PARADYNE CORPORATION 1969-1976

To pursue the DOD angle the company headquarter was located in Washington DC while the engineering facility was located in Clearwater Fl, where the fifth member of our team, Bob Acker, a depart manager under Joe at CDC, knew a number of modem engineers from his days as a manager at Honeywell. Joe and I moved to Washington, occupying a 2500 sq ft facility on Shady Grove Road in Gaithersburg that Joe had signed a three-year lease on. Jim stayed with Control Data in San Diego until such time as Paradyne had the need for a marketing and sales guy.

In short order we started the following activities:

Joe proceeded to raise money since we were operating only on founders money to start. One part of the engineering team started to investigate voice-band modem techniques. Another part started to look into solid state point-of-sale terminals. A third part of the engineering team kicked off a plated wire memory project. As Director of Product Management I had the assignment to do market research and try come up with some product definitions. Glenn Preston for his part decided to drop out of the picture, partly because he had made enough money on selling his company to Control Data and subsequent stock investments, and partly because R&D money from DOD was drying up due to the Vietnam War. At one of our first meetings of the management team Wylie axed the point-of-sale terminal project on the grounds that we were spreading ourselves too thin. Shortly thereafter, Honeywell inadvertently saved the company by threatening to sue us over our plated wire memory activities, forcing us to kill that project. We now had all the resources of the company focused on the remaining products, communications multiplexers and modems. After some time it became clear that we really could handle only the modem development so we dropped the multiplexer idea.

My job was to do market research to support our product development as well as to generate all collateral material for our sales effort. As part of the marketing research I studied the available product literature and gained fairly quickly a working familiarity with the industry: who the players were, and what products they offered. I also visited a number of potential customers to learn their plans and the requirements that they saw in terms of modems. Among the companies I visited were US Timeshare, Tymnet, MCI (which then consisted of a few engineers and lawyers in a small office on K street in Washington DC) Comnet, a few federal government types, such as Pentagon and the state department. I also spent some time doing detailed product specifications for our development group.

One bad mistake that we made, which I have to take main responsibility for was the MARQ-48 modem. Tom Saliga, one of our key engineers and the most creative guy in the group had the idea of combining a very efficient error control system with the modem so that we could offer guaranteed data integrity, which other modems could not. It was implemented by means of a simultaneous reverse channel for sending ack/nak messages and implemented the so-called go-back-two continuous transmission error control protocol. If line hits occurred that caused bit errors, the modem would simply retransmit the block and the only effect that the computers and/or terminals would notice was a momentary slowdown in data transfer (which could be enforced by controlling the clock speed, which was feasible according to my research). The problem with this scheme, which embarrassingly enough was pointed out by a consultant that some VCs hired to do the technical due diligence on us, was that IBM computers and terminals (which made up 80 % of our potential market) could not use such a modem, since there was no way to turn off the inefficient stop-an-wait protocol built into all their hardware and software. This made or modem not only no better than the competition, but because we used automatic equalization with about 6 second training time, it made it unusable in IBM systems. OOOOOPS. Megabad.!!!

To recover from this huge booboo I came up with the idea of the Bisync-48. The concept was what later became known as spoofing. This is one of my two inventions which became relatively known and copied in the industry, and which we did not seek patent protection for. Stupid! It works such that the modem communicates with the computer using the BISYNC protocol, acking every block it receives, thus allowing the computer to immediately send the next block. The modem with its built-in error control handles any retransmissions without line turnarounds by using its reverse channel to handle ack/naks. In case there is a retransmission it stops the clock to the computer to prevent overruns. By eliminating line turnarounds and running at 4800 bps instead of 2000 bps like the 201b did we were able to nearly quadruple the throughput.

We tried to market the basic modem itself as the MARQ-48, an error controlled modem for links that did not use the Bisync protocol, but this did not fly. Instead, I had another idea on how to take advantage of the error-free property of the basic design, which was what we sometimes called the 1000-mile I/O cable. The idea was to add a parallel interface to the modem which was compatible with the IBM Multiplexor channel, i.e. one version looked like a peripheral such as a printer controller which could connect to the channel and the other looked like the channel and could connect to the peripherals. We called this product the PIX-48 and announced to the world. It generated a fair amount of interest and got us some much appreciated publicity. (see story by Ralph Berglund).

One of the flaws of the spoofing design is, of course, that when the modem acks a transmitted block back to the DTE it asserts something which it does not know to be true, i.e. that everything is OK and the sender can go on to the next block. There may be other problems than line errors, which the modem can’t do anything about and which the computer needs to know about, (things like printer out-of-paper, operator interrupt, etc) and these are signaled from the terminal to the computer by means of NAKs or WABTs, or other special characters. When we barged ahead with the Bisync-48 development we did not study the Bisync protocol enough to know about these things and consequently were caught by surprise when they came up to bite us when we started to install the modem in live installations. To troubleshoot these problems, as well as to study the actual behavior of Bisync links in real life, we needed some way to capture traffic across the link. Since we could not find any instrument on the market that could perform this function we had to design our own which Luke Weathers proceeded to do, in addition to continuing the development and modifications of the Bisync modem itself. We found this device so useful that we decided to turn it into a product and put it on the market as the Bisync Analyzer. We also applied for and received a patent for it, on which I was listed as one of the inventors, my first one.

As it became more and more clear that the Bisync modem had too many limitations, both technical and marketing, we bowed to market forces and stripped all the error control doodads from the MARQ-48 and marketed it as the M-48, a plain vanilla leased line 4800 bps modem. We also put resources into the PIX development to try to bring it to market as soon as possible in order to get our revenues stream up and running, because at this stage we were still burning cash at an unsustainable rate.

