

itself is manufactured. Modelling the device and associated programs in this way helps to identify potential problems early on in the development life-cycle so that they can be overcome relatively easily and inexpensively by modification of the original design. Being able to design and simulate the programs in parallel with the development of the hardware upon which they will be executed also reduces the overall development time and makes the process more efficient.

- 5 The applicant itself produces a wide range of modelling tools which enable users to produce so called "cycle accurate" models for simulating microprocessors on a general purpose computer. These models can be used, not only to model the microprocessor itself, but also the program code designed to run thereon. However, problems arise when the simulated model of the microprocessor and/or the program code being subject to simulated execution contain "bugs" or errors which need to be identified and fixed. Difficulty arises in that when the simulation is being run and produces unexpected results, it is often difficult to locate whether the problem lies in the simulated model of the hardware or within the program code.
- 6 Previously, designers would have had to probe the hardware simulation and the program code separately in order to locate the bugs, or alternatively they would have used a "tracing" technique. A technique which is known to be slow and requires the introduction of "trace points" into the model's source code. The invention provides an alternative technique whereby the behaviour of the simulated device and the program code can be tracked simultaneously and the location of any bugs can be more readily identified. This is achieved by an arrangement in which the so called "host debugger" and "target device debugger" are synchronised, and a "debug interface" is provided for enabling access to stored information regarding the state of resources being used by the host simulation and the state of the device being simulated.
- 7 The most recent set of claims which were filed on 28 April 2009 include two independent claims to a method of debugging a simulation system (claim 1) and a corresponding system claim (claim 21). The wording of the claims is as follows
 1. *A method of debugging a simulation system for simulating a target programmable device executing target program code, said method comprising the steps of:
 - executing simulation code upon a host processor of said simulation system to simulate operation of said target programmable device executing said target program code, said simulation code being operable to maintain a store of current values of target resources of said target programmable device being simulated;*
 - executing a simulation host debugger to provide access to host resources of said host processor during execution of said simulation code by said host processor; and*
 - executing a target device debugger to provide access to said target resources during simulated execution of said target program code by said target programmable device; wherein*
 - execution of said simulation host debugger and said target device debugger are synchronised and said target device debugger accesses**

said target resources by sending an access request to a debug interface of said target programmable device provided by said simulation code to trigger return of said target resources to said target device debugger.

21. *A simulation system for simulating a target programmable device executing target program code, said simulation system comprising:*
a host processor for executing simulation code to simulate operation of said target programmable device executing said target program code, said simulation code being operable to store target resources of said target programmable device being simulated;
a simulation host debugger for providing access to host resources of said host processor during execution of said simulation code by said host processor; and
a target device debugger for providing access to said target resources during simulated execution of said target program code by said target programmable device; wherein
execution of said simulation host debugger and said target device debugger are synchronised and said target device debugger is configured to access said target resources by sending an access request to a debug interface of said target programmable device provided by said simulation code to trigger return of said target resources to said target device debugger.

The Law

- 8 The examiner has raised an objection under section 1(2)(c) of the Patents Act 1977 that the invention is not patentable because it relates to a program for a computer as such; the relevant provisions of this section of the Act are shown in bold below:

1(2) It is hereby declared that the following (amongst other things) are not inventions for the purpose of the Act, that is to say, anything which consists of –

- (a)*
- (b)*
- (c) a scheme, rule, or method for performing a mental act, playing a game or doing business, or **a program for a computer**;*
- (d)*

but the foregoing provisions shall prevent anything from being treated as an invention for the purposes of the Act only to the extent that a patent or application for a patent relates to that thing as such.

- 9 As explained in the notice published by the UK Intellectual Property Office on 8 December 2008¹, the starting point for determining whether an invention falls within the exclusions of section 1(2) is the judgment of the Court of Appeal in *Aerotel/Macrossan*².

¹ <http://www.ipo.gov.uk/pro-types/pro-patent/p-law/p-pn/p-pn-computer.htm>

² *Aerotel Ltd v Telco Holdings Ltd and Macrossan's Application* [2006] EWCA Civ 1371; [2007] RPC 7

