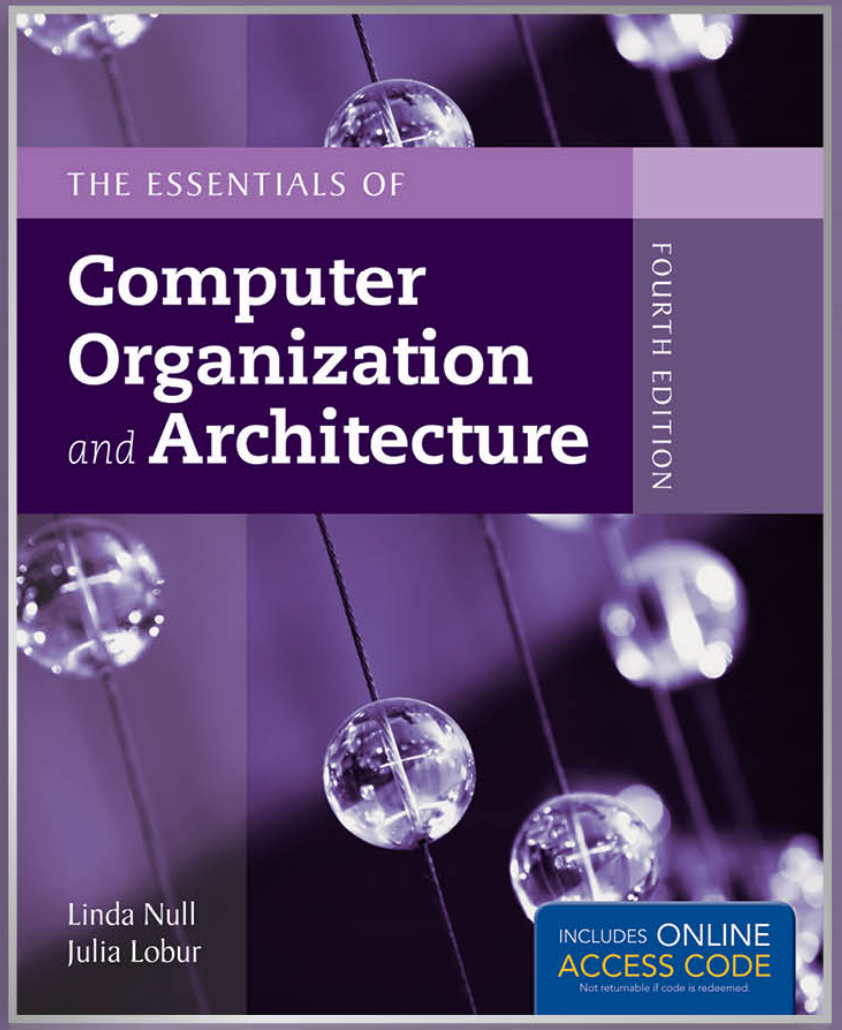
Chapter 2
Special Section

Focus on Codes for Data Recording and Transmission



2.A Introduction

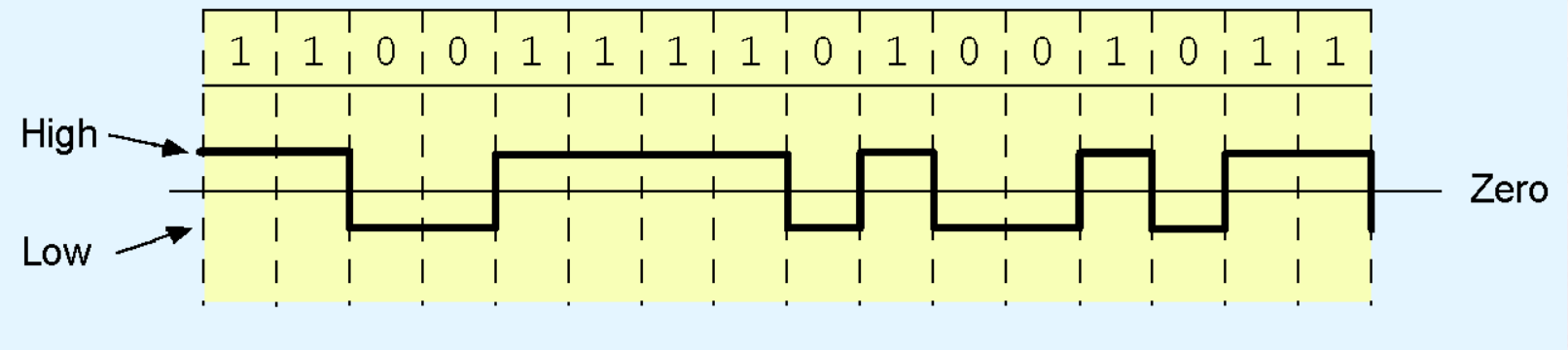
- The main part of Chapter 2 provides great detail about the various ways in which digital computers express numeric and non-numeric values.
- These expressions are an abstraction for the way in which the values are actually stored on computer media and sent over transmission media.

