SYNDEX EXECUTIVE KERNEL DEVELOPMENT FOR DSPs TI C6X APPLIED TO REAL-TIME AND EMBEDDED MULTIPROCESSORS ARCHITECTURES

Y Le Mener[1], M Raulet[1], J-F Nezan[1], A Kountouris[2], C Moy[2]

(1)
ARTIST/FRE URER Laboratory
INSA Rennes
20, av des Buttes de Coësmes, CS 14315
35043 RENNES Cedex, France
Phone: +33/0 223 238 459 Fax: +33/0 223 238 262
Email : jnezan@insa-rennes.fr

(2)
MITSUBISHI ELECTRIC ITE
Telecommunication Laboratory
80, av des Buttes de Coëmes
35700 RENNES, France
Phone: +33/0 299 841 129 Fax: +33/0 299 842 115
Email : kountouris@tcl.ite.mee.com

ABSTRACT

Real time signal, image and control applications have very important time constraints, involving the use of several powerful numerical calculation units. The aim of our project is to develop a fast prototyping process dedicated to parallel architectures made of several last generation Texas Instruments digital signal processors : TMS320C6X DSP. We use SynDEx, a CAD software developed to improve algorithm implementation onto multiprocessor architectures, finding the best matching between an algorithm and an architecture. We have developed a SynDEx executive kernel for the C6X DSP family, in order to automatically generate a distributed and optimized static executive of the specified algorithm onto those processors.

1 Introduction

More and more applications are using digital processing technologies. They will be more and more complex, improving for instance interactions with users, or manipulating multimedia objects. Telephony, radar, audio, multimedia, process control, software radio and more application's fields are concerned with those problems.

The required computational performances should always be improved for a real time use, especially because of the high complexity of the algorithms developed. Because of its versatility and specialized high speed arithmetic, Digital Signal Processors (DSPs) are usually used, and Texas Instruments is the leading supplier in DSP products.

But software solutions based on monoprocessor architectures are not efficient enough in many cases. Chips are developed to achieve this goal, accelerating special calculations, like motion estimation in image coding, turbo codes in digital communications ... Their integration into platforms requires many specialized people, especially because platforms created are mixed, made up of standard programmable processors (software part) and those chips (specific hardware part). Hence, such platforms are usually dedicated to a single application.

Another solution is the use of several processors for one application [3][4]. By this way, better execution times are reached, keeping versatility. Our goal is to perform an automatic implementation of digital signal or image processing line over multi-C6x architectures. We used SynDEx generic executive kernel.

This paper is organized as follows : in Section 2, we describe the SynDEx software. In section 3, C6x DSP specificity is studied. The executive kernel development process is explained in section 4. Finally, conclusions and perspectives are given section 5.

2 SynDEx V5.2

SynDEx [1][2] is a free academic CAD software system, meaning Synchronized Distributed Execution. It supports the AAA methodology [1](Adequation Algorithme Architecture) for distributed processing, which has been developed in INRIA Rocquencourt, France. The goal of adequation, (French word meaning an efficient matching) is to find the best matching between an algorithm and an architecture. The AAA optimization heuristic handle heterogeneous architectures and inter-

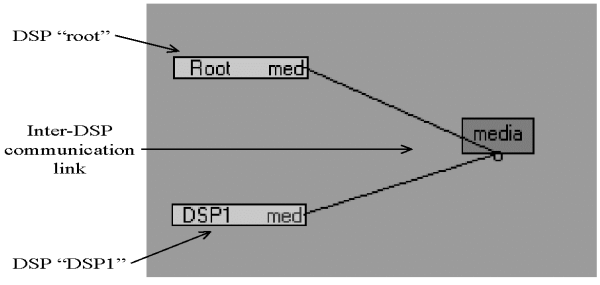

Fig 1 : SynDEx architecture graph

processor communications.

On the one hand, SynDEx use a material graph, which models the multiprocessor architecture. Fig 1 shows an architecture made of two DSP (“root” and “DSP1”) connected each other with one media (called “media”). On the other hand, a software graph describes the dataflow graph (Fig 2).

