Recent trends and modification in core-based design of embedded DSP processor system

Y. Divya[1]
[1]*Department of Electronics and Instrumentation Engineering, CVR College of Engineering, Hyderabad, Telangana, India*

Abstract- This paper considers the design of wireless communications systems which are implemented as highly integrated embedded systems comprised of a mixture of hardware components and software. An introduction part presents digital communications systems, classification of processors, programmable digital signal processing (DSP) processors, and development and implementation of a flexible DSP processor architecture. This introduction is followed by a total of seven publications comprising the research work. In this paper the following topics have been considered. The processor core itself as a bare hardware circuit is not usable without various software tools, function libraries, a C-compiler, and a real-time operating system. The set of development tools was gradually refined, and several architectural enhancements were implemented during further development of the initial processor core. Furthermore, the modified Harvard memory architecture with one program memory bank was replaced with parallel program memory architecture. With this architecture the processor accesses several instructions in parallel to compensate for a potentially slow read access time, a characteristic which is typical of, for example, flash memory devices. The development flow for heterogeneous hardware/software systems is also studied. As a case study, a configurable hardware block performing two trigonometric transforms was embedded into a wireless LAN system described as a dataflow graph. Furthermore, implementation aspects of an emerging communications system were studied. A high-level feasibility study of a W-CDMA radio transceiver for a mobile terminal was carried out to serve as a justification for partitioning various baseband functions into application-specific hardware units and DSP software to be executed on a programmable DSP processor.
Keywords – Wireless LAN system, W-CDMA radio transceiver, DSP processor

I. INTRODUCTION

The field of DSP is as of now the most appealing, quickest developing section of the semiconductor business. As microchip chips impelled the PC time, in like manner streamlined DSP processors now constitute the main thrust behind the broadband communications time as cutting edge remote and wire line frameworks. Cell phones and different remote terminals are a definitive large-scale manufacturing gadget for shopper markets. Keeping in mind the end goal to represent the size of the volume, it has been evaluated that approximately 275 million cell phones were fabricated worldwide in 1999. Notwithstanding traditional voice benefits, the general population will soon have remote access to constant video and information administrations whenever, anyplace. This entrance will mostly be empowered by modern communications motors in light of the most recent innovations incorporated into a framework on a chip. It is obvious that this sort of chip will be a superior multiprocessor framework which joins three to four programmable processor centers, extensive measures of on-chip memory, improved equipment quickening agents, and different interfaces for associating the chip to the off-chip world. Focal parts in these chips are programmable DSP processor centers which, as opposed to application-particular incorporated circuits, give more prominent adaptability and speedier time to market.

A system can be defined as a way of performing task according to some predefined fixed plan, program and a set of rules. It may be also defined as an arrangement in which all units assembled and work together according to plan or program to achieve defined objectives. In the same way embedded system may be also defined consisting of components such as hardware, Software designed to performed specific task according to some plan. Sometime embedded system is designed to perform more than one dedicated function with real-time computing constraints. Embedded systems are the product of hardware and software co design for a specific purpose within a defined time period. A programmable hardware is designed to specific digital circuit to perform a particular task by software. Software which is dedicated for a specific application is embedded in to the memory (FLASH, EEPROM, /EPROM/MASK ROM) of a CPU. In practice embedded system is a combination of computer hardware and software with some additional parts either mechanical or electronic designed to perform a dedicated task. It is usually embedded as part of a complete device including hardware and mechanical parts.

II. COMPONENT OF AN EMBEDDED SYSTEM

Embedded system also having three main components of Input, Process and Output like any other systems. Where process mainly consist of embedded software and Hardware. Input may be in the form of analog or digital. In case of analog appropriate ADC circuit has to be interface for conversion of input into digital form. Output may be through LC Display, Computer Monitor etc. Roughly components of an embedded system may be grouped in to the following figure