2.A Introduction

- To transmit data, pulses of “high” and “low” voltage are sent across communications media.
- To store data, changes are induced in the magnetic polarity of the recording medium.
  - These polarity changes are called *flux reversals.*
- The period of time during which a bit is transmitted, or the area of magnetic storage within which a bit is stored is called a *bit cell*.

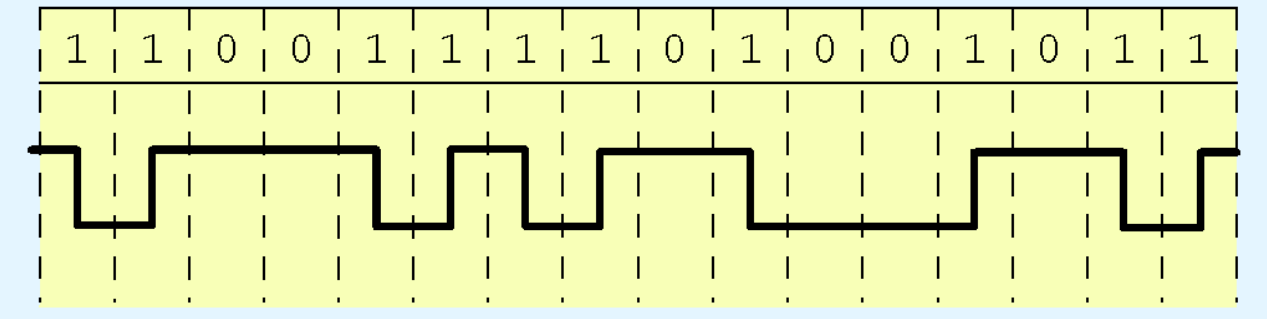
2.A.1 Non-Return-to-Zero Code

- The simplest data recording and transmission code is the non-return-to-zero (NRZ) code.
- NRZ encodes 1 as “high” and 0 as “low.”
- The coding of *OK* (in ASCII) is shown below.



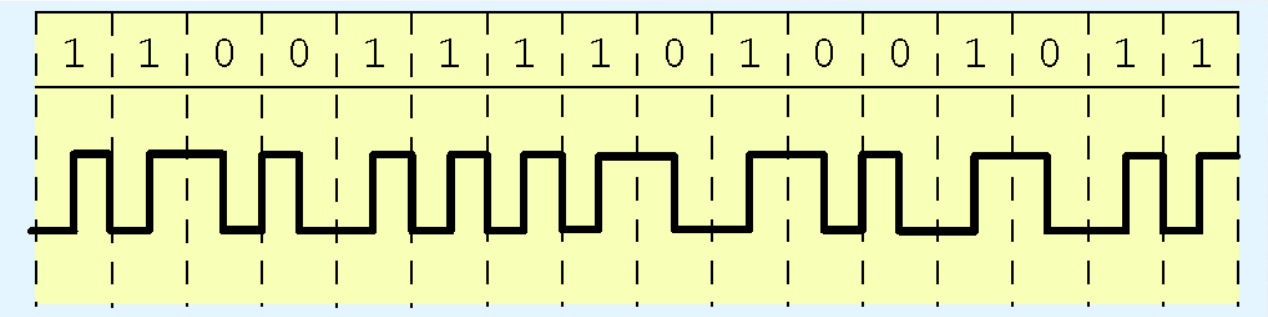
2.A.1 Non-Return-to-Zero Code

- The problem with NRZ code is that long strings of zeros and ones cause synchronization loss.
- Non-return-to-zero-invert (NRZI) reduces this synchronization loss by providing a transition (either low-to-high or high-to-low) for each binary 1.