My role in these two developments had been to come up with the product concepts and then to produce the collateral materials for them, such as brochures, ads, white papers and such. In this activity I worked with our advertising consultant, a woman by the name of Karen Syence, who had a small agency in Washington DC. Besides producing our materials and creating our ads she handled our media relations and space purchasing as well as our convention and show activities. As part of this activity she got us invitations to publish papers in various magazines and give talks and presentations at the different shows. I was given these tasks which I thoroughly enjoyed, although I did have a very tough time with stage fright, which seriously impaired the quality of my presentation. I would get so tense that my throat would constrict and render my voice thin and quavering. If I was lucky I would be able to work up enough excitement about getting my technical points and explanations across that I would forget my nervousness and the tension would release so that from that point it became a more or less normal delivery. Writing the talks and the papers, however, was totally enjoyable; I was especially challenged by explaining technical issues in the most intuitively understandable and readable way while still retaining technical accuracy. Whenever I was able to do this to my satisfaction it gave me great psychic income. Unfortunately, by the time the editors got through with improving my luminous prose the income level was greatly reduced. I have always had a problem with this intense pride of authorship which does not kindly suffer improvement suggestions. One of the papers I wrote for Datamation was an exposition of the WATS telephone billing system in use in the US in the early seventies. I not only wrote it, I also created the color separations for a complicated multi-color chart that displayed the WATS rates in a way similar to a mileage chart found on road maps such that you could look up the WATS rate from any state in the US to any other area. I also devised a slide rule type calculator to do the same thing in a more compact format than a magazine. Karen had this produced and we used it as a handout at shows and customer calls.

In 1972 my right hip which had started bothering me in my teens got so bad that I decided to do something about it. I had read about artificial hip joints and how they seemed to work well for a lot of people. I read an article in the St Petersburg Times about and Orthopedic surgeon in St Pete who was considered the number one man in the field. I tried to make an appointment and was told that he was booked up for years ahead, and they recommended that I contact a Dr Sarmiento in Miami, who was also highly regarded. He turned out to be an outstanding man in his field as well as a great and personable human being whom I treasure as friend. He operated on my hip in August 1972 and installed a Charnley total hip which has worked perfectly ever since.

In 1974 I divorced Anne and Married Anne Berit whom I had met in Washington a couple of years earlier.

As sales were slow to develop and the expense nut kept turning it became necessary to again approach the VCs with a disappointing tale of objectives unmet, and additional needs for funding. With the terms available now drastically impaired, with even the existence of the company possibly in question, the pressure on Joe and Jim grew so severe that their relationship cracked. Joe determined that the problem was due to Jim’s failure of marketing and tried to replace him as VP of Marketing. Jim responded by going to the investors with the request that they fire Joe.

The upshot was that they fired them both (actually, Jim was given the opportunity to stay on as a salesman, which to no-one’s surprise, he declined). The investors then started a search for a new CEO, during which time the company was run by a VC partner by the name of Jim Morgan. (Jim later went on to become CEO of Applied Materials which he has run successfully ever since, and made it into one of the key enabling companies in the IC business during the past quarter century. They design and manufacture some of the precision machinery, such as polishing, dielectric and conductor deposition, and etching systems, as well as wafer and mask processing systems, which make possible the fabrication of devices with millions of transistors per square centimeter.) My position during this conflict was that I took Jim’s side, but I was still on good terms with Joe. He explained to me why he did what he did, and I explained why I thought Jim was still the best guy to head marketing. As a result of this semi-neutral position I became the guy the investors came to learn the unbiased version. When Morgan came on board I worked directly for him in my function as product manager, as we tried to keep the company going during the search period. The search took a long time, as several candidates had a closer look at us and decided that the risk reward relationship at Paradyne was not to their liking. Finally, a guy named Robert Wiggins took the job. He had started his career as a computer salesman for IBM, had founded a timesharing company called Comnet, which had gone belly up, and wound up as GM of a communications subsidiary of Sylvania, before he came to Paradyne. He struck me as cold fish that I had little inclination or ability to warm up to. Shortly after he came onboard he informed me that he was bringing in one of his old cohorts to be my boss as the new Director Of Product Management. My response to this was that I wanted to transfer to Engineering, which he agreed to and I became Manager of Systems Engineering, with zero people reporting to me. My plan was to get up to speed on modem engineering in order to broaden my marketability outside of Paradyne. The first task I took on was to investigate whether our then current modem could be modified for fast train. My thought was that if I could just save the equalizer settings, we might just be lucky enough to recapture the signal with small enough timing offset that the receiver could reacquire timing quickly and be ready for the next block. It took only a few tests to determine that this was a foolish hope and the whole idea is an embarrassing reminder of how naïve I was about modem engineering at that time. (I have never understood why people like Harvey Harris, Tom Saliga, and Dale Walsh let me conduct this experiment, seeing how ridiculous it was.) However, during this time I was reading up on the literature and came across a paper by a guy named Shiv Verma from Bell Labs which described a 201A modem they had implemented in software running on a special microprocessor. I also read some memos written by Tom Armstrong, a PhD in communications engineering, recently hired from Honeywell, which laid out the key elements of a QAM modem in terms that were accessible to me, mostly trig functions and identities.

At this time there was a CCITT standards project underway to come up with a standard for 4800 bps modems, and the leading candidates was some type of QAM structure. Paradyne s modems used a so- called VSB structure which was considered somewhat old fashioned, and which suffered from excessive sensitivity to phase jitter, which was then considered a key line impairment. As the product management guy at Paradyne I had the job of representing the company in standards meetings, and it was now my objective to delay or derail the 4800 bps standard long enough for Paradyne to catch up with the competition in QAM modem development. This was not a task that I was cut out for and my feeble attempts at delaying tactics consisted of supporting Western Union which objected to the standard on the grounds that it would allow people to send data across the international borders using the phone, which violated the rule that Western Union had the monopoly of handling data transmission internationally, and ATT had the monopoly on voice. Obviously, this approach had little chance of succeeding, although it is interesting to note that as late as

1975 we were still comfortable with the idea of the communications business being divided up into government sanctioned monopolies.

Based on my cursory research in the modem literature I concluded that what we should do was to design a special purpose microprocessor and implement a QAM modem as a program running on it. To investigate the processing requirements on such a processor I undertook to write a modem program in Basic which I ran on a time sharing service accessed via 30 cps teletype. From this exercise I estimated the basic parameters of the processor we would need and set out to find a suitable source. Together with Dale Walsh I visited Texas Instruments, and maybe one other potential vendor to hear what they had available, or in the works. TI had obviously focused on this application more than the others, and had developed a chip that was intended for digital signal processing. However, the architecture was somewhat inelegant and did not appeal to me. Since their production schedule was also somewhat iffy, I was able to head off Dale’s tendency to favor TI.

