Interviewee: Stanley Gill
Interviewer: Henry S. Tropp
Date: May 16, 1972
Repository: Archives Center, National Museum of American History

TROPP:

This is... May the 16th, 1972. This is an interview with Mr. Stanley Gill of PA Management Consultants Limited, 2 Albert Gate, Knightsbridge, London, S.W.1. The discussion is being held at the Holiday Inn in Atlantic City, during the Spring Joint Computer Conference of 1972. As I indicated, Mr. Gill, I think probably the best we can start with a general description by you of your involvement, and how you got into the field, your background, training, and things that interested you prior to the 1940's.

GILL:

I guess I first got interested in computers when I was doing a Master's degree at Cambridge towards the end of the Second World War. I'd belonged to a little student society which paid a visit to the Mathematical Laboratory in Cambridge, the one that was later directed by Wilkes, but I think at this time Wilkes hadn't arrived. He was still working on radar. And about the only tool that they had in the laboratory was the differential analyzer, mechanical one…

TROPP:

…modeled after the Bush--modeled after the Bush one.

GILL:

Yes, in fact it was known around the lab as the Bush machine.

TROPP:

mhm.

GILL:

But this intrigued me and then right at the end of the war, we had a regulation in those days that students were allowed two years' study and after that they went on war work. Well the war had finished, in fact, by two days but nevertheless the regulation hadn't been rescinded so I was sent on war work. And about that time, the Mathematical Division of the National Physical Laboratory was just being founded by John Womersley—

TROPP:

I'm sorry, would you repeat that name?

GILL:

John Womersley,

TROPP:

Oh, Womersley, yes.

GILL:

W-o-m-e-r-s-l-e-y, J.R. Womersley. He's dead now, I'm afraid. He emigrated to the States around the mid-1950's but he died about five years later. He was setting up this division. Turing was a member of it, and they were looking for one or two people to work on electronic computing and I was, of course fascinated by this so I indicated swiftly that this would interest me very much. But due to the inevitable hitches in bureaucracy and especially at that time when the whole country was in chaos, the war suddenly having ceased; it was about one year before I finally managed to get an appointment there and when I did it wasn't in the, it wasn't directly in the electronic project. The post that I had been interviewed for by Turing went to Michael Woodger who's another guy that you need to interview, I guess.

TROPP:

Right. He's been talked to by my predecessor on the project.

GILL:

Oh, good. And I joined the punch card section and I worked on punch cards for six months or so, or maybe a little longer, but by that time, Harry Huskey had arrived, and the electronic project began to get moving a bit more swiftly.

TROPP:

So, this would be early 1947?

GILL:

Early '47, that's right. Yes. And I think it was Huskey, in fact, who got me recruited into the electronic project. I worked very closely with him for a few months, but his attempts to get a machine constructed swiftly were balked by the bureaucracy of the place and that was aborted. And I stayed on, working under, I think it was H.A. Thomas who was the head of it towards the end until I then had an opportunity to get back to Cambridge and study a bit longer. So, I did that; I went back to Cambridge in mid-'48 and read physics for a year. And at the end of that I felt I'd like to stay on at Cambridge and do some research. Fortunately, at that time Wilkes had succeeded in constructing the EDSAC (this was the summer of '49) and it had just come into operation, and I was very fortunate in getting a grant to study, do a research degree under Wilkes. And I arrived just at the critical moment. The EDSAC had just begun to operate and we all set to work writing programs for it. It was very exciting in those days because you could write a an entirely new program with several original features and about twenty instructions and have it operating. Well, I wouldn't say operating very quickly because it depended on the machine of course. Sometimes the machine would be out of action weeks on end but certainly it didn't take so long as it takes sometimes now, and we managed to make a lot of progress in a few years.

TROPP:

The book that you wrote with Wheeler –

GILL:

Wilkes, Wheeler, and myself, yes. This was a result of some summer session courses that we began to run. The first one was in 1950, but it was clear then that we needed a text to use with the course and... we wrote it up and got it published by Addison and Wesley [?].

TROPP:

So, you were not only writing programs, but you were also concerned then with training people to write programs…

GILL:

Yes.

TROPP:

…in the future and...

GILL:

Yes.

TROPP:

Let me back off totally then. Let's go all the way back to Cambridge. Who were some of the people that you worked with in mathematics during your undergraduate period at Cambridge?

GILL:

I can only recall one or two and I don't think any of them really went into computing in a big way. There was Joe Manning, who I think is involved in computing for some boot or shoe company somewhere in England now. There was Ralph Tomlinson who is now Operational Research with National ____ of England; Walter Hayman who is now Professor of Mathematics at Imperial College; Roth, who is also I think at Imperial College now and a very well-known researcher in number theory I think.

TROPP:

I would gather that most of your work in Cambridge then would have been mostly in pure mathematics?

GILL:

Pure and applied. I just took the normal mixed, mixed degree which was both pure - well, with an

emphasis on pure, yes.

TROPP:

There was no computational emphasis?

GILL:

None at all.

TROPP:

So, your first involvement in computation then came with the knowledge of the electronic—

GILL:

Well, I should say there's one exception, yes. We took a course in statistics which wasn't regarded by the Cambridge people as a very respectable subject. It fact it was taught by a man from the Department of Agriculture, I believe, but it was one of the requirements of the Government that mathematics students should learn statistics during the war period.

TROPP:

[Laugh]

GILL:

And that in fact, was my first introduction to computing because we had Brunsviga calculators and we actually, did some light calculations.

TROPP:

You heard the discussion earlier this evening about Henry Wallace and his work at Iowa in 1922 and 1923

GILL:

Yes, and that was essentially in the College of Agriculture…

TROPP:

I see.

GILL:

…at Iowa State, and this led ultimately to one of the major statistical departments in the United States, now in existence at Iowa State which grew out of that College of Agriculture and Henry Wallace's need for statistical work in the State of Iowa during that period; development of hybrid corn and various projects. So, the College of Agriculture isn't as farfetched as it may seem.

GILL:

No, true, true. And one of the big users of computers in the UK is the _____Experimental Station north of London…

TROPP:

Mhm

GILL:

…which is an agricultural research station.

TROPP:

When you joined the electronic project in '47, could you describe in some more detail the state of the project in that period?

GILL:

I never—

TROPP:

You mentioned earlier…

GILL:

Yeah.

TROPP:

…Turing's grand design.

GILL:

Turing was put in charge of this and I never understood whether people seriously expected Turing to manage the project and produce some results, because he was a brilliant guy but obviously not the kind of guy to put in charge of a project, partly because he was a rather prickly person to get along with and easily upset people, so that he would have found it very difficult, I think, to keep a project like this going happily.

There was already when I joined it, an arrangement with the Post Office Research Station at Dollis Hill, who were in some way responsible for the electronic design. That, I, I have no idea what the contractual terms of that were. What happened eventually was that the two organizations drifted apart. But the Post Office continued to develop a machine to a very similar design which they installed eventually in a Ministry of Supply establishment in Malvern near the old TRE, which is now the RRE, the Royal Radar Establishment, and it was called the MOSAIC, M-O-S-A-I-C.

TROPP:

Yes. I've run into that name I didn't know anything about it.

GILL:

Yeah. Well if you come across that you'll find that the logical design was very similar to Turning's, called the old ACE, but the engineering was peculiarly Post-Office type, very thoroughgoing and robust. I think they piped three-phase clock signals all around the machine and derived the appropriate phase that they wanted by mixing these three phases.

TROPP:

Lots of safety factors.

GILL:

Lots of safety factors, yes, and it worked, and it worked in fact several years before the Turing's original conception of ACE worked, which the NPO did finally achieve many years later, when all the people had changed but the project was still grinding on. It finally worked, I think, about 1957 and it really was an enormous installation.

TROPP:

Could you characterize the ACE in terms of its, what you saw as major components on its over-all design? Well, for example, let me start with a simpler question. Who would you credit with--what individuals were most involved, most responsible for the over-all architectural and philosophical design of the ACE?

GILL:

One individual: Turing.

TROPP:

Turing.

GILL:

Turing, yes.

TROPP:

So, this in a sense is almost then a direct realization of what we think of as the Turing machine. It was—

GILL:

The Turing machine, of course - what we now know as the Turing machine is abstract idea derived from Turing's original paper of 1936, but Turing realized that that was a little too inappropriate for practical construction. Nevertheless, the machine that he actually did design at the NPL was much more primitive than designs that other people were working on. It did not have the kind of direct instructions that other people were developing. Other machines had distinct instructions for adding, subtracting,

multiplying, and so on. Instead of that Turing devised instructions which were much more directly related to the primitive operations inside the machine itself. And a typical instruction was simply to transfer a word or a string of words from one delay line to another delay line and it was up to the programmer to worry about the phasing of words in the delay lines to arrange for shifts to be introduced to get words into different position or to put them into a short delay line, to hold them, and to transfer into the right place in another delay line, and so on. The basic instruction was simply a transfer from one point to another point.

TROPP:

mhm.

GILL:

Some of these points had special facilities like adders or shifters attached to them and you carried out your processing of information by directing words to places where they would go through one of these processes. So it was really a very primitive kind of instruction code.

TROPP:

Turing conceived the concept of the philosophical-architectural outline…

GILL:

mhm

TROPP:

…in terms of the realization of its components. What role did Turing play in the direction of their construction, their design, their engineering; or were there others involved?

GILL:

Well, Turing liked to pretend that he was master of this project, and I was never quite clear just what the intended relationship with the Post Office was. Since Turing was a prickly character, perhaps that explains why the Post Office eventually drifted away and carried on on their own. But when I joined the project, as I say, I was brought in by Harry Husky but I remember at one stage I found myself being directed by Turing. We were all then trying very hard to make a delay line work, which was a rather

essential achievement on the way to producing the computer, and I don't ever recall myself actually getting to the point of getting signals coming out of the far end of the wretched delay line.

