Research Focus

Configurable Computing (Or, what to do with soft computer hardware)

by Markus Weinhardt

Traditionally, there are two options to implement a computing system, which vary largely in terms of flexibility and performance. The first option is to use general-purpose microprocessors (such as the Pentium chips found in most personal computers). They are very flexible since a program (software) determines which operations are performed. Also, high-level programming languages and sophisticated development environments make it relatively straightforward to develop applications. But microprocessors are relatively slow since they execute most operations sequentially. The second option is the use of custom hardware circuits, or ASICs (application-specific integrated circuits). ASICs are tuned to perform a specific task very fast. This is mainly achieved by performing operations in parallel and avoiding the program decoding overhead. But ASICs are inflexible since they cannot be programmed for other tasks. Examples of ASICs are custom graphic chips for personal computers, which can paint pictures on the screen much faster then a general-purpose processor can.

Recently, a new development in integrated circuits has offered a third option: large, fast FPGAs (field-programmable gate arrays). As opposed to ASICs, the functionality of FPGAs is not determined in the factory. They are rather configured or programmed in the end product. To achieve this flexibility, FPGAs contain arrays of configurable logic blocks, programmable connections between the blocks, and input/output blocks for external communication (see Figure A). The logic blocks can be configured to perform any basic binary operations on its inputs, such as AND, OR, XOR etc. and the results of these operations can be stored in registers and wired to the inputs of any other blocks. Arithmetic operators like adders, comparators, and counters are built by combining several logic blocks. Thus arbitrary digital circuits can be implemented in FPGAs, and they can be reconfigured within milliseconds at any point during use. FPGAs blur the boundary between software and hardware: they approach the performance of ASICs while maintaining the flexibility of programmable processors.

Configurable Computing is the relatively new field of research, which explores the use of configurable hardware, mainly in the form of FPGAs. There are many promising application areas for this technology. FPGAs allow "field hardware upgrades" for devices with fast changing standards, like cable set-top boxes or mobile phones. One can also imagine novel appliances like a mobile phone, which reconfigures itself into an MP3 audio player. And a standard microprocessor and FPGAs can be combined to form hybrid computers in which the FPGAs act as flexible coprocessors. Figure B shows a typical architecture with a coprocessor board containing FPGAs and local memory supporting them. A coprocessor can, for instance, perform hardware data encryption at one point in time, and then be instantly reconfigured for image processing. In general, the coprocessor performs computation intensive, repetitive parts of an application, which would be slower on the microprocessor. This approach is therefore called "hardware acceleration". Even hybrid chips, which contain both a microprocessor and configurable hardware, are already commercially available.

However, there are many problems to overcome in order to make Configurable Computing a mainstream technology. Most importantly, "programming" an FPGA is not as simple as programming a microprocessor: it essentially means designing a digital circuit, i.e. a structural description of operators and wires. But directly stating what operators need to be configured into the FPGA and how to connect them is a very tedious and error-prone process, and is specific to the particular FPGA device. Therefore so-called synthesis tools have been developed to simplify FPGA circuit generation. They generate the circuit structures from higher-level, device independent descriptions. Several researchers take this even further, aiming to synthesize hardware directly from a software programming language like C. This means that the user describes the desired functionality of a system, not its structure, and the tool automatically determines which hardware operators are needed. This approach opens Configurable Computing to software developers without hardware design experience.

My Marie Curie fellowship project "Multi-level Synthesis and Optimisation for Reconfigurable Hardware" involves the use of C programs as well as lower-level specifications. It concentrates on hardware acceleration on hybrid computers. It is useful to specify entire applications for these computers as software programs, and to automatically generate coprocessor configurations for those program parts which can exploit the configurable hardware. This approach -called hardware/software codesign - allows experimenting with different hardware/software partitionings, i.e. allocations of program parts to software (on the microprocessor) or hardware (on FPGAs).

I devised a technique called "Pipeline Vectorization" which adapts vectorization - a method originally developed to simplify the programming of supercomputers - for configurable hardware. It synthesizes pipeline circuits (which efficiently perform many operations in parallel) from suitable software program parts. The design flow is outlined in Figure C. Using a prototype compiler implementing this technique, we accelerated image and signal processing programs significantly, using a standard PC

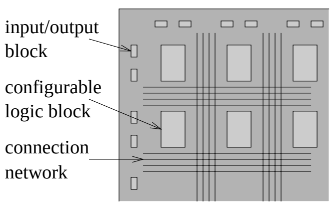

Figure a: FPGA Structure

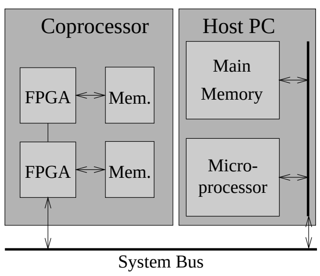

Figure b: Hybrid Computer

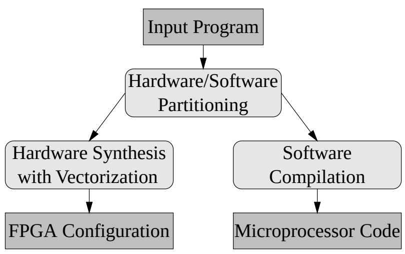

Figure c: Design Flow

and an off-the-shelf FPGA coprocessor board. Other aspects of my project include the automatisation of the hardware/software partitioning and the use of run-time reconfiguration, i.e. reconfiguring FPGAs within an application to overcome hardware resource constraints.

