Capturing the Past, Inspiring the Future

Norman Sanders

Interviewed by

Mark Jones

28th November 2018

At

Home in Suffolk

Copyright
Archives of IT
(Registered Charity 1164198)

Welcome to the Archives of Information and Technology. I'm Mark Jones, an interviewer with the Archives.

Today I'll be talking to Norman Sanders.

It's the twenty-eighth of November 2018 and we're in Norman's home in Suffolk.

We spoke to Norman recently and he told us about his life and career, during which he was rightly referred to as a computer pioneer. Norman started working with computers in the mid 1950s at Cambridge. At that time many of the academics and engineers who created the first computers in the 1940s were still very much alive and active. Today we're going to talk about those early days of computing and Norman's going to share with us some of his memories and opinions of the machines and people who started off this huge industry.

So Norman, once again, thank you for sparing the time to talk to Archives of IT. For me, I think the story of computing, British computing, perhaps starts with Alan Turing in 1936, with a bit of maths. Would that be right?

Yes. I don't think anyone would argue with that at all. Of course, he didn't have a computer, there were no computers, but the idea of the computer had come up in the same way that, when RUR, Rossum's Universal Robots, that play was written, the author just talked about robots as though they existed and that's fine, that's how it is with art and so forth. But the ideas that he had, which basically come down to the store programme, were, you know, twenty years and more ahead of its time. So that puts England, Britain at least, ahead of everybody, really. Of course, we had earlier on than that, we had Charles Babbage and he said, 'I wish to God those calculations had been executed by steam' but he never got the steam. He had the brass and so forth, so the ideas were there at that time. Whether or not, of course, Turing had ever heard of Babbage is, I guess, doubtful, I can't see how he would, but he might have done. But he wasn't mentioned. But Turing's whole idea was, 'If you've got all these numbers, some of these numbers are data, some of these numbers are programme, somehow stuck in a box of some sort, this is what would happen.' And so Turing's name is, in a way, he's the top, he's the leader of the whole thing.

So, for non-mathematicians, it might be hard to understand, I think, I don't understand it entirely either, but I think what Turing did was to provide a theoretical underpinning to say that, if you have a box that works like this, then it is capable of solving any known algorhythmic problem. Is that broadly correct?

Yes. That puts it very well. That's what he did.

So it's a test, really. If you have a computer that acts like a Turing machine then your computer must be able to solve any problem you give it, given enough time and resources and so on?

Yeah, there's a lot of words deeper down in the thing than that but at the top level that's correct.

So, you met Alan Turing, didn't you?

Yes, he came ... very, very sadly this is about two or three weeks before he died in the summer of 1954 ... I was a member of the Quintics which was the mathematical society at Cambridge, he had gone to Manchester University, having been a Cambridge man originally himself, and he was up in Manchester and came down to give us a talk. It was about Fibonacci's Series and so forth and we were all excited about that. I used to go out to the arboretum and collect fir cones and pine cones and count the spirals and make a collection of different sets of spirals. So, you know, we were all aware of Fibonacci's Series. He came, and there were a lot of us, I was just the new kid on the block, I guess, the rest were lecturers. Many of them, later on in life, twenty years later, I discovered, chaps who had been my lecturers, had all been at Bletchley together. It was amazing, it was just a bunch of friends, really, who'd got themselves to Bletchley and then got themselves back to Cambridge afterwards. Anyway, he came up and he gave his talk but it was quite clear to me that there must be something wrong. Here was this great man, walking around and giving a talk about Fibonacci's Series, and never, he just didn't finish his sentences. He was - I didn't know what was wrong then, of course, that all came up later - but he was

clearly a very, very bright guy but he was a terrible lecturer. But there were biological reasons for that.

He was being force fed chemicals, wasn't he?

He was, I'm afraid.

00:06:10

Terrible story.

It was. The way we treated him was terrible. It wouldn't have happened today but ... so, life has improved, we've become more civilised in our treatment of people.

Oh well, that's the father of computing in a theoretical sense anyway, perhaps. So Alan Turing was probably the father of computing, wasn't he?

Yes.

So the next name I found was Konrad Zuse.

Zuse! The funny thing, the Z is an S [corrects pronunciation].

So, had Germany won the war, he'd probably be much better known, wouldn't he, in a nutshell?

Yes, you're quite right. No one at the maths lab ever mentioned Zuse's name. They mentioned lots of others but other names, I picked up quite a few people also who weren't mentioned in Cambridge and among them is Rutishauser. Rutishauser was a Swiss and he was the father of the first ALGOL. ALGOL, of course, followed Fortran as being a sort of arithmetical language for programming. They're almost identical; the big difference is that, in ALGOL, if you wanted to say that something was equal to something you had to have a colon, but in Fortran they didn't care ... x = x+1 and the only solution to that is that x = 0 [laughs], something like that. No, even

that's wrong. But ALGOL anyway became a good academic language. I think Fortran was a good industrial language, ALGOL was an academic language, but you didn't need two, one was enough. Anyway, Rutishauser, he had invented ALGOL 58 and he told me about Zuse.

Right.

And later on I met Zuse, in the summer of 1963, in Helsinki, where there were a whole bunch of computer guys. They were there for a couple of weeks or something like that, and he'd just retired and told us about his machines and his language, strange language, and he was way ahead of anybody in Britain or the United States.

I did read that he's credited with the very first electromechanical relay computer.

Yes, he was, that's correct.

Which presumably, because this is wartime then, developed for the armed forces in some way or another, or something to do with military applications, I would imagine?

Oh, er, No, I haven't heard that. I think his grandmother gave him the money, or something, he did it all at home.

Oh, I see.