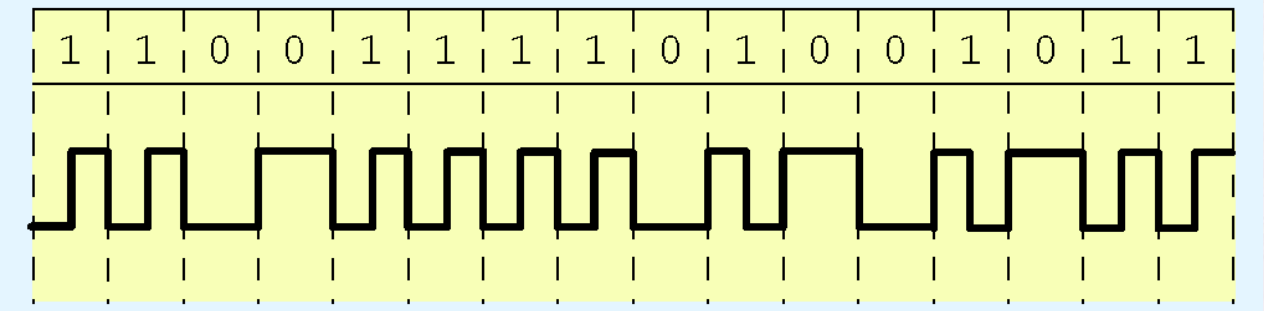
2.A.3 Manchester Code

- Although it prevents loss of synchronization over long strings of binary ones, NRZI coding does nothing to prevent synchronization loss within long strings of zeros.
- Manchester coding (also known as phase modulation) prevents this problem by encoding a binary one with an “up” transition and a binary zero with a “down” transition.



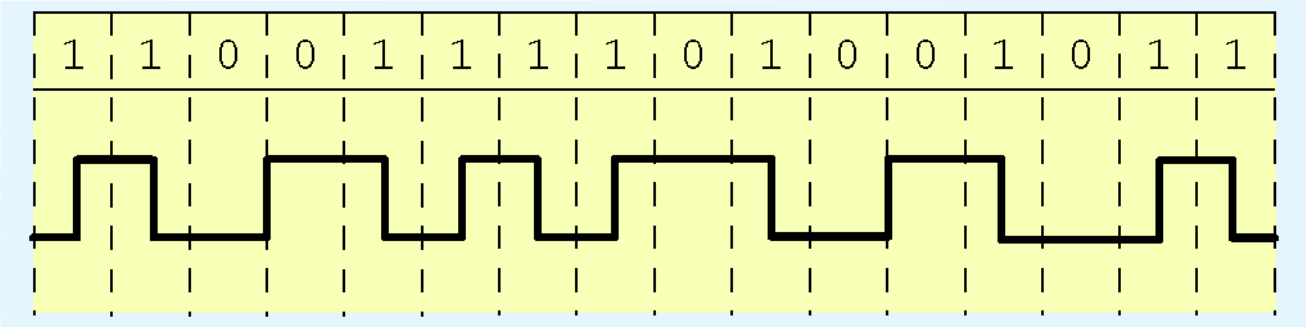
2.A.4 Frequency Modulation

- For many years, Manchester code was the dominant transmission code for local area networks.
- It is, however, wasteful of communications capacity because there is a transition on every bit cell.
- A more efficient coding method is based upon the frequency modulation (FM) code. In FM, a transition is provided at each cell boundary. Cells containing binary ones have a mid-cell transition.



