

Book Review, Eniac in Action

By John Benson

June 2026

1. Introduction

I just finished the title book:

ENIAC in Action

by Thomas Haigh, Mark Priestley and Crispin Rope

©2016, Massachusetts Institute of Technology¹

This took me more than a month to read this, although (or maybe because), early in my career (mid-1970s) I worked with IBM Mainframe-Computers similar in design to ENIAC. ENIAC was conceptualized during WWII, and was operational from 1945 to 1955. However, it served as a bridge-design between the earlier generation of “experimental” large computers and later large commercial mainframe computers like the IBM System/360 Computers (that I worked with in the mid-70s.)²

Also, ENIAC was arguably the first “Supercomputer,” as it was substantially more powerful than any other computer during the earlier parts of its career. It also prompted the development of numerical analysis software in many fields similar to the hydrodynamic-analysis software I used in the mid-70s.

The writing-style of this book does an excellent job of breaking up the dry technical details with many personalities and antidotes. A good example of this was one of the latter when explaining the difference between early leading-edge custom computers (like ENIAC) and later large mainframe computers (like IBM System 360s):

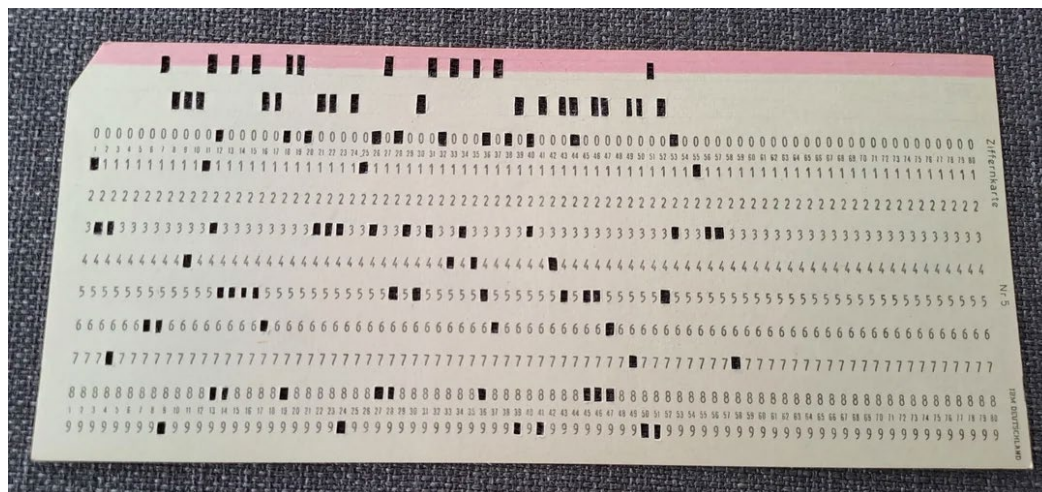
“A computer-industry joke captures the transition: An early computer salesman calls on a potential customer, whose newly established computer team is attracted to the idea of building its own machine. “Well,” says the salesman, “it seems that you face very much the same choice as Noah when he discovered the urgent need for an ark. Do you buy one or build it yourself?” “But sir,” replies a rash member of his audience, “Noah did build his own.” “Yes,” answers the salesman in triumph, “and if you have 60 years and God on your side you should do the same.”

However, although this book is a valuable resource, I believe it would be significantly shorter, and thus more useful, if it didn’t spend so much time delving into the terminology that was evolving at the same time as the early computers. A good example of this terminology was “stored program.” This phrase is not very self-evident, and could mean a number of things, but (per the authors) it was intended to differentiate two types of computer programming/setup. Earlier computers made use of a plug-board to designate which functions were available to a given software-program during a given execution.

¹ <https://www.amazon.com/ENIAC-Action-Remaking-Computer-Computing/dp/0262535173>

² https://en.wikipedia.org/wiki/IBM_System/360

Computers of this era had several types of memory, including punched cards (later “Hollerith Cards,” see image below), punched paper tape, core memory (fixed ferro-magnetic arrays), and later, solid-state memory chips (random access memory (RAM) or read-only memory (ROM)). Collectively these were called “storage,” so “stored program” meant a program or function that resided in some form of “storage”, and was accessed by normal storage-addressing techniques, just like all other variables used by software, rather than one that might be selected through a plug-board. Although all on-line storage of data and programs evolved to use high-speed solid-state storage (initially core memory, but later mainly much faster RAM and ROM). Early “stored programs” might have used slower magnetic media like magnetic hard-disks and (earlier) drums.



Although, this book has a very good set of Notes, a Subject Index, and a Name Index in the End-Matter (at the end of the book), a reader shouldn’t need to refer to these constantly to understand what he/she is reading. Since I spent my early career as a software programmer, I was able to comfortably read this book. Also, even if the only computer you deal with is the one on your desk or in your pocket, it could be that you have absorbed enough computer terminology as a user to be at approximately this level, and you should be OK.

Also, this book will be valuable to me as a reference, so it has definitely earned a place-of-honor on my very small desk-bookshelf, which is where it lives now.