The solution I was enamored with was the AMD 2911 bit slice, still one of the most sensible CPU architectures that I have seen or heard about. (Later in my career I got to know the architect of this chip, a man named Cecil Kaplinsky, whom I met at Cirrus logic, where he was the architect and chief instigator of the Cradle project.) One way or another I was given my head and allowed to proceed with the project on my own. I was to design the processor and then program it to implement the modem algorithms worked out by Gordon Bremer. This was in late 1975 or early 1976 when I had been away from any engineering work since 1965. Then I had done digital design using a line of standard circuit boards at Beckman Systems, now the state of the art was to use standard SSI and MSI chips, such as the TI 7400 series. With more confidence than sense I proceeded to design my processor using the most suitable chips I could find in the TI data book. At the same time the company also launched a project headed by Dale Walsh to do the same thing but using a hardwired logic approach. His team included Dave Springer and Gordon Bremer, while I was alone. However, I did get Dave s help in constructing and checking out the bread board and the development tool and I used Dale’s multiplier design, basically a ROM with the multiplication stored in it. Both projects used the algorithms developed by Gordon. The processor design included of course the instruction set, which I structured to simplify the programming of the algorithms involved. I also defined a type of assembly language which I used to write the programs which I then assembled into hex machine code by hand. The development tool consisted of a panel with bit switches corresponding to the bit positions in the instruction word for single word by word input. There was also a card reader interface which allowed the program to be read in batch wise at 300 CPS via the card reader. There was one instruction per card and the cards were loaded in sequence into successive locations in memory, so that the program could be changed simply by changing the order of the cards. Dropping the deck on the floor was of course disastrous. To facilitate manual assembly I designed a coding sheet format which I had printed up in quantity. The format was something like the following:

```
ADDR  SOURCE CODE     NIA  OP SCR-A  SRC-B     Dest1  Dest
245    01 01  5     6           x add y -> c, mem2; inc        5        44
```

One of the downsides with manual assembly on paper is that there is no convenient way to insert additional instructions into a routine that needs fixing. To alleviate this problem I divided the program up into fairly short paragraphs with ample space between them. As debugging revealed changes and additions needed, the allocated space often got used up and I had to rewrite the whole routine on blank sheets, creating new paragraphs with new spaces. This process of copying code from one sheet to another turned out to have an important side benefit: it was a very effective way for me to desk check the code. The process of reading each line of code carefully in order to copy it correctly on the next page forced me to guarantee that a certain minimum of attention was given to each line. This, combined with the fact that my mind was focused only on a simple repetitive task, set up an almost unconscious process in the background whereby my mind was also following the logical meaning of the instructions, and if something didn't make sense it would be noticed. I found many incipient bugs this way. To my surprise I have never seen or heard of this technique being recommended for desk checking or code walkthroughs. I have formulated this observation into a statement that I will call "Karl's law of attentional misdirection": the best way to check a program or other sequence of text lines is to go over it line by line for some other purpose that requires absolute attention to every line. This works better than a process where the attention is directed at finding errors because, at least for me, it is impossible to guarantee that each line gets the attention it should have. I just don t have the discipline to inspect every line for error in a meaningful way, a certain percentage get brushed over cursorily without making an impact on my thought processes. Only if I have some other un-ignorable reason for looking at every line can I guarantee that it will happen.

After a few days, maybe a couple of weeks, of getting familiar with the system and the development tool, and starting to observe the effects of program execution I started to become absorbed in the task and consequently more and more productive. After a while the program became the central focus of my waking (sometimes even sleeping) attention and it became almost a craving to get back to the system morning, noon, and night. I went back to the lab most days after dinner and was able to continue coding and debugging late into the night without losing my focus and sharpness. The average week I probably spent upwards of 70 hours on the system, writing and debugging code. As time went on it became clear that my project was the winner with the result that the deadline pressure only increased, since now the company had chosen to bet its future on the timely completion of my project.

Toward the end, my manager, Verney Brown, came calling more and more often to learn the status and to re-emphasize the deadline. My response was generally pessimistic, since I felt overwhelmed by the amount of work left to do in the time available. During the time that I was writing the program Dave Springer took care of getting the PCB laid out and the first production prototypes built. When they were ready and checked out by Dave it was my turn to load the program in them and try to make it run. This was finger-crossing, proof-of-the-pudding, smoke-testing, make-or-break time. The program was loaded into the modem by burning it into two 512X8 bipolar PROMs and once they were plugged in, the modem had to run. If it did not there was very limited visibility into what was happening with the program, and debugging would be a time-consuming, tense nightmare that would have nothing pleasant about it. It took all my courage to bite the bullet to set up the PROM programmer, download my code into it, burn the PROM, plug them into their sockets and turn on the power. Fortunately for my nerves the process went off without any hitches that I can recall and the modem ran. With a few fixes and fine-tunings the modem was put into production and marketed as the MP-48, one of the first V27ter modems on the market and THE first implemented as a program running on a general purpose processor, which in theory at least could also be used to run the company's payroll program or whatever. From having been an also-ran in the business and way behind the eight-ball on QAM technology and the V27ter standard, Paradyne was among the first on the market, if not the first, and did so with the most up-to-date architecture in the business. For this achievement I claim that my ideas, my initiative, and my extraordinary effort deserve most of the credit. We filed for a patent on the design with me and Dale Walsh as coinventors.

Toward the end of the project marketing was becoming more and more insistent on the need for a network management system in order to support large networks, and large networks was what we needed to sell to get the volumes up. I volunteered for and was given the task of also developing such a system, with the help of one programmer and one technician, which was the project team I thought would be sufficient. How I could so ridiculously underestimate the size of the task is somewhat understandable, given my lack of management experience, but how senior managers such as Wiggins and Brown could go along is harder to explain. Maybe they truly believed that I had a handle on the project and supported it/me because it looked like a cheap solution to their problem. This seemed to be Bill Seagrist's view at least; he told me later that it was my initial low-ball estimates that made the company think it could handle the project. Howsoever, I did not stick around to do the project so I never found out in person how far off my estimates were. Once the project was started and it became obvious that much additional resources were needed the company put in those resources and got a competitive Network Management System on the market in an appropriate time frame. With the MP-48 modem and Network Management System, plus the follow-on 9600 bps version of the MP-48, Paradyne was now ready to compete in the large network market and soon became a power house in the business.

