Oral History of Steve Furber
2012 Computer History Museum Fellow

Interviewed by:
Doug Fairbairn

Recorded: February 3, 2012
Manchester, United Kingdom
Mountain View, California

CHM Reference number: X6409.2012

Doug Fairbairn: My name is Doug Fairbairn. And I'm interviewing Steve Furber from the University of Manchester. We are talking on February 3, 2012. And this is an oral history covering Steve's background up through the present day. So welcome, Steve and glad to have you here.

Steve Furber: Hello.

Fairbairn: So as a first step in this this recording is being done partly in support of the Fellow award that has been given to Steve Furber from the Computer History Museum in Mountain View California. And it is in recognition specifically of the work he and others have done on the ARM processor. And so we'd like to spend just a few minutes at the beginning of this interview just talking about that. And I'd like to ask Steve what he thinks the most important things about the ARM are and about its impact, the impact that it has had. So just spend a few minutes talking about that, if you could, Steve.

Furber: So the ARM microprocessor was designed at Acorn Computers in England in the early 1980s with the specific objective of giving Acorn a new processor for its desktop computers which at that time were principally sold into education. However, the processor turned out to have pretty impressive power performance properties which were somewhat serendipitous, and so in 1990 it was moved out into a separate company, ARM Limited, that was set up jointly by Acorn, Apple and VLSI Technology. And it turned out to be particularly well suited to the emergent system on chip market in the early nineties and in particular to the rapidly growing mobile telephone handset market. So ARM became the de facto embedded process of the systems on chip, and has now grown to a huge market. I believe to date over 30 billion ARM processors have been shipped and they're in a whole range of products. They're still dominating mobile phone handsets. But they're also in portable music players, Apple's iPad and iPhone, digital cameras, hard disk drives, ADSL terminals and a whole range of other examples of consumer electronics. So the processor has become a very important part of the global electronics industry today.

Fairbairn: Okay. So we were just talking about the impact that the ARM has had on the overall electronics industry. So thank you for your comments on that. Steve, what I'd like to do now is step back and talk about sort of where it all began, where you were born. Tell me about your early family especially as it may have impacted your later choice or career or direction that you took? And just give us a feeling for what it was like for you growing up and where?

Furber: I was born in 1953, so not very long after the end of the Second World War, a few years after. And my parents moved when I was a year-and-a-half to what became my childhood home which was in Marple in Cheshire which is about ten miles south of Manchester. They lived in the middle of Manchester when I was born and then moved out to Marple and I grew up in Marple which is a nice suburb in south Manchester. My father had spent some of his wartime years at Rolls Royce working on aero engines, and then he moved to UMIST in Manchester, one of the technical colleges, before joining the nuclear power industry in the late fifties, and he spent nearly all his career in the U.K. nuclear power industry. I grew up in a comfortable home. My father was an engineer, and so engineering was always an important topic around the dinner table at home. I went to school locally in Marple and then for secondary school I went to Manchester Grammar School in the middle of town which is one of the most academic schools in the area. At MGS I grew to like maths and physics and science, in general. I generally didn't like history or geography, and shifted in the general direction of maths and the sciences. And so when it came to choosing a course to take at university it was a fairly easy choice to go to Cambridge to read maths.

Fairbairn: So what drove your decision to go to Cambridge? Was that just because it was the best school around? Or how did you find yourself going there?

Furber: I went to Cambridge because it was generally viewed as the best U.K. university for maths and the sciences, and my school was very keen on sending its pupils to Oxford and Cambridge. So, I guess it was assumed through school that Cambridge was the right place for me to go, and I fairly readily accepted that general guidance.

Fairbairn: Did you have any hobbies before going off to Cambridge, you know, fiddling with electronics or other engineering tasks? Or you mainly focused on school related activities?

Furber: I had a number of hobbies at home. When I was young I was very keen on Meccano which is an engineering construction set where you can build cranes and cars and gear boxes and things like that, and then in my teenage years I developed an interest in aircraft which expressed itself in some fairly unconvincing attempts to build and fly model aircraft. So I liked aircraft. I liked understanding how they operated. I liked making them. I wasn't very good at flying them. As part of my model aircraft work I did some electronics. So I had a single channel radio control system which, with a bit of help from my physics teacher at school, I converted into a kind of proportional control. So instead of having a single button which you pushed once for left and twice for right I had a stick which would move the rudder almost continuously between left and right. So I did a little bit of electronics but not a lot.

Fairbairn: And did that one fly?

Furber: That one flew, yes. I describe my teenage model aircraft years as radio affected aircraft. I think the term radio controlled probably implies a bit too much control! And I always used to go to the field where I flew them with a couple of supermarket carrier bags to bring the bits home in.

Fairbairn: Having had a little experience in that area myself I totally sympathize with the results that you described.

Furber: Yeah. I did get better at it in my later years, so I actually carried on flying aircraft after university for a bit, and I got into electric powered model aircraft. And then more often than not I took them home in one piece.

Fairbairn: Yeah. So did you have brothers and sisters? Did any of those have any influence on you?

Furber: I was brought up with two sisters, one older, one younger. So I was the meat in the sandwich as they say. And my older sister was pretty independent in her teenager years, but I had quite a lot of time with my younger sister.

Fairbairn: Okay. So you're interested in maths and science. You headed off to Cambridge. Did you have any specific goal in mind in terms of a profession? Or are you going off to university to explore what that might be?

Furber: I didn't have any clear idea of where I would go professionally. I guess I was in the normal state. I had done pretty well at school so going to university and getting a degree seemed the next logical step. Maths was a subject I seemed to be able to cope with fairly easily, so I went off to read maths. Actually, my primary interest in the university was leaning my maths towards the aerodynamics side so I could use my maths skills to understand a bit more about how aircraft worked.

Fairbairn: So you went off to Cambridge. Take me through that and how it developed and how you found yourself moving from aircraft to computers and so forth. How did that all come about?

Furber: So I had a little bit of exposure to computing at school. In my last year of school one of the teacher's who I've met up again with recently, called Neil Sheldon, he introduced computing in 1970 into the school. And there the procedure was we had 80 column cards with-- they were pre punched, only every other column was punched-- so effectively we could use 40 columns of these. And we pushed the little squares out of the 80 column cards with a sharp instrument and that's the way we put our program out on these 80 column cards and they were then posted to Imperial College in London. Fed to a computer and about two weeks later the printout came back.

Fairbairn: Now, that is a long turnaround time.

Furber: That was an early exercise in being careful in how you write code because the debug cycle was very long indeed. Then at Cambridge studying maths there was a little bit more interaction with computers. The maths department had a Modular One computer which was an interesting machine. I remember it had a Tektronix storage scopes so the display would come up and you could write on the display. But then when you wanted to erase something you had to sort of flash the whole display and start again. But I still wasn't really leaning towards computers at that time. And in my maths degree I had some exposure to aerodynamics. I did a project on bird flight which brought a bit of bio inspiration into my work which was going to come back a few decades later. And then I chose to go on to the Engineering Department to do a Ph.D. in aerodynamics at the end of my maths course. So I moved to engineering. And my Ph.D. started off pretty theoretical but gradually turned intp a combination of experiments and computer modeling. So I got more exposure to computing in the course of my Ph.D., and by the end of my Ph.D. I had started building my own computers and using those to support my experiments.

Fairbairn: So what year did you go off to Cambridge? And what was the period you were working on the Ph.D.?