His first machine, the Z1, was in 1936, he just did it at home. When you look back, why were computers constructed? Well, you're quite right, in the United States, it was the army that got ENIAC up and running, they wanted it, they paid for it and gave them everything they needed and so forth, so they were working inside a well-organised situation. In Britain, of course, it was at Bletchley, there the machine, you know, it's on the edge of being a computer. It was a one-time thing, it wasn't really programmable but its job was to help these Wrens decode and, I guess, Turing went to Dollis Hill and said, 'I've got all these girls and we're getting more and more, we're building more and more cabins for them to work in, and we know what they do but want to speed it up.' But because the people in Dollis Hill, they didn't know anything

about decoding but they knew their electronics. And the combination of Turing, on the one hand, and the people at Dollis Hill, on the other, produced firstly the Bombes and then the big one ...

Colossus.

Colossus, yes. You have to forgive me, I'm ancient now and ...[laughs] my own name is written down a piece of paper! [laughs]

00:11:04

So Dollis Hill was the Post Office research centre, wasn't it?

That's right, yes, which has moved here.

Yes, to Martlesham.

To Martlesham. And two of these guys, one lived round the corner and I'll show you if you like, I know that this is not all, but somewhere around here I've got Norman Thurlow's ... [papers rustling], his funeral thing is here somewhere, and Harry Fensom lived out here on the coast and I visited him a few times. One little thing, Churchill said, when the war was finished, 'I want this whole thing totally forgotten, burn everything, dismantle the computers, I don't want anyone to know, I don't want the Russians to know what we were capable of doing.' Ok. And so I was over at Harry's house, and they were rebuilding the Colossus, and I said, 'How could they?' because all the drawings were destroyed. He looked around and he went under the bed, literally, and took out all the drawings. He'd kept the drawings.

Really?

Yeah, but I couldn't tell Churchill, he was dead by then. [laughs] Yes, that was incredible and so that whole Dollis Hill thing moved over to here and, of course, it came away from the Post Office and became Telecoms, in Martlesham.

Yeah, that's a world famous world leading research centre in telecom's technology, isn't it?

Yes, it is, yes. Although a son of mine, over in Norway, he's got his PhD on computer controlled telephone exchanges and, at his PhD dinner, which is what they do over in Scandinavia, I had read everything to check his English. I said, 'Richard, I understood every word in your thesis but I couldn't understand any of the sentences.' It's quite true, it was complicated to me. Em, where did we get to?

Bombes, we were just about to talk about Bombes. So Turing was the sort of leading intellectual figure, was he?

He was. It's very difficult to say what he was because he was sort of everything, you know, they worshipped him as he walked through the corridoors and all that sort of thing. A wonderful chap and he realised that, ok, by that time, '41 or '42, something like that, that they had really, they knew how to do it, and these girls that he had, these very, very bright ladies who could all solve Times crosswords and play chess, they knew how to do it, but they were human beings and human beings aren't as fast as electronics.

I didn't realise so many Bombe machines were built, actually, but according to good old Wikipedia 120 were built all together.

Yes, I've read that number too. I can't imagine where they were but that's what they're saying.

Did you watch the film?

I saw the film.

And it looks like there's one and it's three Bombe machines, isn't it?

That's right.

But obviously it's more complicated than that, as you would imagine. So, there's a guy called Welchman too, did you ever meet him or ...?

No, I'm so sorry I didn't because he has written a superb book. He was in hut six there. And I always wonder how ... you see, you had Turing at sort of the top but he wasn't managing. Turing was not a manager and no one ever asked him to be a manager. He was just there and other people had to do their job in their various huts and would use Turing when they needed to. And Welchman was one of those but clearly he understood how to organise things, he was good with people and all the rest of it. And there were others, like Shaun Whylie, the guy at Cambridge. Shaun Wylie, by the way, and Turing, they were buddies, they were at Princeton together doing their PhDs. And when Turing got this job the first thing he did was ring Shaun Wylie and say, 'Come and join me.' And so Shaun Whylie left Trinity Hall and went down there and came back at the end of the war.

00:15:53

So, am I right in saying that, strictly speaking, the Bombe was an electro-mechanical programmable computer but only programmed for one problem?

That's right. I think you have to say that programmable means you can do ... as long as you can write the programme you can do anything but, of course, all those early ones, the programme was in the mechanical-electro design of the whole thing, and it was like that with the Princeton machine. You had to use ... someone had to go in and turn switches round and all that sort of thing and had to, sort of, rebuild the machine each time, which might have taken two or three weeks.

But, of course, it was designed to solve one particular problem, wasn't it?

Yes, it wasn't a general purpose device.

So we were talking about Bombes, then we get to Colossus which was programmable, but, I think, using patch cords.

That's right, and again, with that one, you couldn't suddenly do something else with it, you had to plan what to do and rework it and do that thing. So it's one programme at a time, including rebuilding the machine, really.

So, obviously, the vast majority of effort during wartime was devoted to solving the problem in hand.

That's right. Yes.

So I would presume there were still people in universities doing research, because research didn't stop during the war, did it?

Probably not but I think the most important thing you can say about Colossus was, they really learned – and the same with the ENIAC in the States – they learned how to handle large numbers of valves, what the Americans call tubes, you know, hundreds of them. I can't imagine hundreds of these things all in a box or along a shelf or something like that, and that was ... all of these early things all led slowly to big things, fast things, you know.

I think I read somewhere that, when they turned ENIAC on, the lights went dim in the nearby town.

[laughs] Yes, I'm sure, I hadn't heard that before ... and they warmed up all the birds in Cambridge with the EDSAC ... [laughs]

So, we're getting through the war now, and Colossus, and I think the ENIAC in the States was about 1945, something like that.

'43, they started.

'43, was it?

Yes, they started working in '43, the bar went up in '45, just at the end of the war it was getting it running.

Do you know if there's much crossover between Britain and the States, or was it two independent streams of people doing their own thing?