I, however, left the company before I had a chance to see the success in person. After finishing the MP-48 I decided to strike out on my own, feeling that designing modems was something I could do pretty much by myself, and did not need a large organization with which to share the rewards. After some abortive tries to get a business started together with Bill Nicholson I started on my own with only some modest consulting opportunities from AMI (where Nicholson then was) as my prospects. Because of AMI's time requirements I had to make a short schedule commitment which forced me to resign with only a two-week notice to Paradyne, and part of that time I had to take vacation so I could visit AMI and start on their project. Verney Brown's view was that I was quitting without notice, which made him upset. I was equally upset when I opened my final paycheck and found an invoice for about $1500 instead of a check. I had borrowed $2500 from the company for the down payment of my house in Florida and this had been deducted plus interest from my final pay, resulting in a negative balance. I could not believe that Wiggins could be such a cold-hearted asshole but he was. From my perspective, I was responsible for the fact that he had the job of Paradyne CEO, because without me there would have been no Paradyne. I was the author of the original market studies and product plans that we used to get our initial investment from Tampa Electric, then I invented the Bisync-48 modem which allowed us to charm the original VCs in New York to the tune of about two million dollars, then I came up with the PIX product which was instrumental in getting us the second round of financing, and finally through mostly my ideas and through solely my sweat and blood I had positioned the company for prime time with the MP-48. Without my ideas and work it is highly likely the investors would have stopped funding long before Wiggins came along to garner the rewards. During the previous year when I had put the rest of my life on hold and given the project all my energy and attention, including virtually all my waking hours and a good percentage of my dreams, good old Bob put in his 40 hours per week, going home at 5 pm, enjoying his weekends, and not even having any idea about what I was doing, or how I was doing it. He knew enough to realize that he could look me in the eye and tell me that he had a right to stiff me and he was doing it. Later I heard that the VCs considered him one of the best CEOs that they had in their teams. But he didn't know enough to keep his company from getting caught by the justice department and winding up in court being prosecuted for fraud. Nor has he been a good enough CEO to pull his present company, Technology Research Corporation, out of the doldrums. Just being a great CEO apparently isn't enough. You sometimes need somebody with ideas and creativity as well.

The financial blow from this parting gesture from Wiggins was devastating. I was starting out in business for myself with negative resources. To show they were really nice guys, however, they gave me an opportunity to make some money. Wiggins put me in touch with a VC named Jean De Leage who bought my 2000 Paradyne shares at about $2.50 each. A couple of years later they were worth around $30 each. John Applegate, who was then an engineering manager gave me a contract to modify the MP-48 for the Nordic Public Data Network, which involved reprogramming it for 6000 bps operation. I did this in a couple of weeks and got paid about $5000. So, in May of 1977, after 390 weeks of intense effort and concentration that seldom were below 60 hours, I left Paradyne with the same 5000 dollars I had put up seven and a half years earlier.

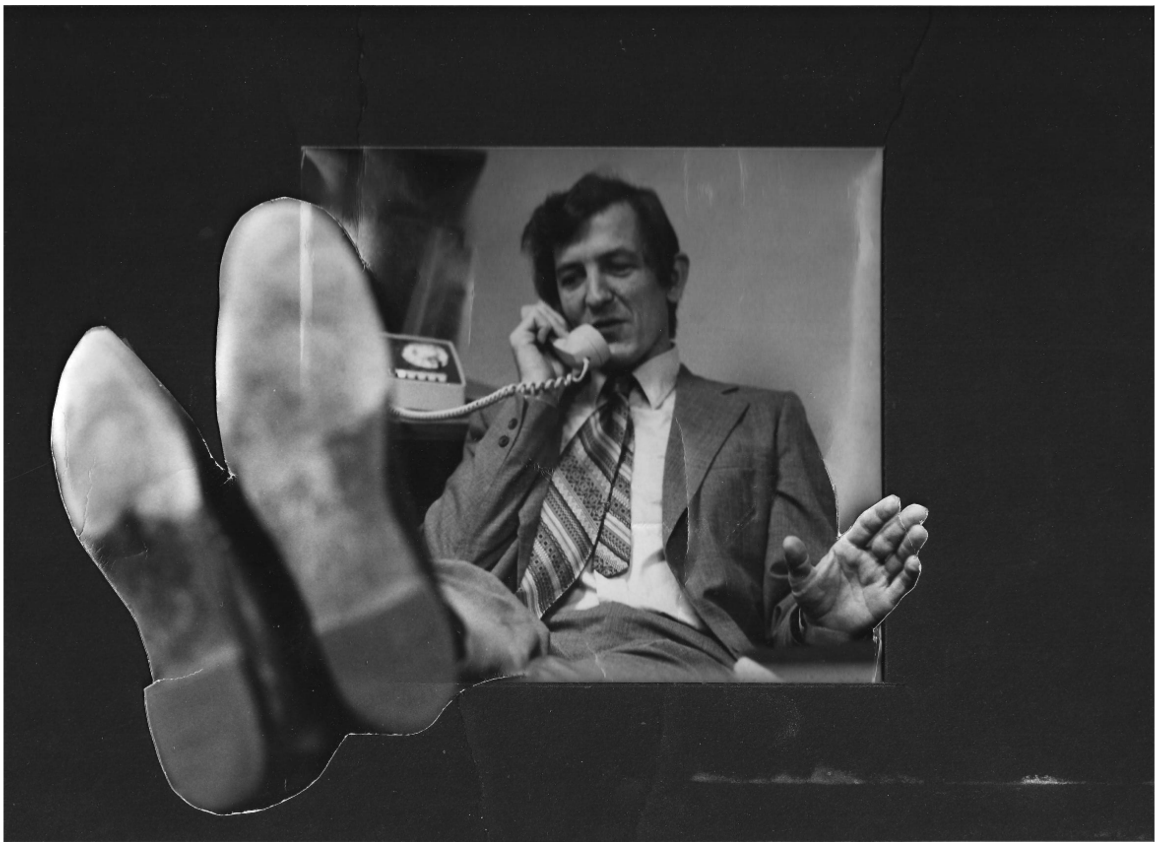
End of part 4

Questions & Answers about the Paradyne BISYNC-48 Modem

paradyne

Bisync-48 brochure

Back



Doing product management

Back

Paradyne as it looks today

Back

INNOVATION IS ALIVE AND WELL

Every so often we find ourselves biting our tongue (though not so often, we suspect, as our patient publisher) over things said in this column. In our January commentary on the state of the industry, we bemoaned the lack of truly innovative and well-conceived new product or service offerings. This month we are pleased to report that innovation is alive and well in at least two cases.

In the first, Western Union continues their transition from singing delivery boys and Candygrams to the Communications Age with their very attractive service offering, Datacom. In the second case, Paradyne has greatly cost-reduced and simplified remote batch or remote job entry with their new peripheral interface extender. We commend both of these to your attention.

