(13)

2-75

THE TECHNOLOGY OF COMPUTING AND TRENDS TOWARD SMALLER, DE-CENTRALIZATION

G. Bell
May 8, 1975
(Presented at UC/Irvine--2nd
Decade Conference)

The Computing Supply/Demand Process

The computing industry can be characterized, quite simplistically as a multi-stage pipelined process, shown in Figure 1, of about 5 stages. The first is a research/technology stage based on phenomena (e.g. materials and formal grammar). This proceeds to an advanced development stage, where in the case of semiconductors, actual prototype devices are developed. As far as the computing industry is concerned, there are 3 stages which should operate in quasi-parallel: architecture, machine hardware implementation, and system software implementation. When these do not operate in parallel, development times increase, and we run the risk of having little feedback, and hence poor design.

It is these stages of development that mainly determine the environment for computing. These then are fed to languages development and on to application's software development. The final stage is the user. If each stage requires about two years, then the time through the whole process is about 8 years, and the user obtains the final output in a much delayed fashion. In some sense, the development time appears to take this long; but in other stages, there can be a great deal of overlap, and development lead times can be reduced by better feedback. The whole feedback process among the stages is nebulous, but in computer organizations, the official channel is denoted as Sales/Marketing. For computing, since the developers are also users for the development process, the feedback process is more closely coupled and direct than in other industries.

Semiconductor density and magnetic storage density increases are the root of most improvements in computing, although improved algorithms also substantially contribute. For example, the fast fourier transform reduces the time from n squared to n log2 time. For 1000 point transforms, this reduces the time by a factor of 100. Assuming yearly performance increases of 40% for hardware technology, it would take 14 years to make the same improvement in a brute force fashion.

Characterizing Measurements of Technology

There are several ways to characterize technological improvement. Improvement is based on the notion of learning, and hence efficiency. The efficiency, En, of the nth unit is characterized as:

EN = Kn^d

where K and d are learning constants.

Generally, technological improvement T(t) has been characterized as exponential growth, with time, of the form:

T(t) = Ke^ct

Fusfeld (1973) shows that technology is best characterized as:

$$T_i = ai^b$$

where:

T_i = technology of the ith unit,
i = ith unit,

a = first unit,
b = technology learning constant.

In the case of computers, b = 2.51, for Ti = memory-size x memory-access-rate. This improvement is significantly higher than any other technology (e.g., contrast with .74 for automobile horsepower). Therefore, for about 100,000 units (the case in 1974), the performance increase has been over 13 orders of magnitude.

Fusfeld (1973) also shows that, if the number of units increase exponentially, the last two forms are identical. Subsequent technology will be measured in terms of the yearly improvement.

Learning and Demand Curves

The previous measures correspond to the notion that cost declines as the number of units produced increase along traditional learning curves. Learning, in nearly all industries, (as measured by cost decline) has been observed to vary between 10% - 20% each time the cummulative number of the units produced doubles.

Finally, a second factor, user demand, affects the units and prices produced along traditional demand curves. For computers, the demand appears completely elastic; the yearly rate of machines supplied doubles each time the price is reduced by about 25%.

Technology of Specific Computer Components and Computers

The technological growth of a computer must be broken into its constituent parts. Therefore, various sized machines will evolve differently depending on the constituent parts. If we look at various parts, we find that some evolve very rapidly and some very slowly. 1965 Armour, (Turn 1974) projected a decline of 80% per year for CPU storage cost per mega word accessed, over a 15 year time scale; whereas his projection for the typewriter cost per mega character printed was roughly constant.

For disks, the price of a given physical structure, e.g. a disk pack with 10 platters, actually increases slightly with time; but the magnetic storage density drastically increases. This is to be expected, since a complete disk mechanism is built up of components which have been "learned"--i.e. sheet metal, motors, power supply, etc. IBM's disk packs with 10 platters (beginning with the 1311) have increased in capacity at the rate of 42% per year, and the price per bit has decreased at 38% per year.