2.A.4 Frequency Modulation

- At first glance, FM is worse than Manchester code, because it requires a transition at each cell boundary.
- If we can eliminate some of these transitions, we would have a more economical code.
- Modified FM does just this. It provides a cell boundary transition only when adjacent cells contain zeros.
- An MFM cell containing a binary one has a transition in the middle as in regular FM.



2.A.5 Run-Length Limited

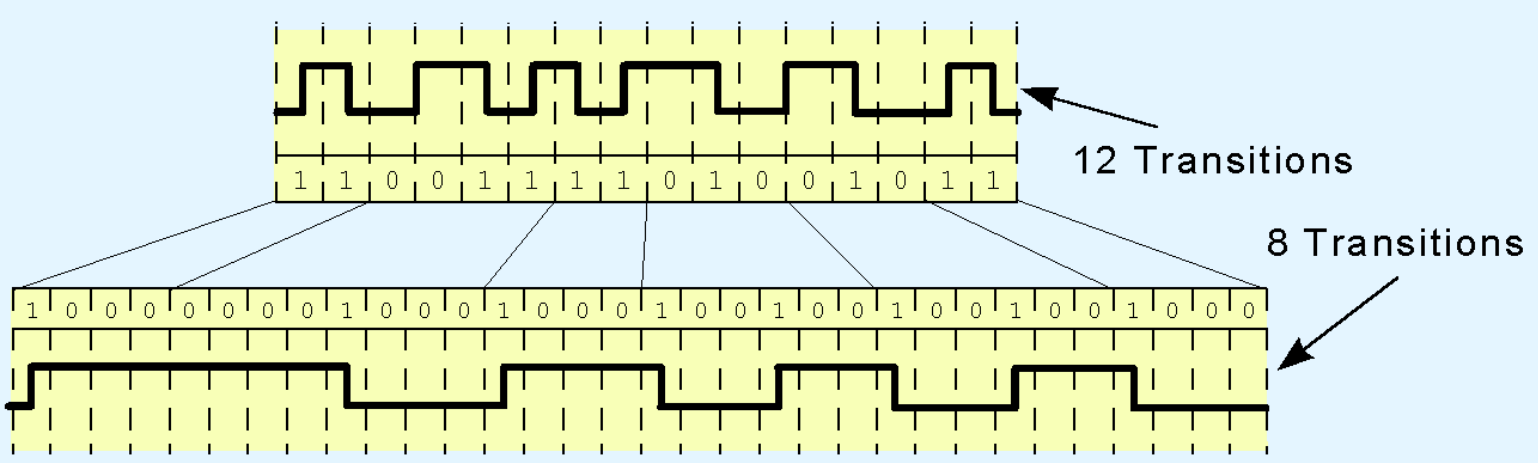
- The main challenge for data recording and transmission is how to retain synchronization without chewing up more resources than necessary.
- Run-length-limited, RLL, is a code specifically designed to reduce the number of consecutive ones and zeros.
  - Some extra bits are inserted into the code.
  - But even with these extra bits RLL is remarkably efficient.

2.A.5 Run-Length Limited

- An RLL(*d,k*) code dictates a minimum of *d* and a maximum of *k* consecutive zeros between any pair of consecutive ones.
  - RLL(2,7) had been the dominant disk storage coding method for many years.
- An RLL(2,7) code contains more bit cells than its corresponding ASCII or EBCDIC character.
- However, the coding method allows bit cells to be smaller, thus closer together, than in MFM or any other code.

2.A.5 Run-Length Limited

- The RLL(2,7) coding for *OK* is shown below, compared to MFM. The RLL code (bottom) contains 25% fewer transitions than the MFM code (top).



