COMPUTER CONTROL PANEL DISCUSSION

Chairman:	H.G. Blosser	Michigan State University
Panelists:	B.M. Bardin	Indiana University
	F. Schutte	Eindhoven University of Technology
	W. Remmer	CERN
	R.R. Johnson	University of British Columbia

BLOSSER:

I now call this session to order. I thought I would take advantage of my position as chairman and just say a few words on what I would like to see, or what I think the goals of a computer system should be, and some of the important considerations. Then I hope that the panelists will land on some or all of them. I also propose to give each panelist seven minutes to discourse on his philosophy, after which we will go into a question period between panelists and between the audience and the panel; comments from the audience will likewise be welcome.

So I wrote down what I thought of as three distinct stages of implementation of computer control systems. The first stage is where you simply put in a computer to do more or less the same things which customarily have been done by the old system--with knobs and pots and things. The possible virtues are that you come up with a cheaper system, you surely come up with a faster system, and I think now everyone is convinced that you come up with a more reliable system. Certainly much of the present focus on implementation of computer control systems is in that stage I area, where the computer just sets things for you and then you go on and run the cyclotron much as before.

But there is a very important stage II where the computer control systems have the potential of taking us way beyond existing control systems by providing much more sophisticated feedback to the operator. In particular, at our cyclotron I always find myself running a chart, taking a turn pattern, counting turns, hauling out a ruler to measure spacing, sitting down to figure out the centring, and all that taking five minutes or so. With a good computer control system you could picture turning a knob and having the computer back there calculating centring just as fast as you turn the knob, and spitting it right back at you on a meter. I think with these much more sophisticated controls the effectiveness with which one is able to ascertain the cause of troubles will just shoot way up. Things like monitoring power supplies are sort of a beginning stage of this--where you can monitor the supply, which is a thing not usually done now, get a Fourier analysis immediately perhaps, and go on from there.

Stage III is the dream world where the operator becomes redundant, the console consists of nothing but a teletype where you type this brief program, and away you go. Stage III, clearly, is still pretty far away although I think we will get there.

So what are the major decision areas which come along in implementing a computer control system? There is, of course, the choice of the kind of computer--whether the computer is intended only for control or whether it is intended to service all the needs of your laboratory. So far we have been, I think, hearing almost all about the single-control computers, though the option of a general purpose computer is certainly also a very viable one.

The human interface is a sort of pet peeve of mine because our computer people keep on making interfaces that don't adapt very well to me. And since I am the director, you can imagine which way the mismatch is adjusted! Anyway, there are these important questions of the input and output fitting well with the person who is going to sit there in front of these things. Typewriters with me are a particular peeve--if I have to type a sentence in order to do something I am just ten times worse off than I was with the old-fashioned system. Push-buttons are a step forward, although if I have to dance back and forth between a complicated array of push-buttons in order to see two things--got to push a button here, the sixth button in the seventh line, to get the other thing I want to look at--that has its problems, too. So it seems to me very important for the person designing the computer control system to actually think, in as detailed a way as possible, about what the human is going to want.

Then, switches, knobs--knobs are still an awfully powerful thing, even if they have a computer behind them. And likewise you have the same set of options for output. Someone mentioned analogue meters this morning. I am certainly one of the people who likes an analogue meter to stare at. When you are concentrating on something, the analogue meter goes into your head an awful lot easier than a digital number up there. You can just watch that meter climb or go down.

A third sort of design area is how much you are going to use your computer as an element in various closed loops. The computer has a great advantage in closing loops in that you can do all kinds of complicated logical operations. It is totally insensitive to noise and things like that. On the other hand, the loops that demand fast response you rather clearly don't want to do with a computer. So I think an important decision is which loops belong in the computer and which don't. We have one very powerful loop, a computer hook-up that comes back and tells us resolution and keeps it going. I think the logic is so involved it would be nearly impossible to do it in an analogue way, and also you don't need it fast. So it becomes an ideal sort of loop for a computer, but clearly you don't want all loops that way.

Then you have a lot of questions on the diagnostic capabilities. It gets very involved with how much computing power you want to put into your control computer. At the lowest level the computer just reads and prints, and then you have to look at this string of numbers or something and decide whether they are good or bad. You go up a notch, redo a simple fitting and print, and then you can do things like Fourier analyses or whatever you want at the output. Or at the highest level you can do things like read, compute orbits, compare and print.

The final thing, which I think is a tremendously powerful capability, that you really ought to build into a computer control system if you are going to have one at all is the matter of super knobs. You know, knobs that do things which an analogue system just almost couldn't do. For example, one which we are trying to set up now, namely, a dispersion knob. If I have, for instance, three quadrupoles in a beam line, it is mathematically possible to have those quadrupoles change dispersion to any value I want and yet maintain a double focus at some point. But in doing that the quads have to move in a complicated, interacting way. One of them goes down or two of them go up, and so on. So my dispersion knob I would want to actually vary all three quads in this complicated way, so that the double focus stayed put and nothing but the dispersion changed. Then I would have a knob which is really tremendously better than anything I have now got in the control system.

Computer control systems certainly open the possibility for this kind of knob, and I strongly think you ought to include provision for them as the systems are being planned.

Maybe I can get the panelists to either add to or shoot at these ideas and tell us things they are interested in.

BARDIN:

Well, what I am going to say is in the nature of very brief comments on some things rather than a formal discussion. I thought when I arrived here that it would be generally conceded that computer control is necessary in the modern age. It appears there is not quite total agreement on that at this point. I think that originally people were very conservative about the way they implemented computer control systems; they were worried about whether or not the computer downtime would cause a large amount of accelerator downtime. I think most people now have had enough experience to feel that it is reasonable to expect a system to be based on a computer in such a way that if the computer goes down, the accelerator is shut down. It is simply because the reliability of these systems has become very good that one can do this. It is, of course, true that there will be some downtime associated with the computer being down, unless you have a very high redundancy in your system, perhaps similar to the way TRIUMF is doing things.

Another interesting aspect of computer control is the question of how many knobs, the question of whether you have one knob per controlled variable or simply one knob. Here again it appears there is no great consensus of opinion. But I think that at least we can say there are a number of experiments under way on both extremes of this situation.