Well, not so much about the detailed design. All the computers were different. In the United States you had a lot of people getting these ideas and the key thing was to find ... how shall we build a memory or a store, you see, and ok, there was the Williams tube idea. Williams, of course, was at Manchester but they were using far more Williams's tubes in the States than they were using at Manchester University. And there were the mercury tubes ... and these ideas ... after 1944 there was a lot of moving around. Maurice Wilkes was constantly taking boats over to the United States. He enjoyed himself but he met lots and lots of people and they exchanged ideas and so forth, and so they were cross-pollinating on ideas, talking about how to handle valves and, you know, what the problem was with Williams's tubes. The problem with Williams's tubes was that you had to engineer it so that your bits, which were on the surface of the tube, [gesturing with hands] I know that no one can see these hands but if they were too close, if one position was being interrogated or a bit was being put there, it could influence the ones close to it, so you had to engineer these things and manufacture them so that they didn't interfere with one another. Whereas, with the mercury tubes it was all crock, it was all bit and crock, so you had a single stream of bits going down the mercury and one didn't interfere with another.

00:20:47

Do you think either the US or the UK had a lead in, sort of, intellectual property, or was it bright people everywhere just working together?

Yes, it's certainly the latter but, with property, in the States there were some acrimonious arguments about patent, but not in Britain. You had a very close co-operation between Ferranti and Manchester University and, once EDSAC was sort of going, Lyons Corner House discovered it and they needed a computer. When they realised that such a thing existed they suddenly knew they needed it, so they started the Leo computing and Leo computer company and Wilkes just handed the whole design over. They just took it, hardware, software and everything.

Yes, so the Americans perhaps saw more opportunities for commercialisation?

That's right. It doesn't lie in the heart of British academics, or it didn't then, to make money on anything. It's all *esprit de corps* stuff, which it wasn't in the States.

Ok. So, round about now, we come to the von Neumann report, 1945, 1946, the Moore School lectures and so on, which, from my little reading of it, and I'm sure you know more than I do, seemed to be quite a catalyst to all sorts of things happening after that. A big conference of lots of bright engineers and computer scientists getting together and things happened as a result.

Yes.

But I think, from what I was reading, the von Neumann report was a little bit controversial, maybe retrospectively rather than at the time, and it would seem as though he kind of trampled on the feet of a few other people, perhaps, and maybe other people didn't get as much credit as they should have done. Hard for you to say, perhaps, but it does sound as though it was ... a little bit, what shall I say, he claimed credit,which perhaps wasn't entirely his, for some of the stuff that was written then.

Yes, now I knew Pres Eckert very well, Pres Eckert and I both worked for Univac, so I would encounter him from time to time and in the Rand symposium which was run every year in California. And he was one of the most impressive guys, Pres Eckert, that I ever met. If you've seen the picture of Pres Eckert he has this high shaved head and it sort of radiates intelligence. He was an incredible person to be with, fast as anything and just bright. You know, through life, when you get to ninety you've met every kind of person there is and he just stands out, absolutely. One thing that surprised me, when I was doing some re-reading recently, was that von Neumann was acting as a sort of consultant to the work they were doing on EDVAC. Well, the last thing in the world that Pres Eckert would need would be a consultant. If anybody wanted to know anything they'd ask Pres Eckert and he'd give them an answer right away. So, then I'd always thought there would have been a protest on the part of Pres Eckert and John Mauchly, when that report came out, and a couple of days ago I read

that there was. The fact is that Maurice Wilkes, in Maurice Wilkes's book, he says quite clearly that they were less than happy with the fact that there was only one name on it, and that was von Neumann's name, and the work involved was Pres Eckert, John Mauchly and the gang.

So von Neumann, of course, is a very gifted mathematician with an awful lot of things to his credit.

Oh yes.

So we'll never know really whether it's just an oversight or how it happened.

Right, whether it was intended that way or not.

Von Neumann may have been thinking he's writing a mathematical paper and just covering that something a bit different, if you know what I mean. He wasn't trying to design a computer, he was trying to do a bit of maths. So maybe he thought he was doing his maths here, but that isn't really how it turned out to be.

You could take that point of view. [laughs]

Or, maybe not. [laughs]

We'll never know. We can't possibly know. I never met von Neumann but, as I say, I did meet ... I knew a lot of these people, all dead now and I'm the last down the line, but ... you know, on the industrial side and ...

So, Pres Eckert and John Mauchly were really seminal characters in the US computer industry, weren't they?

They certainly were, yes, they absolutely were.

Did they get on well, with people like Maurice Wilkes and so on?

Oh yes.

There's no sort of ... there must have been a bit of a competitive streak, I'm sure, but there wasn't any animosity behind it?

Oh, none. I don't think ... no animosity, certainly on Pres Eckert's side, and absolutely not on Maurice Wilkes's side. He was almost the nicest, most generous, wonderful guy. In later years, when I retired, I started doing a bit of mathematics teaching because they were so short of mathematicians, and I was at a school here, in Ipswich, and I happened to say to the class I had, 'Tomorrow I'm going to have dinner with the guy who invented a computer' – slight exaggeration of course. And one of the boys said, 'Oh, please thank him from us for inventing the computer', thinking just something in his pocket, you know, 'and saving us from a boring life'. And so I went over and, of course, I went to dinner and saw Maurice and told him what these boys had said and so he wrote a letter to them. He wrote a letter, which I ran off, there were six in this class, of little bright kids in a scruffy school here in Ipswich, and he said, 'I'm very impressed with the work you're doing, Norman, keep it up and so forth' and so I ran this letter off and I handed it to them and I said, 'Look, this is a historical letter, you guys, never lose them or make a few copies' and so forth, and one copy of that was put up in a glass case in the school.

I knew Maurice Wilkes in 1977, when I was at Cambridge, obviously very much towards the end of his career there. He lectured us and you could tell, he was just a very bright, very able man.

A very nice man.

00:28:11

Yes, who knew a lot of things about a lot of things.

He did. And he writes beautifully.

Yes, I think some of his textbooks were compulsory material.

