This is Interview TC-91 in the IBM Oral Historyof Computer Technology, Larry Saphire interviewing Mr. Fran Underwood June 12, 1968 at Los Gatos.

S. We want to talk about the design of the 1401. You were telling me that you had worked on the WWAM prior to that.

U. That is correct. Along about 1954, I was involved in the Engineering Planning Department in Endicott and about that time I was working on an accounting machine for a system called a VIDOR. VIDOR stood for Video Document Reader. And in 1954 the corporation made a decision to switch from vacuum tubes to transistors and in line with that decision, they instituted a new program in Poughkeepsie called the MAC Program. MAC meant Modular Accounting Calculators. Half a dozen engineers were taken from Endicott and transferred to Poughkeepsie to develop this new line of Modular Accounting Calculator and there was no processor work left in Endicott. Endicott was only going to be responsible for mechanical input/output devices. The MAC Program objectives were to produce a spectrum of equipment that was compatible all the way up and down the line from a small machine that had the capability of a 604 up to the 705, Model 3, which is the largest piece of equipment we had in the product line at that time. They chose a machine logic and started to build the first product in that MAC line. It was the transistorized 608. Endicott had already designed and built a tube version of the 608 and the market that that machine was to cover was going to be taken care of by the transistorized 608 that came out of the MAC line. The MAC Program got under way and the bulk of the effort was concentrated on producing that 608 and very little effort was put on the planning of the entire product line. We in Endicott still had the responsib lity for producing the accounting machine for the VIDOR and the plan was that eventually the MAC line would produce a replacement accounting machine for the VIDOR System. So it was apparent to us in Endicott that we should find out what Poughkeepsie planned to do in the way of machine design for this area so that our design could be compatible. That is, those fifty or so customers who were to get the Endicott design machine . . . they would have their applications all wired up on control panels and then when the MAC Program produced the replacement machine, we wanted to see that the customer would not have to rewire his control panels. He could just take them out of the old tube machine and plug them into the new transistorized machine. In order to do that, we had to understand exactly what Poughkeepsie planned to do. We soon found out that they had no plan. They didn't know what they were going to do in the way of an accounting machine. So we made our proposal. Endicott made a proposal to Poughkeepsie that they accept the Endicott originated design for an accounting machineAt the same time there was a task force sent over to Germany to look at transistor developments in Europe and at magnetic core memory developments that were going on in Europe. And Dr. Ganshorn and one of his men had, we discovered, developed on their own, a design for an accounting machine. We also discovered at about that same time that Gene Estrenz and Maurice Papol in France had developed a different kind of an accounting machine and since the need for an accounting machine was recognized in the MAC Program, these three groups of people were pulled together -- Ganshorn and his man from Germany, Papol and Estrenz from France and the people in Endicott. We met together in Poughkeepsie and after quite a few months of work we put together a single machine concept which finally emerged as the WWAM, the World Wide Accounting Machine. It was based on the CPU data flow that was designed by Estrenz. This was a variable word length concept. The memory developments that were developed by Hans Ganshorn and some of the input/output techniques that were designed by Endicott. I remained in the Engineering Planning Department back in Endicott and had a minor role to play in the development of the WWAM Program. One of the things I did was to help design a control panel for that machine. The original targets for the WWAM were something like $1,250.00 a month rental for the minimum machine and all of the features were planned at the beginning so that the features were simply easily pluggable into the system. Following this design philosophy, it soon turned out that the minimum machine was going to cost about $4, 500. 00 a month, better than three times the original planned objective. So the WWAM machine was not going to meet its objectives, so Estrenz and Papol were sent back to France to rework the WWAM machine and try to get the cost down. They did this and about the best that they were able to do was to get the monthly rental of the basic machine down to about $3, 100. 00 a month.

S. Was this always predicated on the plugboard machine?

U Yes it was. Meanwhile ....

