UNIVERSITY
OF TRENTO

DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY

38050 Povo – Trento (Italy), Via Sommarive 14
http://www.dit.unitn.it

FFT BENCHMARKING FOR DIGITAL SIGNAL PROCESSING TECHNOLOGIES

Fabrizio Stefani, Antonio Moschitta, David Macii, Dario Petri

March 2004

Technical Report # DIT-04-045

FFT BENCHMARKING FOR DIGITAL SIGNAL PROCESSING TECHNOLOGIES

F. Stefani[1], A. Moschitta[1], D. Macii[1], D. Petri[2]

[1]DIEI – Department of Information and Electronic Engineering, University of Perugia, Perugia, Italy
[2]DIT – Department of Informatics and Telecommunications, University of Trento, Trento, Italy

Abstract – An appropriate choice of the computing devices employed in digital signal processing applications requires to characterize and to compare various technologies, so that the best component in terms of cost and performance can be used in a given system design. In this paper, a benchmark strategy is presented to measure the performances of various types of digital signal processing devices. Although different metrics can be used as performance indexes, Fast Fourier Transform (FFT) computation time and Real-Time Bandwidth (*RTBW*) have proved to be excellent and complete performance parameters. Moreover, a new index, measuring the architectural efficiency in computing FFT, is introduced and explained. Both parameters can be used to compare several digital signal processing technologies, thus guiding designers in optimal component selection.

Keywords: Fast Fourier Transform (FFT), digital signal processing, kernel benchmark.

1. INTRODUCTION

The increasing demand for high-speed digital applications has been driving to an extensive usage of low-cost digital signal processing devices, especially in consumer electronic systems. This rapid and ubiquitous diffusion, along with the different computational capabilities of such devices, makes difficult the selection of the best component for a given application, namely the item that is able to meet given design specifications at the lowest cost. This issue is further stressed by a certain lack of univocal techniques for the characterization of computing devices. However, in spite of theoretical limitations and major technological differences, the performances of different digital signal processing devices such as General Purpose Processors (GPPs), Digital Signal Processors (DSPs), Application Specific Integrated Circuits (ASICs), and Field Programmable Gate Arrays (FPGAs) need to be assessed and compared so that a designer can make an optimal component selection. Also, a quantitative characterization of different components belonging to the same family of devices should be carried out. Generally, processing devices' performance are estimated by measuring processing times [1], namely the execution times of appropriate *benchmark programs* [2]. In particular, it has been shown that digital signal processing performances can be determined effectively using *kernel benchmarks*, i.e. special programs aimed at testing single-job attitudes of various computing systems [3]. Among possible benchmarks, Fast Fourier Transform (FFT) algorithms [4], have proved to be quite effective for testing purposes [5].

In the next sections, at first a short discussion about the most appropriate benchmark strategies for digital signal processing devices will be presented. Then, the performance metrics employed to analyze such components will be explained. Finally, criteria to select the best class of devices for a given application in terms of cost and performance will be illustrated on the basis of a collection of data provided by both manufacturers and third parties.

2. BENCHMARK SELECTION

When a designer has to choose a digital signal processing device in a given system, many different aspects (e.g. cost, performance, power consumption, documentation availability, device flexibility and ease of use) should be taken into consideration. Generally, performance metrics are necessary:

- to quantify the single component contribution to the overall system performance;
- to help choosing between different design alternatives;
- to predict system performance, thus identifying possible bottlenecks in a preliminary design stage.

Due to the large number of degrees of freedom involved in this decision-making process, the overall performance of a digital signal processing device cannot be easily either defined or assessed. Several performance metrics, such as Millions of Instructions Per Second (MIPS), Million of Floating-Point Operation Per Second (MFLOPS), clock frequency and processing time, have been proposed so far. Among these metrics, only processing time is a significant index, because it is the only one to keep into consideration the differences in processor internal structure, whereas other metrics provide usually partial and sometimes misleading results [6]. Processing time can be measured by running several benchmark programs on the system or device to be tested. Four types of benchmarks are usually used to estimate the performance of complex computing systems: application, synthetic, kernel, and technology-specific benchmarks.