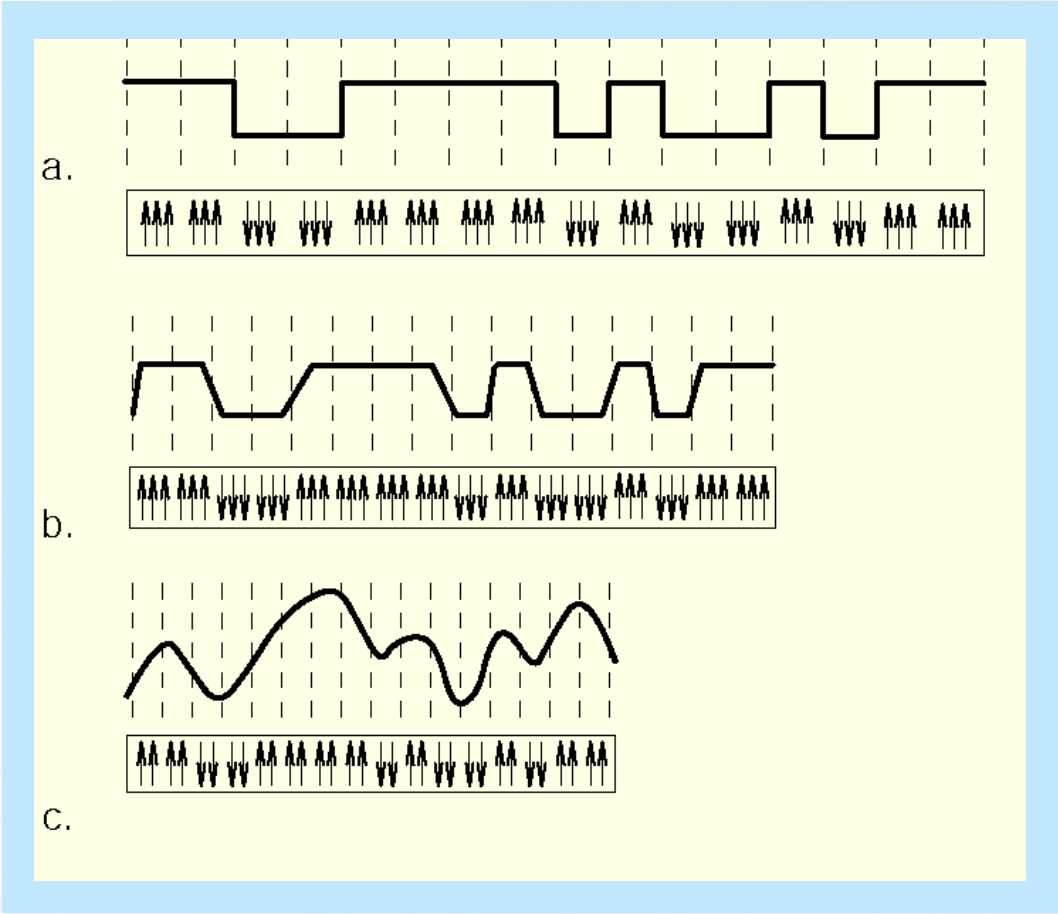
The details as to how this code is derived are given in the text.

2.A.6 Partial Response Maximum Likelihood Coding

- RLL by itself is insufficient for reliable recording on ultra high density media.
- Adjacent bits interfere with each other at very high densities.
  - As fewer magnetic grains are available to each bit cell, the magnetic flux weakens proportionately.
- This phenomenon, called *superpositioning* is shown on the next slide.

2.A.6 Partial Response Maximum Likelihood Coding

- Fortunately, this behavior is well understood and can be used to our advantage.



2.A.6 Partial Response Maximum Likelihood Coding

- The patterns in the previous slide can be made meaningful when each bit cell is sampled several times.
- The sampling determines a “partial response” pattern.
- A Viterbi detector tries to match the partial response with the most likely pattern.
- This technique is stunningly accurate. We describe it in detail in Chapter 3.

Section 2A Conclusion

- Data transmission and storage codes are devised to convey or store bytes reliably and economically.
- A coding scheme that uses fewer magnetic transitions is more efficient than one with more magnetic transitions per character.
- Long strings of zeroes and ones can result in synchronization loss.

Section 2A Conclusion

- RLL(*d,k*) code dictates a minimum of *d* and a maximum of *k* consecutive zeros between any pair of consecutive ones.
- MFM was widely used until PRML and its extensions became widely used.
- PRML requires multiple samplings per bit cell, but permits bit cells to be spaced closer together.
- We return to this subject in Chapter 3.