WESTERN UNION'S DATACOM SERVICE

Last fall Western Union introduced a new and interesting service called Datacom, offered under Tariff [illegible] 2[illegible] The service [illegible] the p[illegible]sion [illegible] d [illegible] e [illegible] direct [illegible] t ser[illegible] y, [illegible] na[illegible] ri[illegible]ation fee is charged according to the schedule shown in Table 1. These fees are per Datacom service. If a customer and an authorized user each needed seven, 75-baud channels, they would pay at the fourteen channel rate.

PARADYNE'S PERIPHERAL EXTENDER

Paradyne Corp., Clearwater, Florida, recently announced a very interesting, unique, and attractive device for use in data communications. For originality in concept, it is very impressive and stands out amongst the chaff of new product releases. Notwithstanding all that hyperbole, it also seems to have economic appeal.

Paradyne's new device, called the PIX-600 Parallel Interface Extender, looks and acts like a peripheral controller. However, it is designed to operate its peripheral device over a communications link. Fig. 1 is a block diagram of the system. Its beauty is three-fold. In terms of hardware, it combines the communications controller, data set adaptors, and high-speed modem in one unit; and the remote terminal can be a standard peripheral device instead of a full-blown remote terminal. From a software point of view, the remote terminal now looks like any locally resident peripheral device. The only difference is that the peripheral seems to operate much slower because of the throughput ceiling imposed by the communication channel. Operating at a data rate of 4800 bps, the PIX-600 appears as a 600 byte-per-second peripheral. Finally, from an operating point of view, the potential throughput rate is substantially improved over many other systems. This is because of two factors. First, the communication sub-system is self-contained within the Paradyne equipment. Hence, the sub-system can block data for optimum transmission efficiency without regard to the blocking characteristics of the terminal. (The effect of optimum block length on throughput is discussed in Saul Stimler's article on Page 68 in this issue.) The second factor in improving throughput is Paradyne's use of a simultaneous reverse channel for conveying ACK/NACK, which eliminates turnaround time on a dialed-up connection.

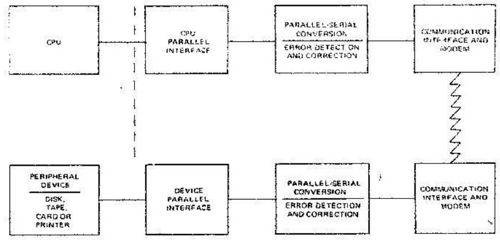


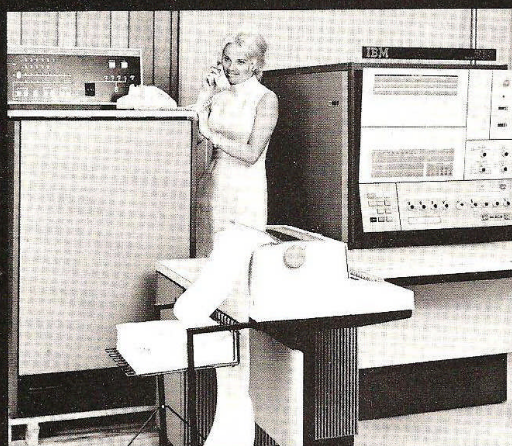
Fig. 1—Block Diagram of Paradyne's PIX-600 Parallel Interface Extender.

Actual line transmission rate is 5112 bps. The difference between 4800 and 5112 is allocated to error-detection coding and to overhead characters for line coordination/control. In a noise-free channel, then, the data throughput is a full 600 bytes/second. Paradyne's press releases guarantee no errors attributable to the communication environment. We debated that one with their marketing VP, James Wylie, giving him a chance to qualify it. He backed off only to the still-remarkable position that the PIX-600 would produce an undetected error rate of 1 x 10^{-12} on a "worst-case" channel. That is, for a system operating at 1 x 10^{-6}, he felt they would operate at 1 x 10^{-12} on the same channel. Of course, the cost of this is throughput degradation, which has to be assessed for all competitive systems in any individual application.

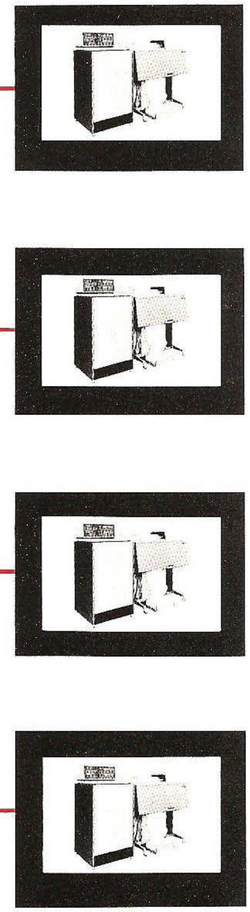
In certain systems (probably those with few remote terminals requiring simultaneous access), the total hardware package of $6000 per end, $12,000 per link, will be very cost-effective. Overlaid on this, however, are the savings in core and software: no telecommunications access method, and no special terminal handlers. Being a more familiar I/O system, it should be infinitely easier to program and to cope with bugs and operating problems. Even in systems requiring "several" simultaneous channels, the $6,000 price will be attractive compared to the $5000-$6,000 necessary for 4800 bps modems alone. ▲

Story in the magazine Modern Data by Ralph Berglund, Back

- Provides highest throughput available on dial-up lines.
- Provides simultaneous card read and line print capability on leased lines.
- Eliminates communications controllers (270X), modems, terminals, and TP software (BTAM).



- Remotes standard IBM peripherals over leased or dial-up telephone lines.
- Drives remote peripherals with standard unit-record I/O software.
- Provides Remote Job Entry by use of standard system Readers and Writers.