Another question which Hank Blosser touched on briefly was the question of how many CPUs, whether you have one CPU which does everything for the laboratory or whether you break up the tasks in such a way that several different computers handle them. We clearly have chosen to break up our system, but I think that as computer architecture improves a network of computers will become more and more favourable.

One thing which I agree fairly strongly with him on is the human engineering aspects of cyclotron control. I think that if we expect computer control to be accepted wholeheartedly by the people who are actually going to operate the machine, we must take those factors into account very strongly. I have to admit that I don't think that there is any ideal solution to that particular aspect of things. We feel that the kind of interactive CRT terminals that we are using have many advantages, and that their disadvantages are fairly minor.

I would like to point out that along with the question of machine operation there is also the question of maintenance of your control hardware, program development and debugging. We have gone very heavily into the interactive approach for that. All of our program development is done using CRT terminals. There is not a card-reader there. We do anticipate using a line printer on the data acquisition machine eventually, and it has been somewhat of an inconvenience during this developmental period not to have one. Basically, an interactive terminal is a better human interface for the programmer. We feel that the development of our system has gone very rapidly because of the approach that we have taken. We also feel that on-line debugging of programs in such a way that you have virtually eliminated the possibility of blowing the control system can be done much more easily with an interactive terminal.

We mentioned cost considerations earlier today. I will just say again that I think we could replace our particular computer with one that was virtually identical for roughly half the cost today.

Our reliability for our system has been extremely good. I am not completely convinced that the reliability of computers using medium-scale integration is actually very much better than that of computers using discrete components, in our experience. But this should come, and perhaps it is simply our lack of experience that convinces us that there may be some problems in this area.

One other comment with regard to software: there are a number of quite reasonable operating systems now available for commercial computers. When we started writing the operating system for our machine, there were none. If we were buying a machine today, I think it would be possible to buy a system where the operating system would be very nearly adequate as it stood to handle the same tasks for which we wrote our own system. I know of at least three small-scale systems which have very similar properties to our operating system. Because of this you could cut down the development cost of the control system, and certainly you could cut down on the development time.

SCHUTTE:

As you may have noticed already from the talk before the break, we have quite another approach concerning automatic control than is the case normally, or you might say with the big machines. I have some slides which might give a better idea on our approach, which actually is Category II of Blosser's introduction.

BEAM PROPERTIES

$\phi(R)$	-	PHASES WITH RESPECT TO THE HF DEE VOLTAGE AT SEVERAL RADII
I_{int}	-	INTENSITY INTERNALLY
I_{ext}	-	INTENSITY EXTERNALLY
X	-	HORIZONTAL POSITION IMMEDIATELY AFTER EXTRACTION
X_i ΔX_i Y_i ΔY_i I_i	-	HORIZONTAL AND VERTICAL WIDTHS AND POSITIONS AND INTENSITIES AT A NUMBER OF LOCATIONS IN THE BEAM TRANSPORT SYSTEM
E	-	ENERGY EXTERNALLY

This is the slide which you probably recognize. We want to obtain as much information from the machine concerning beam properties as possible, to gather all these data in the computer, and let the computer (in our case a PDP 9 16K) decide whether the machine is operated in its optimal condition or not. We intend to measure those beam properties, of which the first two are measured by means of capacitive pick-up probes at eight different radii, and the next two with the position probes. This yields in total now 20 signals. Next, the horizontal and vertical widths and positions and intensities measured at max 26 locations in the beam transport system (which is max 60 m long), concerning about 60 beam properties. Finally, the energy of the external beam is measured by means of an NMR control. This adds some data. So totally we have, let's say, 85 beam properties which we measure at the moment and intend to transmit into the computer.

CLASSIFICATION OF CYCLOTRON PARAMETERS

1. PARAMETERS, SET ACCORDING TO A DEFINED FIXED VALUE
   - 1A. PARAMETERS, THAT CAN NOT OR WILL NOT BE VARIED DURING THE AUTOMATIC CONTROL, SO REAL FIXED SETTINGS
   - 1B. PARAMETERS, WHICH CONTROL A BEAM PROPERTY AND EVALUATE ACCORDING TO A DEFINED FIXED VALUE
2. PARAMETERS, SET ACCORDING TO AN OPTIMAL VALUE OF THE BEAM PROPERTY THEY ACT ON

The next slide shows a classification of cyclotron parameters which we would like to handle. You can divide into: (1) parameters which have to be set according to a defined fixed value, and (2) parameters set according to an optimal value of the beam property they act on. Now the first ones can be subdivided into: (1A) parameters that cannot or will not be varied, and (1B) parameters which actually control the beam property and evaluate it according to a defined fixed value. (1A) on the next slide gives some of the parameters which we don't want to vary: the position of the ion source, the position of the extractor, the position of quadrupole lenses, all positions that are concerned. At the right side you see the beam properties they act on.

REAL FIXED SETTINGS

1. POSITION ION SOURCE HORIZONTALLY AND VERTICALLY	BEAM QUALITY TRANSMISSION
2. POSITION EXTRACTOR	EXTRACTION EFFICIENCY DIRECTION EXTERNAL BEAM
3. POSITION QUADRUPOLE LENSES, CORRECTING AND BENDING MAGNETS	TRANSMISSION BEAM TRANSPORT SYSTEM
4. POSITION AND APERTURE OF ENTRANCE AND EXIT SLITS	ENERGY AND ENERGY SPREAD
5. POSITION AND DIMENSION OF DIAPHRAGMS	BEAM EMITTANCE AT EXPERIMENT

Now we can pass that very rapidly and go on to the next slide. This gives the so-called fixed settings, which are parameters that have to be set according to a fixed value. You see the position of the shorting plate, setting of the trimming coils B11 and B12 (which are the upper and lower concentric correction coils in the centre (which determine the median plane in the central region, the first seven concentric coils, the fringing field gradient (B10 minus the main magnetic field) which determines position and direction of the external beam. The same latter quantitites are also influenced by the extraction voltage and the magnetic induction of some correction magnets in the beam transport system. Finally, the energy is determined mainly by the magnetic induction of the analysing magnets.