In Table 1, different styles of disks are presented (very roughly) in terms of 1975 prices. There are 4 styles of disk packs beginning with the flexible disk and going to the 10 platter disk pack. Here we can see an economy of scale, i.e., the number of bits for larger units increases more rapidly than cost giving a decreased cost per bit. The overhead of motors and packaging increases less rapidly with size. Also, more effort is placed to increasing the densities at the large sizes. This technology is used on the smaller disk pack after a two year delay.

Conservatively, semiconductor technology has improved at the rate of 60 to 80% per year, as measured by cost and/or cost-performance. In fact, semiconductor memory densities have doubled each year, beginning in 1962 with a single bit.

The various technology improvement rates are given in Table 2. We can look at these characteristics in terms of the net effect on minicomputer systems (Figure 2). Minis have declined in price at about 31% per year. Beginning in 1972, the 8 bit micro-computer was introduced, and the name is now synonomous with an 8-bit computer and a processor-on-a-chip (1 single silicon semiconductor chip). These machines are parallel to the 12 bit and 16 bit lines on Figure 2. Figure 3 shows minis in a price performance space. On this chart, lines of constant cost-performance are given in terms of 4 year intervals, where we assume that in 4 years the price performance should increase by a factor of 4--i.e., 41%/year. So with the minicomputer, as we have previously noted, the price-performance objective is to keep performance constant (or increasing) and to have decreases in price.

We can contrast the minicomputer with the large machine, (Turn, 1974) and observe that technology improvement goes to increase performance. However, in the case of several recent large machines, the machines are yet to be operational within 5 years of their scheduled time, therefore, it is hard to characterize them as actual points on technology curves. Indeed, they may never exist as entities as specified.

The small machine, a PDP 8, and the large CDC 6600 can be compared as of 1967, (Table 3). A possible large machine can be compared with a small, board-only, processor-on-a-chip computer costing $1000 for 1975 technology (also Table 3).

Actually, it is perhaps more important to put machine cost into perspective as to other costs of use. (See Table 4.) Note, that the dominate costs are the human, user costs.

Small_Decentralized_Versus_Large_Central_Systems

Figure 4 shows what is likely to be the future structure of computing. Because of the decline of machine prices, we would expect substantially more standalone computers (e.g. the DEC CLASSIC system), which are capable of running any language; in essence a generalization over the HP and WANG BASIC-only language machines. There is, in fact, a strong affinity to have these machines totally standalone and not require external services for their support. We would, however, expect all of these machines to require some support for sharing certain common facilities (e.g. archival storage, classroom programs, printing, plotting) through a switching structure (including closed circuit video) to other systems. At the other extreme, we would expect large general purpose systems to prevail, which are predicated on an economy of scale. These would be capable of exploring varied application spaces where generality is needed. Thus, we would see at the two ends of the cost spectrum: a continued use of larger, general purpose computers; and totally dedicated low cost systems would exist that have been tuned to specific applications.

Table 5 contrasts the two designs: small decentralized systems versus the large general centralized system.

It can be conjectured what eventual systems like this would sell for and how they operate. The standalone systems, e.g. the DEC CLASSIC, and the specialized BASIC-only language terminals, now sell for about $8000 or $200 per month. This price is only a factor of 3 to 8 more expensive than a dumb terminal capable of only accessing a more complex large central system. The total economies are a function of the expected use and the amount of support required in the two systems. As Selwyn, at MIT, has observed, there is a very large economy of scale associated with centralization. The total cost to operate a computer is proportional to the monthly rental raised to the .8 power. But in the very small totally decentralized case, we would not expect this to hold. However, there are clearly hidden costs associated with individual computers. The exact mix is not clear, but both will exist.