TROPP:

[Laugh]

GILL:

We were forever trying to put them in. One of the problems, of course, was the temperature control of the lines in order to keep the velocity of sound sufficiently constant to be able to get the pulses coming out when the clock was expecting them. And in typical British fashion, I suppose, we decided that it would be simpler instead of controlling instead of trying to keep the temperature fixed for all the delay lines, merely to keep all the delay lines at the same temperature and adjust the clock pulse rate as the temperature drifted. And this, of course, meant that one had to give up one of the delay lines to act as a monitor…

TROPP:

mhm.

GILL:

...to control the clock. And Turing had what he thought was a very clever idea of making it control the clock. His plan was simply to inject a square wave into the front end of this monitor line and take the pulses from the far far end of the line, amplify them, clip them against, turn them back into square waves, and put them back into the front end. And he claimed that if you once managed to get this thing set up, oscillating this way, then it would remain in that state stably, producing a clock pulse of the right frequency, because obviously if the temperature of the line changed so the pulses would have to get further apart or closer together, so they still fitted into the line.

And he explained this to me, asked me to construct it on the bench and get it operating. He then went off on a holiday.

[Laugh]

GILL:

Well, I tried for a day or two and I just couldn't get this damn thing to work. And then I fell to wondering about just how this thing was supposed to operate anyway, because if I imagined myself as a pulse that was halfway down this line, if I was going to get myself into the right phase relationship to the other pulses, then I would need to know where other pulses were half a line length away and keep myself the right distance away from them, and I didn't quite see how these pulses could be expected to do this. So, I looked into Turing's analysis. Now Turing had done a typical mathematician's analysis in which he took a small perturbation of the wrong frequency, added this to the signal, and demonstrated mathematically that this small perturbation would get dumped off. But looking at this question of how a pulse would know where it ought to be, I soon worked out that if pulses at one point of the line shifted in phase in relation to pulses in another part of the line, this was equivalent in terms of the perturbation theory to the simultaneous injection of two straight frequencies, one above and one below the frequency. And I then showed that these two frequencies would reinforce each other and would not get dumped off.

TROPP:

In a sense would be added into what was already—

GILL:

Exactly, yes. Right. So, I satisfied myself that I couldn't really be expected to make the circuit work at all and when Turing came from his holiday, I presented him with my mathematical analysis, which he couldn't refute. But he was so annoyed with himself that he refused to speak to me for two weeks after that.

TROPP:

[Laugh] How did you ultimately get the delay line so that you were getting some output?

GILL:

I think that people, in fact, used an oscillator which had some other stabilizing element in it and then they used a feedback circuit which made use of the temperature line in some way to control the frequency of the of the oscillator. I'm pretty sure this is what Wilkes was using in Cambridge when I first got there, but I think both Cambridge and NPO decided fairly quickly that this really wasn't a very good way of doing things and they both of them then put their delay lines into a thermostatically controlled environment and just used a fixed clock frequency.

TROPP:

Who were some of the other individuals connected with the project at that point and what were some of their contributions that you—

GILL:

The National Physical Laboratory we're talking about now or Cambridge?

TROPP:

Cambridge.

GILL:

Cambridge.

TROPP:

Yes.

GILL:

Well, I didn't really get involved in the computer project in Cambridge until '49 when I finished my physics degree.

TROPP:

Well, then let's stay with the National Physical Laboratory.

GILL:

The National Physical Lab, I came in in early '47. Michael Woods we mentioned; Harry Husky we mentioned.

TROPP:

Yes. You might just stop a minute and talk a bit about how you saw Harry Husky's contribution during that period.

GILL:

His contribution was - well, he was a mathematician initially. He was very much the practical man. He was determined that he, come to England to participate in a computer project and he was darn well going to see a computer project and if there wasn't one that was going to work there within his time scale, then he was going to make one. So, he assumed command of as many resources as he could get hold of without the establishment finding out (I was one of them) and set to work to design and build a scaled-down version of the ACE, a simplified one with a limited number of delay lines, simplify all the codes and so on.

TROPP:

You mentioned something about Turing, in his original design for the ACE, something like 100 delay lines.

GILL:

Oh yes, I forget the actual number, but it was some at that time, completely impractical number of delay lines, and a rather complicated instruction code, and Huskey cut this down to, I forget now, something like eight delay lines perhaps.

TROPP:

What were some of the other components of the ACE like?

GILL:

It was based on punched card input-output, which was unusual at that time, at least in England. All the other projects that I knew of at that time were using paper tape for input and output, which was cheaper and more convenient, I suppose, in some ways.

TROPP:

The MARK I, for example, used the stock of the IBM cards.

GILL:

Oh yes, we didn't use anything unconventional like that. It was the ordinary telegraph paper tape that most people used. Turing picked punch cards maybe because he, being accustomed to using them during his work in cryptography, but it had an incidental effect in that the pilot ACE, as it turned out to be after it had been trimmed down by Husky and then we got to it, but would have been quite a suitable commercial data processing machine. In fact, it was never used for that purpose because the instruction code was so primitive that it would have taken a little battery of software to turn it into a suitable machine for businesspeople to use. And in those days, we just didn't have the - I won't say we didn't have the ability, but we didn't have the habit of devising software of that kind and nobody, it never occurred to people or maybe they just didn't think it was worth the effort to turn it into a business data processor.

In fact, English Electric came in in about 1950, I think, to take the place of the Post Office Dollis Hill people and to help to engineer the pilot ACE. I think maybe the original pilot ACE machine itself was built internally at NPL, but English Electric then engineered a copy of it with some modifications, which they called the DEUCE…

TROPP:

Yes.

GILL:

…and this was installed at the NPL. I think the original pilot ACE is, or was at least, on show at the Science Museum in London for many years. And the DEUCE was installed at the NPL probably in about '52. English Electric made a number of copies of the DEUCE, perhaps about ten or twenty. They didn't try very hard to make a business of it, and I think the customers had to work pretty hard on English Electric to be allowed to buy one. They really were rather reluctant to be bothered with it.

TROPP:

What kind of customers were they getting, do you remember?

GILL:

Mostly research establishments and I think the various government-military type establishments.

TROPP:

Nobody saw this then as having business applications?

GILL:

No, no. Curiously enough the Cambridge machine, the EDSAC, which Wilkes finished in '49, although it was designed basically as a scientific computer and didn't have punch card input and output... that, in fact, was to form the basis of the first business computer in England, because there was a very live mathematician called T. R. Thompson, who I think had a good Cambridge Master's degree. He may have become an actuary for a time, I'm not sure, or just an accountant at least. He was Chief Accountant to a London catering firm, J. Lyons and Co.

TROPP:

Oh yes. Which you've probably heard of; and way back in '46, I guess, he persuaded the board of his company that electronic computing was the way they should do their data processing in future. They said, "O.K., go ahead." Well, that was a bit of a problem because nobody had actually built one at that time. He wrote to all the electrical firms in the country and they said, you must be crazy, or words to that effect. Even Ferranti, I think, had either not started or was still at a very early stage and was just unable to make any sort of promises at all. And I remember I was at... the NPL one day when I think Harry Huskey got the letter to comment on from Lyons to the NPL saying, "will you build us an electronic computer?" And the NPL too said, you must be crazy, or words to that effect. [laugh]

TROPP:

What was Harry Huskey's reaction? Same pretty much the same thing?

GILL:

Well, I think he felt quite strongly that they were on the right track. I remember him pointing out to me very clearly the potentialities of computers for business type operations and the possibility of using them to sort data on magnetic tape for example, which I thought was quite crazy at the time, but he was obviously quite aware of all this. But I think he too felt that it really was a little bit early in 1946 or

possibly '47 by then, for a business company really to be wanting to embark on this so soon. But T. R. Thompson was not put off by this. He said, "Right, but if we can't buy one, we'll have to build our own," And at that time Wilkes was looking for more support for his project, and so Lyons teamed up with Wilkes and they had some mutual arrangement, something which Wilkes received some funding for his project and Lyons hired one or two people who had also gone to Cambridge to gain experience with Wilkes and the design of the Cambridge machine was made available to Lyons for them to develop for their own purposes.

They hired a man called John Pinkerton, who is somebody that you might well want to talk to, John M. Pinkerton, who was one of the teachers on the Cambridge physics course when I took it in '48 and '49, but who left I think probably while I was still taking my course and was hired as the Chief Engineer to Lyons to build their copy of the EDSAC, which they were going to call the LEO, by the way - Lyons Electronic Office. And they built this machine and had it operating by probably about 1951, except that they obviously had to extend the Cambridge design... They were going to put magnetic tape onto it. They wanted a line at a time printer. They wanted punch card input-output. And...some of this they were able to do themselves. The magnetic tape they subcontracted to Sun Telephones and Cables, I think, STC... And this was very late on delivery slipped by some two or three years, in fact I think they abandoned that particular problem.

TROPP:

Of course, by then the FERRANTI was built at this point.

GILL:

Oh yes, yes. The FERRANTI MARK I had come out earlier, but again not very appropriate for business use. And for a couple of years or so Lyons had in their head office in London a very successful scientific computer, which was no use to them, so they ran, so far as I know, the first scientific computing bureau on an electronic computer, from the headquarters of a catering company in London. And they had, their clients included British Railways and the Ministry of Supply and various other - the Meteorological Office, I think used it. I remember being called to Cadby Hall to try to sort out some mysterious phenomenon that had crept up in solving some differential equations for one of their customers. And it turned out to be a fault in the hardware which was playing tricks in one of the routines that I had written.

TROPP:

Did they ever get around in this early period to applying it to handling business data processing?