FIXED SETTINGS

1. POSITION SHORTING PLATE IN HF SYSTEM	HF FREQUENCY WITHIN δF
SETTING TRIMMING CONDENSOR	HF FREQUENCY WITHIN 1 : 10^{-6}
2. $B_{11} - B_{12}$	MEDIAN PLANE CENTRAL REGION
3. $B_1 \ldots B_7$	HF PHASE INTERNALLY
4. $B_{10} - B_H$	POSITION AND DIRECTION
V_{EXTR}	EXTERNALLY
$B_{corr.\ magnets}$	
5. $B_{anal.\ magnets}$	ENERGY E

OPTIMAL PROPERTIES

1. TRANSMISSION ACCELERATING REGION BEAM INTENSITY	$B_8 - B_{10}$
2. TRANSMISSION EXTRACTION SYSTEM	B_{10}
3. TRANSMISSION BEAM TRANSPORT SYSTEM	$A_{11} \cdots A_{32}$
4. TRANSMISSION THROUGH ANALYSING SLITS	V_{DEE}
5. BEAM EMITTANCE HORIZONTALLY AND VERTICALLY	V_{EXTR}

Now the next slide gives you an idea of the optimal properties, and that is much more difficult to catalog. The best thing you can do here is to try to have an optimal transmission through the accelerating system, the extraction system, the beam transport system, the analysing slits in the analysing system, and finally, the beam emittance horizontally and vertically has to be optimal. The main parameters which determine that are the eight and tenth (outermost) concentric correction coils--their difference makes the field bump as a function of radius, the outermost one, which gives the isochronism, the six harmonic coils, the dee voltage and the extraction voltage.

FIXED PROPERTIES

- R_K MEASURED VALUE OF BEAM PROPERTY
- R_{KO} CONDITIONED VALUE OF BEAM PROPERTY
- P_J PARAMETER INVOLVED

THE DIFFERENCE $\Delta R_K = R_K - R_{KO}$ HAS TO BE CORRECTED

VARIATION P_J IN P_J CAUSES VARIATIONS IN R_K:

$$\Delta R_K = \frac{\partial R_K}{\partial P_J} \Delta P_J$$

$$\Delta R_K = M_{KJ} \Delta P_J$$

$$\underline{\Delta R} = M \underline{\Delta P}$$

INVERTING M YIELDS

$$\underline{\Delta P} = M^{-1} \underline{\Delta R}$$

M CAN BE MEASURED AND/OR CALCULATED

The next slide gives an example of how to act with the fixed properties. The main idea is that you have a matrix element, which gives the relation between property and parameter, being one of the elements of the big matrix. To find the value of the parameter involved, we make use of an inverting iterative process: we determine the difference between the measured value of the beam property and the conditioned value. Now apply the inverted matrix on that vector and you obtain a correction in the cyclotron parameters. You can adjust them and repeat the measurement. In that way the fixed properties are handled.

The next slide gives you an idea of what to do with optimal properties, which is rather obvious. You just have to make some sort of a least-square fit where you have to minimize the Q, and by differentiating it to the parameter, you obtain another set of matrix elements $(\partial E_i/\partial P_j)$. If these matrix elements do not equal zero, then you have to put a fixed step, which is divided to the several parameters with respect to the gradient involved, and this yields the correction in cyclotron parameter which has to be set.

OPTIMAL PROPERTIES

- E_I MEASURED VALUE OF BEAM PROPERTY
- E_{I0} BEST VALUE OF BEAM PROPERTY
- P_J PARAMETER INVOLVED

THE QUANTITY $Q = \Sigma (E_I - E_{I0})^2$ HAS TO BE MINIMIZED

THIS IMPLIES $\frac{\partial Q}{\partial P_J} = \Sigma\, 2\, (E_I - E_{I0}) \frac{\partial E_I}{\partial P_J} = 0$

IF $\frac{\partial Q}{\partial P_J} \neq 0$ THEN $|\Delta P|$

FROM $|\Delta P|$ FOLLOWS $\Delta P_J = \frac{\partial Q}{\partial P_J} |\Delta P|$

Now, for time reasons, the next two slides I do not want to explain because that is just a combination of fixed and optimal properties.

COMBINATION OF FIXED AND OPTIMAL PROPERTIES

- R_K MEASURED VALUE OF BEAM PROPERTY m
- R_{K0} CONDITIONED VALUE OF BEAM PROPERTY m
- E_I MEASURED VALUE OF BEAM PROPERTY ℓ
- E_{I0} BEST VALUE OF BEAM PROPERTY ℓ
- P_J PARAMETER INVOLVED n

DEFINE QUANTITY $\Delta R_{K0} = R_{K0} - R_{K(i)}$

FURTHER $\Delta R_K = R_{K(i+1)} - R_{K(i)} = \frac{\partial R_K}{\partial P_J} \Delta P_J$

THE QUANTITY $Q = \sum_{I=1}^{\ell} (E_I - E_{I0})^2 + \sum_{K=1}^{m} \lambda_K (\Delta R_K - \Delta R_{K0})$ HAS TO BE MINIMIZED

THE M LAGRANGE MULTIPLIERS λ_K ARE FOUND WITH $\frac{\partial Q}{\partial P_J} = 0 \quad (J = 1 \ldots M)$

IT FOLLOWS $\frac{\partial Q}{\partial P_J} = \sum_{I=1}^{\ell} 2\,(E_I - E_{I0}) \frac{\partial E_I}{\partial P_J} - \sum_{K=1}^{m} \lambda_K \frac{\partial R_K}{\partial P_J} \quad (J = M+1 \ldots N)$