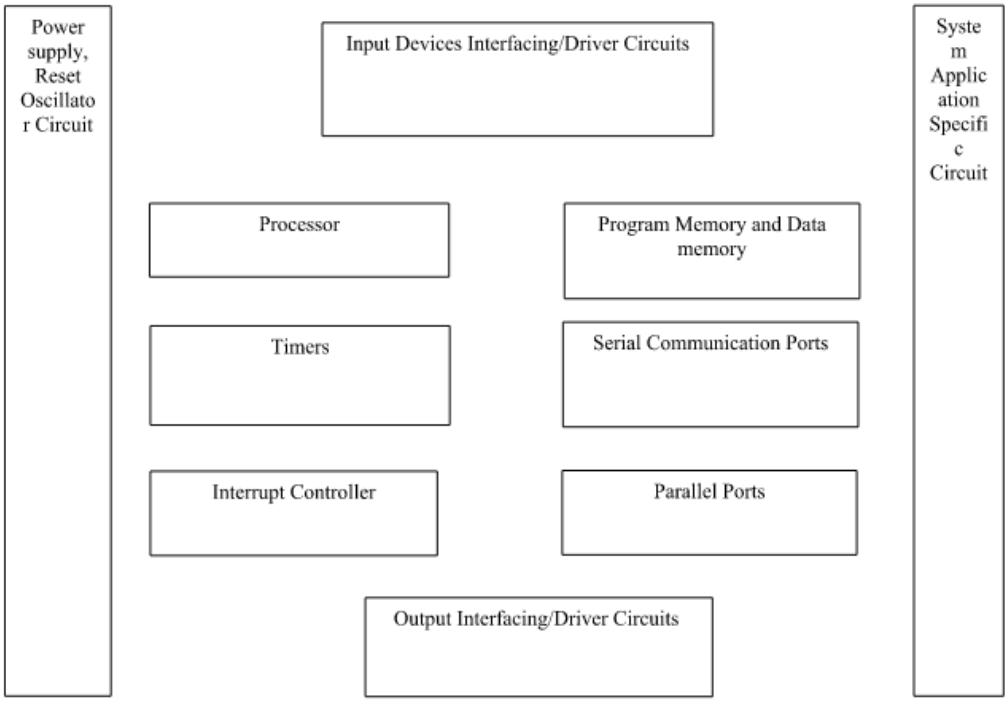

Figure 2.1: The Components of an embedded system hardware

2.1 Wireless Communications System Design

The fields of wireless communications systems and digital signal processing are very broad. Thus, instead of trying to cover these extensive fields in great detail, this chapter prepares the reader with the fundamental concepts behind DSP systems, their primary application area, and the plethora of issues associated with the design of processor core-based wireless communications systems.

2.1.1 Digital Signal Processing

Genuine signs are simple by nature. In any case, computerized PCs work on information spoke to by parallel numbers that are made out of a limited number of bits. In computerized flag preparing (DSP), simple signs are spoken to by groupings of limited accuracy numbers, and handling is actualized utilizing advanced calculations. Accordingly, instead of a constant time, ceaseless abundance simple flag, an advanced flag is portrayed as discrete-time and discrete-sufficiency. Contrasted with simple frameworks, performing signal control with DSP frameworks has various favorable circumstances: frameworks give unsurprising precision, they are not influenced by part maturing and working condition, and they allow propelled operations which might be unrealistic or even difficult to acknowledge with simple segments. For instance, complex versatile separating, information pressure and mistake remedy calculations must be actualized utilizing DSP strategies. DSP frameworks additionally give more noteworthy adaptability since they are frequently acknowledged as programmable frameworks that enable the framework to play out an assortment of capacities without adjusting the computerized equipment itself. Moreover, the gigantic advances in semiconductor advances allow effective equipment executions that are portrayed by high dependability, littler size, bring down cost, low power utilization, and elite.

A square chart of a DSP framework is portrayed in Fig. 1. As appeared in the figure, a DSP framework gets input, forms it, and produces yield as per a given calculation or calculations. The simple and computerized spaces associate by utilizing simple to-advanced (A/D) and advanced to-simple (D/A) converters. A/D change is the way toward changing over a simple