GILL:

Sure, yes. I, I can't tell you exactly when. If you can see Thompson or Pinkerton. Incidentally, T. R. Thompson is now with Shell Max, which is an auditing company in London; and John Pinkerton is with ICL. I think probably by about 1953 they were in shape to use it for their own business and they had some very forward-looking applications.

One in particular, which was a historic application, was the tea shop ordering job. Lyons had a large number of small eating houses around London and they had a problem of keeping them all supplied with food and they implemented a program under which between certain hours every afternoon they had a team of girls equipped with telephone headsets and with a card punch in front of them, that would telephone in turn, the manageresses of all these tea shops in London and just check over their requirements for the following day and punch directly on the cards as they talked. These cards were then collected at the appointed time and taken as input to a program which would summarize them and apply a number of adjustments depending on information which Lyons had available in the head office, about the weather and road repair schedules for the following day and football matches and things of this kind, and come out with printouts of various kinds, one used by the guys at the warehouse who picked the goods off the shelves. Another printout for the truck driver, telling him what to deliver, and printouts, of course, for the manageresses and for summaries for the head office and so on. And this was operating successfully, very quickly. I'm sure they had it in operation by 1954 and it ran with very little trouble for five or ten years after that.

TROPP:

That's interesting because we think in terms of the simplex method and all the problems of linear analysis and operations research and the kinds of problems that I know were solved later on in this country in terms of, say, a baking company…

GILL:

mhm.

TROPP:

like National Biscuit Company, where they are concerned with their warehouse requirements and the ovens…

GILL:

mhm.

TROPP:

…and which one of their crackers or cookies are going to be left.

GILL:

[laugh] I don't think there was anything very cleaver operational research point of view about this. They made some adjustments certainly but I don't think they did any kind of optimization. What they did do was simply a very thorough sorting and, you know, data processing job. And they had it all—

TROPP:

Feeding in all this input about how a football match that was held maybe thirty miles away from the heart of London affected the tea shops.

GILL:

I think maybe you should ask them precisely what the data was. This is the type of story that got around about the program and really just how much of this they did make use of and how they did it, I think is something you should ask them. But certainly, it operated successfully. It was a beautifully conceived and managed job and...

TROPP:

You say that machine was called LEO?

GILL:

LEO, yes, Lyons Electronic Office. Now it was some years before Lyons realized that they ought to exploit this advantage that they had, the know-how they had, by going into the computing business. But after three or four years they did this and they set up a subsidiary, or LEO Limited, which went into the computer manufacturing business and it had quite a degree of success. The Ford Motor Company of England was one of their customers, I remember. The Post Office also had quite a number of their machines. In fact there are some of the old Leo machines still in operation. The Department of National

Savings has been using-- The LEO 326 was the sort of final great successful machine of that range…

TROPP:

mhm.

GILL:

…and there are several of those still in operation. It was absorbed into the English Electric Company. They merged in 19--the end of '61, I think.

TROPP:

So, LEO versions are transistorized machines by then?

GILL:

The 326 would have been transistorized, yes, yes and then,

[END OF TAPE I, SIDE I]

[START OF TAPE I, SIDE II]

of course, English Electric and LEO and several other companies finally wound up together in ICL and so it's all in a part of that company.

TROPP:

Going back to ACE, how, what would you characterize as its significant contributions to future generations of machines that came after it? I can see a number of things but—

GILL:

I find that a difficult question because the architecture was so radically different from all the other machines and I can't really think of any significant features of ACE that were carried over in terms of machines.

TROPP:

In terms of the progeny that came out of ACE. How about the people in terms of the ideas and things that they learned, their experiences on that project as they went on into future developments?

GILL:

I'm just trying to think now who it was - well there was me, of course, but I only spent a year or so with it and my real indoctrination into computing came under Wilkes at Cambridge... All I learned working of ACE really was a bit of electronic engineering. I remember it was either electronics or electromechanical because I had been working on punch card. I designed the original control circuitry for the original punch card input-output of the pilot ACE.

TROPP:

In a sense you're saying that you're finding it difficult to think of the impact that ACE as a machine had on future developments?

GILL:

Yes, yes I do.

TROPP:

It's almost like a dead end. Let's go back to the other end of ACE in terms of the impetus that got it started. What were the kinds of needs that led to the formation of a project like ACE?

GILL:

I think the same as all the other computers at that time, scientific computing demands. NPL was primarily a scientific research establishment and there were several divisions there, particularly engineering divisions who had quite sizeable amounts of computing to do, scientific computing.

TROPP:

It was the basic needs of the scientific community then.

GILL:

Right. Yes, yes.

TROPP:

Okay let's, let's skip then to - unless you care to make any more statements about ACE and some of the people who were there.

GILL:

There's one other thing that's worth mentioning about ACE and that is one of the software developments that. Turing proposed for this machine was what we now call an interpretive routine, although the name had not been coined at that time. Turing obviously had become quite familiar with the potentialities of the idea of simulating one machine by means of another. I mean that was the whole point of his development of the, the abstracting machine back in the 1930's so I suppose it was natural that he should try to apply this idea in a - well, he never actually made it run, but a proposal for a practical software device. And he put forward what he called abbreviated code instructions and he proposed that they should write a routine for the Ace which would interpret these abbreviated code instructions... at run time, during execution, so that made it what we now called an interpretive routine, but he didn't have that name.

And I remember listening to a lecture by him on these abbreviated code instructions, ACI as he called them, sometime in 1947. And that was my first introduction to the idea of the interpretive routine. The first guy to actually run an interpretive was not intended, in fact, as a production machine. It was intended simply as a prototype.

TROPP:

Was there a lot of dialogue between the groups working at NPL and Manchester, at the at Cambridge, and some of the commercial ventures that you mentioned?

GILL:

Oh yes, a fair amount. Cambridge tended to become the center for gatherings of everyone involved, and we had regular seminars during term time, nearly every I forget which day of the week it was now, Thursdays I think we had it, I don't recall exactly.

TROPP:

Pretty regular seminars?

GILL:

Well, that's a good question, and very good question, and one of the things that we recognized very early was that we were not going to be able to have the classes parallel processes just simply because the way we were doing things was just completely out of line, completely different, from the way we were processing under the old system. There were obviously some significant lags that the old system had that the new system wouldn't have and when the month end was over, as an example, we could close the records for the month and do the summary calculation during the summary processes, and be in a position to know where we stood days before the old system could do who had a bell go off which then told one of the workmen to go over and pull the card.

GILL:

Oh yes. That's right. Of course, Babbage was a great railway enthusiast too, wasn't he?

TROPP:

He may have been influenced by the same thing. [laugh]

GILL:

Maybe. [laugh] To get back to my arrival which was the following month, EDSAC had begun to work. David Wheeler had immediately spotted that one needed to have a good loading routine for putting instructions into the machine. I don't know whether David was responsible for the original routine – they were called the initial orders at that time because it was a little routine that was fed in initially before the machine did anything else. They were wired on (Stryber?) switches and the first thing that happened when you pushed the start button, these switches would click around and feed the these set words in the first few words in the –

TROPP:

You mean like putting the coefficients in for an equations…

GILL:

Right.

TROPP:

…this pair constants of some sort.

GILL:

Yeah. That's right, they fill the first forty words or so of the memory and then the machine began to obey this wired-in program. The first set of initial orders that they had wired in had been very crude and the instructions I think were - they might have been in decimal, but they certainly didn't have any arrangement for putting in parameters or coding, functions from a mnemonic code or anything of that kind. And David realized that there were great possibilities by doing a little bit of translation by these initial orders, and so he convinced Wilkes to let him redesign the initial orders and have a fresh lot wired in. And, he'd written this routine, had left a very scrappy page of notes on what it was supposed to do, and nobody was, quite nobody quite understood all of what was in David's mind, but they trusted him and they wired them in. And David had then gone off on holiday. In fact, he belonged to a little club in Cambridge specializing in Scottish dancing…

TROPP:

[laugh]

GILL:

…and they were off on a tour of Norway, demonstrating their Scottish dancing. And I arrived as a new research student, with no idea what it was all about, and I was given this page of notes and a copy of the instructions that had been wired in. I was told, "The machine is now working again; we want these tested. S write some test routines and see if they do what they're supposed to do." Now David is - well, explanation is not one of his strong points... and spelling is also not one of his strong points. These notes of his were very difficult to follow. There was no indication of why he wanted the instructions to do what he said they should do, though of course we all appreciated it very much later on. But nevertheless, I managed to work out what he meant by this and I checked the program and of course it was quite correct and did what it said it should do. And so, we carried on and used them; and then David came back and we all got to, to work together to start exploiting the machine.

TROPP:

Do you remember what the characteristics of the loading procedure was?

GILL:

Well, two main characteristics. One was that it accepted a mnemonic code for the function part of the instruction and it provided for parameters to be incorporated in the address part... as the instructions were loaded. And, of course, the address was in decimal form instead of binary,... so it practically was a much more advanced loading routine than most of the machines were using right on up to about '54 or 5.

TROPP:

Mhm

GILL:

and David conceived this at a time when nobody ever had actually used a stored program machine for more than a day or two.

TROPP:

Who were the people on that project who were doing the main traveling between the U.S. and the United Kingdom at that point in time?

GILL:

Wilkes himself. Possibly Bill Rennick, who was the chief hardware designer under Wilkes, W. Renwick, Rennick. Unfortunately, he died last September so, one of the reasons for pressing ahead with this project.

TROPP:

Yes, that's right. I guess my question is really leading up to the characteristics in the sense EDSAC was the first realization of what was going on at Princeton, and like most of the realizations of the so-called Princeton machine or the Von Neumann machine or whatever anybody calls it, is the fact that everybody had their own wrinkles.

GILL:

mhm.

TROPP:

I mean in principle you had this over-all architecture.