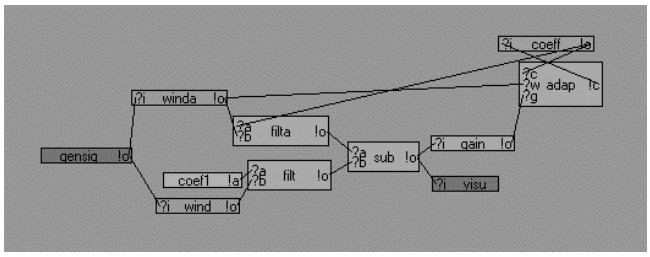

Fig 2 : SynDEx algorithm graph

Then, SynDEx carries out the placement, partitioning, according to the time spent for data transfers between processors, and for each task of the algorithm. The result can be visualized and analyzed thanks the timing diagram generated by SynDEx, shown Fig 3.

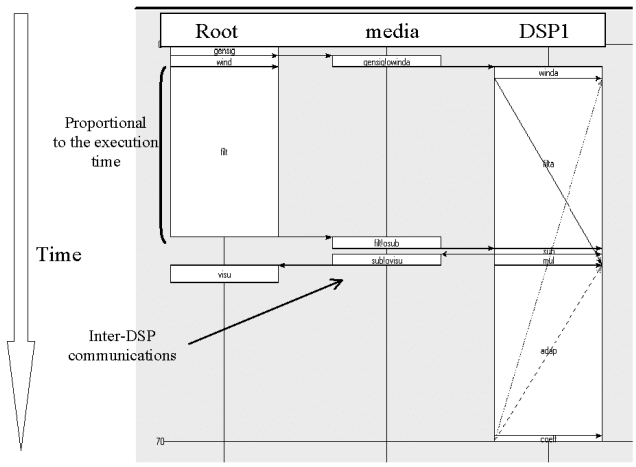

Fig 3 : SynDEx timing graph

SynDEx generates an executive into several source files (fig 4), one for each processor of the architecture, and another one for automating the architecture specific compilation chain. The code generated can insert chronometrical reports for heuristic optimization, or not for the final implementation. The main advantage of Syndex is to avoid operating system like 3L diamond product, implementing the minimum custom-built static executive needed for a given application. Spatial and temporal additional costs are minimized, whereas the order of algorithm tasks is guaranteed and locking is avoided.

Syndex is able to handle different processors : Analog Device ADSP 21060, SHARC, Motorola MPC 555 et MC 68332, Intel i80x86 et i8096, Unix/Linux workstations, and Texas Instruments TMS320C40. This latter was very used, but its computational performance is no longer efficient enough for new applications. Our aim is to couple SynDEx advantages to C6x DSP power creating its SynDEx automatic code generator.

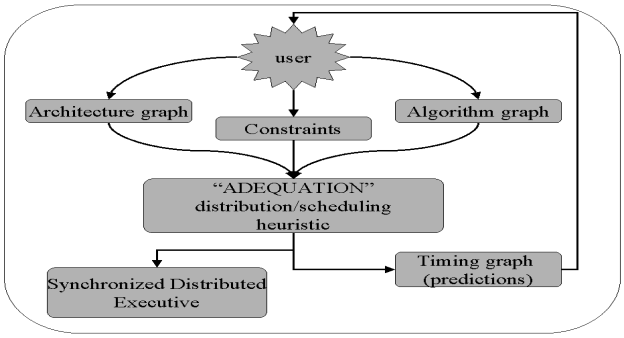

Fig 4 : SynDEx utilization global view

3 Texas Instruments TMS320C6x DSP

3.1 C6x properties [7]

From 150 to 600 Mhz clock rates, C6X are using VelociTI Advanced Very-Long-Instruction-Word (VLIW) architecture, in order to supply up to eight 32-bit instructions to the eight functional units every clock cycle. C6x are high-performance DSPs.

The several peripherals available for the C6x devices, such as various memory configurations, ports, timers, direct-memory access, and power-down logic, make it very interesting in real-time and embedded parallel architectures. C6x peripherals are fully programmable in C language.

Code Composer Studio software [6] speeds and enhances the development process for programmers. It accepts C source code and produce assembly language source code. With its optimizer, the programmer can reach high-performances without using assembly language, specific and complex with the VLIW architecture.

C40 DSPs contain communication ports (CPs), providing inter-C40 communication. This specific interface allowed to construct parallel processor systems very easily. C6x DSPs no longer contain CPs. Manufacturers must insert additional digital resources between two C6x in order to make their communication possible (fig 5). So, inter-C6x communications are architecture dependant. This, and because it was written in the C40 specific assembly language, make that the SynDEx C40 executive kernel can no longer be used for C6x.