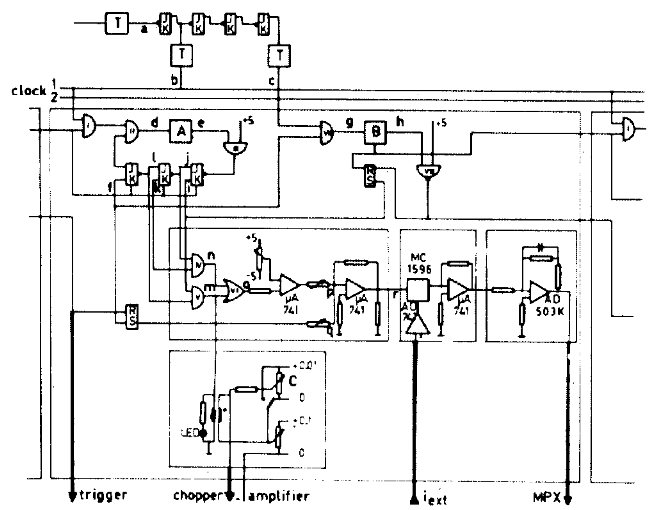
THIS YIELDS $\Delta P_J = \frac{\partial Q}{\partial P_J} |\Delta P| \quad (J = M+1 \ldots N)$

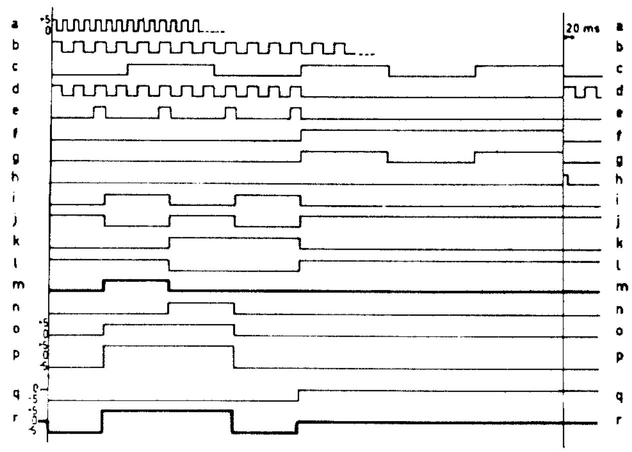
IF $\Delta R_K = \Delta R_{K0}$ THEN $\frac{\partial R_K}{\partial P_J} \Delta P_J = \Delta R_{K0}$

THUS $\underbrace{\sum_{J=1}^{m} \frac{\partial R_K}{\partial P_J} \Delta P_J}_{\text{unknown}} + \underbrace{\sum_{J=M+1}^{n} \frac{\partial R_K}{\partial P_J} \Delta P_J}_{\text{known}} = \Delta R_{K0}$

FINALLY, $\Delta P_J = M^{-1} \left(\Delta R_{K0} - \sum_{J=M+1}^{n} \frac{\partial R_K}{\partial P_J} \Delta P_J \right) \quad (J = 1 \ldots M)$

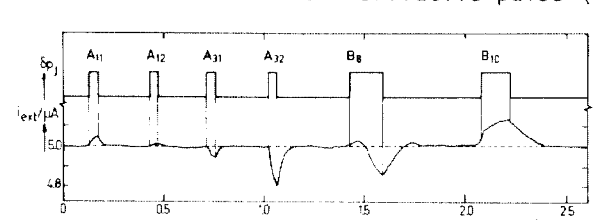
I would like to add these two slides:





The first is a device which gives the possibility to make pertubations on cyclotron parameters and to measure the response on the corresponding beam properties. The second shows the timing. We are able to give a pulse-shaped disturbance (m) on some cyclotron parameter--let's say the outermost concentric correction coil--measure the response, and correlate the response with the second-order derivative pulse (r).

The last slide gives you an idea of how that acts. It shows the pulses and the corresponding responses. The correlation of these responses with the



second-derivative pulses give correlation products which give a measure of the optimal setting of the machine. For a more detailed description see the contribution, "The automatic control of the Eindhoven AVF cyclotron I". This is performed at the moment and turns out to work well. All the previously mentioned data are to be transmitted into the computer, and that has to decide whether the system is in the optimal condition or not.

BLOSSER: Any experimental data on the last slide?

SCHUTTE: It shows experimental data, but the amplitude of the disturbances is in some cases exaggerated to show clearly the response.

BLOSSER: Anyone else want to ask something at this point?

MEADS: I notice you were mentioning a least-squares fit. I wonder if you have considered a Chebychev-type fit as, say, only a programming method? Were you minimizing the maximum error?

SCHUTTE: That is one of the possibilities, yes. Further, adding some constraints yields methods like adding Lagrange multipliers, etc. That turns out to work very well. Yes, I agree with you.

MEADS: One more point to add on that: when you use that you frequently get extremes that go in opposite directions, like adjacent trim coils that run to extremes in opposite directions, and one might then try to minimize the departure from a fixed setting. That way you might get a nice, well-behaved solution.

SCHUTTE: That is pretty dangerous. You can't skip that problem. There are certain combinations which give just a typical response. What you can do then in the pulsing or perturbation scheme which I showed you is, for instance, just increase the size of the perturbation and observe whether the response is changing. If it does, you have an indication that you are only in a relative maximum or in some sort of combination of maxima. If not, then you are on the right way. Further, our system is not designed for starting with a scratch cyclotron. We have the operator who sets the cyclotron according to his best ideas and then we can switch on our proposed automatic control which will maintain it in that way of operation, or make it even somewhat better. But you cannot start with a scratch one; then you might have especially those problems you mentioned.

MEADS: I can just add more constraints to minimize the probability of extraneous solutions.

SCHUTTE: Yes, that is possible, too.

JOHO: I wonder if you have any trouble with the matrix, the response matrix, getting singular? For instance, I could imagine the following situation: that if you have separated turns, obviously by changing the dee voltage you get a periodic modulation of the beam intensity. Let's say if you go from 200 turns to 201 turns, then I could envisage that you run into some trouble with this linear response matrix.

SCHUTTE: Yes, that is also a serious problem and what you can do concerning that is try to divide, to subdivide the main matrix into several parts that lie in some way around the diagonal, and have a closer look at those smaller matrices. Those smaller matrices can be investigated on singularity much easier; they can be more easily handled and you can make a better combination of properties you measure and parameters you change. In that way you can just check the whole system for singularities, and that is the way we have done it also. It especially appears not in the cyclotron but in the beam transport system. For instance, the variation of the isochronism and the variation of the extractor voltage appear to give the same direction variation in the external beam, and that is very dangerous because you have to invert that matrix which gives you very large numbers. You can get rid of that just by making a more proper, more suitable choice of parameters and properties.