I suppose so, yes. Of course, I've just re-read, probably for the third time, his book, his general history of the whole thing, in which he's sort of met everybody, seen everything and so forth.

So, in the UK, after the war, it sounds like there were two real centres of computing, Manchester and Cambridge.

There were.

Well, I know things were happening at Oxford and other places too ...

Not much.

... but the Manchester machines and the EDSAC at Cambridge were the two, kind of, main centres of computing, weren't they?

They were.

So the EDSAC, was that the first computer you ever actually worked on?

Well, funnily enough, I had a bit of a gap year, for a couple of years, between ... I was in the Air Force and I had to do my national service, but when I'd finished I didn't know what to do and so I worked for the Admiralty, to get some good background, and so I did a bit of programming there. But that programming was just patchboard programming for electro-mechanical machines, so if you regarded those as computers, which they very, very basically were, I was probably the first student to have already done some programming before I got up to Cambridge. But when I got there, and then I realised that EDSAC was up and running and so forth, and I just told everybody about it and, 'Come on, you guys' but nobody else did. Not from Trinity Hall, anyway.

So was that because it was thought of as a bit too grubby 'engineeringy' rather than academic, do you think?

No, I don't think grubby engineering came into it, it was just that they didn't know what I was talking about. We were three or four mathematicians at Trinity Hall then, and even they couldn't grab the excitement that there was in getting high speed ... to be able to solve different equations at high speed and get answers out in no time, you know, was heady stuff.

So this would be the mid-fifties, when you started there?

Yes, I went up in '53. I could have gone up in '48, if I hadn't done my National Service and everything, but I went up in '53. So I went to the maths lab in '56 and the EDSAC was running superbly, absolutely superbly, we were writing programmes until the steam came up out of our pens.

So how did they decide what was going to be run on the EDSAC? Was there a whole queue of people trying to use it?

Oh yeah, there was a committee which consisted, on a daily basis, of Eric Mutch – did you know Eric Mutch?

No.

He was dead by the time you came along. Did you know Margaret Mutch?

No.

Ok, she was a programmer. Maurice had known Eric Mutch for years and Eric was a good combination, he had his feet in both camps, he knew about the computer and he knew how to get things done. Eric was a superb – I hate to say that, but – office manager. He was an operations manager of the whole thing and had ladies working day and night, including David's eventual wife, Joyce. I meet Joyce every year at the David Wheeler memorial lecture and we have a good old laugh about the old days and so forth. They came over to America when I was living in Seattle and brought their little kids over and stuff. I'm in mid sentence here ...

It's about how to prioritise work.

That's right, that was it. Yes, so they had this priorities committee and, well, all the usual ... Stan Gill was on it and David was on it. Different people were on this thing and there, in all of that, EDSAC acted as a catalyst to finding out what was going on in science in Cambridge. I don't think anyone has actually written this but, when you get into the detail of what the priorities committee did, anybody who wanted to use that computer had to apply and they had to apply every time they wanted, not to just have a run, but to attack a new problem. So they were taught how to programme, they were taught all about David's subroutines, there was no computer language except there was machine language, of course. And ... what was I saying ... so people would apply to do it and the first thing they had to learn was how to use the fire extinguisher. [laughs] So there we go. I can give you an explanation, some time, on how to use the fire extinguisher at Cambridge University, although it doesn't exist any longer, at least the maths lab doesn't. But, so Wilkes was sitting in the corner, seeing as time went on, seeing more and more, getting into more and more detail of what was going on in Cambridge, and that got to DNA and all sorts of things. I mean, someone or other could have, but hasn't, written a sort of history of post-war science in Cambridge or something like that. And Maurice could have done it.

00:34:40

What did you do on it? What programmes did you write?

I did a lot of work for JCP Miller, Jeff Miller, did you meet Jeff?

No.

Ok, he was a wonderful old chap, just a gorgeous chap. He was '*Mr Numbers'*, he was a numerical analyst and he had a global network of numerical analysts and he knew about all these old people who'd come up years and years ago, centuries ago, with ideas for getting numerical solutions to continuous equation problems. And he gave me these to do, and I just churned these out. I mean, there's not much more you

could do, you only had 1,024 words and say, half of that was data, and half was programme, and I was his sort of numbers slave [laughs]. And he could look at results and say, 'Norman, you've got an error there in your programme' [laughs] and all that sort of thing, so that's what I did. But it wasn't handling data, there was no data. Computers weren't data handlers.

It was calculations ...

They were just calculations and you can do enormous amounts of calculations on a 1K machine. EDSAC 2 was a 1K machine to start with; it wasn't until after two or three years that they realised that they needed to get more core which meant more bits in the address register, so they had to do quite a bit of post-engineering.

00:36:22

So, we're talking about computers in architecture that would be recognisable today, really, store programme computers, input, output, peripherals, albeit of their time, but it's a recognisable architecture, isn't it?

Oh, absolutely, yes.

So, we could simplify it and say that, for the last seventy years, all we've been doing is polishing the same idea?

The same idea, if you come right back to it, is the stored programme. Ok. And if you make it big enough you can handle more and more ... you don't need an enormous computer to do mathematics, you don't need it to solve differential equations and to perform an integration. You can do it on a small machine. Computers ought not to be called computers any longer 'cos, I suppose, and I don't have any figures, but I suppose that the amount of time actually computing anything has gotten very small. It's mostly data searching, it's mostly looking at vast quantitites of data, sorting them and reporting them. And you can go in to your pc or your telephone or something like that, and get something out of a computer that's never been produced before, like -

and I'm just making something up, you know – how many bald-headed eagles know what a cosine is, or something like that. [laughs]

Yes, I know what you mean, searching databases for information as opposed to calculation.

Yes, they are searchers. I think, if I were allowed to rename the computer, I would call it a searcher or a searching machine or something like that.

So, sorry I'm dotting around, but going back to the early fifties, people like Maurice Wilkes, David Wheeler and so on, would you describe them as academics? Were they engineers? Or ...