2.2 Wireless Communications System Design

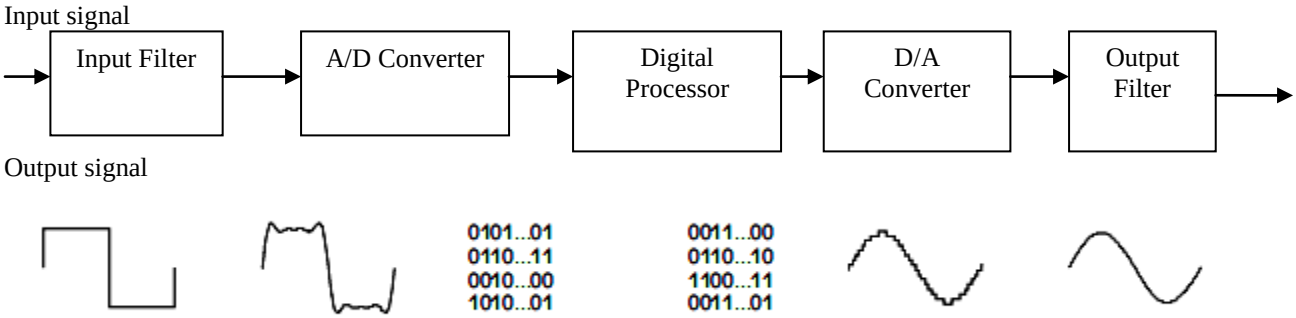

Figure 2.2 . Block diagram of a simplified, generalized digital signal processing system.

The waveforms and digits illustrate signal representation in the system. The A/D converter block includes a sample and-hold circuit [Bat88]. A/D: analog-to-digital, D/A: digital-to analog. Signal, i.e. a voltage or current, into a succession of discrete-time, quantized paired numbers, or tests. In this way, the A/D transformation process and the change rate are alluded to as inspecting and examining rate (on the other hand testing recurrence), individually. Keeping in mind the end goal to abstain from associating of recurrence spectra in A/D change, the info signal data transfer capacity must be constrained in any event to a large portion of the examining recurrence with a simple channel going before the converter. D/A transformation is the inverse procedure in which paired numbers are converted into a simple signal. In D/A change, simple sifting is required to dismiss the rehashed spectra around the whole number products of the testing recurrence since signal generation in just a specific recurrence band is of intrigue. Examining presents some blunder in computerized signals. This mistake is because of quantization clamor and warm commotion created by simple parts.

The fundamental segment of a DSP framework appeared in Fig. 1, is the advanced processor. Practically speaking, this part can be founded on a chip, programmable DSP processor, application-particular equipment, or a blend of these. The advanced processor executes one or a few DSP calculations. The essential DSP operations are convolution, connection, separating, changes, and tweak. Utilizing the essential operations, more mind-boggling DSP calculations can be developed for an assortment of utilizations, for example, discourse and video coding. Ongoing frameworks are obliged by strict prerequisites concerning the reiteration time of a calculation or a capacity. Accordingly, a constant DSP framework is a DSP framework which procedures and produces signals progressively.

III. SYSTEM DESIGN FLOW

Embedded framework outline for remote terminals is emphatically affected by framework level contemplations. At framework level, essential impacts incorporate remote working condition, beneficiary portability, applications, and requirements on framework cost, estimate, control utilization, adaptability, and configuration time.

In, an embedded framework is characterized as a constant framework playing out a devoted, settled capacity or capacities where the accuracy of the outline is vital. Particular and plan of these frameworks comprises of portraying a framework's coveted usefulness and mapping that usefulness for execution by an arrangement of framework segments. As showed in, there are five fundamental plan errands in embedded framework outline: particular catch, outline space investigation, detail refinement, equipment and programming outline, and physical outline.

Amid detail catch the essential objectives are to determine and recognize the vital framework usefulness and to in the long run produce an executable framework display. Utilizing reproductions, this model is utilized to check rectify operation of the coveted framework usefulness. Notwithstanding standard programming dialects, for example, C, generally received devices for demonstrating DSP calculations are graphical piece chart-based dataflow recreation situations and content based specialized processing conditions. These devices are regularly joined by broad pre-outlined model libraries and they give capacities to information investigation and perception. Utilizing these devices, the conduct of a whole framework can be displayed and recreated. For instance, it is conceivable to portray a computerized transmitter-recipient chain and test it by utilizing a practical model of Physical Description (to Manufacturing and testing). The radio transmission channel. Moreover, most information stream reproduction conditions permit heterogeneous reenactments with execution level equipment depictions.