S. Just as a point of information, was that $1, 250. 00 a month objective unrealistic for an electronic machine?

U. I think it was unrealistic at that time certainly. The objective eventually was raised to $2, 500. 00 a month. But Estrenz and Papol could not get the rental down below $3, 150. 00 a month. That was the best they were able to do and at that rental, the machine was really stripped and did very little. So this led to the demise of the World Wide Accounting Machine and one of the things that helped that thing die I think was the death of the MAC Program. The MAC Program never made it either. They couldn't meet their objectives. Meanwhile back in Endicott we were continuing with the development of electro-mechanical input/output devices and one of the most significant developments there of course was the 1403 chain printer. The chain printer was developed to replace the wire printers and the wire printers at that time were being used as peripheral devices on 704's and 703's and 709's as high speed output printers. And a printer control unit was required to drive the printer. So Endicott was developing the chain printer and the control unit to go with that, that would allow that sub-system to be connected to the larger 700 series processors. The basic design for the control unit was based on the concepts of the World Wide Accounting Machine, that is the variable word length big control panel, essentially the same data flow and it was proving to be rather expensive. However, that was the plan and the 1403 was designed specifically for that application, that is peripheral to a very large data processing machine. Well since the MAC Program had died, and the World Wide Accounting Machine died because it couldn't meet the objectives, it was recognized that there was going to be a big hole in our product line. We still did need a very capable data processing machine in the $2, 500.00 a month range and there was no plan to produce it. We were also looking at that time for a 650 replacement. So long about the end of 1956 or early 1957, early 1957, Ralph Mork was made Manager of Accounting Machine

Development, one man, and he needed to get together a plan and pull together a staff to develop a new accounting machine. The objectives were very broad at that time. We needed a system that could replace three 407's and a 604. Something of that sort. Well my experience on the World Wide Accounting Machine and on the MAC Program was recognized and Ralph Mork felt that perhaps I should be the guy to start the development of some new kind of accounting machine. So I accepted the task and in about May of 1957, I started a one man effort and the first thing I did was to review all past developments, look at, that is all past developments for accounting machines, for printers, for card readers, for punches, and I soon came to the conclusion that the World Wide Accounting Machine was the closest that anyone had come in the past to meeting the kinds of objectives we had in mind. So then I took a look to see what was wrong with WWAM. Why had it cost so much? And I soon discovered that about 45 per cent of the electronic cost in the World Wide Accounting Machine was simply there for the purpose of driving the control panel. It had no logical function. Just a lot of electronics in there to permit communication between the control panel and the internal logic and back again. It served no other function and it was very costly.

S. I might ask the question at this point, why hadn't this analysis been made before?

S. I can ask the question at this point, why wasn't this analysis made before?

U. It was over a period of two or three years. No one had ever examined this question before.

S. But Papol and Estrenz must have realized it.

U. No, they never did. After I had taken a look at the WWAM, and discovered that there was that much cost associated with the control panel, I asked the question would it be possible to eliminate all that hardware and with what was left make a stored program machine. It didn't take very much study to show that yes you could. You could strip out all that electronics that goes with control panels and with what little is left, you can fully implement the stored program machine. No one had ever asked that question before. No one had ever made the study.

S. Had they merely gone under the assumption that the plugboard machine was the cheapest to use?

U. I think that is exactly what happened.

S. And that anything would be worse than let that happen.

U. Yes. As a matter of fact, I had precisely those comments made to me not only by Estrenz and Papol, but by Jim Troy, who was Manager of the Endicott Laboratory. Whenever one mentioned stored programs computers, Immediately one would think of a machine like the 705 and say that's too expensive. But I found a way to build a stored program machine very cheaply. I had to .... I did choose to use the CPU organization for data flow that was developed by Estrenz and Papol and I think this was an important point. There was no other data flow or CPU organization that would allow us to achieve that same low cost. Not for a stored program machine. Estrenz and Papol doubted very seriously that I'd be able to really make a stored program machine at the cost that I was talking about. Well it turned out that my first estimate and the final estimates that were made eighteen months apart didn't vary by more than three per cent. So that our first initial estimate on the stored program 1401 was very realistic.