That's a tough one. I don't think they would call themselves engineers but they did a hell of a lot of engineering. I mean, I'm not a tennis player but I've played a lot of tennis, you know. But they were, when you get onto this, they were in the middle of ... you can't hang a label around their neck ... I mean, David ... if you read Wilkes's books ... you see, David Wheeler was this ingenious bloke who showed up for coffee in the morning and tea in the afternoon. To be around David Wheeler, like to be around Pres Eckert, was in itself something from heaven, to be able just to chitchat with them and so forth. I remember, David, when he was staying with me in America, and I was trying to make a bookshelf. He came up and he made this bookshelf for me, he'd probably never made a bookshelf before or since, a superb bookshelf. So they were a combination of things. You wouldn't ask any of those to design an aeroplane, but we had a lot of computer programmers writing programmes that made computers and I think that, when I think of that, I think of the 727 and the invention of CAD, which we did, in about 1961, in Boeing, and here you had chaps who were basically mathematicians, they were good programmers and so forth, and they found out ... 'Explain wings to me.' 'Tell me all about wings and I'll write you a programme and it'll make you the wings' and all that sort of thing. We weren't engineers. I'm not an engineer. But we do a lot of engineering. With David and Maurice and so forth, and Stan, they did a lot of engineering but they weren't engineers. What was your question again?

Well, it's really, would you describe them as academics or engineers?

Well, they weren't academics because they did things. They were lying on the floor [laughs] working the whole night operating the machine and so forth. They were a different ...

Practical people.

They were. And I can tell you this, I'm so glad that I fell in with computer programmers because it's a different type of *homo sapiens*. You can only know this if you do it, it's strange.

Yeah, I can remember some of the silly bits of software I wrote that really don't mean anything to anybody, but I'm quite proud to have written them.

You have terrific ... you're in the middle of Christmas, your mother comes along and says, 'Hey, you're supposed to be enjoying yourself.' I am, but I'm secretly coding instead and stuffing the sheets under the pillows.

I remember writing a silly bit of code that opened a ticket barrier in a car park. That was a project I was given. I remember writing it and thinking, this is amazing really, I'm writing silly bits of lines on a paper and this is going to make a barrier go up and down when I've finished.

I wish you'd go back and redo it because all these youngsters, making a mess of the world because, you know ...

That's because they're not testing it properly.

Not testing it at all.

Another debate probably.

I know, it's horrible. [laughs]

But we're not talking about today, we're talking about the good old days.

Yeah. So this part of the fifties, then, it sounds like quite a lot of fun, actually.

It was terrific fun. I, at one time, I took a fifty per cent cut in salary to go and have fun somewhere else. And my whole life was ... I'm a project man and I lived in Canada, the United States, Italy, Norway, back and forth and round, just enjoying myself and not ever arguing about salary. And when I got promoted to a point where it was boring, because it was managing people, I'd get out and go get down to the bottom again. And that's what it is, terrific fun. Had I been five years older I wouldn't have had this fun because I would have become, I suppose, a mathematics teacher. Too late now, to go and learn anything about programming, and would have spent my life losing my hair and teeth and all that stuff, teaching calculus, yes, which I do of now as a hobby.

00:43:01

So, a couple of other points I came across, about Maurice Wilkes and David Wheeler, is that they invented things that are now self-evident but obviously someone had to think of them first. So, Maurice Wilkes is credited with inventing the concept of micro-programming.

He did. Yes.

Which, of course, every CP works now with micro-programming.

And no one is interested in who invented it. [laughs]

No. And David Wheeler invented the subroutine.

The subroutine, yes.

When I first started work I worked at Kodak, who were early adopters of computers, and the guy I worked for was a friend of David Wheeler.

Oh, what's his name?

I knew you'd say that and I can't remember. [laughs]

So you've got my problem. [laughs]

But he said to me that Wheeler had said to him, 'If he'd had a ha'penny for every time somebody used a subroutine David Wheeler would be sitting on a beach by now.'

Absolutely, absolutely.

And, you know, it's another invention which now seems self-evident – how could you possibly not have subroutines? – but someone had to think of it.

They argued about it because there were some people, for example, at the Institute for Advanced Studies, who just refused even to discuss subroutines, they wanted to write everything out themselves. It's unbelievable. I've been re-reading books I read years ago, all lying around the house and now, I suppose I've read this before but had forgotten, you can't imagine how people could discuss things in this way, how they could argue against using subroutines. Subroutines were the building blocks of programmes.

Yes, exactly, all sorts of benefits and how could we exist without them now?

That's it, all you needed to know was, what sort of data to give it and what to expect out of it, how to test for it.

So, I've got a list of names here, and I think we've mentioned many of them, have we talked about Tommy Flowers? I can't remember.

Yes. I haven't met Tommy Flowers but he gave a lecture, at one time, that I went to, at the University of ... what's it called .... University of Old Age or something! [laughs] which puts on lectures once a month.

And was he an engineer, charged with creating Colossus, that's right isn't it?

Yes.

So mid-forties would be his moment of fame, probably.