- 10 The interpretation of section 1(2) has been considered by the Court of Appeal in *Symbian Ltd's Application*³. *Symbian* arose under the computer program exclusion, but as with its previous decision in *Aerotel*, the Court gave general guidance on section 1(2). Although the Court approached the question of excluded matter primarily on the basis of whether there was a technical contribution, it nevertheless (at paragraph 59) considered its conclusion in the light of the *Aerotel* approach. The Court was quite clear (see paragraphs 8-15) that the structured four-step approach to the question in *Aerotel* was never intended to be a new departure in domestic law; that it remained bound by its previous decisions, particularly *Merrill Lynch*⁴ which rested on whether the contribution was technical; and that any differences in the two approaches should affect neither the applicable principles nor the outcome in any particular case. But the *Symbian* judgment does make it clear, that in deciding whether an invention is excluded, one must ask does it make a technical contribution? If it does then it is not excluded.
- 11 Subject to the clarification provided by *Symbian*, it is therefore still appropriate for me, and Dr Keston did not argue otherwise, to proceed on the basis of the four-step approach explained at paragraphs 40-48 of *Aerotel/Macrossan* namely:
- 1) Properly construe the claim
 - 2) Identify the actual contribution (although at the application stage this might have to be the alleged contribution).
 - 3) Ask whether it falls solely within the excluded matter, which (see paragraph 45) is merely an expression of the "as such" qualification of section 1(2).
 - 4) If the third step has not covered it, check whether the actual or alleged contribution is actually technical.
- 12 The operation of this test is explained at paragraphs 40-48 of the decision. Paragraph 43 confirms that identification of the contribution is essentially a matter of determining what it is the inventor has really added to human knowledge, and involves looking at substance, not form. Paragraph 46 explains that the fourth step of checking whether the contribution is technical may not be necessary because the third step should have covered the point.
- 13 I will deal with the arguments put forward by Dr Keston as I apply the test set out in *Aerotel/Macrossan* to the present case.

Construing the claims

- 14 The first step of the test is to construe the claims. I do not think this presents any real problems since both the applicant and the examiner appear to agree as to the meaning of the claims.

³ *Symbian Ltd v Comptroller-General of Patents*, [2009] RPC 1

⁴ *Merrill Lynch's Application* [1989] RPC 561

Identify the actual contribution

- 15 For the second step, it is necessary to identify the contribution made by the invention. Paragraph 43 of *Aerotel/Macrossan* explains that this is to be determined by asking what it is - as a matter of substance not form - that the invention has really added to human knowledge having regard to the problem to be solved, how the invention works and what its advantages are.
- 16 The examiner has argued throughout the proceedings that systems for the simultaneous simulation of new hardware and associated program code on a host processor are well known. He therefore considers the contribution to reside in a new method of debugging the simulation using two synchronised debugging programs and an associated debug interface, all of which he argues are implemented in software.
- 17 Dr Keston argues that the examiner's analysis is flawed and that he appears to be confusing the means for achieving the contribution with the contribution itself. The contribution of the present invention does not lie in the way in which the simulation and the program it is running are debugged, that is merely the means by which the contribution is achieved and the examiner has failed to consider the "knock-on effect" or advantages resulting from the invention. The simulation system as described provides synchronised access to both host resources and target resources via the target debug interface. Thus, unlike in previously known systems, it is possible to determine the accuracy of both the target program code and the accuracy of the simulation code itself. This means that the simulation is on the whole more accurate and provides better information as to how the target program code is operating when executed on the simulated device. The "knock-on effect" or advantage being that the user of the simulation will be able to locate bugs far more quickly and efficiently than in the past and will not have to waste time looking for bugs in the target program code when the problem actually lies within the simulation code, or vice versa.
- 18 Dr Keston's definition of the contribution, as I understand it, is best summarised in the agent's letter of 1 July 2009 where the contribution is defined as a simulation system "*providing synchronised access to host resources of a host processor during execution of simulation code by the host processor, and to target resources during simulated execution of target program code by the target programmable device being simulated using the host processor, thus providing a better method of debugging a simulation system which provides more accurate diagnostic information enabling bugs in the simulation system to be located at a more fine-grained level*".
- 19 I have some sympathy with Dr Keston's arguments, I do not think the contribution lies solely in a new method of debugging software but goes beyond that and must include some aspect of the advantages associated with the invention. However, I agree with the examiner when he says that systems for the simultaneous simulation of new hardware and associated program code are well known, and that the contribution must lie to some extent in the method by which the two software debuggers and the debug interface are used to locate bugs within either

the simulation code or the target program code. As such, one cannot completely get away from the fact that the contribution must include some aspect of debugging the simulation. So, what has the inventor added to the stock of "human knowledge"? The contribution to my mind, in its simplest form, amounts to a simulation system comprising a new arrangement for debugging software including an associated interface which provides better diagnostic information enabling the user to locate bugs in either the simulation code or target program code more quickly and more effectively.

Does the contribution fall solely within excluded subject matter? Is the contribution technical in nature?