- *Application benchmarks* (e.g. System Performance Evaluation Cooperative or SPEC ratio) are focused on measuring system performance under real workload conditions (i.e. when a set of real applicative programs useful to end-users is running on the system).
- *Synthetic benchmarks* are designed to emulate the functionalities of significant applications, while cutting out additional or less important features (e.g. Whetstone and Dhrystone benchmarks [6]).
- *Kernel benchmarks* are aimed at measuring the performance of simple functions designed to represent

key portions of real applications (e.g. digital filtering algorithms).

- *Technology-specific benchmarks* are devoted to point out the main differences of devices belonging to the same technological family (e.g. Programmable Electronic Performance Cooperative or PREP benchmarks [7]).

Application and synthetic benchmarks tend to assess the behavior of a processing device when different types of operations are carried out. Accordingly, these kinds of benchmarks are suitable to estimate the average performance of very involved, general-purpose systems (e.g. personal computers), but they are unable to provide significant results when single devices have to implement a specific set of algorithms, as it usually occurs in most digital signal processing applications. On the other hand, technology-specific benchmarks are usually so specialized not to be useful for cross-technology comparisons. As a consequence, only kernel benchmarks are suitable to estimate univocally the performances of different digital signal processing solutions. Generally, a well-designed kernel benchmark test has to be:

- *representative*: the performance result should be summarized by a single number, usually referred to as index of performance;
- *reliable*: the larger the index of performance is, the faster the device under test (DUT) has to be;
- *reproducible*: running several times the same benchmark program on a given device under the same conditions, performance results should not change considerably (except for uncertainty contributions);
- *portable*: the benchmark has to be independent of a particular technology or architecture.

Besides these basic properties, a kernel benchmark test should also be:

- *meaningful*, because it is pointless to measure an uninteresting or rarely used feature;
- *linear*: any index should be proportional to real performance;
- *easy to use* so that it can be used frequently and correctly;
- *vendor independent*, i.e. independent as much as possible of the influence of external subjects which may be interested in benchmarking results (e.g. for marketing purposes).

Some published results suggest that performance indexes referring to digital signal processing devices can be effectively measured by means of basic algorithms such as numeric filtering, FFT, matrix calculations or operations on bit streams [3, 4]. Among them, FFT algorithms are most valuable benchmarks because they own all the properties mentioned above. In fact, FFT benchmarking is:

- *representative* because several numerical indexes can be calculated from FFT processing times;
- *reliable* as processing time does not depend on the kind of input data;
- *reproducible* because execution time associated with a given device depends only on the algorithm chosen and on the clock frequency [5];
- *portable*: FFT is basically a mathematical operation;
- *meaningful* because FFT is widely used in many digital signal processing applications;
- *linear*: doubling the performance means halving the execution time (or doubling data rate as explained in next section);
- *easy to use* because, once FFT is implemented, only transformation times need to be measured regardless of input values;
- *vendor independent* as standard FFT algorithms are not proprietary.

Besides these features, plenty of documentation is available about different FFT implementations both in terms of design choices and performance analyses. Hence, using a given FFT algorithm as a kernel benchmark not only eases the comparison between devices belonging to different technologies, but allows also verifying the truthfulness of the performance claimed by device manufacturers.

3. PERFORMANCE METRICS

Basically, the performance of any digital signal processing application can be measured in terms of data rate (*DR*), which is referred to as:

$$DR = \frac{N}{t_{proc}} \quad \text{[samples/s]}, \qquad (1)$$

where N is the amount of processed samples and t_{proc} is the processing time (e.g. the time to compute an FFT algorithm on N=1024 complex samples). Notice that this parameter is reliable because it is inversely proportional to the processing time, so that its value grows linearly with performance, as expected intuitively. An equivalent index to express the processing capabilities of a given digital signal processing device is the so-called Real-Time Bandwidth (*RTBW*) that represents the maximum bandwidth with which an effective analog input signal can be processed in real-time, without loss of information. According to the Nyquist theorem [4], *RTBW* is numerically equal to half of *DR*, provided that t_{proc} is the effective FFT processing time, i.e. it is not affected by bus overhead or latencies associated with external memory operations. Under these assumptions, once FFT algorithm has been chosen, the estimated *RTBW* value depends only on the characteristics of the DUT, such as the clock frequency and the operation execution speed. Conversely, if FFT computation time is slowed down by poor bus performance or by other system bottlenecks, the *RTBW* value provided by (1) could be appreciably overestimated.