BLOSSER: It does strike me that there is usually a big message, too, when the thing goes singular. It is usually saying the trouble isn't in the realm of things that are in the matrix. Then you just want to ring a bell.

PARKER: I thought you might be interested to know that during the long nights at Los Alamos prior to our 800 MeV beam an effort was started to write the software, to determine the transfer matrix for our system, and when that is finished we will put the machine on line that way.

REMMER:

Rather than go through the single points that we have mentioned in the introduction, I would like to make some comments on man-machine interface via computers. It is not so easy to make a man-machine interface which is effective. The operators must use them very easily, and even professors have to understand them! I think I just have to stimulate discussion, and so I want to describe a man-machine interface implemented at CERN in Geneva, and tell you what we have and mention explicitly what we don't have.

Now and again at CERN we build a new accelerator. The newest one is the booster. We have there to acquire 700 16-bit data and we control 400 16-bit data. It was decided to make that machine completely controlled via the computer. No manual back-up is provided. Therefore, we must have a good and handy man-machine interface. We do this mainly using push-buttons. We can set and control the process parameters by pushing buttons. We assembled buttons in a special console. We have several accesses--or consoles, if you like--four. For injection, for the rings and for the extraction system, and one console where we have all parameters available. A special console is built by a selection matrix. You have one button per parameter. They are assembled in 16 parameters per row. They are labelled so that the advantage is you don't have to remember the mnemonics. You find more or less immediately the parameters you like. You push a button and then this parameter is displayed. We have on the consoles four lines of 32-character displays, and we display there all the relevant information concerning that parameter. For instance, the name and the resolution, what it means--the least significant bit means 2.5V, for instance--then we display immediately the value of that parameter. And one row is dedicated to error messages. Now this parameter can be controlled. We do this again with push-buttons. To increment these parameters, we have two rows of push-buttons, one for positive values and one for negative values: +1 + 10 + 100 + 1000 and -1 and so on. So we can display only four digits, and we can very easily control them.

This can be done at three consoles and, of course, from one console which works via a keyboard and a display. There you have access to all parameters. The software approach for that is that we have a computer, a big data bank, where every parameter and its associated information is stored. We have 20 entries or information pieces for every parameter. For instance, the mnemonic name, the address of the power transmission system, where you acquire the data, where you control. Then you have the resolution there, you have limits--when you control a parameter you have not to go manually higher than a certain voltage, let's say. This is done software-wise, because a computer should be considered as a tool to cut costs in hardware.

We have other push-buttons. For instance, to call programs we have so-called program request units. You have 32 buttons, and you can call 32 programs. This is very convenient if the operator is tired; he just has to push the buttons, which are labelled. At the same time, called programs may have certain options. In output you can print the results you received, or you can display them and you can punch them in cards or you can store them for later reference, and so on. Also, to control the programs you have options--to repeat the programs, to abort the program, and so on.

Push-buttons are a common tool, as was mentioned earlier. The push-button philosophy is so appealing many persons thought about it and they did some clever hardware tricks. For instance, they still have the buttons but not the labels. The labels are on separate cards, and you have such a card in your pocket and on the card you have an identification of that card. Everybody has all cards and inserts it between the buttons so that you have any number of programs and you have your individual programs on cards. Or because operators are used to push-buttons, you even display the push-buttons on a display. And then you touch them with a finger and certain hardware finds out what program you mean. So push-buttons are very attractive.

On the console where we have all parameters available there is a keyboard and a display, and you can have different displays, both refreshed displays and storage scopes. For discussions you better have refreshed display; otherwise you can't change information. And we don't have a track ball and we don't have a light pan. We use a check list. You know by the program on how many spots you want to change the program. Let's say there are 10 locations to introduce an address or insert a value, and then we have only one button which we push to go through that list. This is very convenient.

Another point of interaction is language. You can have an interactive language in many different ways. For instance, there are arithmetic interactive languages like BASIC and FORTRAN, and for control purposes those languages are changed to make them sort of a control language. We go a different way. We create our own interactive language only by providing some commands which are easy to use by the operator like TYPE, DISPLAY, SET and HOOK. We also have knobs which you can hook--only one or two--and you can hook any process parameter to these knobs.

NEED: There are times that you will want to be able to control one parameter or in some cases two parameters while observing a third. What is the maximum number of parameters that you have under display?

REMMER: In seven minutes I couldn't mention everything! A console with push-buttons where you can select one single parameter consists of two halves, so per console you can select two parameters. We have three consoles, so you can select six parameters at the same time. Two for the injection, two for the ring and two for the extraction. Then we have a console where all parameters are available and you can select--this is done by program--any number you want. You can select elements of single parameters, or groups of elements, vectors and so on. In the data bank which co-ordinates all the activities on the computer we have a busy flag. If this busy flag is set by one console, you can't control the same parameter by another console.

NEED: You have six available. Are these all within the purview of one person?

REMMER: No, the booster is controlled in a little corner, and you have per console one person. So there are four persons who can control or choose the parameter.

NEED: If you have two available, can you display a third on one console?

REMMER: No.

BLOSSER: My bias is certainly to have a meter that I can also attach to anything.

WEITCAMP: What order of magnitude of software effort in round numbers, in man-years, was required to start from scratch, essentially, on this booster program?

REMMER: Six man-years. There are three programmers working two years.

SLOBODRIAN: I have the impression your system is not automatic; you always need the operator to interact with the system. Or can you set it on an automatic course once it is started?

REMMER: Yes, the booster was just finished, and they are in the period of running in; it is most valuable for that period to have an open loop control, so the operator makes the settings and increases and decreases the values. Certainly later we will have closed loops and optimization techniques. The three programmers are still there. They have to produce programs.

BLOSSER: Or else!

JOHNSON:

I was invited to give a statement of my views on computer control. Since it is an invitation, these need not be the views of the TRIUMF Director, nor for that matter the Controls Group.