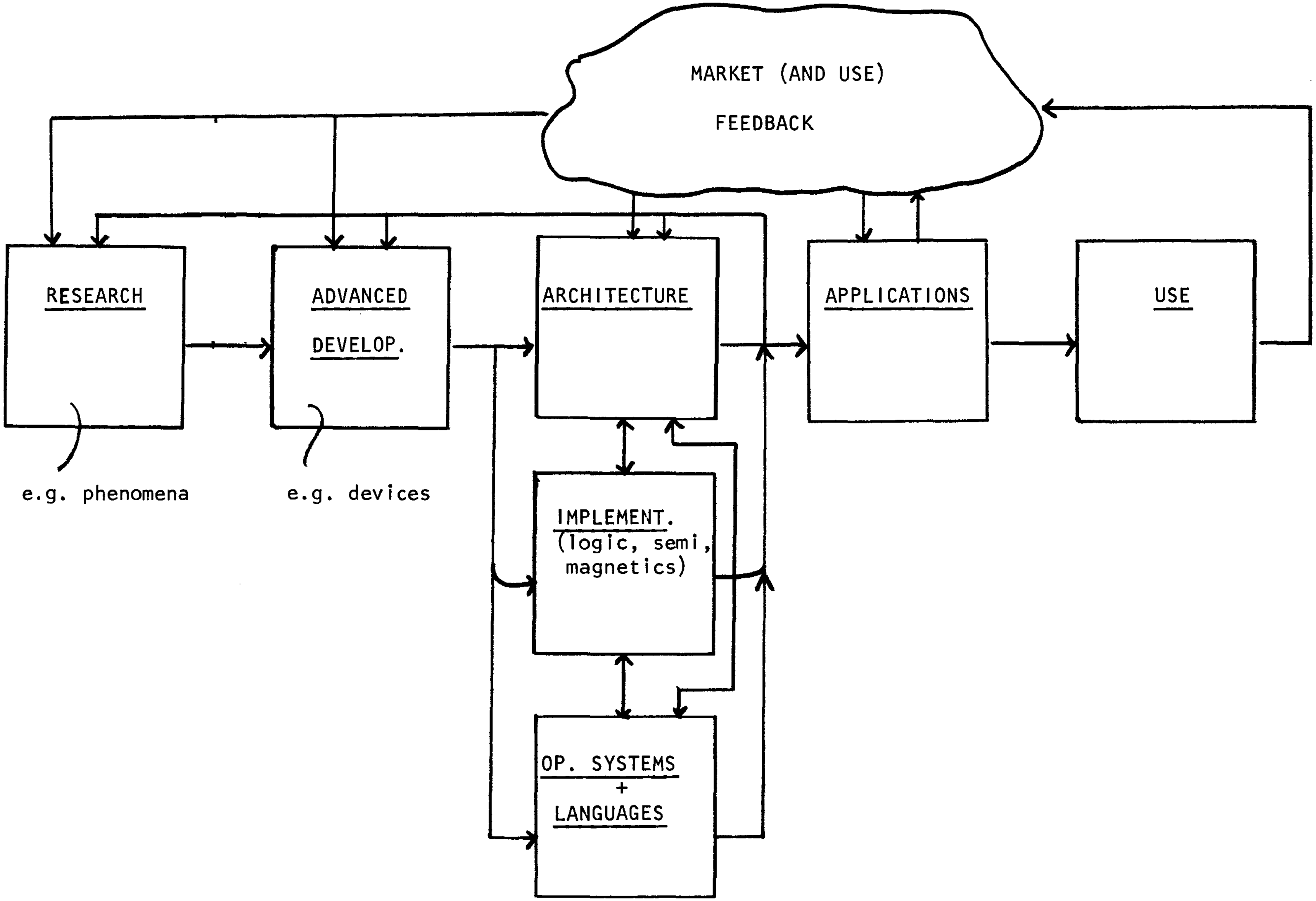
The new exciting areas of application for computers is clearly in the small systems and the coupling with other machines.

Bibliography

Fusfeld, Alan R., "The Technological Progress Function", Technology Review, Feb. 1973, pp 29-38.

Turn, Rein, "Computers in the 1980's", Columbia University Press, 1974.