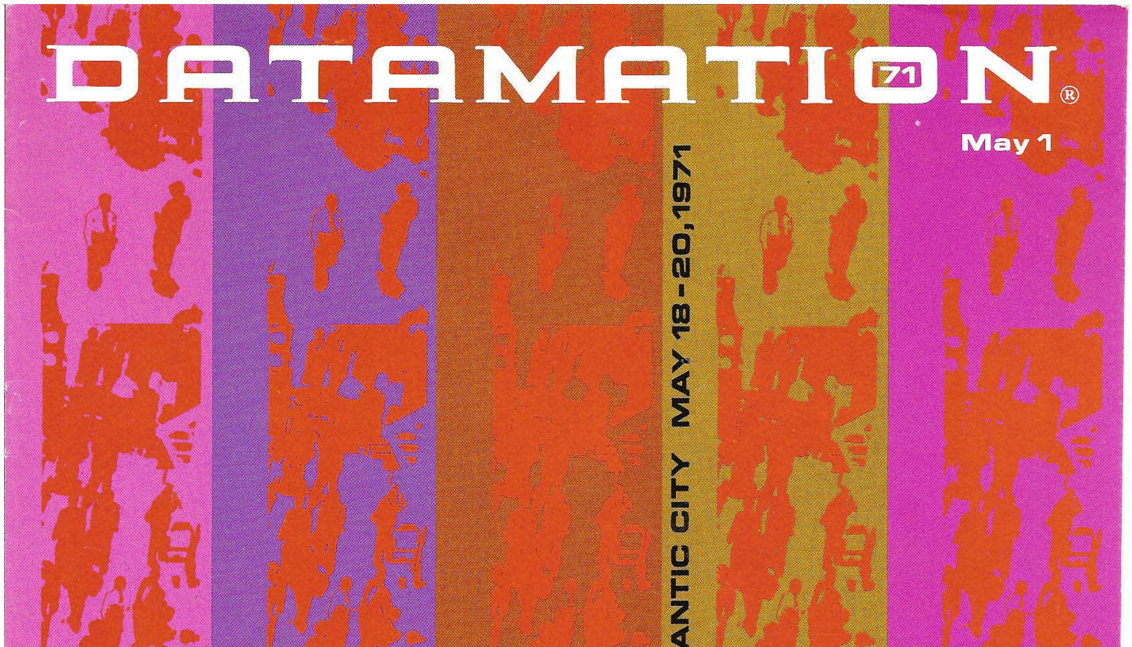


PIX...
Remote Processing and I/O for System/360/370

paradyne

PIX brochure

Back



Analysis of Common Carrier Tariff Rates

With communications becoming an increasingly common feature of computer systems, data processing managers and their staffs are now faced with the task of evaluating common carriers and their services. In addition to the technical complexities of grafting digital equipment into the analog telephone system, there is the not inconsiderable problem of making informed price comparisons of various services offered. The rate structure of the telephone system is a complex affair whose complete details are not readily available. Granted, the WATS rates for service between a given state and any other state are available from telephone company representatives in that state, and the operator in a given city can supply the long distance rates from that city to any other. But neither of these sources is sufficient for full information about the various rates between points which are both distant to the inquirer. Such information is found in the files of the FCC, and presumably in certain telephone company files.

This paper contains a summary of the tariffs on file as of the end of 1970[1] for three types of voice grade service: (1) private line, (2) the dial-up network, and (3) WATS. Further it presents some techniques for making rapid rough cost comparisons of the different types of service.

1. AT&T has filed increased tariffs with FCC for some types of dial-up service, effective January 21, 1971. These rate increases mainly involve operator-handled calls and therefore should have negligible effect on this analysis.

The information presented herein, and the results based on the comparison techniques, are not meant to be used in lieu of quotes from the telephone company, but can be used for preliminary planning purposes, prior to contacting local operating companies for the specific costs of a planned data communication system.

Types of service available

Private line service. The private line service most applicable to data communications use is the type 3002 channel. Although basically of voice grade, this type of channel is somewhat better suited for data than dial-up connections, because its characteristics are better controlled. In addition, channel-conditioning arrangements are available for improving its suitability for data transmission. Such conditioning adjusts the frequency and phase response characteristics of the channel to meet closer tolerance specifications.

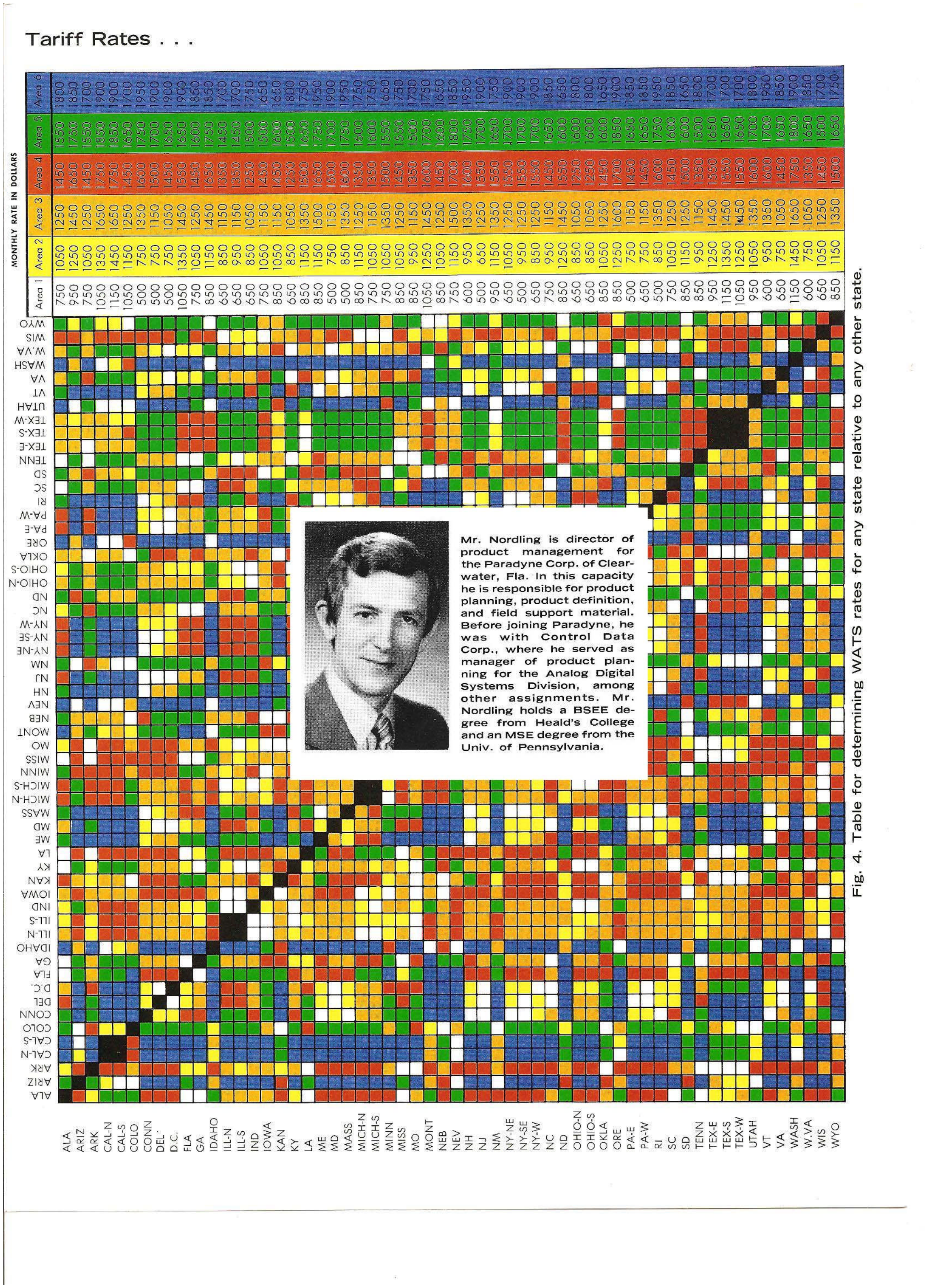
To the user, the effect of the different degrees of channel conditioning is roughly as follows:

1. Unconditioned or C1 conditioned lines allow 2000 to 2400 bps operation with modems using fixed equalizers.

2. C2 conditioning allows 4800 bps operation with modems using manually adjustable equalizers.

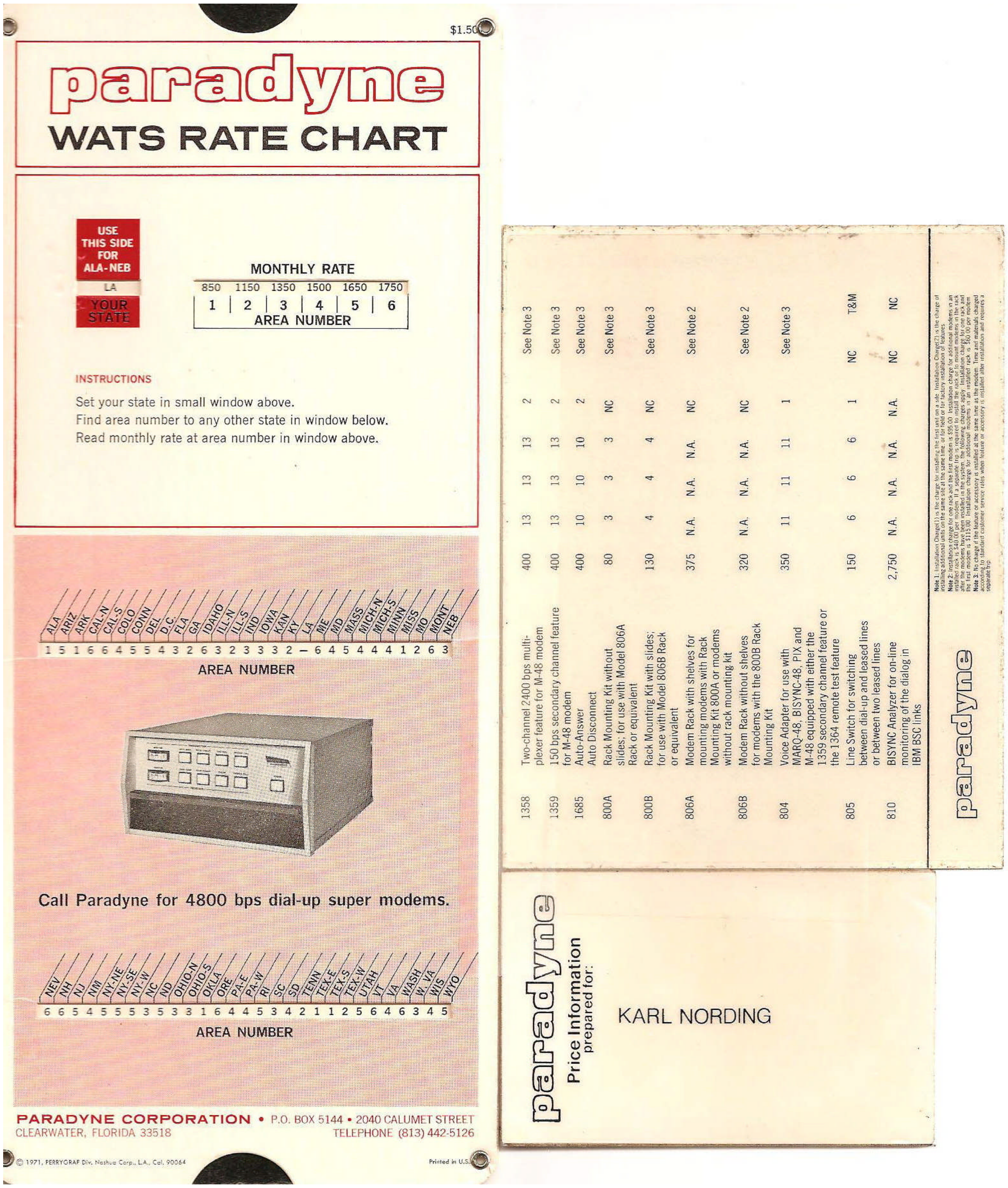
3. C4 conditioning allows 7200 to 9600 bps operation with some modems using adaptive equalizers. (The equalizers in the modem perform basically the

My feature article in Datamation Magazine Back



My colorful WATS chart

Back



WATS slide rule and laminated price list

Back

PROGRAMMING A MODEM

Karl I. Nordling   Dale M. Walsh

Paradyne Corporation
Largo, Florida

SUMMARY

A 4800 bps voice grade modem is described which is implemented in its entirety by means of a special purpose programmable digital computer. The signal processing load is analyzed and the resulting requirements for processing capability in the computer is discussed.

The computer architecture is described and related to the special nature of the processing load. The approach to programming typical signal processing functions is discussed and an example of a detection routine is given to illustrate the software solution of a typical hardware problem. Potential user benefits resulting from the programmed approach are discussed.

INTRODUCTION

From a hardware viewpoint, the device described here is a special purpose programmable computer equipped with analog and digital I/O and an associated clock-system whose count-down length can be controlled (over a limited range) by the computer. The computer is a special, as opposed to general, purpose machine only in the sense that it is optimized for modem signal processing, not in the sense that it is restricted to that type of processing. The product implementation for which it was designed, and which is described here, is a 4800 bps voice grade modem designed to handle all popular signalling methods, including CCITT V27 bis , V27 ter , the Bell 208 A and 208 B and anything else that is deemed interesting or worthwhile. The methods which the first product offering uses are those defined by the CCITT V27 bis and V27 ter recommendations. In the implementation of these modems the processor handles all modem functions except A/D and D/A conversion, clock count-down and EDP interface driving.