GILL:

Yes.

TROPP:

But what were some of the main wrinkles, in a sense, that you saw going on in EDSAC when it was operative, that were different and unique to the work in Cambridge?

GILL:

Well, I guess this loading routine of David Wheeler's was one of the most unique features. But this wasn't a hardware feature at all; this was purely a software feature.

TROPP:

mhm.

GILL:

I remembered when I first looked at the EDSAC and compared it with the ACE that I had been familiar with I was struck by the comparatively little attempt that had been made at Cambridge to optimize speed wise. Obviously, there wasn't much point in those days in any case in squeezing the ultimate in speed out.

TROPP:

That's interesting because Von Neumann did have a thing about speed. He wanted more speed…

GILL:

mhm.

TROPP:

…because of the ability to get better and better approximations for certain kinds of problems.

GILL:

Sure, yes. Well, Wilkes said, "To hell, let's make the damn thing work and we'll worry about speed later on." So although Turing's design would enable one, if one were clever enough, to make very efficient use of the time by hacking some kind of transfer into every cycle of time, nevertheless, Wilkes was quite happy to design all the codes such that they may well be several cycles during which nothing is happening and we're waiting for a number to come out of the delay line or something. And the instruction code was designed much more directly for the programmer to use so he didn't have to worry himself about the precise arrangement of words in delay lines.

TROPP:

Which may be one of the reasons the machine got finished so soon, is he was concerned with getting a running machine and not worried about other factors.

GILL:

Well, hardware-wise it didn't make it any easier. In fact, the hardware design of the ACE was probably simpler than that of the EDSAC. But of course, Wilkes and his people were much more practical when it came to getting on with the job. Turing really had no idea how to drive forward a project of this kind and that's probably why the NPL thing didn't move fast. It was certainly more convenient software-wise though. One obviously, with all the knowledge that we now have of software, the ACE would have been a wonderful thing for a programmer to get his teeth into and show how clever he could be with software and make it really programmable. In those days this wasn't conceived of, and the fact that the EDSAC instruction code was much more directly meaningful to the users meant that it was possible to get ahead with using it and learning about software much more easily. So that is one of the reasons why the EDSAC became the first real software center of the world.

TROPP:

mhm.

GILL:

There is, of course, a great argument between Cambridge and Manchester about which machine was working first. I guess probably Williams and Kilburn did have their hairy experimental setup operating pretty much before—

TROPP:

I wasn't thinking of that conflict so much as the fact that EDSAC is really the first realization of this particular type of machine from the standpoint of its over-all design.

GILL:

Well, I would have thought that the Manchester machine was very similar.

TROPP:

Very similar.

GILL:

Oh yes, yes. The logical design was very similar. It had a different kind of memory, of course, which meant that it didn't, the question of whether to use the kind of logical design that Turing adopted didn't arise. Turing's design was really very much built around the delay line memory. The Manchester machine had electrostatic cathode ray tube memory, and so it was more sensible to design directly to the kind of instructions so the programmer would naturally think of. What Wilkes did was to use the same kind of instruction repertoire but to use a delay line memory and accept the wastage of time that one would get with this kind of memory, with this kind of architecture.

But I think Manchester machine was probably working, very poorly though, some months before the EDSAC. But it was it was never a healthy machine. It was, engineering-wise, it was much too marginal. It was i

GILL:

Pretty regular seminars.

TROPP:

Pretty open to people from…

GILL:

Sure.

TROPP:

All the other projects?

GILL:

Yes, yes. There was quite a bit of rivalry, of course, but Kilburn, in particular, tended to be standoffish.

And Turing was a bit prickly and Wilkes and Turing didn't get on too well. I think, in fact, Turing had applied for the position that Wilkes got, so that was---

TROPP:

[Laugh]

GILL:

But, nevertheless, these meetings did take place and they mixed together pretty freely.

TROPP:

So, Cambridge then was pretty much the mixing pot?

GILL:

Yes, yes. Mainly for geographical reasons. It was sort of nearer the middle of London, I suppose.

TROPP:

The atmosphere, I'm sure, of the campus had a lot to do with it.

GILL:

Yes, yes, right.

TROPP:

Now--

GILL:

And there, there was a meeting of the Royal Society which you must have come across the proceedings of this too of which all of these projects were described. Exactly what the date would be, I guess, probably the spring of '49.

TROPP:

Spring of '49?

GILL:

I think so....

TROPP:

Okay. I'll check the proceedings during that period. I can check them for a couple of years through the---

GILL:

Mhm. It was a symposium and took up half a day, maybe a whole day, and there were talks about—

TROPP:

And it was held in Cambridge?

GILL:

No, in the Royal Society - this one was - in London. Wilkes was there and Kilburn and Williams were there, and Utley had a paper and probably—

TROPP:

I may have that. I guess it's just maybe the name escapes me at the moment. Well, let's get on then to the programming aspects and get involved—

GILL:

Can I, before I forget,

TROPP:

Yes.

GILL:

…just speak of another feature of the Manchester machine which was rather crucial. And I'm not clear myself what the original reason was for introducing this, but Manchester was the first to introduce what are now called index registers. And at that time, the index registers, or register, I think they only had one I don't remember was called a B tube.

TROPP:

Oh, yes, Jack Good talked about this.

GILL:

Right. They had separate tubes for the central registers of the machine - the accumulator, the control register, and so on - which were known as the A tube and the C tube. And then, for some reason, which I still don't understand fully, they introduced another register which they called a B tube. And, its

function was to hold a number which could be added to an instruction as it was obeyed. I, when I heard about this I didn't appreciate its implications. The man who pointed out to me the significance of it was Jim Wilkinson, J.H. Wilkinson of the NPL. And he felt this really was a very important development, and he was right.

They were known as B registers in England for many years after that. Ferranti was still calling them B registers right up to, oh, to the ATLAS I guess.

[End of Side II]

Wilkes added a B register to the EDSAC in about '54 probably. So, software, what did we discuss so far?

TROPP:

Well, we talked very briefly about the book and about your involvement, and I think, how you got into software area and beginning of the programming course, the kinds of things that you felt were necessary in terms of training and instruction of future people who were going to have work with these machines.

GILL:

Mhm. I regarded it as just great fun in those days. I must confess I didn't think very deeply about the needs of the community and the future development of the subject and so on. I could see it had great potentialities but I was only a research student and I assumed that people who were older and wiser than I am, I was were worrying about these things and doing something about them. It was just fun to play my part. We were very busy during the end of '49 and '50 in writing little routines to compute all the standard mathematical functions. And I was developing these diagnostic routines, of course. And we ran these summer courses, I suppose because we thought we should. I was asked to do some of the lecturing, which I enjoyed.

TROPP:

Did this start about 1950?

GILL:

Yea, the first one was in September 1950 and I think I gave - or did Davy Wheeler give it - David and I did most of the lecturing on programming. Hartrey, I think, did a bit. Eric Munch probably did some.

TROPP:

One question I guess (pardon my ignorance), did EDSAC have built into it functions which the programmer could call upon?

GILL:

No, there were no microprograms or built-in sub-routines at all. The only thing that was in any sense built in in the way of program were the initial orders, which were wired on Stratler switches and loaded into the delay line—

TROPP:

Each mathematical function that you wanted to generate an approximation on had to be—

GILL:

Had a routine which you had to include in your program. And Wilkes was very keen on this idea of the library of subroutines. It almost seemed to be an obsession with him. The idea of a library became very much the Cambridge idea and Wilkes wanted to see this as one of the contributions to the subject being made by Cambridge, to such an extent that I remember - I think it must have been in the second summer school probably, in 1951 - I had spent some time and trouble in developing what I thought were some very good exercises for the students. We divided the students up into little groups, of course, for the practical work and I had worked out some little tasks which were a little bit complex for one man to program in the available time, but I had indicated how the job should be broken down into perhaps five or six sub-routines and I had suggested specifications for these sub-routines in such a way that they should interlock correctly and the whole thing could be put together and make it work. And, I'd written all this up with a little write-up for each member of the team and I felt very pleased with this. But before we ran the summer school Wilkes insisted on looking at these exercises and he was very upset when he saw the way that I had designed these, because it was giving people the wrong idea entirely.

"The idea of subroutines," he said, "was that you should be able to have standard subroutines in a library so that people wouldn't have to keep writing them over and over again for every job; and you should not encourage people to write their own subroutines." So, my exercises were ruled out of order and I wasn't able to use them.

TROPP:

That's interesting because as I looked over some notes of Jack Nash's, of course, he was teaching at the University of Illinois in the same period.

GILL:

Mhm.

TROPP:

He was doing similar things to what you're talking about designing,…

GILL:

Yeah.

TROPP:

…and he was talking about contests that they had from class to class about particular subroutines, like the square root of two to so many places, which finally some student set the record for in some few number of steps.

GILL:

mhm.

TROPP:

And the challenge to students was to try to accomplish this job in the shortest possible, most compacted way.

GILL:

Yes,

TROPP:

And essentially, as I look and think back over the notes that he has since given me, he was doing essentially the same thing in design that you now say Wilkes rejected.

GILL:

Yes.

TROPP:

Although, philosophically that idea is a very sound one, no question about it.

GILL:

Oh yes, I felt very hurt because I felt this was the right way to go but...

TROPP:

But as a teaching device, this other...

GILL:

[laugh]

TROPP:

in that -- when a student is faced with the need for a subroutine that isn't in the library—

GILL:

He's got to learn how to write it.

TROPP:

He's got to know how to be able to write it.