Fig. 1 COMPUTER INDUSTRY/USE PROCESS



GB
5/7/75

G.Bell

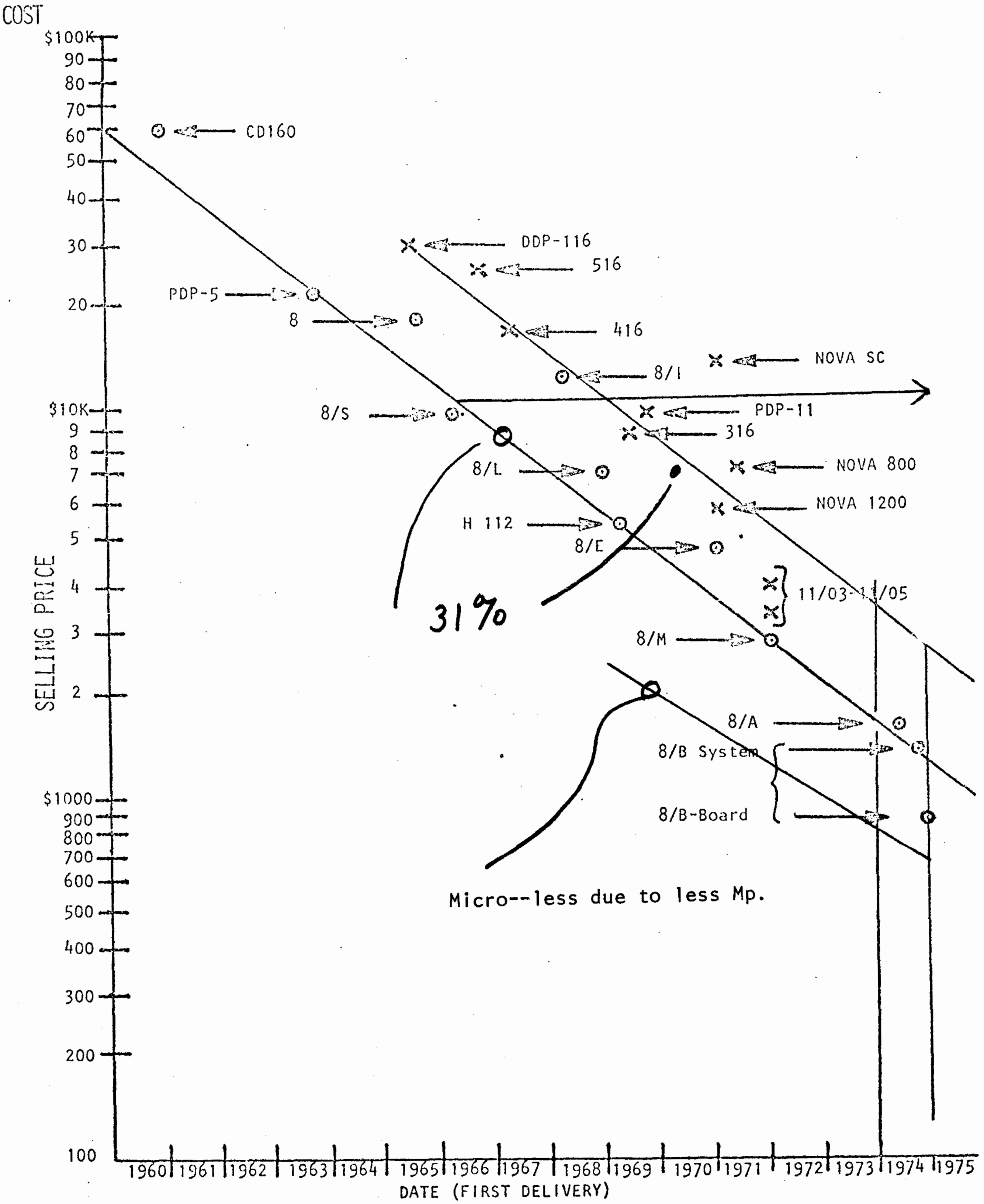


FIGURE 2 SELLING PRICE VERSUS FIRST DELIVERY DATE FOR MINICOMPUTERS

FIGURE 3 MINICOMPUTER PRICE VS PERFORMANCE FOR VARIOUS TECHNOLOGIES. LINES OF CONSTANT COST/PERFORMANCE ($/ACCESS/SEC) ARE PLOTTED FOR EACH FOUR YEAR'S (FACTOR OF $4 = 1.41^4$) ASSUMING IMPROVEMENT OF 41% PER YEAR