In addition to *RBTW*, another valuable parameter to assess the performance of a digital signal processing device is the Architectural Efficiency (*AE*), which is referred to as:

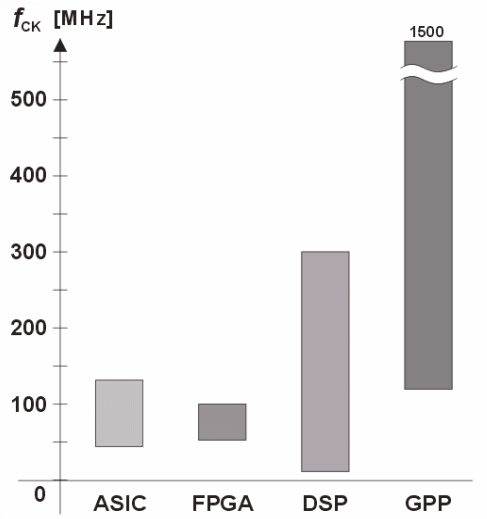

Fig. 1. Range of clock frequencies for the four technologies under test.

$$AE = 2\frac{RTBW}{f_{CK}} = \frac{N}{t_{proc}\, f_{CK}}. \qquad (2)$$

Since *AE* represents the sample rate processed in real-time per clock cycle, this index provides an estimate of the performance related only to the internal structure of the device. Notice that *AE* can be estimated without knowing in advance the total number of clock cycles required to complete the benchmark computation, which is usually not provided by device manufacturers. From designers' point of view, when different components have to be compared, equation (2) can be used to estimate the *RTBW* associated with devices having similar architectural features (i.e. similar values of *AE*), but running at different clock frequencies. On the other hand, if a given value of *DR* or *RTBW* is required by design specifications, an efficient architecture allows the system designer to achieve the wanted performance at a lower clock frequency and, accordingly, with a lower power consumption. This feature can be preferable, for instance, in embedded or portable systems.

4. A PERFORMANCE COMPARISON BETWEEN DIFFERENT DIGITAL SIGNAL PROCESSING TECHNOLOGIES

In this section, the performance indexes described above are used to compare different digital signal processing technologies. To this purpose, FFT processing times of many families and models of FPGA cores, DSPs, GPPs and dedicated ASICs have been collected resorting to either manufacturers' documentation or experimental results obtained from independent third parties [8]. These data concern respectively 5 FPGA cores by Altera, Amphion, Mentor Graphics, and Xilinx, 27 distinct DSP processors by Analog Devices, Motorola, and Texas Instruments, 24 GPPs by AMD, Digital, IBM, Intel, Motorola, Samsung, SGI, and Sun and, finally, 6 ASICs by DSP Architectures, Catalina

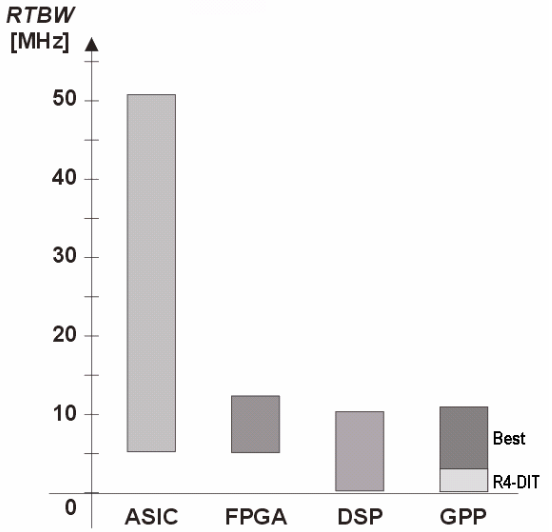

Fig. 2. *RTBW* ranges associated with the calculation of a 1024-points complex FFT, implemented in GPP, DSP, FPGA and ASIC devices.

Research, doubleBW, Radix Technologies and Zarlink. In order to achieve significant and comparable results, the FFT program employed as a benchmark in most tests is the decimation-in-time, Radix-4 butterfly (R4-DIT) Cooley-Tukey algorithm, albeit even other algorithms have been occasionally implemented. Reproducibility of experimental results depends largely upon the considered technology. On one hand, data concerning ASIC devices are highly reproducible due to the dedicated, fixed structure of this kind of components. On the other, programmable devices such as FPGAs, DSPs and, above all, GPPs present so many degrees of freedom that it is difficult achieve always the same results, unless some further auxiliary constraints are set (e.g. layout constraints in FPGA implementation). The majority of the devices under test considered in the current analysis are based on floating-point architectures. Only FPGA cores consist only of fixed-point implementations.