- 20 The examiner argues that because the simulation, the program running on it and the debuggers running on the host are all programs for a computer "as such" the contribution falls solely within the computer program exclusion. He goes on to say, in the Official Letter dated 20 May 2009, that even if the contribution were considered to lie in:

"a better debugged simulation system, it is still just a better program (or set of programs) executing on a known computer system, with any contribution just being what would be expected from running a better program on a computer. Also, the nature of simulations is that they are in general mathematical methods and the nature of debugging is a mental act."

- 21 He concludes that not only does the contribution fall solely within the field of computer programs but it is also excluded as a mathematical method and/or a mental act. Furthermore, the examiner does not consider the contribution to be technical in nature stating that:

" the alleged contribution can be seen to be the synchronising of the execution of two debugging programs for two other linked programs (the simulation and the program running on the simulation) to produce a better simulation system that is more easily debugged. This is a method of executing a program on a known system which is a problem of program execution and linking. The underlying host does not work any better, any improvement lies in a better simulation debugging program i.e. is due to better general programming and is not an improvement to the host system i.e. firmware programming. Also, the improvement to the design of any devices produced by your simulation are not part of the proposed system and thus cannot be considered part of the contribution. Thus your contribution is just a better way of executing computer programs and not a better general purpose computer per se as in Symbian. I believe that the proposed system and method of this application is not technical in nature, as a matter of practical reality."

- 22 Dr Keston again considers the examiner's argument to be flawed. She argues that the debug interface, which is allegedly part of the contribution cannot be considered to be a computer program as such, at least because it can be implemented in either hardware (e.g. a defined area within the memory of the host processor) or software (e.g. as application program interface within the simulation code). She argues that because the invention has an aspect that can

be implemented in hardware it should be equally as patentable as similar arrangements in other technical fields e.g. telecoms, engine management, process control systems etc.

23 She then went on to address the issue of technical contribution. She argues that whilst the simulation, the target test program and the debuggers may well be computer programs, the examiner has not demonstrated that they are computer programs “as such” i.e. that each of these programs and any interplay between them have no “*further technical effect*” which goes beyond the normal physical interactions between program and computer. In fact, she would argue that, the synchronisation of the debuggers and the provision of the debug interface do provide a further technical effect in the form of a more efficient and more accurate diagnostic tool for processor architecture design and testing.

24 Furthermore, Dr Keston would have me believe that the invention should not be excluded from patentability on the basis that it solves both a technical problem external to the computer and one within the computer itself. In particular, she argues that:

“(i) it solves the problem external to the computer of troubleshooting a programmable device (e.g. processor) architecture in a more efficient and reliable manner; and

“(ii) it solves the problem within the computer of how to provide a dual view of performance of the simulator and the programmable device in a single testbed simulation.”

25 Additionally, Dr Keston argues that, as stated by Lord Neuberger in paragraph 56 of *Symbian*, one should consider what is achieved by the program as a matter of practical reality. If the program operates differently on a technical level to previously known programs, then it should be patentable as the invention is then more than just a computer program as such. In support of her argument, she referred me to the decision in *ARM Limited*⁵ where the Hearing Officer determined that a technically improved simulator makes a technical contribution and so is more than a program for a computer as such. Unlike prior art systems, she says the simulation system of the present invention can provide mechanisms for accessing both the host resources and the target resources and hence can pinpoint the location of the bugs in the simulation code and target program code more precisely. This difference means that the present invention operates differently at a technical level, not merely at a program level, and so does not fall within the computer program exclusion.

26 Does the contribution relate solely to a mathematical method or a mental act? The examiner has said little more than that “...*the nature of simulations is that they are in general mathematical methods and the nature of debugging is a mental act.*”

27 Dr Keston argues that because the invention requires a cycle accurate model of a programmable device and simulators by their very nature exclude the

⁵ ARM Limited’s application GB002696.6 BL 0/292/04

involvement of the human operator then the contribution cannot be considered to be a mental act. She also argues that the contribution cannot be considered to constitute solely a mathematical method, because at most only the simulation code itself could be considered a mathematical method. All other features provide synchronised access to host resources or target resources, and such synchronised access to resources cannot be implemented by a sequence of mathematical operations. Moreover, the actual contribution of the invention, the knock-on effect of better diagnostic information which can identify locations of bugs at a more fine-grained level and hence enable progress in device development, is clearly not a mathematical method in Dr Keston's opinion.