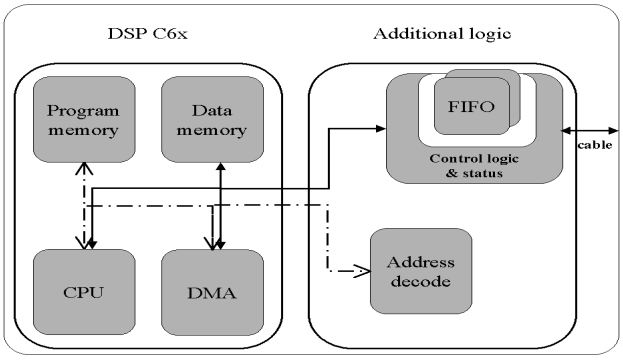

Fig 5 : C6x additional logic for communications

Additional logic added by manufacturers is generally a FIFO buffer which communicate with another FIFO connected to the other DSP. The size, the control and status information of FIFOs can change from a platform to another one. A FIFO can create interruptions in the DSP program, for instance at the end of a transfer. The number and the meaning of those interruptions can change too. FIFO's specifications are always done by the manufacturers with examples.

3.2 Development method :

The executive generated by SynDEx is divided into several source files, each of them contains an intermediate code composed of a list of macro-calls of the intermediate generic kernel SynDEx. Those macro-calls will be translated by the macro-processor M4 [5] into a source code in the compilable language for target processor. We have created M4 libraries for C6x forming the SynDEx C6x Kernel.

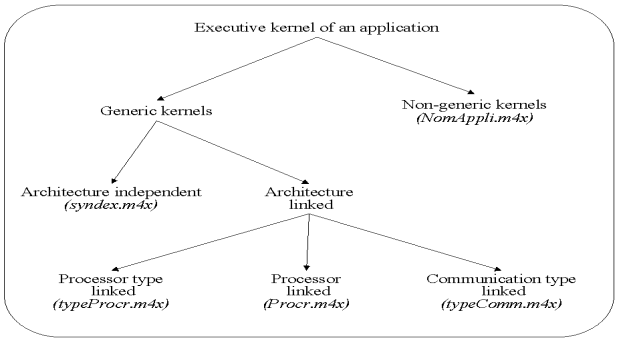

Fig 6 : Executive kernel organisation

We decided to develop C6x Kernel in C language, in order to make it partly reusable for any C programmable DSP, and because it is not an important waste of time on a complete application.

The C6x executive kernel has been divided into several libraries (fig 6), enabling its easy adaptation to a new architecture. Non-generic library contains M4 macros for the application, such as specific input/output functions. Architecture independent library contains macros used whatever the architecture target. The others are architecture dependent : processor type, processor or communication type dependent.

We developed kernels for two different platforms to ensure that macros can be reused. The adaptation of our work for another multi-C6x architecture is limited to the communication sequence adaptation for a new media, if needed.

4 Developed kernel

4.1 Inter-C6x communications developed

Syndex macrocode creates two interleaved schedulers : one for computation tasks and the other for communications, allowing parallelism of those actions. We have chosen the use of multi-channel DMA transfers (fig 7), maximizing this parallelism and timing performances. The DMA contains four different channels, we have linked each communication media to one channel. SynDEx architecture graph can have four connection for each C6x.

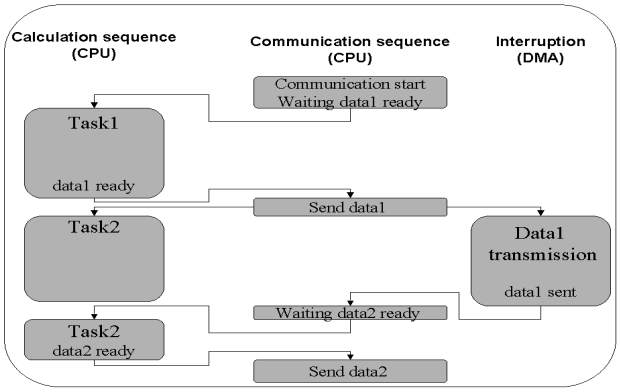

Fig 7 : parallelism computation/communication

We have created communication sequences for each different media, but they are always structured as follow :

- Computation of the number of packets (For a communication, we have to split data into FIFO length packets).
- FIFO selection (depending on the FIFO)
- DMA Configuration
- Permission of interruption (depending on the FIFO)
- DMA start

4.2 Sundance

We first develop a C6x kernel for a SUNDANCE SMT320 motherboard with two Texas Instruments Modules SMT335. Each SMT335 contains one C6x TMS320C6201, whose clock frequency is 200MHz, and one FPGA (*Field Programmable Gate Array*) to connect the two C6201. Each FPGA manages six communication ports (CP: C40 communication standard) et 2 *Sundance Digital Buses* (SDB : Sundance proprietary solution) which allows each 200 Mbytes/sec communications.