Furber: I went to Cambridge in 1971. I finished my undergraduate degree in '74. I did a fourth maths year which was called part III maths from '74 to '75. And then my Ph.D. was '75 to '78. After my Ph.D. I got a research fellowship to continue with the aerodynamics and that was '78 to '81.

Fairbairn: So were there certain professors or other people that particularly influenced you during your Cambridge years?

Furber: There were quite a number of characters at Cambridge. I remember John Conway, a lecturer who was early on in the course, who was responsible for a lot of work on cellular automata. And he would give lectures wearing a large black gown which was fairly common in those days. And he'd give a lecture and then he'd come in the next lecture and he would say, "Forget everything I said last time. I got it all wrong. This is how I should have done it." I also had some teaching in my fourth year maths course from Professor Sir James Lighthill who was a very big name in the U.K at the time. He was Lucasian professor at Cambridge, which is the chair that Stephen Hawking occupied subsequently, and he was very dramatic. He was quite elderly by the time he was teaching me, but he used to demonstrate how to de-crab an aircraft when you land it by applying a differentiated delta function to the rudder. He used to sit on the front bench and illustrate how you kicked the left pedal and then the right rudder pedal to de-crab the plane as it touched down. And he was also somebody who always found the blackboard as supplied by the university a bit limiting, and his chalk used to end up going around the walls as well. The final influence at university was my Ph.D. supervisor who was another great character. It was Shon Ffowcs Williams who was, I think, best known for his involvement in the attempt to reduce the noise produced by the Concorde. He had an aero-acoustic outfit in the Engineering Department. But he was somebody who did everything he did with great enthusiasm whether it was work or the bowls played in college after lunch. He always put a great deal of energy and enthusiasm into everything he did, and I've always found that a fairly inspirational approach to life.

Fairbairn: That's great. So you mentioned earlier that you became better at your aircraft flying in the university. Is there any particular stories or interesting things?

Furber: Well, it was really after university that had I gone back into the model aircraft flying. While an undergraduate at the university I tried some real flying so I joined the university glider club where I spent Wednesday afternoon for most of the year standing out at Duxford Airfield, often in the freezing cold, spending a lot of time watching other people fly guiders and very little time doing it myself. I think by the end of the year I racked up about 14 flights and a grand total of about 54 flying minutes. So I just about got to the point where the trainer would leave me to do the entire flight, this was take-off, circuit, and landing. But I decided for my final year that this had cost me too much time and it was my final year and I should concentrate on my university work a bit more.

Fairbairn: So what was your thesis on, your Ph.D. thesis?

Furber: My Ph.D. thesis has the interesting title of "Is the Weis-Fogh principle exploitable in turbo machines?" That probably requires a bit of explanation. Torkel Weis-Fogh was a zoologist at Cambridge who had done lots of research into the flying motions of different animals and insects. The Weis-Fogh effect was an aerodynamic mechanism employed by a very small insect, the chalcid wasp which is used in greenhouses to control aphids, and it's so small that its Reynolds number is too small for conventional flying mechanisms to work. It overcomes this problem by clapping its wings together and effectively using each wing as the starting vortex for the other, and my Ph.D. was really trying to see if one could apply that principle in jet engine compressors, because generally in a compressor, interference between consecutive blade rows is considered a nuisance, something to be avoided. But what I was able to demonstrate is that, in fact, under some circumstances the interference between consecutive blades can actually improve the operation of the engine.

Fairbairn: So did that find some practical use in engine manufacturing?

Furber: I don't know where it went on to be applied. I think the fundamental difficulty with applying it was that in order to get this effect you had to make the blades operate with very low clearance. And, of course, low clearance is a problem when the engine ingests a bird, for instance. So I suspect they would have problems passing the flying chicken test with the blades that were very close together. But I think it offered some insights which are still being studied. So I was looking a month or two ago and I saw, in fact, that the papers that we wrote back around 1980 are still being cited occasionally. So people are still thinking about this area and thinking about what the implications are.

Fairbairn: Interesting. So throughout this university period you were using computers as tools to do whatever calculations or create models or whatever, is that correct?

Furber: Yes, that's right. And, of course, the turning point was when a few enterprising students decided they wanted to form a university society which was called the Cambridge University Processor Group, and that was for people who liked building computers for fun. While I wasn't a founder of that society I did attend the inaugural meetings and was one of the first members. So that introduced me to a bunch of people who had the knowledge and skills to put computers together, and I started building computers as a hobby in connection with that society.

Fairbairn: Throughout doing your Ph.D. work did you, at any point, sort of have a clear vision of going to work in that area. What was your thoughts as you were doing your Ph.D. as to your future career?

Furber: In my teens I had done work at school studying maths and played with planes as a kind of hobby, and then at university eventually the aerodynamics had become the day job and the computers had become the hobby. So I was used to shifting something from a hobby end of activity to my mainstream day job. But there was a sequence of events that caused me, eventually, to become primarily involved in computing. The University Processor Group was undoubtedly the start of this process, and then Chris Curry and Hermann Hauser decided they wanted to start a venture in the microprocessor area and they went looking for technical people to help them with this and the University Processor Group was an obvious place to look. So Hermann approached me, I guess in about ’77, and said he was thinking of starting a company, was I interested? To which I said, I only do this as a hobby. I don’t claim to have much skill in this area. But if you think I can help I’m willing to have a go. So I got involved in the embryonic Acorn basically as a side activity. I was doing it evenings and weekends with aerodynamics still being the day job, and Acorn then began to grow. In my final year as a research fellow, Acorn got the contract to build the BBC Microcomputer. So at the end of my research fellowship, where I had to take a decision as to whether to go and find another research position in aerodynamics or move subject and join Acorn, it was actually a fairly easy decision because Acorn’s position was very interesting at that point.

Fairbairn: So what were your initial-- you said you started working part time at Acorn in ’77. What kind of activities were you involved in?

Furber: I did various things for Acorn in the early days. They weren’t called Acorn at the beginning - they were called Cambridge Processor Unit Limited, or CPU Limited - and the first job they got was building a microprocessor-based controller for a fruit machine, a one-armed bandit, and this was the time when fruit machines were making the transition from electromechanical control to electronic control. So this was very early in that development.

Fairbairn: So what do fruit machines do?

Furber: Well, a fruit machine is a one-armed bandit.

Fairbairn: Okay. Got it. It’s a slot machine. We call it slot machine.

Furber: Slot machine, yeah, that's right. In the U.K. they're often called fruit machines because usually the dials have pictures of fruit on them.

Fairbairn: Yes, I understand exactly where the name came from now.

Furber: Yeah, so it's the same machine. At that time CPU Limited got good connections with National Semiconductor, and so the hardware was based around the National Semiconductor SC/MP microprocessor, and we used two of them in this controller. And I did a fair amount of the software writing, I think, for that.

Fairbairn: So you weren't involved in any hardware design? It was purely software programming?

Furber: I was involved in the software at that stage, yes. I had met Sophie Wilson through the Cambridge University Processor Group, and Sophie got involved in that stage too. Sophie's first contribution was a piece of hardware to make the fruit machine less prone to paying out if somebody flashed an electronic cigarette lighter near its socket. So interestingly it started off with Sophie doing a bit of hardware and me doing a bit of software.

Fairbairn: So you had gotten your Ph.D. and you were actually still at the university as a research fellow, is that correct?

Furber: That's correct, yes.

Fairbairn: So you were still, at least from that point of view, still focused on your major path. So tell me, again, what year as it that you decided to make the switch and how exactly did that happen? You decided to work with Hermann Hauser.