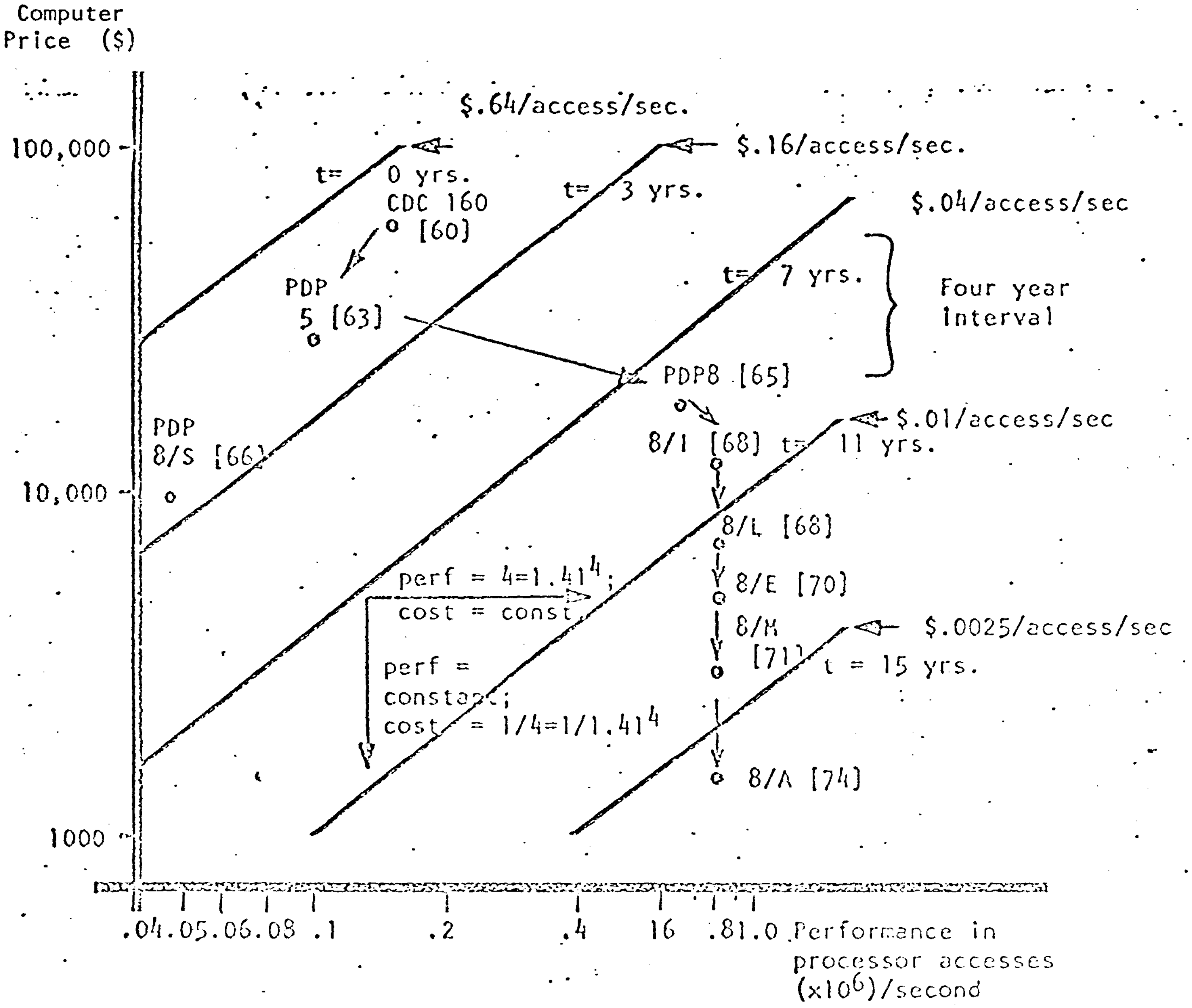