GILL:

He's got to see how one can structure a complete program into subroutines so that you can tackle them

one at a time. So, I was a little bit hurt to find that I wasn't able to use this. I worked on differential equations, I remember. I was put on to that by David Wheeler actually. It was his suggestion that I should take that as a subject. My supervisor was Wilkes, who had suggested, I think, that I might look at simultaneous linear equations, but they didn't inspire me very much. Hartrey had tried to get me interested in boundary layer theory which he had been very much concerned with during the war, but which I found heavy going and not very…it, it just didn't turn me on very much.

TROPP:

It never has been a very exciting subject.

GILL:

[Laugh]

TROPP:

When one thinks to an introductory course in complex variables that's one of the dullest areas.

GILL:

Yes. So, Davy Wheeler said, "Why don't you look at differential equations?" He said, "It looks to me as though the process is the most simple one to use in computers so I looked at the process and felt that one ought to be able to refine it a little bit so that it would go a bit more streamlined way on a computer. And so, I developed the Gill version of this which I probably did in, I suppose, at the beginning of 1950.

I don't recall exactly.

TROPP:

So, this was an attempt then to take the process, and…

GILL:

mhm.

TROPP

…apply it to machine technology, simplify it.

GILL:

Yes. The way that I wrote up the paper on this explained the way I derived the process by laying a restriction on the storage requirements for the process. And it's true that at that time storage restrictions were pretty tight. On the other hand, in fact, the pressure on storage was hardly a critical one and I realized at the time that saving of storage was not in itself a very important thing, but that I, I felt that by looking around for some such criterion as this, one might be led mathematically to a formulation which had other nice properties and it turned out in fact that that it did. So that was how I arrived at that particular form of the process. But it was very popular for many years. It's still used a lot, I guess, although it's now tending to be superseded by a number of other processes and I think it was probably overestimated quite a bit by some of the people who used it. In fact, obviously some of the people who used it really didn't understand it at all because one of the features of it was a formulation which enabled one to improve the precision of the process on the assumption that it was being programmed for a fixed-point machine. You had to be a little bit careful how you used it on a floating point machine if this feature was going to operate properly. Now, I'm pretty damn sure that from about 1956 or '7 onwards it was used on a vast scale, almost entirely on floating point machines and I bet 99% of the routines just didn't allow for this feature properly, …

TROPP:

[Laugh]

GILL:

…so that it just, it didn't operate correctly.

TROPP:

I guess it's the standard thing that one runs across in applications of many mathematical theorems.

GILL:

Oh, yes.

TROPP:

People always looking for conclusions and never worry…

GILL:

Exactly.

TROPP:…

about whether they…

GILL:

…apply them out of context.

TROPP:

…apply to the hypothesis or not.

GILL:

That's right.

TROPP:

And the, the, conceptual aspect is totally lost…

GILL:

mhm. And the reason for your total routine on the fixed-point machine then was, as you say, lost in the process. The motivation you mentioned earlier I think is worth repeating, I don't remember, on tape, for writing the text with Wheeler and with Wilkes.

GILL:

Oh yes. We ran our first summer school in '49, sorry, '50,

TROPP:'

50.

GILL:

1950, and it was a very ad hoc affair without very well-prepared notes. Wilkes said, "Obviously, if we're going to run this again, we must prepare the material more carefully." So we wrote up a very thorough text in the end of '50 and early '51, which we used for the summer school in 1951, and which we also got published by Addison and Wesley. In fact, the guy who fixed the publication by us with Wesley and Kopal, (Zdenek?) Kopal, the Czechoslovakian astronomer.

TROPP:

Oh, yes, K-o-t-o-w?

GILL:

No, K-o-p-a-l.

TROPP:

K-o-p-a-l.

GILL:

In recent years he's done a lot of work for NASA specializing in the properties of the moon. At that time he was at MIT and he was a friend of Wilkes for some reason that I'm not quite clear about. And the reason we didn't get published in England at that time was that all the European publishers had a great long backlog of manuscripts, and it would have taken us far too long to get it published over there.

TROPP:

Addison and Wesley did publish simultaneously and –

GILL:

Very quickly, they had it out. Yes, I think within six months, they had it printed and published. It was

then picked up by the Russians too, who published a Russian translation of it in about '53, I think.

TROPP:

What was the - how would you describe the underlying philosophy or main theme of that initial work, from the standpoint of programming, which is a very new, almost a brand-new word in the language, at this time?

GILL:

Well, the thing that, are you talking about Cambridge specifically or about programming generally?

TROPP:

No, Cambridge specifically, your work.

GILL:

Cambridge specifically. I think the big theme was the library of subroutines. In fact Wilkes insisted that the title of this book should be "The Preparation of Programs for an Electronic Digital Computer, the special Reference to the EDSAC and the Use of a Library of Subroutines." I don't remember the exact wording.

TROPP:

...would just love that title! [laugh]

GILL:

Of course, the last half of it was always left off, but Wilkes insisted that that be the full title.

TROPP:

He insisted that, so that was the main, the idea of writing programs with the future notion of building what would ultimately be a total library of any subroutines that anyone might wish to call upon…

GILL:

That's right.

TROPP:

…to do all the various programs.

GILL:

Yes, Yes.

TROPP:

And he saw this, then, as a way to solve that particular problem…

GILL:

mhm

TROPP:

…within the kind of machine, the philosophy of machines that were then available…

GILL:

Right.

TROPP:

…or on the horizon.

GILL:

Yes, Yes.

TROPP:

That's interesting.

GILL:

Then I suppose allied with that were the features which David Wheeler put into the initial orders for EDSAC I, which were the ability to incorporate parameters into instructions as they were loaded so that one could then have a library tape for a routine which was not its final absolute form.

TROPP:

mhm. That you could then modify?

GILL:

It was modified on loading so as to locate it in the appropriate position in store and locate its parameters where they had to be and so on. And without that feature which Davy Wheeler put in…obviously the concept would be far more restricted in—

TROPP:

mhm. How do you see the impact of the work at Cambridge on future developments?

GILL:

Oh, I think software-wise it had a tremendous impact, because the techniques that we developed, that were written up by Wilkes, Wheeler, and Gill obviously inspired pretty well everyone in software development from then on. I'm talking about software; actually, of course, the word software didn't come until about 1960.

TROPP:

Right, right. That's a fairly recent word. But the same kinds of developments are going on simultaneously on the West Coast of the United States.

GILL:

Similar, but I don't think the- I may be wrong but I don't think the idea of incorporating parameters on Loading…

TROPP:

No.

GILL:

…as David Wheeler…

TROPP:

No.

GILL:

…had it. I don't think they came in—

TROPP:

I'm not thinking of specific things but again the emphasis on software because of the needs…

GILL:

Same effect.

TROPP:

…of the aircraft industry…

GILL:

Yes.

TROPP:

…in solving similar kinds of problems that you're…

GILL:

Right.

TROPP:

…working on in Cambridge.

GILL:

Yes. In fact, I don't remember now which came first, whether it was SEAC or SWAC?

TROPP:

I think SEAC.

GILL:

I think SEAC was probably only a few months behind EDSAC, and Jack Miller was visiting SEAC. He's another guy you ought to talk to, by the way, J.C. Miller. I gave his name to Bob (Northoff?) because Bob was interested in L.J. Comrie…

TROPP:

mhm.

GILL:

…and Miller was one of Comrie's associates. In fact, Miller had been working with Comrie right up to about the summer of '49. He then spent a few months with SEAC and when he came back he joined Wilkes's laboratory at Cambridge and he's been there ever since. Miller is a numerical analyst with a great deal of interest in numbers theory as well, and somewhat incidental interest in electronic computing, but he's contributed to the subject to some degree.

TROPP:

Maybe this would be best done on paper. But I'm hoping after you've attended our session this evening and heard the conversation, that along with the names you've already given me, at some future date you'll be able to almost give me a, a fixed itinerary of people that are absolute musts.

GILL:

Yes, I've jotted a few more down here; we must go through that later on. I remember Claude Shannon visiting Cambridge. I'm just trying to date this now. I guess it was probably late '50. And at that time we'd just got very excited about interpretive routines because Tony Brooker had joined us. I think he probably joined us in the beginning of 1950 and he had begun to develop an interpretive routine for floating point arithmetic.

TROPP:

B-r-o-o-k-e-r?

GILL:

That's right, yes. And we were, I remember, enthusing to Claude Shannon about the horizons opened up by the idea of an interpretive routine. And I remember saying to Claude, "It's, it's as if we had several computers here. We have the EDSAC and then we have the floating-point computer designed by Tony Brooker which happens to run on EDSAC."

TROPP:

[laugh]

GILL:

I mean it was such fun to be able to bend these languages. In fact –

TROPP:

The EDSAC was a fixed-point machine?

GILL:

Oh sure, fixed point machine, but we could do floating point arithmetic on it using Tony Brooker's routine for floating point arithmetic.

TROPP:

It was a synchronous machine, was it not?

GILL:

Oh yes.

TROPP:

Where you had to wait for—

GILL:

Yes, yes being a delay line machine, it had to be synchronous, yes. And of course, the floating point routines were very slow indeed. But—

TROPP:

But it's interesting that you were floating, it really was like having a second machine.

GILL:

Yes, exactly, we, it was, it was exciting you know to feel that you had this power over the thing. You could turn it into another machine.

TROPP:

Right. Well, since then the EDSAC is a derivative of the work at the Institute at Princeton, with its modifications by Wilkes…

GILL:

Right.

TROPP:

…in terms of his needs and how we saw the thing, with the modification of this loading routine and with things that ultimately developed, so that in a sense it went off pretty much on its own…

GILL:

Yes.

TROPP:

…from the initial concept that originated at Princeton.

GILL:

Can I mention another visitor we had, too? I remember this one very clearly. The date I'm not quite so sure of, but I was discussing it with Eckert a few months ago and I think that pinned it down fairly precisely to around about February 1950. Leslie Groves…

TROPP:

Oh yes.