SOFTWARE ARCHITECTURE

Using a programmable, sequential processor to perform the signal processing in a voice grade modem is one of the latest applications of the now well established art of real-time programming. As in any such application, the parameters that determine the hardware and software requirements are the length of the compute cycle and the amount of processing that must be

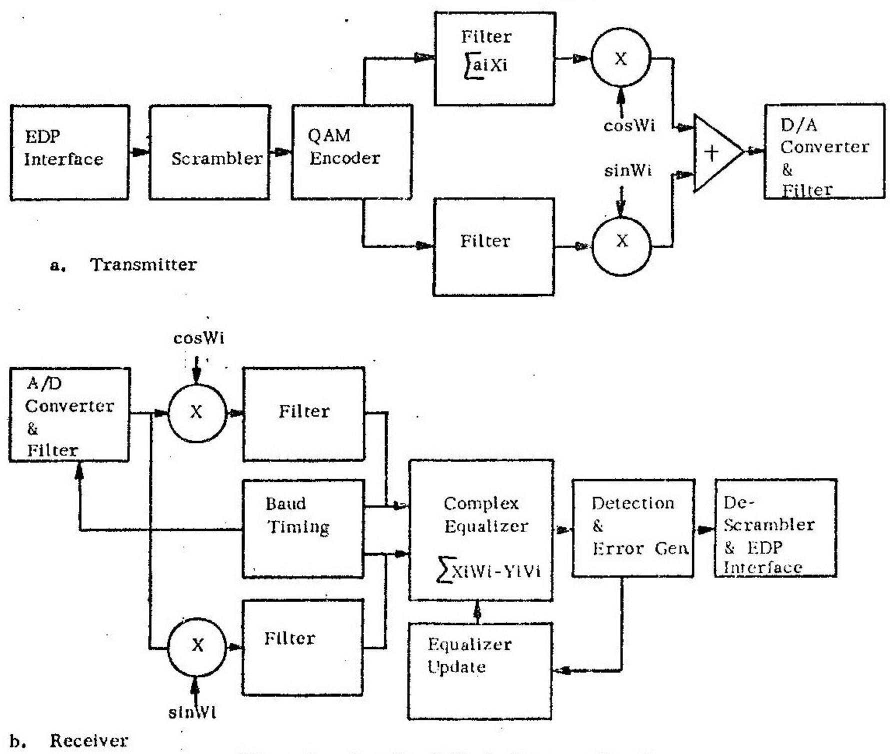
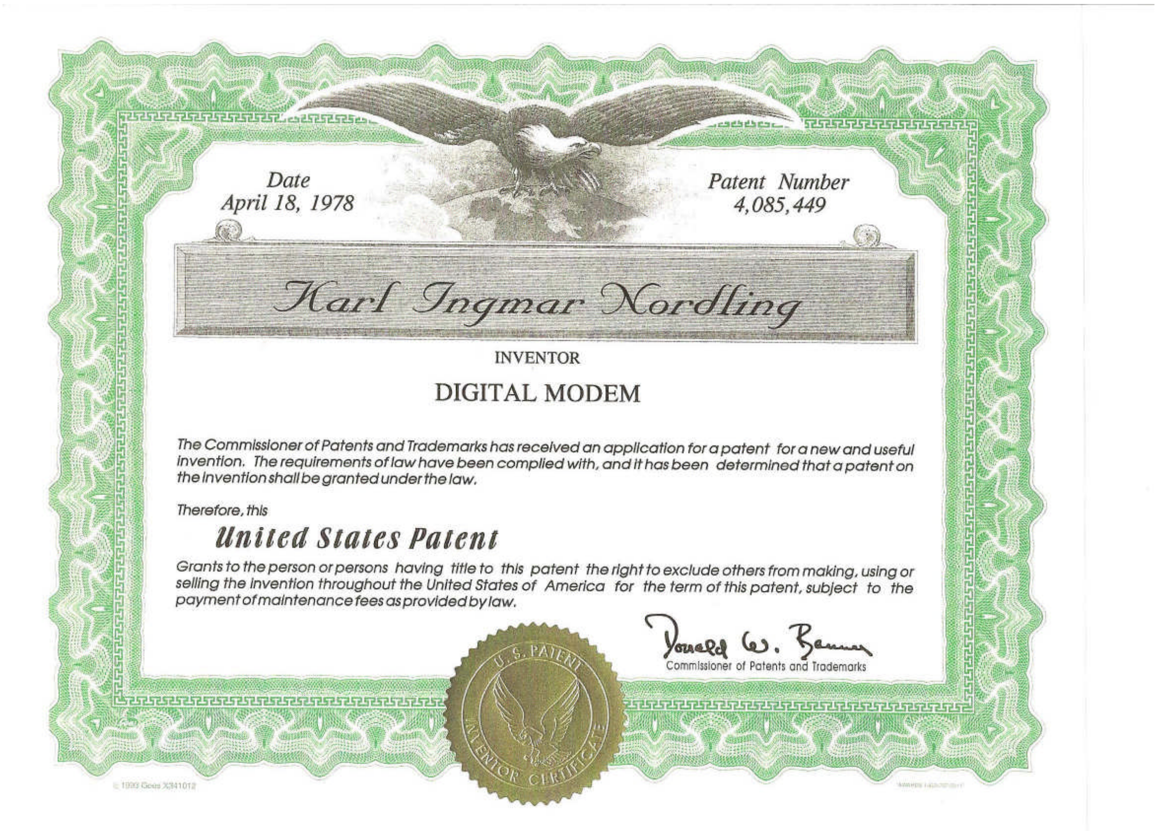


Figure 1. Functional block diagram of modem.

A paper I presented at a computer communications conference in Dallas

Back



Date
April 18, 1978

Patent Number
4,085,449

Karl Ingmar Nordling

INVENTOR

DIGITAL MODEM

The Commissioner of Patents and Trademarks has received an application for a patent for a new and useful invention. The requirements of law have been complied with, and it has been determined that a patent on the invention shall be granted under the law.

Therefore, this

United States Patent

Grants to the person or persons having title to this patent the right to exclude others from making, using or selling the invention throughout the United States of America for the term of this patent, subject to the payment of maintenance fees as provided by law.

Donald W. Banner
Commissioner of Patents and Trademarks

U.S. PATENT INVENTOR CERTIFICATE

My first modem patent

Back