In configuration space investigation the displayed usefulness is changed and apportioned into various target structures, or stages, that contain distinctive arrangements of assigned framework parts, for example, programmable

processors, ASICs, and memory. Utilizing estimation, the objective is to locate a possible engineering that meets the criteria for constant operation, performance, cost, and power utilization.

IV. PROGRAMMABLE PROCESSOR ARCHITECTURES

An instruction-set architecture (ISA) can be viewed as a set of resources provided by a processor architecture that are available to the programmer or a compiler designer. These resources are defined in terms of memory organization, size and type of register sets, and the way instructions access their operands both from registers and memory. In the early phases of processor evolution, designers began to develop instruction sets so that the processor directly supported many complex constructs found in high-level languages. This approach lead to very complex instruction sets. Often execution of an instruction was a long sequence of operations carried out sequentially in a processor that had very restricted hardware resources. An execution sequence was essentially stored as a set of macrocodes that correspond to low-level control programs. In retrospect today these types of processors are referred to as complex instruction-set computer (CISC) machines. CISC-type processors are typically characterized by long and variable-length instruction words, a wide range of addressing modes, one arithmetic-logic unit, and a single main memory that is used to store both program code and data.

Due to the very complex control flow, the performance of CISC machines was very difficult to improve. It was shown that by decomposing one complex instruction into a number of simple instructions and by allowing parallel execution of these instructions, the performance could be improved significantly. Moreover, data memory accesses use distinct register loads and stores, and data operations have only register operands.

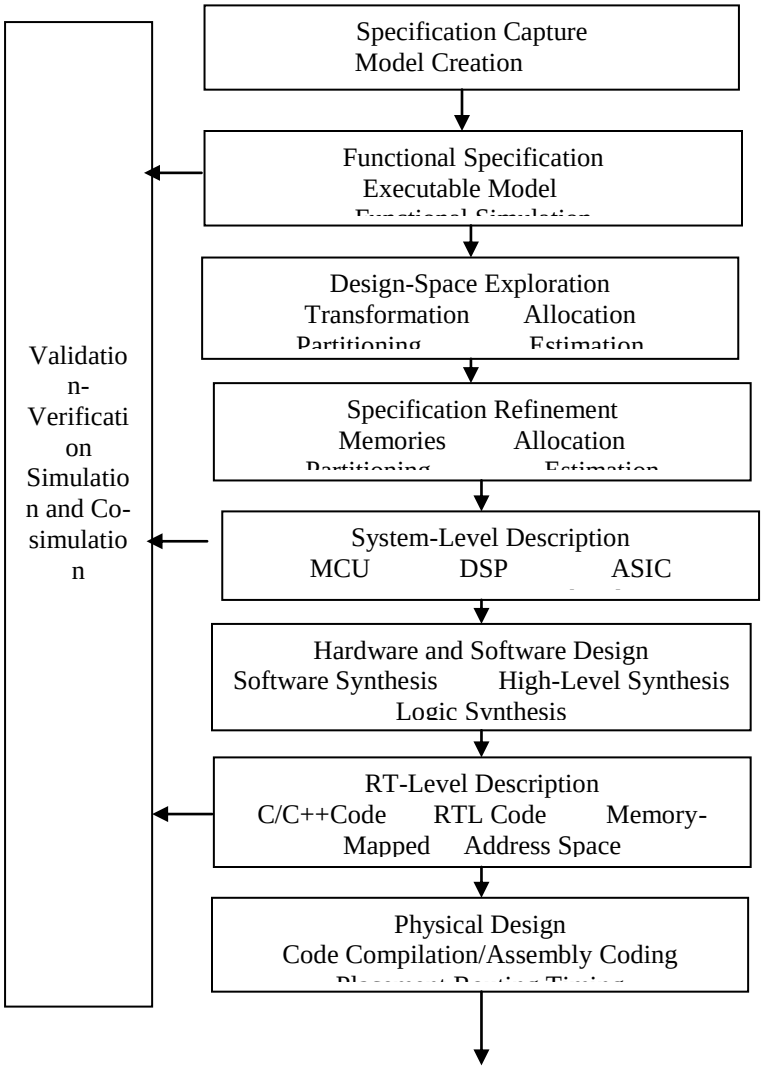

Figure 3.1 System-level design process of embedded systems.