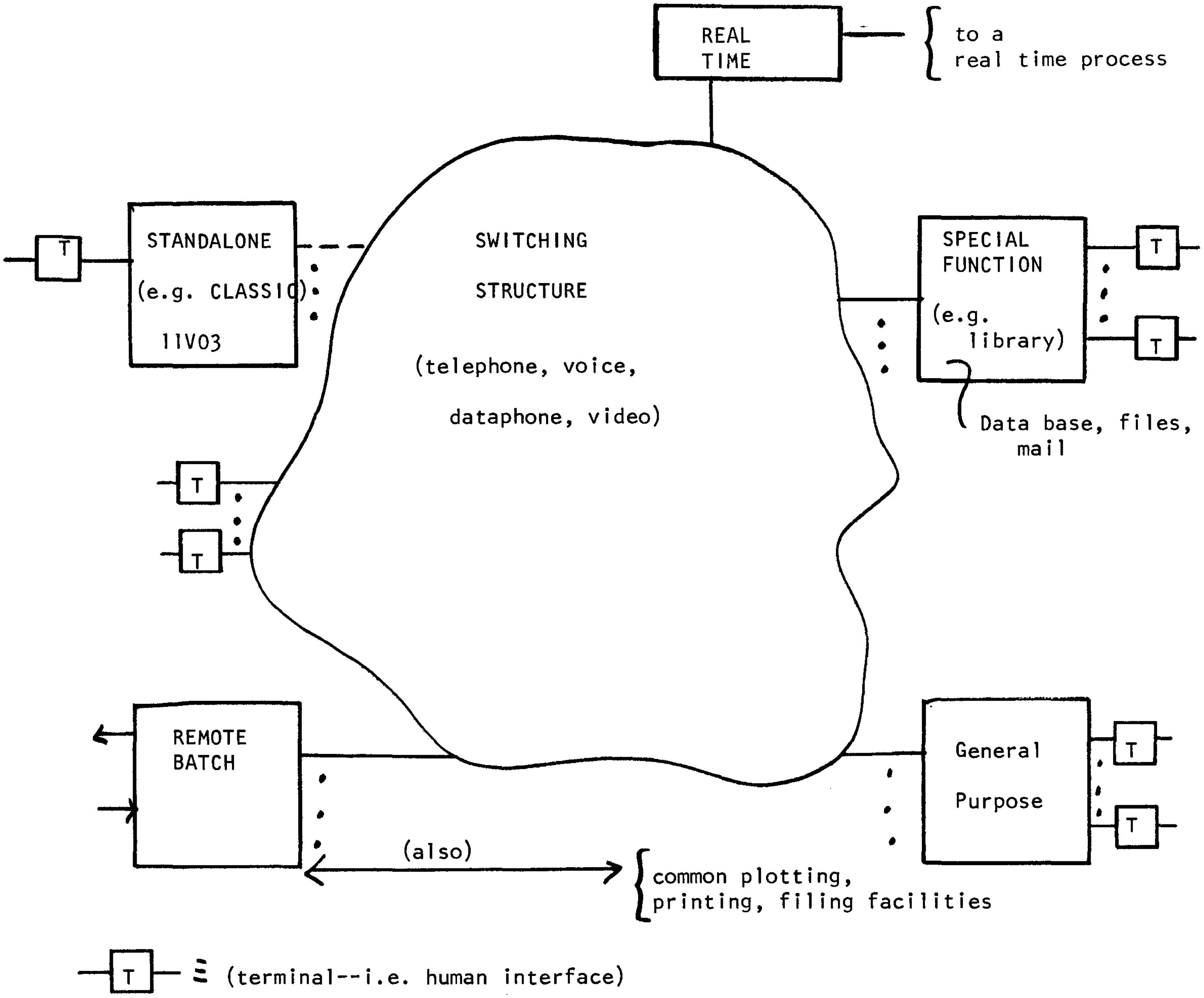