S. Did you come up with a new idea that allowed you to make .... an inexpensive stored program?

U. Yes. I think the word mark concept was an important concept. The variable word length idea and the instruction length idea were important in that it permitted us to pack a lot more program into a given amount of core storage than was otherwise possible. An interesting fact, I spent about nine months by myself developing the concepts that went into the 1401 and before that nine month period was up I discovered that San Jose was developing a successor to the 305 RAMAC and I was invited to come out to San Jose andhavea discussion with these people. So I spent three weeks in San Jose in about 1958, and we made some very serious studiescon cerning the worth or the effectiveness or the efficiency of variable word length and variable instruction length and it turned out to be quite significant. You can get by with approximately 40 per cent less core using the 1401 concept comparedtothe amount of core required using the 305 successor concept. They used a 12 digit fixed instruction length.

S. This is another interesting point that you are implying that the old idea or let's say the stereotyped idea was that fixed length words are a lot cheaper than variable length.

U. That is correct. But I found a different way to do it. I chose to put an eighth plane of cores in core memory. I called them wood marksThat allowed us to find a very cheap answer to the problem of the variable word length and variable instruction length.

S. Somewhere along in the WWAM discussion, I'd like you to explain how you came to these conclusions that that reduced the cost so much.

U. I had two target systems or two systems to compare against. There was the World Wide Accounting Machine and the hardware was built and the logics were drawn and the cost estimates were there and the component counts were there and you could actually see and use that as a bogey.

The 305 replacement I think it was called or the 306 or the 310. That design work was well along here in San Jose. They had component counts and they had cost estimates. So it was easy to see exactly what the comparisons were. We had some programming kernels, some example programs that would program for both systems and showed time and time again that the 1401 would pack more power in a given amount of space than the other concepts.

S. Well how did you arrive at the word mark concept in the eighth plane?

U. Well I don't know. I had an overwhelming desire to pack as much as I could into magnetic core storage because at that time core storage was expensive and it just irked me to see wasted core positions between fields which would be necessary if you had fixed word length machines. You are continually wasting a character here and two characters there all over the place and that just bothered me and I wanted to find a way to pack everything up tight. I looked at programs written with variable length instructions or with fixed length instructions and I saw a lot of wasted space in the instructions, address space in an instruction and we didn't need it. It bothered me. I was just determined to find a way to pack everything up tight and so I think it just naturally occurred to me that if one went through an instruction stream or a data stream, there had to be something that tells you that this is the end of a field, this is the beginning of a new field. This is the end of an instruction, this is the beginning of a new instruction.

S. And it would immediately start storing that new instruction.

U. Well a flag at the beginning of each new field of each new instruction and the natural thing to do was, there were two choices, a special character code or an extra bit. Now the extra character code bothered me because that was extra positions of core going to waste and so I chose to go with the extra bit.

S. Well why hadn't this idea been thought of before?

U. Who knows? I have no idea. I don't know why not. Perhaps ... you see I was trying to work down at the very low end of the cost spectrum at that time, trying to save every penny, nickel and dime. Motivations are quite different. When you are working on a large system and trying to get every ounce of performance, and you're not really trying to save money, the motivations are quite different. When you are working down at the low end you are not terribly concerned about performance but you are concerned about cost.

S. Well how would the performance be affected by attacking the memory, simply reducing the flexibility?

U. Well if I had used a unique character code as a word mark or a field boundary, that would have eaten up performance because one would have to scan over those things and you are wasting time scanning over a control character. So I just chose to go to the word mark and pursue that, just intuitively believing that that was the way to go.

S. How did you see the word mark as being implemented?

U. As an eight bit in core, an extra bit with each character.

S. And the programmer?

U. Would then set and clear word marks with a machine instruction, with a housekeeping instruction. Well that was one aspect of the 1401 development. Another thing that I did that was I thought somewhat unique, I tried to make the instruction code neumonic. I used alphabetic characters, A for add, S for subtract, M for move, E for edit and so on, the motivation being that it would be easy for a customer to learn machine language if I gave them the neumonic op code. Another thing that was definitely significant in the 1401 was the edit command which allowed one very easily to edit out reading 0's, to insert floating dollar signs, to fill with asterisks, to edit out unwanted comments and so on. I struggled with that problem for several months before I finally stumbled on the particular idea that was finally implemented. I can't tell you how it came about except that I worked very hard on the problem and thought about it constantly and tried various approaches and then finally hit on the one that seemed perfect. Another thing that is interesting about the 1401 development, I concentrated primarily on the development of the CPU and gave secondary thought to the development of the input/output equipment. I knew generally what was required. I think my first thought was to somehow modify the print mechanism of the 407 to give us a printer for the 1401 but fortunately the 1403 was in development and that provided the answer. It's a strange thing that the printer development people and the other product planners failed to recognize the importance of the 1401 CPU to the 1403 printers and it wasn't until the week before the 1401 announcement that all the other product plans for the 1403 were dropped. They were still trying to build a control unit for the 1403 to attach it to largescale data processing equipment. Just before announcement, it was finally recognized that the 1401 system itself was a far better control unit for the printer, far more powerful and at about the same price than what they were trying to do.