Furber: The major event that caused me to switch to Acorn as my day job was during 1981. We had been playing with ideas for machines to follow on from the success of the Atom and we came up with this idea for the Proton which was a kind of dual processor. And then Chris Curry had heard that the BBC was looking for a machine to support a series of TV programs, so we fairly quickly converted the Proton into a single processor machine and we used the front end as a design that we submitted to the BBC for the BBC Micro. We turned that around from a sort of a sketch into a working machine extremely quickly. It has been a fairly well documented week where Hermann rang Sophie at the weekend and said, "The BBC are coming on Friday. Do you think we can show them a prototype?" To which Sophie said, "No, we can't build it in a week." And then Hermann range me and said, "The BBC are coming on Friday. Sophie thinks we can show them a prototype." And I said, "Well, if Sophie thinks it's worth a go, I'm prepared to have a go." And then he phoned Sophie back. So he played us off against each other quite beautifully. And that week, remember, my day job was still supposed to be in the university but we ended up working day and night that week to get the prototype going for BBC on Friday. The end result was Acorn got the contract for the BBC Micro, and that happened around Easter 1981.

Fairbairn: So what was your job for that whole week? You were starting from a-- you didn't even have a diagram, a schematic for the computer?

Furber: I had the beginnings of a schematic at the beginning of the week, and I detailed the schematic. We then got some help in to wire up the prototype, and then I was involved with Sophie in debugging the hardware and the software together, which finally came into life at about seven o'clock on the Friday morning with the BBC arriving at ten o'clock. I was still a research fellow, and aerodynamics was still my day job. That BBC Micro contract rush rather took over for a days, and Acorn got the contract. The BBC were confident that with the support of their TV program that Acorn would be able to sell 12,000 of these machines, which was quite a big number for Acorn - that was a business worth going for. Of course, it turned out to be a hopeless underestimate. Eventually about 1.5 million BBC Micros were sold.

Fairbairn: Had you ever done logic design before of the scale that was required for this computer? Or were you treading new ground even there?

Furber: I had done logic design before for the machines I had built at home and, in fact, the BBC Micro bears more than a passing resemblance to a machine I had built the previous year at home. I had built it on several discrete cards with a back plane, OS and so on. But the principles of operation of the BBC Micro hardware, which were having processor and display access multiplexed into the same memory bank, those were all lifted directly from my home machine, the difference being for the BBC Micro we ran it at twice the speed. So we pushed the speed, a bit. So I had done a reasonable amount of logic design, but entirely as a hobbyist. And in retrospect, the somewhat frightening thing is that I had very little experience of producing logic for large-scale production, and there were aspects of the BBC Micro logic design which were fairly close to the edge, let's say. If I had known at that time that somebody was going to try to make a million of these things I might have been quite worried about the prospect.

Fairbairn: Okay. So we were talking about the fact that you had done the logic design for this BBC Micro without actually having done that sort of for any kind of major production machine in the past. Tell me about the design. You've told me something about the memory access. What kind of microprocessor did you use? And how did you come to choose that microprocessor?

Furber: The BBC Micro was based around the 6502. And we chose the 6502 because it had been used in a previous Acorn design – that I didn't have much of a hand in – which was the Acorn Atom. So we had quite a lot of software ready to go on the 6502, and quite a lot of experience of using it. We ran the 6502 at 2 megahertz, and we interfaced it to a memory system that basically had a cycle time corresponding to four megahertz. So we could get two processor accesses and two video accesses into the same memory in each microsecond, and they didn't interfere at all.

Fairbairn: So the BBC Micro as you designed it that first time was it higher performance than any of the other similar sort of small personal computers at the time?

Furber: It was a bit faster than the competition. I think the Apple II was out some years before that and that was a 1 megahertz 6502. So the BBC Micro had a bit more performance. But the thing that really sold it was it had a very wide range of interfaces, and it had some very user friendly software built on it. So the BBC Micro had a whole range of connectors that you could connect other bits of kit to, including second processors. Acorn produced a stream of second processor boxes. You could add a Z80 running CP/M, a National Semiconductor 32016 running Unix and ultimately, of course, we built ARM second processors as the major development environment for the ARM itself.

Fairbairn: And did you sort of provide those interfaces because of specific requirements from the BBC? Or you just thought that that would be a good feature to include?

Furber: The spec was the outcome of quite long negotiations with the BBC, and the BBC had fairly clear ideas of the sort of things they wanted to do. There were one or two things that we wanted to do, and in the end we put all of the interfaces on for all of the agreed functions, and it gave us a pretty broad range of ways of connecting to the machine.

Fairbairn: So when did you actually demonstrate this first working version to the BBC? And when did you ship sort of the first production unit? Do you recall those times?

Furber: Yes, the working version was when the BBC visited which was around Easter time in 1981. My research fellowship ended at the end of September '81. I joined the company full time in October '81, and the BBC machine shipped to customers in January '82. So the whole development from first working prototype, which was a very rushed prototype, to the machine in production was about nine months.

Fairbairn: And when you designed it for those first production units, you were expecting to sell a few thousand. And obviously it sold vastly more than that. Did you have to go through redesign? When did you discover that it was going to have a much higher volume run? And did that require that you redesign the machine in some way? Or how did that transpire?

Furber: The machine transitioned from the initial plans into much higher volume without any fundamental redesign. So I think it was probably the order of a million BBC Micro's sold to the original design. One of the things we did have to change was the power supply. The BBC were very averse to switch-mode power supplies because switch-mode power supplies switch at radio frequencies, and in the U.K. the BBC owns quite a lot of the radio spectrum. It didn't like a bit of electronics running at radio frequencies when they didn't have to. So they insisted that we launched with linear supplies. But the space available in the case for the supply meant the linear supply was very prone to overheating, and we had a lot of trouble with that. So fairly early in production of the linear supply-- the BBC was persuaded that we should switch and the linear supply was replaced by a switch-mode supply which completely fixed that problem.

Fairbairn: Okay. Steve, so you've got this first prototype working and then you had to go into volume production nine months later, were you involved in that phase of the project? Or how did that go?

Furber: I was pretty heavily involved in the process of getting the thing into production; not at the manufacturing end, but at the end of getting the design sorted for production. A significant aspect of the BBC Micro design was that we used two gate array chips to replace quite a lot of logic, and my responsibility was primarily to get those gate array chips to work. Initially, those chips were made by Ferranti in Oldham in the U.K. But we had quite a lot of difficulty with them overheating inside the machine, and subsequently we got them second-sourced from VLSI Technology and they ran NMOS chips and they were much cooler and fixed the problem.

Fairbairn: So the question I had was when did Acorn realize that it was going to be producing machines in the tens or hundreds thousands to millions versus a few thousand?

Furber: I think we realized that the demand was going to be much higher than at first had been thought even before we had shipped the first machine. I think before we had shipped the machine we had orders for more than the 12,000 that was originally anticipated, so it was clear there was a wave of interest. It took some years before it became clear how big it was. I suspect '83 was probably the biggest production year for the machines. But there were a number of other things that happened that told us how strong the interest was. Three of us were asked to give a seminar on the BBC Micro at the IEE in Savoy Place in London, which was the U.K. Institution of Electrical Engineers. We gave this lecture in the hall that would accommodate six or seven hundred people and, in fact, three times that number of people turned up and most of them had to be sent home because there was just no capacity in the room. IEE seminars usually weren't like that. So this was another signal that there was a lot of interest, that really the general public was very ready to get involved in microprocessors when they saw something that they felt they could trust. And, of course, the key for Acorn was having the BBC badge on the machine gave the machine an extremely trusted brand.