That's right. Those guys were engineers. What's interesting, if I can just pick up what you said there, computers started with university people. All those, the great names of the first flush, were all university people and computers were built in universities. And why were they built? Because there were people like Wilkes and so forth who wanted high speed arithmetic done. And they got the money from here, there and everywhere, to do it. And Wilkes was fantastic at getting money. He would apply ... oh I've forgotten what the places were called ... but he would apply and the money would come immediately. And he even sometimes spent the money before he even applied and still got the money. [laughs] You see, a profit motive was no way of getting computing started because it was pretty clear to anyone who thought about it, and there probably weren't many, that computers weren't profitable. One example of that is drafting machines. When we invented CAD there were no drafting machines on this planet, but we turned an enormous, a multi-ton Kerning trekker cutting machine ... oh, what do you call it ... anyway, it was used for cutting blocks of aluminium into different shapes and so forth. We used a thing like that as a drafting machine, we replaced the cutter head ... a milling machine, that's what it was, I'm sorry ... these milling machines were enormous, and we replaced the cutting head with a diamond scribe and so, as we got our programmes to the point that the numbers looked ok, we tested them by feeding those numbers back into a D to A converter attached to the milling machine, and ... here's a piece of history that no one knows about. The first engineering drawings ever made with CAD were not made with a pen on paper, they were made with a diamond scribe on great big sheets of aluminium. They were brought over from the factory, so the factory did all the drawings for us, and we emptied the 1085 building of all of its furniture, got all the desks out, got great

big clear floors, and put these sheets of aluminium on the floors, and the engineers, who knew – these were aeronautical engineers – they knew what wings were and knew what to expect, they spent three or four weeks, at least a whole month, on their hands and knees, on the floor, looking at these scribed lines with little tiny microscopes, comparing. And we, first of all, ran ... for the second time, the 707 was running before computers were used. We had computers at Boeing but there was not much engineering being done. By the time we computer people came along and started implementing the ideas of CAD that we had, they, rightly so, decided to give us the 707 key points, you know, the highest point of the wing and one or two points along the trailing edge and all that sort of thing. And we had to re-draw, not re-engineer or design, but re-draw, the 707. And this went on for four or five weeks, we had no idea how our drawings were compared with the 707 pen and ink drawings that these guys had worked with ten years earlier. This was testing a computer programme. The only way to test it is to run it. And so we did, by using this diamond scribe thing on a multi-ton machine. The chief engineer of the 727 had us in to the theatre and we all sat there, in our rows, not knowing what to expect. And I remember his first sentence, 'Gentlemen, lines, engineering lines of this accuracy, have never been produced before in the history of engineering.' That's incredible. We had no idea how things were going, they kept it all to themselves, and said, you know, 'this is what we are now doing at Boeing, we are now using the computer' – these are his words, not mine – 'to draw and cut the parts of the 727 aircraft'. And that's how the whole thing started.

00:51:22

Vindication. Yes.

I mean, one of the most exciting days of my life was hearing that, I can tell you.

Which is the point you made earlier, really, isn't it, about computing itself is quite interesting. Computers are quite interesting but what you use them for is really where it gets interesting.

Absolutely, and anyone who has stayed, sort of, academically, they would have wonderful ideas, great fun in working with ... what do you call it ... with block diagrams and that sort of thing, or giving lectures about computing and so forth, but actually using a computer to make things, and things that fly and all the rest of it, is ... when I think back on it, it's been an incredible time, an incredible life.

Have I missed anybody out, of the famous people that you wanted to talk about, or well-known people?

Oh, you want some more famous people? Just a second. Oh, shall I tell you about what I did with Harold Wilson?

Yes, ok, that would be good.

Shall I do that?

That was mid-sixties, I guess, was it?

Yes, that was. I've got a letter from him, here, for example, I've got a bunch of letters from him. What date is this? ... ah, November the fifth '64. From the Prime Minister, 'Dear Norman,' - how's that!

It's a good start. [laughs]

So, that's a little story. I got involved with a lot of things, for example, I was the United Nations Consultant to the Hungarian Government, and then Yugoslav Government, on computing, and used to go down there and talk to them about computers and so forth. But the most important thing I've done outside actually working in industry is with Harold Wilson, the Prime Minister. Of course, he's probably forgotten by now, or the youngsters have never heard of him, but anyway, I just bumped into him in Seattle and he said, 'I'd like to come round and have a chat with you.' So I said, 'Ok, come on round, come round tomorrow if you like – tomorrow's Sunday, I've got nothing to do.' So he came round and I thought, well, you don't get a chance like this every day to chat with a Prime Minister, I wonder

what he wants to talk about. But I'll tell him what I'm doing. And what I was doing was things that I ought to have been doing in England. And it wasn't just me, I had quite a few Englishmen, I'd brought a lot of Englishmen over by advertising, which I did in the Observer on Sundays, computer programmers. I couldn't find enough home-grown, ie American, programmers. We would hire people with engineering degrees or mathematics degrees and train them in, well, in Fortran really, and tell them what aeroplanes were, and put them to work. And bright people were ok, but it would be much better to get seasoned programmers who come in with ideas and so forth who would really like the excitement of working with computers. So I told Wilson. First of all, I found him an easy bloke to talk to and he would puff away at his pipe and drink my wine, and encourage me to go on telling him. And I didn't know what was going to happen but he really understood that Britain ought to bloody well use computers. One problem was that the computer industry in Britain ... if you wanted to buy a British computer they were half-a-dozen very, very small companies, and if you were a big thing, if you were making aeroplanes and so forth, you want a supplier of equipment to be able to suddenly deliver something. So the safer thing to do was to buy an American computer, not to buy a British computer. But if you're going to buy a British computer then let's take all these small companies and put them into a single company and, in that way, establish a British computer industry. And I didn't know what he was going to do, I was just chatting to him and he could have gone away and forgotten all about it, but he didn't. He asked me to write him a, he called it a 'memorandum'. I call it a blueprint [showing papers], this green thing here, and I sat down ... he wanted to know, 'What shall I do about computers in Britain?' A memorandum, what's that, half a page or something like that? I don't know ... and so I thought, 'Well, here I am, I don't get an opportunity to write a memorandum to a Prime Minister ... even though I'm not going to get any money from it ... he paid all his other guys ... so I sat down. And I was living then in Trondheim, in Norway, building up and sort of creating the computing department at the university. It was a good university, marvellous fun, wonderful students and all the rest of it. And, up on the fourth floor of the bank there, there was a baroque ... what would you call it, their boardroom, where they had their meetings ... one long shiny table and pictures of the old bankers around hanging on the wall. And I knew the bank manager and he let me have this room and this table for a whole weekend. I had no 'phone, I didn't need it, there was no one to ring, but during the week I was just too busy building up the