These are the fundamental concepts of the reduced instruction-set computer (RISC) design philosophy. Other key characteristics of a RISC machine are: fixed-length 32-bit instruction word, large general-purpose register files, simplified addressing modes, pipelining, and program code generation with sophisticated software compilers [Bet97, Heu97].

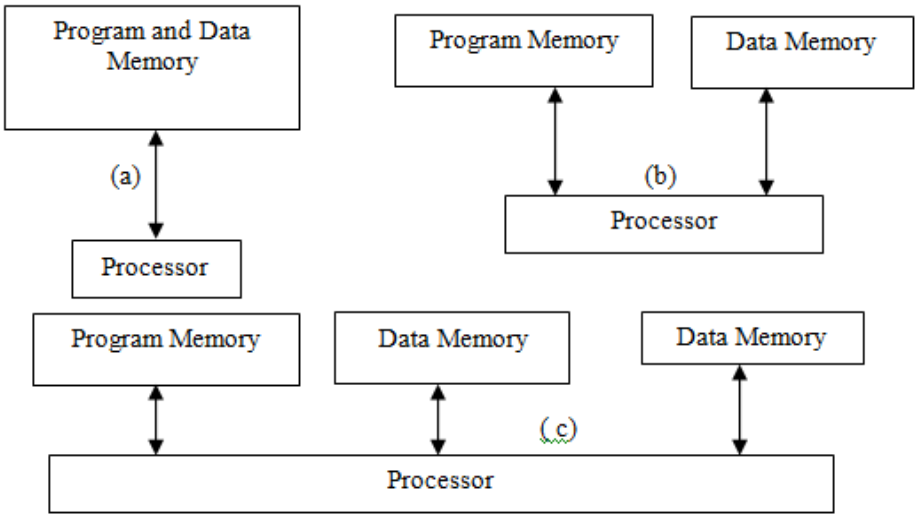

Figure 4.1 Processor memory architectures: a) von Neumann architecture, b) basic Harvard architecture, and c) modified Harvard architecture.

4.1 Memory Organization

All programmable processor models require memory for two fundamental purposes: to store information esteems and direction words constituting executable projects. In this unique circumstance, distinctive memory associations are sorted into three sorts of structures: von Neumann, fundamental Harvard, and altered Harvard.

The arrangement of these memory structures is shown in Fig. 6. Before, a single memory was utilized for the two information and projects. This design is known as von Neumann engineering. In any case, the memory design represents a bottleneck in memory gets to since a direction get requires a different access and, in this way, dependably hinders a potential information memory get to. Thus, the obvious bottleneck was bypassed with Harvard engineering that holds isolate recollections for both program and information. In the essential Harvard design, program and information memory gets to can be influenced all the while and, in this manner, to program execution does not block information memory exchanges. This engineering is right now found in for all intents and purposes all elite chips as discrete store recollections for directions and information [Hen90]. Notwithstanding, the changed Harvard design is the overwhelming memory engineering utilized in DSP processors. The memory design consolidates two information recollections to allow concurrent bring of two operands.

What's more, various varieties have been accounted for in DSP processor frameworks. For instance, utilizing an exceptional DSP direction in a single guideline rehash circle, a third operand can be brought from the program memory, consequently viably getting a sum of three info operands at any given moment [Tex97a]. Memory designs supporting four parallel information memory exchanges have been accounted for in [Suc98]. Besides, some current DSP processor structures consolidate a supplementary program memory which contains wide macrocodes to acknowledge very parallel guidelines without augmenting the width of the local direction word [Kie98, Suc98].

4.2 Operand Location

Regarding areas of source and goal operands, processors can be separated into two classes: stack store and memory-enlist structures [Hen90, Goo97]. Load-store engineering (then again enlist design) performs information operations utilizing processor enlists as source and goal operands and information memory exchanges are completed with partitioned enlist load and store guidelines. This engineering is one of the key ideas in the RISC processor models, yet it is additionally basic in DSP processors as the source operand loads amid DSP operations are frequently executed in parallel with number juggling operations.