Fairbairn: So you had this first machine in production then. And obviously much higher volume than you had anticipated. Lead me through sort of the evolution of the company and the machines that took you up to deciding to start on the creation of the ARM processor itself.

Furber: The BBC Micro was selling in huge numbers and the company had grown from a few tens of people to about 400 people in just a couple of years. We, in what was called the Advanced Research and Development end of the company, started to think about how we would build on this success and what we needed for a follow-on machine. It was pretty clear to us that we progressed from eight bits at least to 16, and we spent quite a lot of time looking at the 16-bit microprocessors that we could go out and buy from the established manufacturers. Each one we looked at didn't really meet our expectations. There were two things we found somewhat frustrating with the standard offerings; these were microprocessors such as the Motorola 68,000 and the National Semiconductor 32016, both of which had very rich complex instruction sets. The two problems we found were firstly their real-time performance was not as good as the 6502 - they couldn't handle interrupts at the rate that we got used to running the 6502 with - and secondly, these microprocessors couldn't really keep up with commodity memory. The major cost component in a small computer then was the memory, and it seemed to us obvious that the most important thing for the processor to do was to make the most of that memory. Neither of these processors would exploit the full bandwidth capability of the memory. So we were wondering which processors to adopt, not being very happy with any of the options available to us, when Hermann brought us a couple of papers describing the Berkeley RISC and the Stanford MIPS processors. We looked at these and we saw in them something that seemed to address our issues. The Berkeley paper used the term reduced instruction set computer, and it talked about making processors quite a lot simpler, making better use of the silicon than you got if you just put in a very complex instruction set.

Fairbairn: What year was this? When did you get to see these papers?

Furber: This would have been 1983, early in 1983. Sophie started doodling with these ideas and producing draft instruction sets for the kind of processors that would meet the requirements that we felt we had at that time. Based on a lot of RISC ideas, but also based on Sophie's deep understanding of what it takes to build an instruction set that's nice for targeting your BASIC compiler to, because Sophie had been responsible for all of Acorn's BASIC interpreters. So what she was coming up with, while it had a very strong flavor of the Stanford and Berkeley RISC ideas, it also had some features that were slightly different that Sophie felt made it better suited to supporting high-level languages.

Fairbairn: Sophie was at work trying to fashion an instruction set that would be very efficient and could run on one of these reduced instruction set processors is that correct?

Furber: That's right, yes.

Fairbairn: So continue your description of those developments.

Furber: Sophie was playing with these instruction set ideas and, to us as a team at the time, the idea of building our own microprocessor didn't seem a terribly likely solution to our problem. We had spent quite a lot of time looking at and talking to the major microprocessor suppliers. We had been to visit National Semiconductor's design center in Israel. And what we had seen is that these companies had very large teams building these processors, and it took them a long time to get them right. So we still viewed microprocessors as a bit of a black art and not something that companies like Acorn would take on. But the other feature of the Berkeley and Stanford stories was that they had managed to produce reasonably competitive microprocessors just using a class of graduate students for a year. So they had a lot less experience and a lot less resource than the big companies had used, and therefore we thought maybe, just maybe, with these ideas if we set off we'll come up with something interesting possibly. But if we don't we'll learn quite a lot, and that what we learn will help us choose the right processor for the next product. So we talked to Hermann about this and Hermann encouraged us to continue working with this design, and we set off to build the processor. The company had recently invested in a chip design team. We had gone around the world looking for tools and selected those from VLSI Technology. We had appointed professional chip designers, and this had been done as a strategic investment - we didn't have any chips for them to design at the time. So it was a natural fit between Sophie and me playing with ideas for a microprocessor and having some chip designers looking for a job at the time, and we started the project off. We expected at any point to discover why designing our own microprocessor was a bad idea, and really the history of the project is that that point never arrived. We never got to the point where it became clear that this project was a bad idea and therefore it carried on to a conclusion and left us with a working microprocessor at the end after only 18 months from when we seriously started on the project, to when we had working silicon.

Fairbairn: It's a heck of an achievement. Tell me about your role in that. Sophie was designing instruction sets. How did you take it from there?

Furber: My responsibility was the microarchitecture, and that really goes all the way from taking Sophie's instruction set down to having logic-level implementations. I built the reference model, so the way the design was done was by building a program, and that was all built in BBC BASIC. And, indeed, a couple of years ago I found the original ARM reference model sitting on a floppy disk at the back of my garage somewhere, and the complete development environment is about 850 lines of BBC BASIC. So I put together the microarchitecture, which is a three-stage pipeline, built the reference model and then turned the components of the reference model into what we called block specs, which were basically two- or three-page written descriptions of particular sub units, so there would be a block spec for the ALU and a block spec for the register file, and so on. These block specs were somewhere between schematics and the slightly higher levels of the tabular description of the logic, and they were given to the VLSI group who put the blocks together in bits of physical design and assembled them into a chip.

Fairbairn: So they took the blocks and did circuit design as necessary and created the physical design.

Furber: That's right, yes.

Fairbairn: And how big was the VLSI design team?

Furber: I think there were three people who worked directly on the ARM. The team grew a little bit and, of course, the ARM was never conceived as a single chip. It was always one of four chips that we needed to build the Archimedes system. By the end of the project there were four or five people in the VLSI team, but I think only the first three of them were directly involved in the ARM.

Fairbairn: So here you were designing what has become, perhaps, one of the largest selling microprocessors. And you had actually no formal training in computer architecture or even logic design. All of this was learned at the seat of the pants, is that correct?

Furber: That's right. I had no formal training in any relevant subject, and none of us had built a microprocessor before.

Fairbairn: So just coming back, is there a particular day or event or something that happens and says, we're going to go do this? When you went from sort of experimentation and playing to this is a real project, we can do this and we're going to make it happen? How did that…

Furber: I don't think there was a single point where this project went from just a sort of a random idea to a formal project. Clearly, it had to have backing from higher up in the company and Hermann encouraged people to pursue ideas, and if the ideas were sound enough he encouraged them to put them on a formal footing. One of the major points in this was in October 1983. Sophie and I went to Phoenix, Arizona to visit the team that was developing a successor to the 6502. It was a processor that became the 65C816, I think that's the right number, which was an extended-address version of the 6502, and Acorn used that in some of their products. Sophie and I went to Phoenix expecting to find our image of an American company in large shiny office blocks, and what we found was the 65C816 design was being done in a fairly modest bungalow in the suburbs of Phoenix. Admittedly inside this bungalow it was kitted out with some fairly heavy duty pieces of equipment, but basically the design was being done by a small team with local school kids contributing to some of the physical cell design in their summer holidays, and so on.

Fairbairn: So you were visiting the team in Arizona that had designed the 6502 and now the successor. And you found that they really didn't have all-- the team was not of a size that was similar to the others. And this, perhaps, gave you confidence that you could do it yourself?

Furber: That's right. I distinctly remember returning from Phoenix talking to Sophie and saying, well, if they can design the microprocessor then maybe we can too. So it was really when we came back from Phoenix that we thought we should get this thing on a proper footing and really try and drive the project through, although until we had working silicon in our hands we always expected to discover why it wasn't a good idea. The other nervousness we had, of course, was that the RISC idea had come out of Berkeley and Stanford and had been widely published. So everybody knew about RISC. The information was out there, and we figured that some of the big companies must pick up on this idea - it was obviously a good idea, so somebody was bound to pick up on it, and when they did we essentially get trampled underfoot because they will be able to put much more resource on to turning this idea into a commercial processor. So we were very surprised when we got to April 1985; we had working silicon, and still nobody else had used this idea to develop a commercial processor.