The first contentious point is some system design problems. Several of the cyclotrons here are under development, and maybe I can tell you what I learned about system design in the last few years. Computer control approaches should be included in development, commissioning and operation. Initial installations need not be exhorbitant. One should make a statement of fact, though, that in the full run of time it costs you just as much money to use computer control as any other approach. It is not really possible to specify a cyclotron controls system completely. It is possible to specify a paper mill, an oil refinery, and something that fits within the framework of an engineering task. The research facility growth as well as technology growth over the life of the facility prohibit such an engineering specification. Therefore you have to have a flexible controls system. Now the hit is that a control system structured after the facility organization will accommodate the growth pattern. If you try to go your own way, the control system will not work. The site standards, and even the Controls Group themselves, must be controlled. Computerized control systems don't have as their basis a computer, rather the fundamental idea is digital data handling. We use Supernovas, they are well suited to the application. So are all the other 16,000 minicomputers. Digital transmission of data affords rapid, reliable, flexible handling.

The second part of the talk is examples, and that is the other side of the picture. In development we have done two significant things. Our control system is now spread over two provinces and about 1000 miles. The University of Victoria is involved in quadrupole field measurement; the apparatus that they are using there is controls equipment that will be installed on the cyclotron when there is a beam line to be installed on the cyclotron.

The other significant thing that we have done in the last year is to control the central region cyclotron. You have all seen it. It doesn't really look like a computer control system. It is not meant to. It is meant to have honest-to-goodness knobs and dials and displays.

Standard languages like BASIC and FORTRAN have been mentioned already here. We have taken the Nova standard BASIC and included CAMAC commands in it. This has one very, very important point. It means the engineer who is involved in the system development can actually write his own program to start out with. An engineer, even a scientist, cannot communicate with a program. A program does. We are using CAMAC, we are using it very heavily, perhaps more heavily than we should. There are drawbacks, but it does allow us a certain freedom. We do not have to amortize the development of this digital standard. And I would like to make a specific announcement: I'll trade two TRIUMF DAC modules for one 256-bit digital input multiplex.

The drawbacks to the CAMAC standard are: the number of interrupts are limited, there are only 24 of them. There is a way around that, described in paper N9 in the Proceedings. And trivial interfaces--input gates--are expensive.

I'll skip everything and get to the outstanding problems in computer control. I am in complete agreement with Dr. Schutte, and also support Dr. Kashy and his paper. The most significant development is actually in the set-up and tuning of the cyclotron. Scanning, limit checking, absorbing large quantities of data, is again a well-defined engineering problem and very trivial to do. The outstanding area, then, is in tuning. Diagnostics must be assembled by a computer, results dealt with in such a way as to reset the cyclotron to more optimum conditions. This requires things like phase probes. It also requires a regression technique that is rapid, and this has been alluded to before.

I can't do it. The Controls Group at TRIUMF can't do it. We are supplying FORTRAN. Again, this is something that a beam diagnostician can use with CAMAC, and he can drive the diagnostic equipment he needs. He can go away and do the regression calculations, and then if it is permitted--again, by the Controls Group, mind you--he can set the power supplies, trim coils, harmonic coils, to a setting.

MEADS: I was curious as to whether or not any of you have been dissatisfied with the speed associated with compiler-generated programs using BASIC, FORTRAN, etc. as opposed to the much faster speed you might accomplish with an assembler language program.

JOHNSON: For the last year we have been using BASIC for commissioning operations. This does not involve any significant time drawback. I would not advise using BASIC for actual control--the fundamental structure of BASIC does not really allow it to operate rapidly. However, the FORTRAN, at least on the Data General machines, does not have a significant overhead. It is about 1.5 x longer (the program in FORTRAN) than straightaway in assembler. There may be an executive problem with making the FORTRAN compatible with what you are doing. We use an Asynchronous Task Supervisor in our programming for the garden-variety things that I dismissed quickly. Beam diagnostics is a dynamic-type operation. It doesn't really fit into an asynchronous supervisor.

BARDIN: I will just simply say that many modern FORTRAN compilers are exceedingly efficient. The inefficiencies which have traditionally been associated with them have largely disappeared.

MEADS: I have another comment on that. That's what everybody gets to think, but just recently, by putting a small routine into machine language, a problem that took an hour to do on the 6600 was cut down to half an hour.

BARDIN: It is clear there will always be cases where you can do a better job because the compiler isn't as smart as you are.

SCHUTTE: I think it is a bad example. Concerning the language, Johnson has already stated that for the Supernova or Nova there is some sort of CAMAC language, and that is due to the fact that this type of computer was the first one which was linked to the CAMAC system. Now the ESONE Committee is very busy writing a Euratom report concerning CAMAC languages. When it will appear I cannot guarantee.

REMMER: I would like to make a comment on programming languages. Up to now I have heard only assemblers and FORTRAN, but for some years there has been an intermediate language available that is called P.L. There is P.L. 360, and we have in Europe P.L. 15 for the Metraquinze computer. We also have a compiler P.L. 11 for the PDP 11/45. This is compiler with a syntax like ALGOL and FORTRAN, but with the full flexibility that assembler offers. If I dare to make a guess, in several years nobody will make a program in assembler, and maybe even the FORTRAN compiler will be replaced by that.

BLOSSER: I want to comment, too. This question of where the interface goes between the systems people and the cyclotron people. You know, it is highly dependent on your personnel. We have professors who know nothing but FORTRAN and they want to sit there and develop control systems. So our computer people are asked to furnish FORTRAN instructions for each of the devices--"Call Probe" or something like that--and then when a good FORTRAN program is developed, as it is, of course, one you are going to regularly use, it is perfectly possible to hand it back to the systems people and say make it efficient.

FRASER: I wonder if the members of the panel would like to comment on the desirability or otherwise of the operation of the computer in a multi-programming capacity, and also in that connection, in the event they wouldn't recommend it, the desirability of or the importance of a hardware memory protection on the system.

BLOSSER: Multi-programming means time-sharing?

FRASER: For example, program preparation--editing and so on--as a background job, while the computer is engaged in scanning and other control operations on a higher priority.

BARDIN: Our system is a multi-programming system with time-sharing at a rather primitive level. It is a very small system so it does not correspond to a big central computing facility's monitor. But it is clear that we wouldn't be able to do the task with the system we have if we were not doing multi-programming. Multi-programming means that you keep several programs, or at least parts of those programs, resident simultaneously and execute--in this case on a priority basis--until one of the programs either terminates or exits for I/O or some other situation in which the CPU can be used by a lower priority program. In that kind of environment, of course, you *must* have a hardware protection scheme. There is no way you can maintain the system's stability without such a scheme. I might say that such schemes exist. The architecture of many small machines provides for this, and in practice it is our experience that with such a small operating system, the system stability is exceedingly good, and we do program development as a routine thing simultaneously with control and monitoring of the accelerator.