GILL:

…came over to England. This was at a time when he had quit Project Manhattan and joined the board of Remington Rand. Eckert and Mauchly were still running the Eckert-Mauchly Computer Corporation and trying to build UNIVAC but they'd run into financial problems and Remington Rand was considering whether to buy them out.

TROPP:

Which they did.

GILL:

Which they did, and Groves was the guy who looked into this. And he came over to Europe with an assistant, and he was obviously a tough military gentleman. And he spent half a day or so looking around the lab. Wilkes was out at the time for some reason and Hartrey acted as host. Hartrey had known Groves during the war, of course, on the atomic weapons work. And Groves saw all the equipment we were using and discussed with us how we were using it and how we felt it would be used in the future. He was particularly interested, I remember, in business applications.

TROPP:

Groves was?

GILL:

Groves was, yes. And I remember him asking me how he felt programs should be written for business use. Would one expect business people themselves to learn how to program, or would we have to have specialist programmers who would have to find out about the business? Set about writing programs for them? And I was quite emphatic, I remember, that it ought to be possible for business to be taught how to program. The programming should be made easy enough, and businesspeople would be capable of being taught how to do it. And I didn't like the idea of having to have specialists doing this all the time.

TROPP:

Were you really thinking in terms of a special language then, or were you…

GILL:

mhm.

TROPP:

…thinking in terms of these libraries of subroutines that they would just sort of package together?

GILL:

I don't recall what I had in mind exactly for that. No, I don't think, we weren't thinking in terms of special languages much except through interpretive routines. The idea of compilers and translators came along later.

TROPP:

The compilers, I guess, first on the UNIVAC.

GILL:

I guess so, yes. It was really Grace Hopper…

TROPP:

Right.

GILL:

…who pushed that, wasn't it? They were a bit slow in the UK. The guy who pushed them first in the UK was Alec Glennie. He's worth speaking to. A.E. G-l-e-double n-i-e, who was working at a Ministry of Supply establishment southeast of London at the time, Fort Halstead, where they, I think, had perhaps placed an order for one of the Ferranti machines. I'm not too sure. He came to spend some months with us at Cambridge to study programming, and in fact he wrote what was the best description of the EDSAC ever to be written. I think it was, it was more clearly readable than Wilkes, Wheeler, and Gill. It did go so far into the available subroutines and some of the more advanced applications, but it was a much better introduction than Wilkes, Wheeler and Gill, and it was simply published as an internal document at Fort Halstead.

TROPP:

Fort, is that Fort Halstead?

GILL:

Yes. H-a-l-s-t-e-a-d, I think.

TROPP:

e-a-d. Would that be a kind of publication that I would be able to obtain if I wrote to the Ministry of Supply in Fort Halstead?

GILL:

Well, the Ministry of Supply doesn't exist anymore.

TROPP:

Oh, I see.

GILL:

It comes under the Ministry of Defense I think. I'm not quite sure.

TROPP:

Well, that's not important. It's something that I can write a note about.

GILL:

It wasn't classified. I would guess, I guess Wilkes probably has a copy, or if you can get hold of Glennie himself he may still have one.

TROPP:

These internal documents are always difficult…

GILL:

I know, yes.

TROPP:

…to find because they tend to vanish rather rapidly.

GILL:

Yes. Glennie is now at the Atomic Weapons Research Establishment at Aldernaston.

TROPP:

Atomic Weapons Research. Where is that?

GILL:

Aldernaston, A-l-d-e-r-n-a-s-t-o-n, Berkshire. He still is one of the world's leading experts on compiler writing and he began promoting the idea way back in about '51 or so. I think he went to work at Manchester for a time, too, when they got the first Ferranti-engineered Manchester machine operating there.

TROPP:

He sounds like an interesting person to talk to.

GILL:

He is, yes. He, when I went up to Manchester for the inaugural meeting of the Manchester machine, he very proudly showed me a diagnostic routine that he'd had running there on several lines...

TROPP:

On the EDSAC. Were you at all concerned about error, error checking? This does not seem, the reason I ask is, it doesn't seem to have been a concern of that particular generation of machines as an over-all thing that people worried about. You find them in the early relay machines. If you look at the Bell Labs, they have all kinds of checking methods, the excess-three code, ways of checking as you're going, ways of checking results so that you don't ever get wrong answers, but one looks at ENIAC and the early electronic machines of that same vintage, and there doesn't seem to be much worry…

GILL:

No.

TROPP:

…about it.

GILL:

The ones that actually worked were the ones where people hadn't worried about this too much. I mentioned this remark of Freddy Williams…

TROPP:

Right, right.

GILL:

…about the propounded plans of Utley, but Utley's machine never got built. Eventually there was a TRE computer built which may have had some parentage from Utley's designs, but it was pretty dissimilar I think. And, of course, the original UNIVAC planned extensive duplication of the….

TROPP:

Right. That's where the checking was going to be done by doing the same thing over a number of times.

GILL:

That's right, yes. They were going to duplicate the process entirely.

TROPP:

Right. This was the, of that whole period this was the general - the machines were fast enough…

GILL:

mhm.

TROPP:

…that you could duplicate and check the results at the end.

GILL:

Oh, you mean by using the same…

TROPP:

Right.

GILL:

the same process, but repeating the calculations? No, I understood it—

TROPP:

I mean dual machines, I'm sorry.

GILL:

Oh yes, dual machines. Right, yes.

TROPP:

BINAC, for example, is essentially two machines…

GILL:

Yes, yes.

TROPP:

…and this seems to be the theory that this was faster and cheaper and quicker…

GILL:

mhm.

TROPP:

…than worrying about the complexities of checking routines and diagnosis of malfunctions.

GILL:

Yes. Right.

TROPP:

It was much easier to test the tubes ahead of time, give them sufficient hours, and then plug them in and know they would run for a long time.

GILL:

Yes, yes. Anyway, to come back to Alec Glennie. He began writing these translators in about '51 or '52, and there was a conference at the NPL in spring of '53, where you'll find a paper by Alex Glennie. At that time, I had written an improved routine for EDSAC and I don't really recall very much about it, I must confess at the present time. It was on lines that Wilkes was anxious to pursue, and it gave more facilities than David Wheeler's (auxiliary?) routine, but just exactly what the facilities were I must confess I don't, I can't recall at this time. But at Cambridge at that time we weren't very impressed by the need to be able to accept algebraic formulae and translate these into strings of instructions. And I seem to remember saying at the NPL conference that I didn't think that this was a real requirement, that it was no great problem for programmers to be able to convert formulae into instructions. What a programmer needed was some sort of clerical assistance in editing his program, and assembling it, and making it all fit together, and that was the kind of thing that we had been concentrating on at Cambridge.

TROPP:

EDSAC ran for quite a long time.

GILL:

Oh, yes. EDSAC I, I think, was probably taken out of service in about 1958. EDSAC II went into service probably in '56 or '7 and was finally scrapped somewhere in the middle sixties, '65 or '66, I guess.

TROPP:

What were most of the mag-major modifications on EDSAC I during that decade of operation?

GILL:

B registers, one or two of those went in - I forget now, no, only one I think. A telephone dial was added. This is, yes, I should have mentioned this - another thing that Wilkes was rather firm about was that there should be no attempt to have any kind of manual input or intervention with the machine, because this was a very inefficient development. One had got to accept the fact that humans were far too slow compared with computers and it was bad practice for a human to hold a computer up.

TROPP:

[Laugh]

GILL:

If he was in any doubt about which button to push next he should get the hell out of the way and let somebody else come in. So that we had a single shot button for a single instruction operation. This, I suppose, was officially put in for the engineers to use. In fact, of course, the programmers frequently did use it, and I saw Wilkes using it many times himself.

TROPP:

[laugh]

GILL:

But this was known as peeping. And it was bad practice and frowned on by the authorities. Well then, in about '53 we were in the process of attaching magnetic tape, packing storage to the machine and one of the things I did was to write a program for adding this. I wrote a routine for the magnetic tape. And obviously, we were planning to put the library onto the magnetic tape so that it could be pulled off there and, in fact, people could complete programs. Test programs, all sorts of things could go onto this tape. And we needed a way of being able to.. instruct the machine manually to retrieve on of these routines off the tape and execute it, but we were stuck. We had no manual input. We had no way of feeding even one digit into the machine telling it what to do. So I said, “Well, the simplest thing is to have a telephone dial; just stick a telephone dial on the machine and count the pips.”

TROPP:

mhm.

GILL:

And so finally that's how we got our manual input. And of course, having brought it on then, a lot of programmers found this a very useful tool just for inserting the occasional parameter manually into a program that they designed.

TROPP:

When did you begin to get to the realization of - I guess the phrase is now stored program concept - the idea of storing a program, modifying a program within the machine itself?

GILL:

Oh, right from the beginning. This was the great advance which Von Neumann - it's attributed to Von Neumann. Of course, you suggested that maybe he got it from Turing. I don't know.

TROPP:

I don't know. But this was inherent in that idea, and it was exploited right from the very beginning.

TROPP:

So, the EDSAC that you saw operative had that capability from—

GILL:

Well yes. It was inevitable. Once you'd designed a machine to take instructions from the same store that contains the data from which it operates, then you learn, you know, you can do it.

TROPP:

It was there from the beginning.

GILL:

From the beginning. It was inherent in the design of the thing.

TROPP:

I see. That was really a stupid question. Von Neumann talks about this, I guess, as early as 1944.

GILL:

I see, yes.

TROPP:

I haven't -- as an abstract concept, but the realization is somewhat later and I guess I just got my time confused in terms of the realizations…

GILL:

mhm.

TROPP:

…because of that other machine being so late in coming along.