In memory-enlist design (on the other hand memory-memory), input operands are brought from the memory, an information operation is executed, and after that the outcome is composed back either to a memory area or a goal enlist. Conversely with the heap store design, the processor pipeline needs to contain an extra stage for perusing source operands. Besides, another stage is required for memory compose get to if an information memory area can

go about as a goal operand. Memory-enroll design can cause an asset strife in a pipelined processor. Such a contention happens if an area in a memory bank ought to be composed when, in the meantime, a similar memory bank ought to be gotten to peruse an operand. The contention can be evaded utilizing pipeline interlocking in which the compose operation is completed regularly however the execution of the operand bring is deferred.

V. MAIN RESULTS

In this postulation, the improvement of an adaptable DSP processor center engineering has been exhibited. The processor advancement envelops three eras, all sharing the base design layout initially displayed in [P1]. In this publication the main utilitarian units and center parameters for the Gepard processor were depicted. Using a GSM full-rate discourse codec calculation, it was exhibited that it is conceivable to enhance the processor performance by adjusting the center parameters and the highlights of the processor data path.

A nonexclusive ASIC configuration stream for utilization of the DSP processor center was formed in [P2]. In view of the licensable processor center approach, the means in the framework advancement were isolated into assignments did by the center merchant and the DSP framework engineer. In the publication, the GSM full-rate discourse codec application was given a more point by point examination. The exchange off investigation secured four cases beginning with an essential center and ending with an advanced center that has an equipment looping unit, immersion mode, and include with-convey ability. Rather than the essential center, the streamlined center lessened the instruction cycle tally by 43 % and thusly the evaluated control utilization by 37 %. Interestingly, the aggregate kick the bucket zone remained for all intents and purposes the same, 17 mm2, on the grounds that the region increase in the center was remunerated by the decreased program memory measure.

Execution of a MPEG sound decoder for the VS-DSP1 processor was exhibited in [P3]. The decoder programming depended on an orderly approach in which a floating-point C-dialect source code was first changed over to a rendition that precisely copies 16-bit settled point number juggling operations. After this alteration the changed over C-dialect source code filled in as somewhat exact portrayal of the calculation conduct in the DSP processor. The usage additionally delineates the utilization of broadened exactness 16×32-piece MAC operations which were required for certain parts in the decoding calculation. The program code required 2.3 k words and the information memory utilization was 12.4 k words, of which 74 % was utilized for different settled esteemed information esteems. A broad investigation performed on the dynamic conduct of the application code uncovered that a 25 MHz processor clock recurrence was adequate for 192 k bit/s, 44.1 kHz stereo sound streams.

In [P4] a parallel program memory engineering was depicted. The proposed parallel design was examined with a GSM discourse codec and the sound decoder that was displayed in [P3]. The main issues experienced were the instruction cycle punishments related with branching and equipment looping. The outcomes demonstrate that the GSM discourse codec was, actually, very ineffective with the memory engineering. In any case, because of the exceptionally consecutive program code, the MPEG sound decoder could gain a linear accelerate. From the down to earth point of view, memory structures with two or four parallel memory banks appeared to be sensible.

In addition to upgrades to the DSP processor center itself, [P5] exhibited a few themes emphasizing the significance of the advancement condition. During the course of advancement, it had turned out to be certain that an uncovered DSP processor center is very a long way from a reusable, licensable IP segment. The key zone of worry for a DSP framework designer is the infrastructure gave by a DSP processor center merchant. Before committing to a certain processor engineering, potential framework designers should be convinced that they approach all the help important to fulfill the improvement work. This infrastructure contains an extensive variety of issues: programming and equipment improvement instruments, operating frameworks, abnormal state EDA apparatuses, programming and calculation libraries, and broad specialized help. A set up DSP processor center merchant needs to construct all the fitting infrastructure set up, with the goal that framework engineers can quickly profit by this infrastructure.