SCHUTTE: Yes, I agree with that. On the other hand, it depends very much on what view you have, what approach you have on the automatic control. If you have a system like Indiana or TRIUMF, you just need this. If you have a system like ours--we just have a rather small system--you can have just a processor which is linked to CAMAC and which has some magnetic tape device and a control, and perhaps a display. This is sufficient. You need 4K for the CAMAC system and the rest, say 16K, you need a monitor for the other peripheral devices. If you would like to add to that system, which is able to do, for instance, data acquisition of the nuclear physics experiments, then it is obvious you need some sort of multi-program or multi-processing. If you are able to have a separate computer to do that, I would prefer that.

REMMER: Yes, the question of whether a bigger computer or a system of small computers. If you have a system of small computers, you add tasks which were previously not there, for instance conversation between computers. Multi-programming is very, very useful if you have a suitable computer for that. We have several years' experience with an IBM 1800. We use a multi-programming system. We have about 15 programmers to make programs for that computer, and every 10 days the system breaks down. We have a memory protect scheme, a bit per word, but nobody cares to protect his programs carefully. Nowadays better computers are available. For instance, they have three operating modes where some instructions are simply not allowed for application programs, and they have also memory segmentation where you can't get out of your segment. These computers are more suitable for multi-programming. I don't recommend it with all machines.

BLOSSER: Bob de Forest, would you like to comment on that question? I know you have worked a lot on this.

De FOREST: I don't know. Maybe some of you people know that we have had a multi-programming system at Michigan State which I guess one can say has been doing all the things one wanted it to for some years now. On the other hand, you were talking about big machines and little machines, and I know Indiana has a Sigma 2 which doesn't have the capacity of the Sigma 7 CPU and was not designed to be as elegant as ours. I think one of the problems you run into is that when you have a little machine like the Sigma 2, or if you want to get into one of the mini-minis, the monitor has so much work to do in operating a multi-programming environment that your overhead gets kind of high. So you are spending a lot of time keeping track of things and not getting much done. In other words, if you want multi-programming on a grand scale, you are already talking about a very expensive machine. I remember when the Sigma 7 went in originally it was somewhere on the order of half a million dollars. And so that is already a big machine, and it was justified on the basis of the scientific programming it could do. I hope that says something about the multi-programming problem.

BARDIN: I would just like to say that our system overhead is on the order of 2%.

JOHNSON: I would like to contribute. It is obvious that we are not quite in favour of multi-programming. Let me point out the very human aspect to it. When I am operating a control system, I do not want any software freak within a mile of me!!

BLOSSER: I'll toss in the comment there, though, that with a proper program, a proper monitor back there or something, some super thing, you are protected.

SLOBODRIAN: All information that is furnished by most of the measurements circuits comes in an analogue form, if I understand you correctly, and then you transform it into digital. I know that there are some control systems which have been developed for other purposes where what one does is a symbiosis of an analogue computer with a digital one, particularly when one wants a fast response of the system.

I believe some of these control programs may be rather sophisticated, and it may take quite a bit of time to come to answers for parameters that have to be changed quickly. I wonder if you could comment on that?

JOHNSON: The outstanding example was Dr. Schutte, who gets to talk after I do. His phase probe has a great deal of information in it, and really as it stands now it needs to be interfaced to a computer. In the meantime, the information that is presented in an analogue fashion to the operator is enough to get the operator going. That is an analogue computer. The fast multiplexing and handling of data is a longer-range program, as described by Dr. Blosser.

SCHUTTE: Perhaps I may add that according to my opinion you don't need a hybrid computer for these purposes. There is another thing you should take care of, and that is the time constant of the closed loop or total system. For instance, with the mentioned phase system you might gather the information as fast as you would like to, but if you closed the loop and want to have a trim coil set to another value, you just have to wait for, let's say, at least one second until you can rely upon your new data. So it doesn't matter so much if you have a very fast system.

SLOBODRIAN: I still have a question for Dr. Bardin. You said that the record of your computer--you have a single one if I understand correctly--was extremely good. But what I would like to know is what are the statistics on that extremely good performance.

BARDIN: What particular aspect are you interested in?

SLOBODRIAN: The downtime of the computer in these two years.

BARDIN: There is a regularly-scheduled maintenance period once per week for normally-scheduled maintenance. I would say that, in terms of CPU failures, we have had zero. We have had some difficulty with the teletype. Of course, that is the usual component that fails. And a few troubles with disc errors. But other than that it has been remarkably good. In terms of memory parity and CPU errors, it has essentially a clean slate.

BLOSSER: Could I interpose a question for the audience? How many people starting now to build a cyclotron would feel badly about having the cyclotron off when a computer was off? And how many people would be not bothered? I see, it comes out very even, interestingly.

RICHARDSON: The time which was mentioned just a minute ago is the thing that worries me about this business, i.e. the time of response to an action. A response time of 1 sec is just too damned slow. An operator should be able to get information at a human reaction time rate, which is something of the order of 1/20 of a second. One second is just too damned slow!

BLOSSER: For changing a magnet?

SCHUTTE: It seems to be fairly slow, I agree on that. But, for instance, if you change isochronism in a machine or want to optimize that, and you change manually the knob of the corresponding correction coil, then you don't get information, if you obtain the phase at the outermost radius, within 1/20 sec. You just have to wait until the magnetic induction is set according to the value you have.

RICHARDSON: You obviously have not done that experiment. The reaction is extremely fast.

BLOSSER: I think it is such a function of how your trim coils are buried in copper.

SCHUTTE: Our experience is that the trimming coils are much slower than the harmonic coils. The time constant of the harmonic coils is on the order of 0.1 sec and the time constant of trimming coils is 0.4-0.5 sec, and that is the reason why I mentioned that time.