GILL:

Yeah. It's rather interesting though that the usage made of it has dropped off since the early days. The first programs for the EDSAC made very heavy use of this, especially for operating on vectors and matrices, because without a B register, if you needed to have an instruction loop that operated on successive elements of a vector, the only way to do this was by programming the modification of those instructions as they stood in the store. And frequently we'd write loops which had maybe three or four business instructions in the middle and then about ten instructions around them which modified the three or four in the middle and obviously this was inefficient. It was the only way we could do it at that time.

TROPP:

It was the only way you could do it at that point.

[End of Tape Two, Side One]

[Start of Tape Two, Side Two]

Then, of course, index registers became more common and these loops became shorter and the instructions as they stood in the store were not directly modified. And then later of course with time sharing, and multiprogramming systems, the idea of the pure program, the pure procedure, which was deliberately designed so that it never got charged during the execution, became an essential feature of these systems.

TROPP:

I've never seen the Ferranti machine. I've only had one discussion about it. But how does it relate in its conceptual aspect to the EDSAC?

GILL:

Well, it's much more closely related to the Manchester machine; of course, it was really just an engineer copy to the Kilburn & Williams machine. A similar kind of instruction code except that it had this B tube in it... The way of writing instructions was far more primitive. If you look at any of the earlier programming manuals for the Manchester MARK I, you'll wonder how any programmer ever managed to stay sane long enough to complete a program.

TROPP:

[Laugh]

GILL:

That's because it was really set up by Turing. Turing moved to Cambridge for about a year. I think he took a sabbatical year from the NPL, I'm not sure what year that would be. It was '48, '49 I guess. I remember visiting him in Kings College, Cambridge.

TROPP:

You think it was '48 or '49, so it would have been long, after you, during the period that you first arrived in Cambridge?

GILL:

Yes. Yes. But he never went back to the NPL; he went on to Manchester and he was the, he was primarily responsible for the programming conventions of that machine, so they bore all the marks of incomprehensibility that one normally associated with Turing.

TROPP:

In terms of business applications, you mentioned the Lyons Tea Company. How about, I guess I should call them the assurance companies? Were they thinking, were the ones in the United Kingdom thinking in terms of the electronic machine as being something that could help them?

GILL:

Not quite so early as in the states, but fairly soon after that, I remember representatives from the Prudential Assurance Company, which was our biggest, attending one of the summer courses at Cambridge in 1955. And also around that time John Bennett, with Ferranti Limited, was collaborating with Tony Baker, who was working with the Royal Assurance Company based in Liverpool and jointly they developed the design of a business computer which logically was very similar to UNIVAC, in fact.

And I think it was going to use delay lines, wire delay lines which Ferranti developed at that time, in fact similar technology to the Ferranti PEGASUS which was being designed at that time, this is '55. And this machine eventually was manufactured and it was called PERSEUS, but unfortunately the Royal Assurance Company never in fact bought one in the end. After allowing Tony Baker to collaborate with Bennett to designing it they decided not to have one. However, Ferranti succeeded in selling two of these PERSEUS machines, both of them overseas. You mentioned the problems they had in trying to sell this thing, deliver it, and commission it overseas, but they did it. One went to Sweden to a joint operating company owned by two insurance companies. I think the insurance companies were Trugg, T-r-u-double g, and Fylgia, F-y-l-g-i-a, I think and they set up a joint committee, company which they called - sorry, I don't remember the name…

TROPP:

That's all right, I can trace that down.

GILL:

…and the second PERSEUS went to the South African Mutual Insurance Company in Capetown.

TROPP:

Love that name.

GILL:

Incidentally, one of the best films made at that time, one of the best movies about electronic computers was made by this Swedish insurance consortium.

TROPP:

Is it in English?

GILL:

Oh yes, well certainly there was a, a version produced with an English soundtrack.

TROPP:

You don't have any idea where one might find a (copy?) of that?

GILL:

I'll tell you a man who'll almost certainly find one for you. He might be a guy that would be worth talking to perhaps. Bernard Swann, B.B. Swann, Bernard B. Swann with two n's who was the sales manager for the Ferranti Computer Department for many years, in fact until the department was sold to ICT. And he stayed with Ferranti in various other capacities, recently retired from full-time employment, but I understand he's now being retained by Ferranti to write a history…

TROPP:

He's worth talking to.

GILL:

…so he would be worth talking to, yes.

TROPP:

What would be the address to write to Ferranti, if I were just to address the letter—

GILL:

Well, I should write to him c/o Ferranti Ltd, at Millbank Tower.

TROPP:

And that's one L?

GILL:
No, two L's, M-i-double l-b-a-n-k. Millbank Tower, Millbank, London. I don't know what the postal district is. It's probably S.W.1, I would guess.

TROPP:

That'll get us close enough. Write in care of Ferranti at Millbank Tower. It'd be interesting, I'd be interested to se his historical work because I told you about my involvement in a little bit of history at the University of Toronto and they bought, I think, the second…

GILL:

Indeed, yes.

TROPP:

…Ferranti which became the - what they called FERUN, Ferranti-University of Toronto.

GILL:

That's right, yes. The first computer ever to be exported.

TROPP:

Right. The story behind that, as I understand it, is marvelous. Because of the election, the change of government in England, suddenly all contracts are canceled. Nobody could buy machines or anything over so many hundred thousand pounds. The machine became available; Toronto had the money which they had originally earmarked for research and a brand-new machine. There's the Ferranti. They want computability…

GILL:

I can believe that too.

TROPP:

…because they've got the St. Lawrence Seaway project. They've got atomic energy work going on, and they bought it.

GILL:

I'll tell you, the guy who can confirm that story, Swann's predecessor was Vivien B. - Vivien Bowdenoh, in fact, the guy who first, in thought…

TROPP:

Right.

GILL:

…he's now Lord Bodin, of course, and he is the principal of the Manchester University Institute of Science and Technology. Which is the sort of technology school at Manchester University. In fact my son is studying chemical engineering there right now.

TROPP:

That's in Manchester, of course.

GILL:

Yes.

TROPP:

That'll be interesting.

GILL:

Bodin is a great talker.

TROPP:

mhm.

GILL:

He'll talk to you interminably.

TROPP:

Did you, were you cognizant at all of Lord Halsbury's role in this period?

GILL:

Not very, no. The people that I met up with were mostly somewhat critical of the part being played by Halsbury and his NRDC. I think they had a feeling that although he was ostensibly trying to support computing there seemed to be so many blockages in the way that people felt that he was being more obstructive than constructive. So much so - I think this is probably one of the things that I shall have to mark for being kept private for some years yet - but when the British Computer Society came to be set up in '57, well I think we were discussing it in '56 and we were considering who should be picked as the first president. Halsbury's name was put forward quite naturally because in fact NRDC through Halsbury's support had done a lot to help the BCS get started. I mean we were using their premises for our meetings and they were providing secretarial services and so on. Nevertheless, when Halsbury's name was put forward as the first president there were so many people who'd been critical of his role in this that he was voted out, and, in fact, Wilkes became our first president. So, I don't, I wasn't.

TROPP:

Personalities play such a strong role…

GILL:

They do, yes.

TROPP:

…in the development of anything that was as exciting and vibrant as this whole conflict.

GILL:

But I wasn't close enough to the financial and political side... [GAP IN TAPE HERE]

TROPP:

Williams, who runs a company in Wales, and his rather abortive attempt to set up a conference in 1970 on the history of computers.

GILL:

Right. Now present at that meeting were Grace Hopper, David Edwards (Di Edwards) who is one of Kilburn's people, and a number of others. I don't recall all the names. They had somebody over from the Soviet Union maybe. I'm not sure, somebody from over there, and they presented a number of papers on the history of the subject. And it may be possible for you to get hold of these. Knowing Williams, he'll probably charge you fifty guineas a copy or something of that kind.

TROPP:

[Laugh] Maybe if I publicize this meeting he'll donate them.

GILL & TROPP:

[Laugh]

GILL:

Maybe, and then F.C. Williams was the one I mentioned from Manchester, Freddy Williams. He is now a senior official of the University. I forget his exact title - I think they call it pro-vice-Chancellor or Vice-Chancellor—

TROPP:

This is the Williams of the Williams memory tube?

GILL:

That's right, yes. He was initially leader of that project, but he dropped out of it pretty soon after they got the first computer to work and he left it with Kilburn,Tom Killburn,

TROPP:

Killburn, I know of, and MHA Newman...

GILL:

Right.

TROPP:

…you've mentioned. I'd like to get all the names who is –

GILL:

Next Donald Michie, M-i-c-h-i-e. Donald was working along with Turing and the other guys during the war on cryptography. After the war he dropped out of that game for some time and he went into medicine. I don't quite know why he went into medicine. Maybe he'd had a medical training before the war; I don't know. But he came back into computing again in the late 1950's and set up a project in the University of Edinburgh to experiment with way out uses of computers. He called it the Department of Artificial Intelligence and Perception, I think, something like that, which is now still operating.

TROPP:

Is Michie still there? I've heard the name.

GILL:

Yes, oh, yes. He was a great friend of Christopher Strachey...

TROPP:

Is that Christopher Strachey related to Lionel?

GILL:

Yea, I think he is; I'm not quite sure what the relationship is but most of the Stracheys that one hears of in England are related to each other some way or another.

TROPP:

A. M.--

GILL:

A.M. Uttley.

TROPP:

Uttley.

GILL:

That's two t's U-double t-l-e-y. Yes.

TROPP:

You mentioned him earlier.

GILL:

Right. He was at TRE and he's now at the University of Sussex.

TROPP:

Now at the University of Sussex.

GILL:

Right. Strachey is at Oxford now if you want to get ahold of him. And I think he's got a chair there; I think he's Professor Strachey now. Oh, here's another guy that you should track down - Andrew Donald Duve.

TROPP:

Oh, yes, oh, of course.

GILL:

He's now at the University of Saskatchewan.