S. And so all the big machines have the 1401 and the 1403.

U. That's right.

S. Well let me ask you a question about the CPU. Were you concentrating on that more or less because that was where all the problems had occurred in cost and that you had gotten on to the fact that that was really the key element in reducing cost?

U. At that time that was our belief and that same belief has persisted for a good many years in the corporation. People believing that the real problem was with the CPU. Fortunately a lot of us now recognize that that's not really where the problem is. The problem now is with the input/output equipment. The CPU is quite secondary. Anybody can design a CPU to any level of performance pretty easily. But the real trick in the trade now is to design low cost, high performance input/output gear. We solved our input/output problem very nicely. The 1403 came along at just the right time, with just the right performance, at just the right cost and we were able to pick up the collator, the 088 collator base and change one of the feeds to a punch station and convert that into our card reader and punch. This at a very good performance and good price.

S. Well it is rather interesting that the development of the 1401 and subsequent machines should kind of depend on getting rdi of those old ideas and looking at things freshly. And the 1401 and correct me if I m wrong, with that the CPU was really the seat of the problem at that time in the WWAM.

U. Yes it was.

S. And therefore you concentrated on the CPU in the 1401 and later on as you say, the CPU became considered more generally a simple problem and people always concentrated on the CPU so that input/output was the key. Is that correct?

. . . . .

U. Well I don't know if that was particularly true. My particular skill and my particular duty brought me to concentrate upon the CPU. I remember the World Wide Accounting Machine development. They had very serious problems with input/output, well recognized serious problems. The printer that they were going to use on the World Wide Accounting Machine was being developed in Germany was a stick printer and it was just no good, very expensive, very p very poor performance, the maintenance problem would have been unbearable. The card reader and the card punch, these were being developed I think in France. They were very poor performers, very expensive. So when the WWAM Program was dropped, those input/output developments were dropped fortunately. The timing on the 1401 was beautiful. The timing had a great deal to do with with the success of the 1401. First of all, there was the 1403 printer that was being developed, not for the 1401 but for another purpose but it was done at the right time. The 088 collator, that development was complete and we had a product and we were to build on that. It came through at the right time at the right price. The circuit family that we used on the 1401 was being developed for the 7070. I don't recall that we .... that the 1401 development shared in the development cost of SMS but the SMS circuit family was developed and it just fit our needs beautifully. Design automation was developed and was usable at the time we wanted to use it in the 1401 and we didn't have to wait for that or struggle with that. So as I say, timing was extremely important to its success. All of the elements that we needed came at the right time and all that was lacking was the CPU and that's where I concentrated.

S. And the key element in the CPU was the word mark?

U. Well I think primarily the key was stored programming first, getting away from control panels, recognizing that control panels really limit what one can do on a system, recognizing that the stored program is many, many times more powerful than a control panel machine because one can use high level languages to express the problem and then they can use the computer to develop the machine language the machine language program. You can develop a great many applications in a short time on a machine that has stored programming and you can't do that on a machine that has control panels. So the stored program was the first thing. Secondarily the techniques of variable word length, variable instruction length, the use of the word mark to achieve those and the edit function I think are the th ngs that are unique to the 1401 and help to achieve its low cost.