Fairbairn: And was it obvious fairly early on that you could provide the-- you could achieve the kind of performance that you were looking for? That this really was an answer to the problem that you were facing?

Furber: We knew that the processor would meet our requirements because it was fairly easy to do the sums. One of the reasons ARM came out as a 32-bit processor and Acorn effectively went from 8 to 32 without stopping at 16 along the way was because we had this strong idea that memory bandwidth was the key to getting processors to perform, and the 32-bit processor had an easy access to twice the bandwidth of a 16-bit processor so why not use it? We were not talking high clock rates here. The original ARM processor that went out in the Archimedes product was capable of operating at about eight megahertz, and that was, you know, not too difficult to achieve; and if you achieved eight megahertz then we had a very good idea of how much bandwidth it would use. The other thing we did with the processor which, again, I thought was remarkably obvious at the time, and so we didn’t patent it which, in retrospect, might have been a mistake, was we had this idea of just exposing a little bit of the internal operations of the processor to enable the memories to operate in high-speed page mode when the processor was generating sequential memory request addresses, which it does quite a lot of the time. That gave us about 50 percent more bandwidth out of the memory and really for no additional cost. It was a very cheap thing to do. In terms of complexity it cost maybe half-a-dozen logic gates on the chip, and one pin, to give the memory controller enough information to deliver 50 percent higher bandwidth and therefore performance. The other reason we were confident the design would deliver was, being a reduced construction set computer, it had no complicated instructions, and therefore we knew exactly how long it would run before-- the maximum time it would run-- before it would take an interrupt, and we could design that to meet our real time requirements.

Fairbairn: So you said earlier on that the lower power aspect of the risk was sort of serendipitous. Was there any discussion or any issue about power as you were going through the design?

Furber: Yeah, I said the low power properties were to some degree serendipitous but they were not accidental. The goal for the ARM processor was to go in Acorn’s desktop product, and that was quite a cost sensitive application, so our goal was to get the ARM processor into a plastic package because plastic packages were much cheaper than ceramic packages. And the plastic package had a power limit of around a watt. If you went above a watt you’d have to go to a ceramic package and that brought the cost up a lot. Now, the tools we had available at the time – and we were using the tools from VLSI technology – they didn't really give the kind of accuracy of power prediction that we now take for granted with today's tools. So the way we made sure that we came in under a watt was to use the tools conservatively. Effectively, we used Victorian engineering margins on the power consumption of the chip. So in being very careful to make sure it came in under a watt, we actually produced a design which came in under a tenth of a watt, and as I say some of that was down to the fact that the tools didn't allow us to get particularly close to what we thought the power limit was.

Fairbairn: So what was-- you completed the design. What was it like, you sent it off, and what happened when it came back? What was the…

Furber: Well, the design was fabbed by VLSI Technology. We took the design to VLSI Technology's offices in Munich in January 1985. I think the records will show that January 1985 was especially cold in Munich, and one of my memories is that all of the diesel taxis were immobilized because the fuel had frozen in the tanks and pipes, and so the town it was pretty snowy and it was littered with immobile taxis. That's where we did the final checks on the design, at the VLSI offices. And then, of course, as with all fab runs, the design goes off, and as far as the designers are concerned it goes into a black hole. The chips came back on April 26, 1985 and they arrived in the morning, and by the afternoon they were running BBC BASIC. We had one or two minor hiccups getting the test card to work, so we had some PCB level problems we had to fix, but once those were fixed the chips just ran as expected and we had our own microprocessor, much to our surprise!

Fairbairn: It must have been quite a celebration at that time?

Furber: Yeah. Acorn had this tradition of cracking a bottle of champagne once a chip was actually really showing that it worked, so the champagne was opened later that afternoon, and I think you'd find the bottle that was signed on the day – probably by Sophie, I'm not sure - but that bottle is still on proud display on the shelves at ARM Limited's facilities in Cambridge, alongside quite a few hundred other bottles, now, I think.

Fairbairn: So you said that there were other chips that needed to go into this Archimedes product, is that correct? Did you design further custom chips?

Furber: Yes. So there were three other chips, and they were known as MEMC, VIDC, and IOC, which simply stood for the memory controller, the video controller, and the IO controller. I was design lead on the memory controller, and then Tudor Brown, who's been a very senior figure at ARM Limited for the last 20 years, was design lead on the video controller, and Mike Muller, another very senior person at ARM, was lead on the IO controller. So the four chips together took us another couple of years, and by then we had the second generation of the processor itself. So the chip set that went out in the Archimedes had ARM2 in it with MEMC, VIDC and IOC, and those four chips greatly simplified the design of the machine.

The Archimedes was a fair success as a product, and definitely when that machine came out it was the fastest computer around for quite some time.

Fairbairn: So it sounds like you were involved in the development of these chips and so forth for some time. Tell me about the transition from those days to when-- and how did it happen that ARM was split out from Acorn? How did it lead up to that and what role did you play in that?

Furber: Following the success of the ARM, my principal role at Acorn throughout the eighties was I led the team that looked after the ARM both on the hardware and software side, and we developed further versions of the ARM chip leading up to the ARM3, which was effectively an ARM2 processor but with the cache memory on the same chip, and the ARM3, again, went out in Acorn's products. At the end of the 1990s, Acorn's business was getting pretty flat because they principally sold into U.K. education, and the U.K. education market size was decided by the government on an annual basis, so Acorn's business was not growing. And the cost of maintaining a viable processor line was increasing; I couldn't see that this was going to go anywhere particularly positive, and the opportunity came for me to take up a chair at Manchester in computer engineering. So I left Acorn in August 1990 to take up my current position at Manchester. About a couple of months after I had left, Apple came knocking on the door at Acorn and said they wanted to use the ARM in their Newton product, but they were a little uncomfortable with ARM being owned by a competitor. So they came to Acorn with the proposal to set up a joint venture, and Acorn had been trying to offload the overhead of maintaining ARM for a couple of years, so Apple was pushing on an open door, and within a couple of months ARM Limited was set up. Most of the people that I worked with, apart from Sophie, were moved out into ARM Limited. Robin Saxby was brought in to head up the company, and ARM Limited was formed by the end of 1990 and was in operation.

Fairbairn: So were you involved in any of that process? Or you had left and you were just sort of keeping up with the development through contacts with friends and so forth?

Furber: By the time Apple came knocking I had left, and I was not involved in the process of setting up what became ARM Limited. I had spent some of my time in my last two years at Acorn drawing up business plans to try to and find ways to set up the ARM activity as a separate company, but none of these business plans had ended up making very much sense because the fundamental problem was that the advantage of ARM is it's a small chip. The business model was to license this design and get royalties and if you get royalties on a small part of a chip you get a fairly small income per chip, and you'd have to sell very large numbers to get the books to balance. So none of my attempts at producing a business model, with the Technical Director of Acorn at the time, looked very promising, and really it was when Robin Saxby was brought in to head up ARM Limited – once it was formed – he was responsible for developing the business model that actually turned these business plans around and made them successful. Royalties are also difficult for cash flow reasons because the royalty stream comes in very late, and Saxby introduced the idea of an upfront payment for a license followed by royalties downstream, and that's what made the business succeed because, of course, the upfront license is wonderful for cash flow.

Fairbairn: Yes. So in those later years, I guess the second half of the eighties at Acorn is there-- are there some things that you were involved in that you'd like to highlight? Or that you feel particularly proud of were important in the evolution of the processor, or other elements of Acorn?