TROPP:

That's the reason for this becoming sort of the center for the current history of Babbage I think.

GILL:

Could be. He was at Birkbeck College in London for many years, late forties and through most of the fifties. And most people didn't get along with him too well because he was inclined to be cynical of other people's efforts, but his own were not particularly conspicuous by success. But he has some good ideas and he really pioneered the idea of the drum-based computer; many years before the 650 he was designing a computer around a drum as its main memory. And in fact, his designs were copied in the initial faltering attempts of the British Tabulating Machine Company to get into the computer business.

TROPP:

Now, when would this have been? When he was working on drum memory?

GILL:

Around about 1948 and '49, I guess.

TROPP:

You heard the discussion tonight about the drum condensers?

GILL:

Oh, yes.

TROPP:

Drum memory which had

GILL:

Yes. Which I've seen pictures

TROPP:

But I've not seen a, which I gather there is one on exhibit in a museum at Ames, Iowa.

GILL:

I see. mhm.

TROPP:

And we've had many discussions about the...

GILL:

…J. H. Wilkinson, Jim Wilkinson.

TROPP:

Right.

GILL:

I think you mentioned his name this afternoon.

TROPP:

Right.

GILL:

He's at the National Physical Laboratory and has been there since the days of Turing, along with Mike Woodger. K.D. Tocher.

TROPP:

K. D. T-o-c-k…

GILL:

c-h-e-r.

TROPP:

c-h. He is now with the British Steel Corporation, concerned with operational research and computer systems. He started out as a statistician but immediately got very interested in computers. I first met him at the NPL in '46, '47. He left there about '47 and went to Imperial College, Mathematics Department, where he stayed for about ten years and went quite a long way along the road to building in fact I think they even had it operating, but they never did very much work with it, their own computer called the Imperial College Computing Engine, I-C-C-E, "Icky".

TROPP:

I-C-C-E, right.

GILL:

But the math department at Imperial was not a very comfortable environment for this kind of work and he finally gave up and cleared out. Incidentally, collaborating on that with him was Sydney Michaelson. Syd Michaelson stayed on several years after Tocher left and did quite a bit to develop the teaching of computing at Imperial College, but he too finally was defeated by the establishment and he left and went to the University of Edinburgh, where he is now.

TROPP:

Edinburgh University. Is he tied in at all with this Artificial Intelligence—

GILL:

No, he and Donald Michie are not on speaking terms. The University has to go to great lengths to appease them both. In fact, I think probably they had to arrange that they both received chairs on the same day.

TROPP:

Then I'll have to arrange to see them on separate days...

GILL & TROPP:

[laugh]

TROPP:

…since I can't be there at the same time at both places. Let's see, you've mentioned…

GILL:

I've mentioned Pollard, Brian, Brian W. Pollard. He was the manager of the Ferranti Computer Department from the beginning up until about 1958, I think, when he quit and came over to the States. When I last heard of him he was with Burroughs, I think, but that was some years back.

TROPP:

I'm sure that if he's here I'll find somebody who knows him.

GILL:

mhm.

TROPP:

If he was with Burroughs, Ike Auerbach might be able to chase him down.

GILL:

mhm. Has he been talking to you about the Ferranti version of the –

TROPP:

How would you describe the atmosphere at Cambridge during that period when you came back in '49, as an intellectual atmosphere within which you do this kind of creative work?

GILL:

Very very exciting - not highly intellectual. It was a very friendly atmosphere at Cambridge, very exciting. We were all falling over each other to try and do cleverer and cleverer things with the computer.

TROPP:

I would just normally, if I had not known anything ahead of time, just not thought of Cambridge, any more than I would the Institute for Advanced Studies, as a place where a machine would evolve and be built and so on.

GILL:

It's interesting, yes. It was somewhat detached from the normal academic life of Cambridge. We didn't really mix in an intellectual sense with most of the University at all. It was a rather self-contained little band of people. Comparatively few users around the University had got converted to the idea of using

an electronic computer. One of the biggest groups outside the laboratory itself that was using it was the... theoretical chemistry group under Dr. Boys, B-o-y-s.

TROPP:

Is he still alive? And still at Cambridge?

GILL:

mhm. I can't tell you that. Wilkes would be able to tell you. And there's a story about Wilkes and Boys happening to meet each other in a tea shop in Kensington. And Wilkes described to Boys the computer that we were building. Boys, who was a very talkative man, spent pretty well the whole session explaining to Wilkes that the machine really couldn't be of any help to him in his work because the algebra was too complicated, but within a year or two he was the biggest user.

TROPP:

[Laugh] It solved his algebraic problem. That's interesting…

GILL:

And the—

TROPP:

because I think that's a fairly early use by chemists of—

GILL:

Oh, indeed. Yes, very early. This was already by 1950 they were fairly big users. The people in the Cavendish Laboratory, the radio physics people in particular, were heavy users. Wilkes himself had been a radio physicist…

TROPP:

mhm.

GILL:

…and he worked on radar during the war. He wrote a book. Even after he'd got deeply stuck into the EDSAC project he wrote a book on oscillations of the earth's atmosphere, which was a subject I think he had done research on before the war.

TROPP:

How about the astronomers?

GILL:

Astronomers, yes. Tom Gold, Fred Hoyle, and Bondi, Herman Bondi, were all at Cambridge at that time and certainly Bondi and Hoyle worked together on some early models of stellar evolution. I remember collaborating with a couple of students of Hoyle's, one working on a stellar model and one working on a model of the acceleration of particles at neutral points in a magnetic field. This was Jim Gungary who is now at Imperial College. Hoyle went on then, of course, to become deeply involved in computing power in Cambridge and got himself his own computer in his own institute. Recently, within the last few months, he's become disgusted with the politics of Cambridge and has announced that he will finally emigrate permanently to California.

TROPP:

[laugh]

GILL:

He already spends half his time in California anyway. It was exciting at that time. There seemed to be less tension around than there was in most of these other places. At NPL we seemed to be too strung up in bureaucracy. At Manchester they were a little too arrogant and self-centered and didn't seem to take much account of what happened elsewhere, which was a pity because they did, in fact, generate a lot of good ideas of their own.

TROPP:

It was a one direction kind of thing.

GILL:

It was very much one-directional, I'm afraid, yes. Well, I enjoyed it. Certainly was the happiest time of my life.

TROPP:

Of course, just the whole atmosphere of Cambridge is one that to this day is so pleasant that it's hard to…

GILL:

Mhm

TROPP:

discount it as one of the…

GILL:

Yes.

TROPP:

…pleasures of life.

GILL:

Yes, yes.

TROPP:

In terms of the documents of this period, are they housed primarily in the main University library or would there be some other special library?

GILL:

Wilkes has a pretty extensive collection of his own and he's very good at keeping documents of this

kind, so I think... he would be the best source of the material.

TROPP:

Best source of getting material which can be copied and…

GILL:

Right.

TROPP:

…indexed…

GILL:

Yes.

TROPP:

and identified. Very, very good. Well, I'm not going to tire you anymore, and I don't know how to thank you enough. This has really been great.

GILL:

Well, it's so nice to know that somebody at last is really getting down seriously to documenting this period, because I've been feeling for so many years that it ought to be done. And over and over again suggestions of this kind have just disappeared without a trace.

[Recorder off]

TROPP:

…say that again.

GILL:

Yea. The one country that I know of apart from the U.S., and the U.K., and Germany that was operating

in electronic computing very early on, almost as early as the U.K., was Australia, particularly in Melbourne. The CSIRO, the Commonwealth Scientific and Industrial Research Organization, built a computer called the SIREC MARK I…

TROPP:

[laugh]

GILL:

…which had some interesting features. I don't recall at present just what they were but I remember when I first read the accounts of the design of this, probably in about 1950, I was quite fascinated by some of the features that they put into this.

TROPP:

So, it was already built by 1950?

GILL:

Well, no, I, at that, when I first read about it; it wasn't built. I think probably they had it operating by '51 or 52 and it was in operation there for probably about eight or ten years from then. I think it was moved to the University of Melbourne. One of the guys who can tell you about this is Trevor Pearcey, P-e-a-r-c-e-y, who is an Englishman in fact, who had gone out to Australia.

TROPP:

He's now at Melbourne?

GILL:

Well, I'm never quite sure. He keeps moving back and forth between Australia and England, but I think at the present time he's probably in Australia, yes. But I just—

TROPP:

Do you know, in a sense, the message that I'm begging you to carry to England is one that you might carry also to Australia on your current trip.

GILL:

Indeed, yes, yes I will.

TROPP:

…because the same thing holds there, because it's a small group in a fairly few concentrated areas…

GILL:

mhm.

TROPP:

…in terms of these earlier developments, and they may already have done the history.

GILL:

Yes.

TROPP:

It's conceivable, knowing of their interest in historical work. I know there's an institute in the capital city that was founded eight or ten years ago for the history of things related to Australia in particular.

GILL:

I see, in Canberra.

TROPP:

Right. And I don't know what its orientation is or what its emphasis is. I merely read - I remember vaguely reading eight or ten years ago about the founding of such an institute…

GILL:

mhm.

TROPP:

…and their interest in such things. And it may be that as you talk to people, you may find this is already done.

GILL:

True.

TROPP:

And what I would like to do then is to incorporate what they have into our source to enable us to become a research source for people in that topical area as well. And so, if you have a chance in casual conversation during your –

GILL:

Sure yes, I will.

TROPP:

The same offer holds for Australia…

GILL:

Okay, fine.

TROPP:

…because the chances are fairly good that I'll get to England within the next six or eight months, but Australia seems very remote to me.

GILL:

Yes, yes.

TROPP:

It's going to have to be long distance.

GILL:

OK.

TROPP:

OK, thank you.

[End of Interview]