SLOBODRIAN: I still have a question concerning the Stage III that you mentioned. I think that only then would I say the computer is 100% useful. But then there are several questions that come to one's mind, particularly in optimization. The computer will act always in a series fashion and that may become very slow in order to optimize. Our human mind works in parallel, and a good operator sort of watches simultaneously a large number of instruments and probably comes to an optimization criterion which will avoid going through certain procedures that a computer will have to check in a sequence. Like any program you may, of course, have branch points, but it may be very slow in the end. I would like to hear some comments about optimization in the Stage III sort of situation.

BLOSSER: Maybe I'll make one, and Martin Rickey wants to make one, too. I feel it depends so much on whether you have arrived at a proper recipe. If it is just a matter of doing a sequence of things that you know, the computer will certainly do it faster.

SLOBODRIAN: If you know the sequence.

BLOSSER: Yes, it is when you have these situations that haven't been thought of that the human brain is so much better.

SLOBODRIAN: Assume that one constructs the machine to be computer controlled right away, and you have no parallel buttons except the computer. Now suppose the machine is built in a Stage III situation and it is the very first machine of its kind. Do you think that one can let the computer find this optimum?

BLOSSER: I certainly don't think so.

RICKEY: As a matter of fact, we are in the process of doing that in Bloomington right now, and I might say that I am of the old school. When I went to Bloomington in 1965 it would have taken me six months to add 2 and 2 on the computer, and I progressed enormously--I can do it in three months by now. Had I voted in your little election a few minutes go, I am afraid I would have voted against the computer as far as shutting the machine down goes. On the other hand, in the talk which you gave a couple of days ago, I think there is an excellent point for the phase III, for the closure of slow loops, slow intricate loops, and one of the things you mentioned, which to me is very much the state of the art for tomorrow for cyclotrons, is in flat-topping. There is, of course, the very, very deep question as to whether flat-topping is practical from the standpoint of regulation of amplitude and of phase of the signals, and my own feeling, after considerable thought and looking into the alternatives, is that the only possibility of its being viable, of its being useful, is by means of a computer and the sophistication one has at one's disposal through a computer. I don't think it is practical without closing a slow feedback loop through an exceedingly intricate device.

BLOSSER: Anyone on the panel want to comment on the question?

SCHUTTE: Concerning reaching the optimum, that is fairly difficult to do, I agree. That is the reason why I stated that you shouldn't start with a scratch cyclotron. Just have a cyclotron which is already almost set, and if you have that as a starting point, then it should be possible in this way to maintain the optimum, or even to find the optimum if you aren't quite there. If you are not in that region, then it is very, very difficult. I think you can't succeed.

BLOSSER: I think it hinges on the fact that you put into your proposition that it was a brand-new machine. Then I can't dream of getting it going or developing the programs without that human interaction. But then, you know, you should be able to advance.

SLOBODRIAN: All right, but then it means you need a parallel manual possibility, the possibility of manual access to the machine, even if it is through a computer. I mean these buttons which you may sort of touch on a screen.

BLOSSER: Yes.

BARDIN: I think that is the basic irreducible minimum of any computer control system. Everybody that I know of is doing it.

SLOBODRIAN: Yes, I agree with you but Stage III was one where you would simply instruct the computer of your beam energy, your beam resolution and your current. In that case you should touch nothing.

BLOSSER: That's after I have been using the machine a while.

NOMEN NESCIO: In this business of strategies for optimization you have sometimes the matter of beam on a surface, and you will find a local maximum in this surface and the real maximum goes somewhere away. A lot of the matrix inversion things restrict you to the area of your starting parameters and don't permit this extra step out of this. Can anybody comment on this? You may start at a fairly optimum region but the one that you want is through a valley or something else in the response.

BLOSSER: What do you do to recognize this?

NN: I don't like my results.

BLOSSER: But if you can answer that question, then you can proceed to program it.

SLOBODRIAN: In any multi-parametric search you have a basic grid with a certain coarseness and then you search over this. But this is a very long procedure. There are some of these routines for search which may take half an hour before you find out the gridding.

SCHUTTE: Yes, you should be fairly sure that you are in that region if you start the machine for nuclear physics experiments or whatever you like to do with it. If the machine is set, then you can start the perturbation scheme, which I briefly gave, and to be sure that you are in the right optimum, just increase the height of the pulse which perturbs. That gives you a fairly good idea of being in the right or the wrong optimum. You are only allowed to do that during that time. If you have a machine delivered to the experimentalists, you only are allowed to make such small perturbations that the responses are in the order of magnitude of the instability of the ion source. Then the experimentalist can't decide whether "I am disturbing the cyclotron or the ion source is performing a little badly."

BARDIN: I am not sure that the goal of control on the cyclotron should be to be able to run when the operator is asleep. I think that you still want the human being there to make higher-order judgements. Now it is true that you may be able to do a very good job with the sophisticated kind of systems which Dr. Schutte has been discussing, but you still will have an operator there who can look at the situation and see if there is something grossly wrong with what the computer is doing. Therefore, perhaps a rather simple system may achieve the same ends.

SCHUTTE: It is just adding a human aspect--in these times you shouldn't fire an operator.

BLOSSER: I was going to say a minute ago that in these times we *need* to fire the operators!

HEYWOOD: The computer apparently is a bogeyman in the system, whereas if you look at the failure mode that is going to cause you the most trouble, in our system at least, it would be a DAC. If the computer fails and the DAC doesn't fail, the power supply retains its value, and the beam continues to flow. Whereas if the DAC controlling a bending magnet fails, the beam is through the pipe. So I think there is perhaps too much emphasis on system failures due to CPU problems.

Secondly, there is very active research in the optimization of multi-parameter systems. Very sophisticated methods of functional minimization and so on for multi-parameter systems are actually being realized in areas other than cyclotron control. And if I may say, I think that most cyclotrons are designed and built by physicists, and they perhaps may not be aware of the activity that is going on in the control area with respect to digital computers and their implementation.

BLOSSER: Dave Judd gets the last word.

JUDD: I would just like to point out to the chairman that in taking his straw vote he neglected to use the proper weighting factors. Because I believe that it is true that in a well-run laboratory if the Director is unhappy, almost everybody is unhappy whereas the converse is not true!