FIGURE 4 GENERAL STRUCTURE OF COMPUTATION FACILITIES



GB
5/8/75

TABLE 1 DISK TECHNOLOGY (1975)

DISK CONFIGURATION	PRICE (Roughly)	CAPACITY (Bits)	COST/ BIT	ACCESS TIME
Flexible (floppy)	3	2.5	1.2	1 s
1 Platter	6	30.0	.2	50 ms
3 - 5 Platters	12	160.0	.075	50 ms
10 Platters	24	800.0	.03	20 ms

GB
5/8/75

TABLE 2 YEARLY IMPROVEMENTS FOR VARIOUS COMPUTER TECHNOLOGY COMPONENTS

Technology Indicator	% Improvement
Semiconductor density	60 - 80
Semiconductor memory density (bits/chip--leading edge)	100 (1962 - 1974)
Magnetic recording density (disks)	41 (1962 - 1975)
Core (price)	30
Terminals	25
Magnetic tape (density)	23 (1952 - 1973)
(data-rate)	29
Power Supply (cost/watt)	-3
Packaging (cost/in↑3)	-3
Minicomputers	31 (1960-1975)

GB
5/8/75

TABLE 3 PARAMETERS OF LARGE AND SMALL COMPUTERS 1967 AND 1975

ACTUAL AND (RELATIVE)

1967	COST (K$)	WL	Mp.Size	$/bitx100	MIPs	MIPS X WL	REAL (MIPS)	PERF/COST (MIPS/MS)
8	10(1)	12 (1)	.05Mb(1)	20	.3 (1)	(1)	.002 (1)	30
6600	3K(300)	60 (5)	8Mb(160)	38	3(10)	(50)	3 (1500)	1
1975								
LSI-11	1(1)	16 (1)	4Kw(1)	1.6	.3 (1)	(1)	.04 (1)	300
Large	10K(10^4)	64 (4)	1Mw(10^3)	16	100 (10K)	(40K)	100(2500)	10

TABLE 4 OPERATING COSTS FOR COMPUTING

	(1K) COST/Year	COST/HR. @ 2400 HR.
Human	0, 5, 10, 20, 40	0, 2, 4, 8, 16
Computer	1.2 ~ 2.5	.5 ~ 1.
Terminal	.25 ~ .75	.1 ~ .4
Service	.05	.02
Power	.005 ~ .01	.002 ~ .004
Line	0 ~ 2.4	0 ~ 2.
Paper	0 ~ .1	1/3¢ ~ 3¢
Space	.05 ~ .1	.02 ~ .04

GB
5/8/75

TABLE 5 CHARACTERISTIC DIFFERENCES OF SMALL (decentralized) VERSUS LARGE (central) FOR COMPUTATION

Attribute	Small (Decentral)	Large (Central)
Performance	Greater average	Greater peak Large memory (programs)
Cost	Economies through production f(COMM line, terminal, memory costs, and utilization)	Economy of scale for disk Cheaper when user utilization is low
	Production limited	Design limited
	Overhead of maintenance on individual (hidden)	Explicit, central maintenance costs
Use	Small (or 0) data bases	Large, shared data base; and/or general (undefined) large computational tasks
	Fixed, (well-defined) computation; or small computational (e.g., text publishing, program preparation, CAI, statistical)	
Security	Private	Public (but easy to share programs, data, etc.)
Reliability	Distributed	Central + Comm. System