- 28 In summary, Dr Keston's argues that simulation of programmable device is a technical field and it would be inequitable to preclude patentability of all inventions in this field of technology. While some elements of the present invention are implemented using a computer program, the actual contribution made by the invention is not solely a computer program. Indeed, the debug interface used to enable synchronism between simulation host debugger and target device debugger can be implemented in either hardware or software. Nor can the contribution be considered to be a mental act or a mathematical method as such. The present invention provides a more efficient and powerful diagnostic simulation tool for processor architecture exploration (e.g. instruction set design and pipeline design). Furthermore, the improved diagnostic information provided by the invention enables the design of the target programmable device (processor architecture and software compatibility) to be provided as to reduce the error occurring when the target programmable device is actually fabricated. The invention therefore provides a technical contribution to save it from exclusion.
- 29 It is quite clear from the description, that whilst the debug interface could be implemented in hardware, it is more likely than not to be implemented entirely within the computer program of the simulator or debugging software. It is worth stating, that a technical contribution does not exist merely because the invention or an aspect of the invention, in this case, the debug interface may be embodied in hardware, something else is required. It is the substance of the invention that matters and, I do not think in this case it is sufficient merely to state that because the debug interface could be implemented in hardware that the invention should avoid exclusion.
- 30 That said, there is no doubt in my mind that the contribution requires a computer program for its implementation. However, the mere fact that the invention is effected in software does not mean that it should be immediately excluded as a computer program as such. What matters is whether or not the program provides a technical contribution.
- 31 Does the invention make the computer a better computer or solve a technical problem within the computer itself? It is clear to me that if the invention made the computer a better computer in the sense of running more efficiently and effectively as a computer, then this would constitute a valid technical contribution sufficient to save it from exclusion. That was the case in *Symbian*. However, this does not appear to be the case here, where the host computer is entirely conventional. What the applicant has done is to introduce additional debugging

software into the simulation to enable the user to better identify any bugs which may be present in the simulation or target program code. Whilst this will obviously have an effect within the operation of the computer in the sense that any new program will, I think something more is required to avoid the exclusion, and (in this context) that “more” is something which makes the computer work better. The invention does not have this effect. It makes the computer, as a computer, work differently in the sense of processing data in a different way, but it does not make it work better, faster or more reliably in terms of its performance. What the applicant has effectively done is to produce, brilliant though it may be, a new computer program or suite of programs for debugging a simulation.

- 32 Does the invention provide a technical effect external to the computer? Dr Keston argues that it does, in the sense that “*it solves the problem of troubleshooting a programmable device (e.g. processor) architecture in a more efficient and reliable manner*”. However, I am not convinced by her arguments. What the invention does is provides a new means for troubleshooting a simulated processor architecture, i.e. one which does not exist outside the confines of the computer or the simulation software. The invention does not necessarily produce a better programmable device at the end of the simulation even though it may reduce the time it takes to get to the final design. There is therefore no technical contribution outside of the computer or the simulation running on the computer.
- 33 Whilst the previous *ARM* case referred to by Dr Keston relates to a simulator for a programmable device, that is where the similarity ends, and the fact that the Hearing Officer found the invention in *ARM* to be patentable is of little bearing here where the facts are entirely different. However, I note that the Hearing Officer in that case found the invention to provide a technical improvement in the construction and operation of the simulator. Indeed, he referred to the way in which data was transferred between stages in the pipeline processor as “*a “neat” technical solution to a technical problem*”. He found the invention to operate at a much “*deeper level than the computer program that implements the simulator*” and for that reason, was prepared to accept that the invention provided a technical contribution and was therefore patentable. In this case, the invention does not provide a technical improvement in the construction or operation of the simulator. Indeed, I do not think that the operation of the simulator as such is improved in terms of its performance and reliability. What the applicant has done is to add functionality to the simulation software by the addition of new software to enable bugs to be located more easily. To my mind this does not provide the same “deep level” technical improvement as was the case in the previous application and hence does not provide a technical contribution in that sense.
- 34 Does this mean that simulators per se are not patentable? No, what matters is do they provide a technical contribution; if they do then they will be patentable. Whilst I do not think the many granted European patents referred to me by Dr Keston at the hearing are of any real relevance to the facts of the current case, they do provide evidence that in the right circumstances simulators are patentable.
- 35 Having considered all the evidence made available to me, and all the arguments put to me at the hearing, I do not consider the invention to provide a technical contribution, and as such it would seem to fall squarely within the computer

program exemption of section 1(2)(c).

- 36 The examiner has also argued that the invention is excluded as a mathematical method and a mental act. However, having found the invention to be excluded as a computer program, I have no need to decide this issue.

Conclusion

- 37 In the light of my findings above, I conclude that the invention as claimed is excluded under section 1(2) because it relates to a computer program as such. Having read the specification I do not think that any saving amendment is possible. I therefore refuse the application under section 18(3).

Appeal

- 38 Under the Practice Direction to Part 52 of the Civil Procedure Rules, any Appeal must be lodged within 28 days of the receipt of this decision.

P Slater

Deputy Director acting for the Comptroller