S. Was programming difficult for the 1401?

U. No. We didn't have systems programming people like we do today. At the time we started the 1401 development, there was no group of people that we could go to and say, here's the proposed design for the CPU. Program it for us and tell us where its deficiencies are. We didn't have such a group so I had to do the programming myself. I had to find representative problems and program those applications and then modify the CPU design so that the programming was okay. Eventually there was a programming group formed and they went on to do all the systems programming for the 1401. Following the 1401 there was the development of the 1410. The 1410 was a much larger system, file-oriented and was essentially the next step beyond the 1401. In the initial stages of the development of the 1410 the question was should one be precisely compatible with the 1401, just add new instructions and more powerful instructions to facilitate working with the disks, or should one depart? Well they made a partial departure from the 1401. They went to fixed length instructions. They didn't use word marks and there was a debate about that point and several independent studies were made and it was shown that the 1401 organization was certainly as efficient if not more efficient. It was hard to tell really, than the organization that they planned for the 1410, but there were other factors involved and I don't remember what they are.

S. I take it you weren't designing the 1410.

U. No, I was involved in it and I was consulted but I wasn't directly responsible for anything onto the 1410.

S. So what finally happened?

U. Well they did depart, but as you know, compatibility is the big thing in the corporation today and they insist upon compatibility now because of growth problems. When one grows out of a 1401 and grows into a 1410, you've got compatibility problems, both for the customer and for IBM and they don't like to see that happen anymore. So they really should have been compatible.

S. Well actually was the 1410 in the real family with the 1401 ...... as you described it it seems to be radically different.

U. Well it was a serial by character machine as was the 1401. The instruction format was the same as a 1401 except that it was fixed length and it had a five digit address instead of a three digit address. I don't recall enough about the 1410 but it was enough different that you had to write all new application programs and programming systems all had to be done over again for the 1410.

S. Did that have a print edit function?

U. Yes it did. They copied the edit function from the 1401.

S. And by eliminating word marks and by switching to a fixed length word, what did that accomplish?

U. I don't know. I don't know what it accomplished for them.

S. I thought you always had to have variable length for commercial type machines.

U. No you don't need to have that.

S. Just make the fixed length long enough to accommodate them.

U. Uh huh.

S. So the 1410 must have been much more wasteful than the core storage in the 1401.

U. Oh yes.

S. And since you had shown that you could make it cheaper and cheaper, the other way, am I right in assuming that the 1410 was more expensive?

U. I think so, but they had different kinds of problems. They had to work with a file and they had to work with larger core memory. The addressing capability of the 1401 was limited. They had to go beyond the addressing capability in the 1410 and they only could do that by lengthening the address part and once they decided to do that, it seemed to make sense I guess at that time to go to fixed length instructions. It is so far back that I don't really remember any more.

S. Who was the designer of the 1410?

U. Of the 1410?

S. Yes.

U. Well Dick Case. Do you know Dick Case?

S. No I don't.

U. He is the Divisional Director of Systems Architecture now and he is the guy that is insisting on compatibility for all the entire product line. There are a few comments I'd like to make in retrospect with regard to the 1401 program. The initial objectives of the 1401 were very simple, very broad, not very specific. We started with this objective. We want a machine that will replace three 407's at a rental of $2,500.00 a month. That's to include a card reader, card punch, printer and the CPU. We started from there. As we proceeded through the program, of course the specifications became more and more firm, more detailed as they always do. The point is that we did not envision at the beginning the tremendous power of the 1401 as it finally evolved. We didn't see the tape system. We didn't see the 1401 being used as a peripheral system to large system. Those concepts developed much later. In other words, we got some tremendous fallout from the 1401 program, new concepts in data processing.

S. Could you be more specific on that?

U. Well the idea of having a small processor controlling the I/O devices communicating with the large system through tape, we didn't have that concept when we started the 1401. That proved to be a very important concept. It was really great. I mentioned at the beginning that the 1401 started at about the time the corporation was looking for a replacement to the 650. We never found a replacement to the 650. There were many, many efforts made to find such a replacement but when the 1401 came out, we automatically had one. The 1401 was a far better machine than the 650 and it hadn't been planned that way. It was another big fallout in the program.

S. Did 650 users buy the 1401?

U They certainly did. They certainly did.

S. And the 7070 was a larger version of the 650, was it not?

U. Yes for larger users.

U. And it turned out that the 1401 was the I/O control unit for the 7070. It hadn't been planned that way in the beginning. They had quite a different approach to the problem. But as the 1401 matured, and the new concepts came into mind, we suddenly saw that we really had more than we had bartered or bargained for.