Furber: Well, the four chip set got into product first in the Archimedes in 1987, which is already getting to the back end of the '80s. So the first product launch based on the silicon that we developed was '87. After that, the ARM3, I think, was a very fundamental improvement. I was not design lead on the ARM3. I oversaw that in a managerial capacity, and Alasdair Thomas, who worked for me, led the detail design on that. I was particularly pleased with ARM3 because it effectively increased the performance of the processor by a factor of five in real Archimedes systems even though the clock rate only went up by two-and-a-half. The reason it did this is it prevented the processor from conflicting with the video system for memory bandwidth. Although the processor was running effectively at five times the throughput, its memory requirements were reduced and, therefore, it shared the memory bandwidth with the video system much more happily. We also did one or two bespoke designs. One was for Hermann's Active Book Company where they got quite close to producing an ARM-based product, and the last ARM processor that I was involved in in a very hands-on way was the ARM2aS, which was the first fully static implementation of the ARM architecture. In many ways, that particular version of the processor was the smallest and tightest that was ever produced. But unfortunately, the Active Book Company changed direction and that processor never found its way into product.

Fairbairn: So tell me the story of how you ended up at the University of Manchester. How did they come to find you? And what goal were they trying to-- or what need were they trying to fill at the university?

Furber: The University of Manchester has a long and honorable history in building computers from the Manchester Baby Machine in June 1948, which was the first computer to implement the stored program concept, through the Atlas in the early sixties which was a real trailblazing computer; it introduced virtual memory to the world, for instance. And, if you like, the first wave of academics had gone through Manchester and by the end of the eighties Dai Edwards had retired as the last professor of computer engineering. So Manchester was very keen to continue its traditional strength in computer engineering and had been looking to appoint to this chair for at least a couple of years and not found it easy. I found out about this chair somewhat indirectly, but I think it was Andy Hopper that drew it to my attention. He said, "Manchester is looking for a chair." I had made it reasonably public at Acorn that I was minded to return to academia at some point, without any particularly definite plans, but I think Andy new this, and he said, "You know, Manchester is looking for a chair in computer engineering, do you?" I didn't know, but when he told me I followed this up, applied for the chair, and went through the fairly normal, rather tortuous academic interview procedures and was offered the position. So I guess thereafter it was a fairly normal process. I gave in my notice at Acorn and moved to Manchester in August 1990.

Fairbairn: So we haven't talked anything about your family. Are you married or what's your status? And did that influence any of your decisions?

Furber: I married – in 1977 I married Valerie, who is my wife. She's from the south of England, and so when I started thinking about this move back to Manchester I had to handle things fairly carefully because in England there's a fairly strong north/south divide. I was born and brought up in the north, so Manchester held few fears for me, but Val had been up to Manchester several times to visit my family and so she was not too worried about this. During the eighties, in addition to BBC Micros and ARMs, I also had two daughters. In fact Alison, my oldest, was born in January 1982, the month the BBC Micro went on sale. And Catherine, my younger daughter, was born in April 1984, which was, I guess, about halfway through the ARM design period. So that's my family. I seem to have spent a lot of my working life predominantly with men, and nearly all my home life with women, because I have sisters and no brother, and daughters and no son, so maybe that's a balance of some sort.

Fairbairn: Yeah. So tell me about the work that you've done at University of Manchester, what has been your focus of research, and what do you find particularly appealing about the academic world, versus the commercial world that you left?

Furber: So I returned to Manchester at the end of the '80s. At the end of my research fellowship I'd been doing research for a number of years, and starting at the fairly junior end, and I felt it was, at that time, it was quite difficult to work out what I should be doing. It was quite difficult to find my own motivation and direction. So moving to the commercial world in 1981 was a fairly comfortable move, because in the commercial world, it's much clearer, day by day, what are the important things that you need to get on with. But as the ARM project had developed, and grown in strength, I'd begun to develop a whole lot of other ideas that I wanted to pursue, and of course in the commercial world, you can only pursue these ideas if they make commercial sense and you can get backing from above. So one of the attractions of the university position was that I knew that in that position, I could effectively set my own agenda., and I was ready to do that again, after ten years at Acorn. So I moved to Manchester. Acorn was highly supportive, they gave me some machines to take with me and, in fact, they allowed me to move part of the European funded project to the university, which I'd been working on setting up while at Acorn. So I arrived at the university with some funds already in place, and some equipment, and really quite well positioned to go on from there. The interest that I'd been developing towards the end of my time at Acorn, was in understanding how to build processors and chips that work without clock signals, so this was the generic - the general – area of asynchronous design. This funding gave me the opportunity to pursue this, and I spent most of the '90s understanding how to build asynchronous circuits and systems, processors, on-chip buses, peripherals, and so on. And by the end of the '90s, we we'd got a pretty complete armory of techniques for building clockless circuits. It became clear, I think, to me, at the end of the '90s, that industry didn't feel a strong need for abandoning clocks, that clocks play a very central role in the design flow that's used in most of industry, and the problems that we believed, and in fact I still believe, face chip design in the future, were really not hurting enough for industry to be very interested in asynchronous design. So even today, if you look around industry, the number of companies active in asynchronous design is still quite small, and the impact of asynchronous design is quite marginal. So towards the end of the '90s, I decided I would switch direction a bit, and tackle a different problem. This one's really even-- arguably even less commercial, because it's really about fundamental science, but it's motivated by the fact that, after 20 years building processors and seeing them get enormously faster than they were when I started, they still couldn't do some of the things that, you know, today my two year old granddaughter can do easily. I got increasingly interested in the idea that we'll really only get computers to do different classes of things when we understand how our own brains work a lot better. And so my research slowly changed direction towards building machines to understand how the human brain works, and that project has been more than ten years in gestation. Over the last five years, we've been designing silicon to build the machine. In May 2011 we received the full silicon and it's all working, and so now my goal is to build a machine with about a million ARM processors in it, for brain modeling purposes. And to give you a feel for how big the problem is of modeling the brain, I reckon with a million ARM cores, using fairly simplified models, I can get to about one percent of the scale of the human brain.

Fairbairn: So getting to-- you're actually physically going to build a million processor machine, is that-- and how many cores per chip, and, sort of, what is the scale of this machine that you're building?

Furber: The million core machine will be physically quite large. We're putting 18 ARM cores on a chip, and each of those cores has quite a lot of local memory, as well as some shared external memory, and so we put 18 cores on a chip, and then we need about 50-60,000 chips to build the million core machine. And we're-- right now, we're building a circuit board that will have 48 of these chips on, so that's 864 processors, and the final machine will be about 1,200 of these boards. We'll get about 24 boards in a rack, about five racks in a cabinet, and then about 10 cabinets to make the full machine. So it's a physically large machine. It's not something that's going to fit inside your mobile phone any time soon.

Fairbairn: When do you expect to be able to build that?

Furber: Well, we're doing the circuit board design now. Once we have the circuit board, then assembling those boards to build bigger machines will progress roughly at a factor of ten every three months, so around a year's time, I would expect to have the million processor machine.

Fairbairn: And you have a team of people working on the software problem, as well as the hardware problem, I presume.