In Fig. 1 the clock frequency ranges of different FFT designs are compared by means of a bar diagram. As stated previously, nominal clock frequency is a quite misleading performance metric. For example, although GPP maximum clock frequencies are about one order of magnitude higher than working frequencies of all others technologies, ASIC performance are usually considerably better than any last-generation GPP implementation. This is due to the fact that the internal device architecture is completely neglected. This consideration becomes even more critical when FPGAs are concerned. In fact, the working clock frequency of any FPGA FFT core is usually much lower than the nominal maximum frequency of the chip employed for the implementation.

A more objective performance analysis is shown in Fig. 2, in which the *RTBW* values related to the computation of a 1024-point complex FFT processed with different models of FPGAs, DSPs, GPPs and ASICs, are displayed. Each bar represents the performance range of the corresponding class of devices. Quite obviously, dedicated ASICs outperform

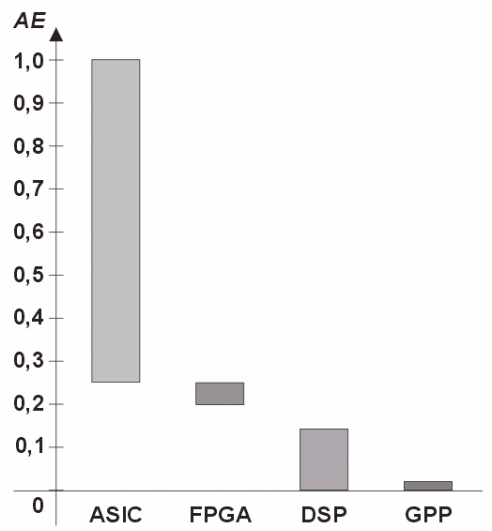

Fig. 3. Architectural Efficiency (*AE*) index range for 1024-points complex FFTs implemented in GPP, DSP, FPGA and ASIC devices.

any other technology. Furthermore, their maximum clock frequency (i.e. 133 MHz) is considerably lower than that of fastest DSP considered (i.e. 300 MHz). By using the more efficient FFT algorithm for a given kind of processors, namely the algorithm that exploits in the best way a GPP's resources, GPP performance proves to be on the same level, or even better, than the fastest DSP solutions, even though they usually work at lower clock frequencies. When an optimal FFT algorithm is employed, the GPP performance can be noticeably higher than the one obtained with the R4-DIT Cooley-Tukey FFT algorithm. In particular, in some cases using the best algorithm has increased the *RBTW* by a factor of 9. FPGA cores are characterized by a *RTBW* that ranges from about 5 MHz to about 12 MHz. Therefore, FPGA cores' maximum *RTBW* is slightly higher than the maximum value achievable by typical DSP implementations even though, as stated previously, only FPGA fixed-point implementations have been considered in the current analysis. Consider also that FPGA performance is strongly related to the features of the software employed for the synthesis.

In Fig. 3 the architectural efficiency (*AE*) ranges are plotted by dividing the *RTBW* value of any DUT by its corresponding clock frequency. As expected intuitively, GPPs are very inefficient in implementing FFT algorithms, due to their general purpose structure. In practice, they reach high *RTBW* values only because of their high clock frequencies. DSPs are more efficient, but they are surpassed by FPGAs and ASICs, which rely on hardware algorithm implementation. On the other hand, since GPPs and DSPs are software-programmable devices, they allow FFT computation of data records of any size, whereas FPGAs and ASICs can elaborate only records of limited dimensions (e.g. some powers of 2). Moreover, GPP- and DSP-based solutions can be upgraded quickly whenever a more effective algorithm is discovered, thus ensuring more flexible performances. For instance, when the best FFT algorithm for a given GPP is chosen, the maximum *AE* can be twice the value achievable with a R4-DIT Cooley-Tukey implementation. By comparing carefully Fig. 2 and Fig. 3, it turns out immediately that there is a strong correlation between performance and architectural efficiency. This means that architecture influence on overall performance is much stronger than the clock frequency, although clock frequency and data rate are linearly related.