S. Were a lot of the enormous sales of the 1401 due to the use of the ....

U. Less than 50 per cent. I think the bulk of the 1401's were sold as we originally planned, as a step up from unit record. The 1401 very effectively bridged the gap beginning... well between the unit record, the EAM accounting machine user and the largescale data processing user.

That was a very difficult gap to bridge. A typical installation might have half a dozen 407's and many sorters and many key punches and collators and reproducers, so many that the installation would be unmanagable. They would be forced to go to a 705 or a 709 and to bridge that gap was very difficult before the 1401. The 1401 made it very easy to do. Well the point is that you never know at the beginning of a program just what's going to come out of the program and for instance, at the time the 1401 was announced and we had some of these important concepts in mind and essentially we knew that we were going to have a real winner on our hands, Market Forecasting in DP forecast 3,150 systems, 3,150 1401's to be sold. We in engineering had predicted 7,500 and we were very disappointed with the DP forecast. We sold 750 systems the first week. Within the first three or four months I think we sold a couple thousand and it wasn't until we sold 4,000 systems that DP agreed to up their forecast to 5,000 and when we sold 6,000, they agreed to up their forecast to 6,500. Then we sold 7,000 and then we sold 8,000 and 9,000 and we ended up selling over 10,000 1401 systems. It was a real winner. It could easily have been killed.

S. By the wrong forecast.

U. That's right. It could easily have been killed by the low forecast. It could easily have been killed by doubting Thomases. I think we owe a great deal to people like Jim Troy and Ralph Mork who kept the program going and kept it, I really shouldn't say concealed from the eyes of other people, but they nurtured the program. They didn't pick at it you know. They encouraged the development of the program. If they had chosen to be the other way, we wouldn't have had a 1401 and I'm sure you wouldn't have seen the 360 either. Because I really, truly believe that the 1401 led the way to the 360.

S. Organizationally or technologically?

U. Both.

S. Can you tell me a little bit about the development of the 1130 and the 1800?

U. Well the 1800 and 1130 essentially started about early 1962, early 1963 I believe. The 1130 was designed to replace the 1620 and responsibility for that development was a small scientific group here in San Jose. There were a bynber of machines proposed and a number of different designs and at the same time here in San Jose, there was an effort started to replace the 1710 process control system. There was myself, I transferred out here from Endicott, and there was Roy Harper and Bryan Ottley and Bill Peck and Chuck Probster and Fred Jones. We sort of formed a team to develop the architecture for the new process control system which eventually turned out to be the 1800.

So we had two parallel new processor developments going on out here in San Jose. And if we had continued as two separate groups, we would have had two separate machines entirely different with a need for the development of two separate programming systems developments. Now that sort of thing is frowned on. The question always comes up why can't these two groups that are doing two different things, why can't they get together and do the same thing? So we made an effort and the groups were merged together in a task force with the objective of designing one CPU that would work for both process control and for the small scientific 1620 replacement. It was recognized that the requirements were different but hopefully one processor could emerge that would serve both requirements. Small Scientific needed a low cost system. The process control requirements was for a high performance. We had proposed a number of different designs that would potentially meet both requirements. When we received an edict from back East that we were to make a design which was "lean and hard," now that means very low cost, stripped of all frills, and was designed to meet the performance requirements of the 1800 and meet the cost requirements of the 1130. We did that. We made such a design and we proceeded to implement that design and it wasn't very long ... a few weeks later that we were asked to investigate the possib lity of making a System/360 compatible

CPU to meet those two requirements. We worked very hard on that problem trying to develop a piece of hardware that uses the 360 instruction set, that is low cost for the 1130 and had high enough performance for the process control. We made several efforts and time and time again we could show that the best that could be done with a 360 compatible design, that it would cost about fifteen per cent more now and its performance would be 30 per cent less. Now it would be possible of course to increase the performance by adding cost or reduce the cost by decreasing performance. But this particular compromise was about the best we could do.

S. Well what was it about the 360 compatibility created these undesirable effects?

U. The 360 instruction set is very complex, very large. The channel concept, the standard interface channel concept in the 360 is very involved and very complex, very expensive ......