We used an M4 [5] macro file already developed for other processors : C.m4x. We create four new M4 libraries : one dedicated for specific C6x macros (C6x.m4x), one for each media (CP.m4x and SDB.m4x) and another one for platform macro (SMT335.m4x). The last one is needed for the initialization of the board parameters. We can notice that the two media specific files are very close, and it is a good example for future developments.

4.3 Pentek

Pentek platform [9] is made of four DSP TMS320C6201. Each C6x have three communication links: two bi-directional inter-C6x links and one for the I/O interface.

This stage allowed us to verify architecture independence of our work, and to develop another media M4 macro file : BIFO.m4x, for pentek specific inter-C6x communications. Similarly the SMT335, we have created PENTEK.m4x for our Pentek platform. The adaptation for this new architecture was very fast, validating our choices for future developments.

4.4 Application

We have tested the executive generator on a FM decoder application. The main problem was to handle the application and to create its dataflow graph on SynDEx. Then, the placement, partitioning and the executive generation was performed automatically.

5 Conclusions et perspectives

We have created an automatic distributed executive generator for multi-C6x DSP architectures using SynDEx. It was tested onto Texas Instruments TMS320C6201, but it is suitable for other C6x family's DSPs. Furthermore, because of the use of C language, it can serve as a basis for other DSP kernel developments. Static executives generated are custom-built, avoiding to add operating systems and saving platform resources.

This project ensures the fast prototyping process of digital signal applications over parallel architectures, in many technological fields. Mpeg-4 decoder in image processing [8], and software radio [9] applications will soon take advantage of it.

We are working with Syndex V6 developers on the automatic code generator with new features : shared memory, conditional nodes and hierarchy in the SynDEx algorithm graph description. The last issue will allow the description and implementation of application with partial or full reconfigurability.

The additional logic added between two C6x DSPs is often integrated in a FPGA. The implementation of elementary and regular operations onto this material part would give higher performances. So, we plan to study the adding of this material element in the SynDEx material graph, and the generation code for FPGA.

References

[1] C. Lavarenne, Y. Sorel : "Specification, performance optimization and executive generation for real-time embedded multiprocessor application with Syndex" , Proc. *of Real Time Embedded Processing for Space Applications*, CNES International Symposium.

[2] T.Grandpierre, C.Lavarenne, Y.Sorel : "optimized rapid prototyping for real time embedded heterogeneous multiprocessors". *In 7th International workshop on Hardware/Software Co-Design*, pages 74-78, Rome, Italy, May 3-5 1999. IEEE Computer Society, ACM SIGSOFT, IFIP.

[3] M.Cosnard, A.Ferreira : "the real power of loosely coupled parallel architectures". In *parallel processing letters*, pages 103-112, 1991.

[4] S.Srikanteswara, J.H.Reed, P.Athanas, R.Boyle : "A soft radio architecture for reconfigurable platforms". In *IEEE Communications Magazine*, Feb 00.

[5] M.Loukides, A.ORAM. "Programming with GNU software". *O'Reilly*, 1996.

[6] Texas Instruments technical documents : "TMS320C6000 Code Composer Studio – tutorial". Ref spru301c. Available at http://www.dspvillage.ti.com. Feb 00.

[7] Texas Instruments technical documents: "TMS320C6000 programmer's guide". Ref spru198f. Available at http://www.dspvillage.ti.com. Feb 01.

[8] J.F.Nezan, V.Fresse, O.Deforge : "Fast prototyping of parallel architectures : an Mpeg-2 coding application. In CISST'2001 proceedings, Las Vegas, USA, June 01.

[9] A.Kountouris, C.Moy, L.Rambaud, P.Lecorre : "Reconfigurable radio case study : A software based multistandard transceiver for UMTS, GSM, EDGE and Bluetooth". In *IEEE VTC (Vehicular Technology Conference) Fall'2001*, Atlantic City, USA, Oct 01.