computing centre, which involved not only doing things inside the computer but bringing industry in, and telling them what computers were, and getting jobs for them. A lot of my salary, half of my salary, was with doing jobs for industry, yet here I was doing ... I had my feet in two camps already, so I couldn't do that during the week. But I sat a whole weekend and I wrote this thing here, and this was then the blueprint to the British computer industry because in it I said, 'You don't need a computer industry, because America is producing lots of really good computers. The thing to do, your top priority, is to use computers.' Ok. Not to build them. 'But if you absolutely have to build them then don't build them with half a dozen small companies, put them all together into one company.' And they did and I called it the BCC, the British Computer Company; they called it ICL, International Computers Limited, which I felt was a bit deflating. And I've got a date here, when that was created, if you'll just give me a moment to consult my notes here. I had to ring up an old IT person to ask him [laughs] and he said that he quit when they were bought out eventually, just a second, ... [flipping through papers] it doesn't matter, really, it was about 1964 or '65. It was put together, and eventually the Japanese bought it up and he told me that it finished [looking at papers], if I can see it very, very quickly, I can't, I've got too much written.

Was it Fujitsu, who bought it?

Fujitsu, you're quite right, yes, it was Fujitsu that did it, anyway, then it disappeared so then no longer did we have a British computer building industry. It was the same in Scandinavia, they had two, first of all, a Swedish university, then a Danish one, that started building computers and sitting around twiddling their thumbs looking at them and not getting anything done. And I was working in Norway, at the time, and people had the idea of building computers there and one of them actually started but it didn't come to anything. But ... stop trying to build computers, take the best possible people you've got and put them out into industry, and that's what they did and it really worked well. That's a long answer to a short question, what was the short question?

1:01:03

Famous people, other famous people you met.

Ah, other famous people, yes, sorry.

So, Harold Wilson.

Ok, and so he called it the leading edge of ...

...Whitehall technology

Yes, those weren't my words but I said, 'Jolly good, Harold'. So I did a lot of things writing to him, though, I had other ideas, non-computing ideas, but I won't bore you with that. Ok, let's see, I've got a list of names here, somewhere. Ah, Peter Naur, he's not just hardware but software. Peter Naur was a Dane. So, I told you about Rutishauser, who invented ALGOL 58. Well, Peter Naur came on and changed that to ALGOL 60 and had a team of people around him, I don't know who else was in the team, and got that ALGOL thing going. And I knew Peter quite well because he was down in Copenhagen when I was up in Trondheim. Two or three years ago, I had to go over to Denmark, to talk about how computers got started, to a bunch of Danish graduates of one sort and another. And if you go back in time, Peter Naur's name was well-known all round the world, in Australia, in Germany and everywhere, but when I started off, I said, 'Does anyone here remember a chap called Peter Naur?' and nobody did. So his name is forgotten, and I think a lot of names are forgotten.

1:02:53
He was the Naur of the Backus-Naur form.

That's right. John Backus, of course, invented Fortran. Naur, you can say, invented ALGOL and BNF was the Backus-Naur form. Some people say the N stands for normal form, but anyway, it might have bounced back and forth between the two. I'll try and give you a few names, here. Just a second. There's Newman ...

Max Newman, is it?

Max Newman, yes. He had an office either in Manchester University or in Ferranti, and it was described by Maurice Wilkes as 'late lavatorials' [laughs]. Yes, computers came from the weirdest of places, I mean, in Cambridge, up in the maths lab, it was up on the third floor, in the physics laboratory, in a scruffy ... you see, a computer comes along late in the game of building things and so forth, and they didn't have any money, much, so we took the premises that we could find. And, as a result, you had to learn how to use a fire extinguisher. And there's Newman and ... Freddie Williams of the Williams tube, Tom Kilburn...

So Williams and Kilburn were Manchester people, weren't they?

They were Manchester, both of them.

Manchester Baby and that sort of thing ...

That was the Baby, yes. The Baby, I've got a Baby bloke who lives just around the corner here, [laughs] still alive but dead, living around here, either because they've been to a local university or had moved over from Dollis Hill. But, anyway, there's a chap just round the corner here, he worked in Manchester and the Baby was nothing much more than just a test bed for the Williams tube. Whereas, once the EDSAC really got going in Cambridge, more and more scientists, one way and another, got to use it and so, I'm only repeating myself by saying that the computer made Cambridge science sort of well-known and describable, except that I don't know that anyone ever did. Kilburn ... yes, now, right, just a second. [looking at papers] I've got them here. Oh, Cobol, yes, there was another language called Cobol, Common Business Oriented Language, 1959. That was invented by a strange lady, a very, very famous lady called Grace Murray Hopper. She was a Rear-Admiral, she started off as a programmer and invented the Cobol language and she was well-known for carrying a piece of string around in her jacket pocket and, in the middle of a discussion, would produce it. It was a foot and a half long, or something like that, and she would hold it up and say, 'This is how far an electron moves in a microsecond.' [laughs]

1:07:15

Did you meet her, ever?

Yes. Oh yes, because she also worked at Univac.

Right.

She was a DAR. She was a Daughter of the American Revolution.

She strikes me as a little eccentric, perhaps?

Oh yes, everybody was. [laughs] You wouldn't design a computer if you weren't, or certainly that piece of software. Tommy Flowers, you mentioned him, just a second [looking through papers] oh yes, John Carr, Marvin Minsky, did you know him?

No.

Marvin Minsky and Jackson W Granholm. Have you heard of a Kludge?

Yes.