References:
General:
J. Villasenor and W. H. Mangione-Smith: "Configurable Computing", Scientific American, June 1997
IEEE Computer magazine, Special Issue on Configurable Computing, IEEE Computer Society Press, April 2000

My work:
M. Weinhardt and W. Luk: "Pipeline Vectorization for Reconfigurable Systems", Proceedings Int. Symposium on FPGAs for Custom Computing Machines 1999 (FCCM'99), IEEE Computer Society Press, 1999
M. Weinhardt and W. Luk: "Pipeline Vectorization", submitted to IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (for pre-print see www.doc.ic.ac.uk/~mw8/#publ)

Markus Weinhardt
m.weinhard@computer.org
is a Marie Curie fellow at Imperial College, London, where he is working as a post-doc on Configurable Computing. He received a Doctor of Engineering degree in Informatics from the University of Karlsruhe, Germany.

MCFA local groups

A Day at the National Contact Point

Camilla Lauren-Määttä visited the UK National Contact Point

The UK National Contact Point for Marie Curie Fellowships is located in the Office of Science and Technology (OST) at the Department of Trade and Industry in London. Within the OST office, the International Directorate is responsible for planning, co-ordinating and managing UK involvement in the European Union's science and technology activities including the European Union's Fifth Framework Programme. The directorate is divided into four sections called Europe I, containing the National Contact, Europe II, Europe III and The Rest of the World.

I meet up with Paul Wright and Jane Grady from Europe I in the big glass building where they work, centrally located in Westminster. In all, there are five people employed on Europe I, which also has responsibility for other areas of work aside from the Improving Human Potential (IHP) Programme. Jane and Paul are both civil servants and Paul, who is on loan from the Department for Social Security for two years, is the UK's National Contact Point. If you have contacted the UK National Contact Point, you will probably have spoken to Paul.

Most of Paul's work consists of replying to enquiries, for instance by phone or email. Every university also has a European Liaison Officer of some kind as well as people dealing with the administration of fellowships. As a consequence, most of the day-to-day problems are sorted out locally, but research fellows usually turn to the National Contact for more specific problems relating to working and earning in the UK. The Contact Point also has its own web-site: (www.dti.gov.uk/ostinternational).

In 1999, the National Contact Point had 1,000 enquiries, and Paul expects an increase this year. He attends information meetings at the European Commission (EC) in Brussels a couple of times per year, but most of his contacts with the EC are by telephone. During an ordinary working day, two thirds of his time is spent dealing with enquiries. Typically, researchers want to know what is available or how to fill in an application form. Also, researchers coming to the UK may have practical problems relating to taxation or difficulties with their supervisor. In addition, they are worried about exchange rates, about the amount of money that the university is getting or the fact that they will have to pay National Insurance contributions.

Although Paul's main role is to provide information, he has also produced a model contract for use between institutions and fellows, which can be downloaded from the Contact Point's website. There is no requirement for institutions to use the contract, but it has been approved by the EC contract and legal staff. In addition, fellows can ask Paul to check whether obligations set out in the institution's contract with the EC are fulfilled. However, any disputes are for the fellow to take up with the institution, and then with the EC. As a complement to MCFA's Welcome Pack, Paul has also produced his own document on taxation and social security matters and he can also advise fellows about who to contact at their university or at the EC if they are having problems.

I asked Paul about the confusing fact that some fellows are paying tax in Britain, while others are not. He replied that tax in the UK is very different from most other European countries. The written contract is only one part of the picture. The Inland Revenue, the Ministry responsible for taxation matters, will look at the way a person does his work, how he is supervised and how the contract is set out in order to determine his/her tax liability.

Most fellows pay tax, but there are circumstances when fellows can be exempt from tax. For example, one fellow may not be required to pay tax because his or her work conditions are like those of a student (i.e., he/she has a contract of training, not a contract of employment).

Another fellow may choose to pay tax in his home country, if he has another income at home. A very independent fellow may be self-employed; in which case his tax arrangement will be totally different. Altogether, there are no rigid regulations and most cases will be looked at according to individual merits based on past practice and case law. However, the allowances paid to fellows are designed to take into account the tax and social security costs and fellows in UK should be receiving a similar amount to that of a UK national, with the same experience, working in the UK.

Paul also receives enquiries about the use of the Commission's allocation to the host institution (1200 euros a month for lab-based work or 900 euros for non-lab-based work). The host is granted an allocation to cover its administration costs, overheads, etc., the inevitable extra costs of hosting a fellow and running his/her project. The way in which this host's allocation is spent is up to the host to decide. Much of this money may be used in administration costs, but a host may also be able to dedicate some of it more directly towards the project. If fellows feel that the host's allocation could be spent in a particular way, they should approach their supervisor, department or host's administrative offices.

In addition to dealing with enquiries made to the Contact Point, Jane has attended a series of seminars round the country, most recently at Brighton and Sheffield Universities. At the