Furber: Oh yes, the chip design took a lot of effort. Chips have become a lot more complex than they were in the days of ARM1. In fact, one of my reflections here is that I'm probably using ARM processors rather more wantonly in this machine, than I was using transistors in the first ARM processor, so the level of abstraction has moved up a bit. But yes, the software problem is pretty enormous. We have small circuit boards now with four chips on 72 processors, and we have software that will run real time neural networks on those boards. So we have been working on the software for a fair time, too. But as the machine scales up, the software gets harder in interesting ways. So we still have some problems to address. The key to the whole machine now is how those processors interconnect, and the innovation in the machine is the way we've been able to emulate the very high connectivity that's found in the brain. So in the brain, each neuron connects to, on average, about 10,000 targets, and we've had to build a highly specialized comms system for this machine, that allows a neuron model on one processor to send information, when it spikes, to 10,000 other neurons across the machine, and effectively, we have a comms system that implements arbitrary graphs that connect processors together to get these messages across in real time.

Fairbairn: Is there any particular human-like task that you want to accomplish with this new machine, to kind of prove its capabilities?

Furber: In terms of what we'll do with the machine, we're looking very much to collaborators from the worlds of neuroscience and psychology to direct us toward interesting problems. So we have a current collaboration with our psychology colleagues in Manchester, and they are interested in problems that some people have, particularly in older age, with reading. If you suffer a minor stroke in the wrong part of the brain, then your reading ability can suffer in all sorts of strange ways. The psychologists have neural network models of the reading process, and they can damage these models, and they can reproduce the clinical symptoms and so the hope is that they'll be able to use these computer models to test different choices of therapy and find the right therapy for the computer model before taking it to the patient.

Fairbairn: I see.

Furber: So that's a fairly near-in application for which there's specific expertise.

Fairbairn: So have we missed, sort of in going through your professional career, have we missed any sort of major developments or major things that you'd like to mention, before we go on to the next stage of the interview?

Furber: I don't think we've missed anything. I mean, my career has been very simple. Following my PhD, I was a research fellow for three years, I was at Acorn for nine years, and then I've been a professor at Manchester for 20 years, and that's my entire career.

Fairbairn: And quite a noteworthy one it is.

Furber: I beg your pardon?

Fairbairn: I said, quite a noteworthy one that it is.

Furber: Yeah, it's pretty simple. I've done my term as department chair in Manchester, I do a number of things outside the university, so I've recently completed chairing the Royal Society's study of computing in schools, which is quite an interesting issue because, of course, back in the 1980s, the BBC Micro was the thing that really introduced computing into schools in the UK, but then in the '90s, the BBC Micro gave way to the IBM PC, and you know, a machine designed for education gave way to a machine that's really designed for office and business use. And with the introduction of the PC, a lot of computing lessons at school have become rather mundane office productivity lessons, and recently there's been a very significant upsurge of interest in trying to regenerate some of the creativity and excitement that there was in the '80s, in computing in schools, today. So I've chaired this study, and we've made some fairly strong recommendations about really how computing education in the UK ought to be completely overhauled and revised, and we want to see much more exciting activities in the schools. And as I say, it seems slightly odd, because the report was launched in January this year, exactly 30 years, to the month, after the launch of the BBC Micro, and it seems, 30 years later, you know, we have to start again and find the right way to do this. Now I know this issue of how you deliver computing in schools is not limited to the UK, and there have been similar moves in the US and many other countries, but you know, I'm back involved in education again at the schools level.

Fairbairn: Well that's an interesting topic. I'd-- perhaps when we get together in April, we can have a further conversation. That's an interesting topic for myself as well. So I'm partly struck in terms of your academic background, your PhD thesis and so forth, and its connection or disconnection with any of the work you did subsequently. Is there threads of connections, or is it really-- was it really different?

Furber: I think there is a connection, in the sense that I think mathematicians and how they think, and the ways you need to think to build computers, are strongly related. In terms of how we select undergraduates in Manchester, we don't currently mind if they've done any computing at school, what we ask is that they've done good maths at school, and we think an ability at maths is the strongest indicator of an ability to cope with what's in an undergraduate computer science course. So I guess that the fact that my background was very strongly mathematical through school and university gave me some of the right mental tools for coping in the computer industry. But it is true that most of the practical details were learned just through hobbyist societies. Now, of course the society I joined, Cambridge, was a student group with some very expert people in it, so maybe I was taught through the society, rather than being taught through classes. I certainly had a lot of help in the university Processor Group from people who knew a lot more than I did, and that was very helpful.

Fairbairn: Right. So interesting connections there. So you know, as you reflect back on your career, what are you most proud of, what do you feel best about in terms of the various activities you've been involved in?

Furber: I think, reflecting back on my career, and picking things that I'm proud of having achieved, it's interesting, because I think I've only got a good sense of the significance of some of the things I did a very long time after I did them. So it's easy to look back now with 30 billion ARM processors in the world and say, you know, we were doing something really important back then in the early '80s, but I don't think we had any idea that we were doing something that would become really important. And you know, I think we did a good job - had we screwed up getting the ARM chip out, then none of what follows would have happened, so in that sense, it was important - but on the other hand, many other people have contributed many other ideas and lots of work and effort, and the success of the ARM today depends even more on how the company was managed in the '90s by Robin Saxby, how all the people that used to work for me carried the technology forward, and the ideas that have been brought in by thousands of people who worked on the architecture since I moved up to Manchester. So I'm very proud to have sown the seeds of that important development, but I'm a long way from believing that it's all down to me. Likewise, this is probably less evident in the US, but in the UK, as we-- as the computing education agenda has come back to the surface, there's more reflection going on back to what we did with the BBC Micro in the 1980s, and the significance of that, I think, has become much clearer recently than it perhaps was in the '80s, or even the '90s. There really was a generation of people who were brought into computing and technology by the availability of that machine, and of course there were competitors around as well, so it wasn't the only machine at the time, but it did cause a significant change in the approach to computing in education, and that's had a lasting influence on the lives and careers of many people. So again, from 30 years away, one can see that what we did then was really quite important, even though our sense of its importance at the time, was probably much lower than it is now.

Fairbairn: So you're in an interesting position, as a university professor, working with new students coming in and so forth. What counsel, what advice do you give to students that you come in contact with, about their career. They come to you saying, gee, I don't know what I want to do, or what direction should I go, or things like that. I presume you have those conversations periodically, if not frequently. What's your counsel to kids entering university these days, and especially those interested in the technical fields?

Furber: Yeah, so I, myself, never really had a grand plan as to how the rest of my career was going to pan out. So my advice, really, is to make choices that maximize the number of options; to do subjects that-- study subjects that give you many different ways you can go at the end; choose bits of research which create opportunities for you to move in a number of different directions, when you've done them. My experience is that it's not possible, when you're 18, 19, or even 25, to sit down and plan the rest of your career, but what you can do is work on keeping more of the better doors open. Other choices may close doors, and in general, I'd avoid those.

Fairbairn: Excellent. Is there any-- you know, as you've made these decisions, throughout your career, is there anything you would have done differently, any regrets? Do you feel happy with the choices you've made along the way?

Furber: Yeah, there's no decision I've made at any point, that I really go back to and say, you know, I wish I'd done something different at that point. As I say, I've followed fairly easy choices most of the time, the opportunity to join Acorn was wide open, and the timing was right, so it was an easy decision to take. Probably the toughest decision was the one I took when I decided to leave Acorn and move to Manchester. But of course that was a decision that was taken with the knowledge available at the time, and Acorn's business had gone pretty flat, and it was getting increasingly difficult to do interesting things there. So at the time, it felt like the right thing to do. And I'm often asked, you know, do you wish you'd stayed at Acorn, and then you'd have gone with ARM, and I don't regret that decision. It's fairly clear that it was probably an expensive decision to make, to go back to academia at that point in time, so economically, it was not the most favorable. But at the university, I've had lots of opportunities to pursue interesting things. I've probably had far more scope to decide what I do, than I would have had, had I stayed at Acorn, and then probably gone on to join ARM. So on balance, I think I've come out of that pretty well.