Jack Granholm invented the word 'Kludge' and what he was good at was joining all sorts of incompatible machinery together. He was another Boeing chap. Incidentally, I think most well-known people are university people, because they publish and they get read and they go to meetings. Of course, the one who did it more than anybody else was Maurice himself, he was always off to things. He had terrific contacts, he had this global network of people. He would go to New York and he'd pop off in Pittsburgh and pop off somewhere else and so forth. Everyone knew him, he was welcome everywhere. We industrial users of computers don't publish. I've published a lot of books but there's nothing ... I've never published papers.

Yes, it's a corporate memory, isn't it, rather than individual memory, if you like?

That's right, yes.

Yes, the intellectual property belongs to your employer and all that sort of stuff.

That's right, and that stopped me, stopped one of my books. Have you heard of Tony Jay?

Yes.

He wrote 'Yes Minister'. Ok, he's dead now, he died a couple of years ago ... Because I was the chap down in the engine room of Boeing, the people at the top ... Boeing was a member of the American Management Association and they had to go and give lectures, give talks and that sort of thing, but they didn't know what to talk about because they were asked to talk about computers. So I became ...

... a speech writer ...

...a speech writer [laughs], exactly. I wrote all these speeches, and I've got them all printed up somewhere in the house, but at the time I was turning it into a book to be published by Prentice Hall. And I wanted to invoke the name of Machiavelli in it because I though that ... so one thing, in Cambridge, certainly in my day, there was no discussion of management because there were no professors or no academics at all who knew anything about management. That may still be true, I just don't know, but it certainly was then. And so, I came into Boeing totally innocent of any knowledge of management and, of course, high up above me there were these blokes who'd been engineers during the war, on the B17 and the B29 and everything, and they had become managers. And, of course, they knew nothing, they were paying the bills of computing, and they were doing this because the engineers said that they wanted to use these computers, and our job as the programmers was to write the programmes for them to use. Ok, what was I saying ...

This was about senior managers using computers.

Yes, so I became a speech writer and then, I had all these things that Prentice Hall were going to publish. But Boeing wouldn't let me do it because each chapter in the book was basically a speech, given by a Boeing manager to managers of computer

companies. And all the others all signed a document saying they didn't mind ... because these were very critical, all very critical. So each of the other chapters in my book about management and Machiavelli, they said, 'Well, all right, you can have it, you publish it, that's all right' except IBM. IBM wouldn't allow us. That's not quite true, they said, 'Look, we don't own your speech, you own your speech, we were the recipients of it, and if you want to publish it it's up to you.' But Boeing wouldn't do that. Boeing's top management wouldn't allow me to publish that book, and I still to this day haven't published it. But it got around, if I can just tell you this little story. It sort of got around the world somehow, as it was bootlegged. Someone or other made copies of all these speeches [laughs] and I was over in Copenhagen, and I was talking to the head of computing for SAS, the Scandinavian Airline System. Ok. And this chap had all the management books on his shelves. We were chatting away and I said, 'Ah, you've got all the management books.' 'Aah', he said, 'they're nothing. Look, this is what I've got.' He opened a drawer and he took my speeches out.

Yours!

Yes, I said, 'I wrote that.' 'You did?' 'Yes.' It's got my name on it, you know, probably on the first page, but all these others, these were speeches given by Charlie and friends and so forth. And he said, 'It is vastly better than the other books.' [laughs] But then, one Friday in Time Magazine, there was this article about a book written by a bloke called Antony Jay ... this is a long time ago, around fifty years ago or something like that ... and it was called 'Management and Machiavelli'. And I've got it on these shelves here. When I came up with this thought of Machiavelli, like, 'No one else will think of this', and so I wrote to him, through the publisher, and I said, 'Dear Mr Jay, you rotter, you've stolen my book, you've stolen my jokes' and all this. His book was superb. His book was absolutely superb. It wasn't a bunch of speeches like mine was, it was a proper book, probably the best book ever written about managers, management and so forth. And I got a very kind letter back and he said, 'When I first had the idea of bringing Machiavelli's name in, as modern management is no different to Machiavelli, I thought, someone, somewhere, somebody will have come up with this idea.' And that was fifty years ago and we've been friends ever since.

Marvellous.

He has, well, he's dead now, but he and Jill, his wife, Lady Jill Jay now, they had a couple of barns down in Somerset and we have an annual RAF reunion from 1949 down there. And I've just had one.

Splendid. Sadly, we probably ought to bring it to a close, as we're running out of time.

Yes, a lot of things are probably more outside, you'll have to cut a lot of this junk out.

Oh, I don't know. I was thinking, actually, on the way over. The strapline of Archives of IT is 'capturing the past, inspiring the future'. And I was thinking that we've captured the past, that's for sure, but I think there's a lot you can talk about here to do with inspiring the future too, because the stories you've told about the fifties and the sixties, they're just as applicable today.

They are.

You know, there'll be a university department somewhere that's inventing something we don't know yet.

That's right.

And, if we roll with that, it's going to be just as exciting as the times you had in the early fifties, we just don't know what it is yet. Politicians need advice still, goodness knows ...

Do they [laughs]

... and this is going to be an exciting job for somebody in the IT industry at some point, advising politicians. So, they say history repeats itself and I think that's perhaps how to inspire the future here. You know, the things you did, fifty, sixty and

seventy years ago in some cases, are still things young people getting into IT now could find themselves doing in the next fifty years.

They could. And they will. And it's going to happen. We're just not clever enough to know. You see, little Billy Gates, when he was sixteen, lived very close to me and he was the son of a chap I'd worked with who I'd put as the project manager of the CAD project. His son Dan, who's now Professor of Physics down in southern California, Dan Irwin and Billy Gates were together at this school, in Seattle, and ... I should have bought shares in Bill Gates when he was sixteen, they would have been quite cheap then, but who would have known ...

Who would have known? Who could possibly have told at tat time what he would later go on to do, so yes. Right, well, thank you ever so much again, for your time Norman.

Well thank you, and I hope I haven't bored you too badly.

Not at all, it's been absolutely fascinating, like last time, so thank you very much.

Thank you.

[recording ends at 1:17:49]