S. Its complexity is due to the desire for compatibility which you have to add on to make it compatible.

U. Well the 360 itself is complex because it wants to do so many different kinds of data processing jobs, commercial and scientific and it wasn't designed to do process control. Now there's a unique requirement in process control.

Now there's a unique requirement in process control. One has to sense a large number of sensors out in the process. It can be contacts or the voltage sensors, temperature sensors, flow meters and so on. And another thing the system has to do is respond very rapidly to an interrupt. The 360 is not designed for fast response to an interrupt. It is just inherent in the system. It can't do it. Not unless you put an awful lot of money into the hardware. So the decision was made...oh and another thing, the 1130 didn't need the complexity of System/360. That is it didn't need decimal arithmetic, editing capabilities and so on that is in 360. So our studies showed time and time again that the best compromise would be a machine which had 30 per cent less performance, and 15 per cent more cost and so the decision was, okay don't go compatible. Build this new processor which turned out to be a processor that is common to both the 1130 and the 1800 and not compatible with 360. And believe me, that was a mistake. In retrospect, that was a mistake.

S. Why?

U. We have compatibility problems today that just won't stop. People do want to grow out of 1130's into Model 30 and Model 40's and Model 44's and they can't do it without a great deal of reprogramming.

S. What about this process control machine?

U. Well you see we had to develop a whole new set of software for the 1800 and we couldn't capitalize on anything that was already developed for 360.

It is hard to measure the size of that error so I can't really say that it was an error but it appears to be today.

S. Well what you are implying that the whole philosophy of the 360 has been going on for a long time and that standardization was really a desirable feature and that it would eventually pay for itself.

U. This probably is true. I have to believe that it is true. It is unfortunate though that the 360 wasn't designed for process control also. They designed it for data processing, commercial data processing and for scientific work but they didn't design it for process control. The interruptability is very poor in System/360 and channels are expensive. The response is bad. I think it will be interesting to find out what some of these other people have to say about this same subject.

S. Well in the design of the 1130 and the 1800 CPU which is common you say, what did that stem from? Was this entirely new?

U It was entirely new. There was nothing like it before and those two systems have the same processes because they were both developed in San Jose at the same time and the corporation said you can only have one CPU out there, not two.

S. So that CPU I take it was designed both for the qualities of process control which equips interrupts in a lot of channels. Are those good for a small scientific machine also?

U. Yes. The 1130 and the 1800 are pretty good for their application.

S. Was there anything novel in the design of this equipment?

U. There was one thing that is novel in the design of the CPU in the 1800 and that is the arithmetic section. That's novel, it is low cost but it is a poor performer and I think we made a mistake by tryhng to do things as cheaply as we possibly could. I really think we made a mistake. The nature of the arithmetic unit in the 1800 is such that it is hard to predict how long it is going to take to add two numbers together. You're never really sure. You know that it won't be longer than X number of microseconds but it might be as short as two microseconds and you don't really know it and you really should know when you are trying to control a real time process. We have a number of complaints from our customers about the fact that you can't predict how long it is go ng to take to execute a program. But it was cheap and I think that was the mistake.

S. Technologically is there anything in it that we should talk about that differentiates it from other CPU's in particular?

U. No, other than the fact that we tried to pack all of the information relevant to an instruction in one or two words. Now 360 has variable size instructions and they are either two bite or four bite or eight bites long or six bites long, I forget. They vary in length. And it takes more core to store a program to do a job in 360 than it does to do the same program in 1130. The requirements for 360 are quite different. They have to address large cores. They have so many high powered instructions in the machine. The requirements are quite different. I wish though that we could have found a way to make the 1130 and the 1800 360 compatible.

S. Well you did but it wasn't considered worth while, is that right?

U. That's right.

S. Do you wish you could have found a way to make it compatible at a 15 per cent increase or a 30 per cent increase in performance and a 15 per cent reduction in cost rather than the other way around?

U. Uh huh.

S. Is there anything more about that... about those machines that we should mention from a technological point of view?

U. I don't think so.

S. So the philosophical end of it is apparently the key thing in the design of those machines.

U. That's right.

S. Otherwise it was a straight forward engineering job.

U. That's right, it was quite a straight forward engineering job.