Fairbairn: How do you feel about sort of just in general, about the interesting challenges in computing, you know, if you were to start over today, would you choose a path in math and computing? Do you have other areas that you think might be equally interesting to pursue? What's your, on a broader scope, you know, starting over, what-- and given what's going on today in the development and so forth, what-- anything particularly strike you?

Furber: Well, one thing that does strike me a lot is the fact that computing has become far more deeply technical than it was back in the 1980s. In the early '80s, you could basically start a company, buy a few chips off a shelf, make a few products and sell it, and build a good business that way. These days, I think that's much, much harder, and designing a chip that's anywhere near the leading edge is much, much harder than it used to be, because there's so much more resource to play with. You know, the first ARM chip had 25,000 transistors, my latest SpiNNaker chip has 100 million transistors, and it's not actually particularly near the leading edge there. The leading edge chips are now in billions of transistors, and these just take formidable amounts of resource and effort to design, so it's far harder for a newcomer to get into chip design and make an impact than it was 30 years ago, and this concerns me a bit. It concerns me, actually, that it's becoming close to impossible to design interesting chips with university groups, because you know, our research grants will buy us 10, 20 man years of resource, but state of the art chips take hundreds of man years of resource, so there's a tricky issue there that I'm currently wrestling with. If I were advising somebody who is starting out on this course, then I think there's still plenty of opportunities for ideas, there's lots of places where you can add value, not by designing whole chips, which is what I found interesting, but by providing technologies that contribute to the design of chips. So in the chip area, I would say the really exciting topic for the next decade or so, is going to be understanding how to do 3D packaging well. I think we-- it's now time to go into the third dimension. The two dimensions have become too constraining. So that's one area I'd look at. But of course there's also a lot of excitement around new materials, and in my office is about 20 feet from where my two Russian colleagues earned their Nobel Prize, recently, for the discovery of graphene.

Fairbairn: Oh yes.

Furber: And graphene is a material with significant potential, yet to be fully understood. So whether, if I was starting again, I would go into microelectronic design, or I would find something in the more materials end, and nanotech end, I'm not sure. You know, I never planned my way into computing, it's just as I moved through education, there were clear opportunities in computing that I followed up on. Today, if you go through university, when you get to the end, you have to look at where things are moving in interesting ways, and seeing where you can contribute. Had you asked me at school what I would do for my career, I would probably have said something to do with aircraft.

Fairbairn: So one of the areas that you're-- I mean, the area you're focused on now, having to do with modeling the brain, and understanding the nature of the brain, sort of brings up the whole topic of biology, and obviously there's a huge amount of computing work going into the area of, you know, the study of genes and everything having to do with the biological sciences. Are-- did you find yourself collaborating with people in those domains, and how do you see the world of computation and the world of biology progressing in the future?

Furber: I think that is a very exciting area, and there's no doubt that biology itself has been accelerated by adopting more computing technology. The major factor that enabled the human genome project to come in ahead of schedule, and on budget or below budget, was the way it adopted computer technology to industrialize the process of extracting gene information, and the biologists have learned quite a lot from that, and they also have huge databases of, for example, proteins, where, again, they're highly dependent on computer technology to make sense of the information they've already got. So the biologists are already pulling computer technology into their world, at quite an impressive rate, and quite a lot of my computing colleagues spend most of their research activity working out how to improve the efficiency of biologists. The feedback the other way, I think is going to become stronger over future years, so the biologists are developing understanding of how genomes lead to behavioral properties of the brain, for example, they're getting quite fundamental insights. They go all the way from genomes to behavior, and I think we will know a lot more about how the brain works, even in the next ten years, so I'm optimistic and excited that, with the current research I'm doing, that in ten years time, there will be much greater understanding. Whether it will be complete understanding of how information processing runs in the brain or not, I don't know, and clearly that's the ultimate goal, to really get a good model of what's happening inside our heads. I don't know whether that will happen or not, but we will certainly advance towards that. And I think that will give us some quite important answers to the problems facing computers over the next ten years, so the variability of components, the reliability of components, are all big issues that biology has handled eons ago, and if we understood how they were handled, we'd know how to build better machines on the technology that's coming our way in the future.

Fairbairn: Mm-hmm. Okay, well I think that's a good note to end things on. Is there any other comments you'd like to make, or summaries or final points to conclude with?

Furber: I'm not sure, you haven't asked me any of the museum questions.

Fairbairn: Well partly it was, I wasn't sure what your background and understanding associated with that was, and so I-- obviously the major goal of the Computer History Museum is to document and collect the history and use it as a teaching tool for generations to come. And this interview is one of those steps in that process. And so I'm curious-- I don't know if you've had a chance, I don't know if you travel to the US often, or if you've seen the museum in the past, or this will be your first time this year?

Furber: I was out in the Valley for a couple of months in 2005. I had a sabbatical when I stood down as head of department, and I spent a couple of months at Sun Labs in Menlo Park, and I went to one or two talks at The Computer History Museum, when I was out there. Now, back in 2005, the museum was not as well established as it's become since then, I understand, so I'm looking forward to seeing it again. But I have a reasonable idea of the setup there. And yes, I think recording the history of computing is an extremely important function. It's all so new, that, in some sense, we tend not to think of it as history. But a lot of how computers are today can only be properly understood if you know the path they've taken to get there. It's very hard to explain some of the features of today's computers without going back through the sequence of events that have got us to where we are today, so I'm very keen to see computer museums established all around the world, and there are a couple of fairly significant computer museums coming together in the UK, which I'm also keen to support and encourage as far as I can.

Fairbairn: Well that will be great. Yes, the museum has advanced tremendously. It now is a world class organization and facility that is worth traveling to and seeing, for anybody who has an interest in the history of computing, and I'm curious in that regard, how do you see the interest in computing amongst people entering the university, in the UK, is it growing, is it waning, you know, what role, what needs to be done to perhaps promote interest in this area?

Furber: I-- my observation is that the interest in computer history, among the current generation of undergraduates, is generally very weak indeed. They are looking at technology as a way to the future, rather than being interested in how it got to where it is. So they're quite resistant to history lessons, as part of their computing courses. And it may be that interest in where the technology has come from is something that only does grow later in career, where you're looking for explanations of obvious peculiarities of the current state of affairs. But some students are very keen, so one of our undergraduates actually works as a volunteer at the local Museum of Science and Industry, which hosts a replica of the original Manchester Baby Machine, and that machine is still operational. So interest seems to be there, but only in a very small proportion of students.

Fairbairn: What about just general interest in entering the field of computing. Do you find that growing or static or waning, or--

Furber: Well, there are various measures of that, and this has come up in this study I've been doing with the Royal Society. One of the bits of data that we cite as evidence of the problem is the fact that, since 2000, the number of applications for university positions in computer science has halved, so there's a measurable decline in the interest in technology of school leavers, and that's a big worry. But on the other hand, the students that we get are as keen as ever, and as capable as ever, so if you take a slightly longer historic look, you can actually see the 2000 figure as a bit of a bubble. So interest grew very dramatically in the '90s, and then at the time of the dot-com bubble burst, we saw a similar burst in student applications. Maybe those two facts are related, we don't know.

Fairbairn: Mm-hmm. Okay, well we certainly look forward to your visit, and delighted with the interview, and thank you very much for taking the time and going to the effort to make it all possible.

Furber: Okay, you're welcome, and I look forward to meeting you out there in April.

Fairbairn: Okay, thank you very much.

END OF INTERVIEW