Since best performance, in terms of maximum processing speed, are generally related to expensive devices, component costs have also to be taken into consideration when different technologies need comparing. ASIC unit prices range from some hundreds to some thousands dollars per chip, albeit they can decrease considerably for volume purchases. DSP unit prices, instead, range from few dollars to few hundreds when high performance all-in-one chips are considered. FPGA prices are usually included between few tens and few thousands of dollars, depending both on cores and chips costs. Finally, as far as GPP are concerned, their prices are so affected by the technological evolution, that they vary too quickly to be significant. Indeed, the cost of an actual top-performance GPP may be halved within few months.

If the unit price of each DUT is divided by the corresponding *RTBW*, a new interesting parameter, referred to as cost-to-performance ratio, is determined. This index represents the cost per bandwidth unit and it shows, quite surprisingly, that even though DSPs are usually cheaper than FPGAs and ASICs, their cost can be very high compared with their achievable performances. This is probably due to the fact that certain manufacturers integrate the same processing basic architecture in several DSP models which differ mostly in memory capacity, fabrication process, data path width or other secondary characteristics. As a result, devices with similar performances in FFT computation may have very different prices. Similar consideration can be repeated also for GPPs, whereas FPGA cost-to-performance ratio is less variable due to the stronger correlation between device costs and core performances. Such a correlation is even more evident in ASIC devices whose range of cost-to-performance ratio is the narrowest among the 4 technological families considered.

5. CONCLUSIONS

A performance metric analysis and a comparison between four different classes of devices for digital signal processing applications is presented in this paper. According to the proposed approach, by using FFT algorithms as kernel benchmarks, performance metrics such as Real-Time Bandwidth and Architectural Efficiency enable an objective comparison between different technologies, thus helping system designer to select the best component for a given application. Since, in some context such as embedded or portable systems, economic or energetic issues could be more critical than execution speed, clock frequency and cost-to-performance ratio should also be assessed to make the best design decision. The main results deriving from the proposed analysis can be summarized as it follows:

- ASICs are the best choice when the primary design goal is either the performance optimization or the power consumption reduction, especially in floating-point applications.
- DSPs have to be preferred when high flexibility, low

system fabrication cost and medium power consumption are the main design specifications.

- GPPs are the simplest solution when high performance Personal Computers are available. In fact, GPPs are characterized by the best flexibility, and provide also excellent performance if devices working at very high frequencies are employed.
- FPGAs represent a good tradeoff between flexibility, performance and costs as long as the digital signal processing design is implemented in fixed-point arithmetic.

REFERENCES

[1] S. Sahni, V. Thanvantri, "*Performance Metrics: Keeping the Focus on Runtime*", IEEE Parallel & Distributed Technology: Systems & Applications, Vol. 4, Issue 1, pp. 43-56, Spring 1996.

[2] D. J. Lilja, "*Measuring Computer Performance – A Practitioner's Guide*", Cambridge University Press, Cambridge, UK, pp. 111-137, 2000.

[3] J. Eyre, "The digital Signal Processor Derby", *IEEE Spectrum*, Vol. 38, pp. 62-68, June 2001.

[4] A. V. Oppenheim, R. W. Schafer, "*Discrete Time Signal Processing*," Prentice Hall, pp. 587-618, 1989.

[5] "Evaluating DSP Processor Performance", Berkeley Design Technology Inc. (BDTI), (http://www.bdti.com/articles/benchmk_2000.pdf).

[6] D. A. Patterson, J. L. Hennessy, "Computer Organization and Design – The Hardware/Software Interface," *Morgan Kaufmann Publishers*, San Francisco, CA, 1998.

[7] U. Meyer-Baese, "Digital Signal Processing with Field Programmable Gate Arrays," *Springer Verlag*, Heidelberg, Germany, 2001.

[8] www.miteq.diei.unipg.it/file/prestazioni/performance.pdf.

AUTHORS:

[1]Fabrizio Stefani, Antonio Moschitta and David Macii are with the Department of Information and Electronic Engineering, University of Perugia, via Duranti 93, 06125, Perugia (ITALY).
Phone: +39 075 5853933;
Fax: +39 075 5853654;
E-mail: moschitta@diei.unipg.it.

[2]Dario Petri is with the Department of Informatics and Telecommunications, University of Trento, via Sommarive 14, 38050, Povo, Trento (ITALY).
Phone: +39 0461 883902;
Fax: +39 0461 882093;
E-mail: petri@dit.unitn.it.