Moreover, the exploration covers two diverse ways to deal with abnormal state determination of remote communications frameworks. As of now, reenactment situations in light of the dataflow worldview have an increasingly imperative part in determination of complex signal processing frameworks. As exhibited in [P6], these instruments can be misused to quickly outline an executable framework detail using a library of practical models. Afterward, this determination was reused for co-confirmation purposes where two utilitarian models were acknowledged with an execution level portrayal of a multi-useful equipment unit. In spite of the fact that the resulting framework demonstrate was somewhat entangled, the reenactment condition gave superb intends to formulating framework level ideas, for example, operation scheduling. Additionally, the framework recreation with the equipment unit increased the reenactment time by no less than two requests of extent, in this manner distinctly demonstrating exchange offs between reproduction exactness and speed.

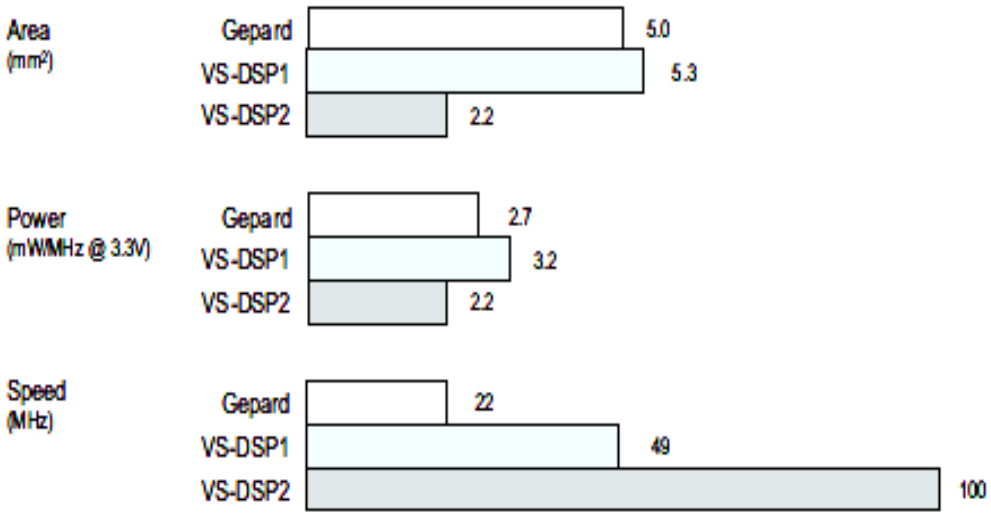

Figure 5.1 Comparison of three DSP processor core versions. For Gepard, the area estimate is based on a gate-level netlist and the power consumption is for a processor that does not contain hardware looping and modulo addressing. [AMS98b, Ofn97, Tak98],[P5].

Publication [P7] exhibited an abnormal state attainability investigation of the framework capacities and different execution viewpoints related with a W-CDMA radio handset. The accentuation was on the beneficiary baseband usage which, rather than the transmitter, has impressively higher unpredictability. In the publication, initial introductions are given of the calculated partitioning into capacities acknowledged as programming executed by an elite DSP processor or as devoted equipment units. As closed, a W-CDMA handset will mainly be equipment based for capacities performed at test and chip rates. In any case, a superior DSP processor (or processors) can give the adaptability and computational power required for the operations at the image rates.

To finish up, the exploration work has fulfilled the objectives of the examination. An adaptable DSP processor design was produced and effectively executed as three center variants. The Gepard processor had a kick the bucket region, most extreme operating velocity, and power utilization of 5 mm2, 22 MHz, and 2.7 mW/MHz at 3.3 V, individually [AMS98b, Ofn97]. The corresponding figures were 5.3 mm2, 49 MHz, and 6 mW/MHz at 4.5 V for the VS-DSP1 processor [Tak98] and 2.2 mm2, 100 MHz, and 0.65 mW/MHz at 1.8 V for the VS-DSP2 processor [P5]. Contrasted with the VS-DSP1, the VS-DSP2 execution exhibits a 100 % increase in performance while the power utilization was lessened by a factor of 9. These attributes were mainly accomplished by the move from 0.6 □ m to 0.35 m CMOS process. Moreover, the VS-DSP2 processor incorporated other profitable functionality, for example, the low-control sit out of gear mode and new instructions [P5]. In Fig. 25 the Gepard, VS-DSP1, and VS-DSP2 processor centers are contrasted with deference with the center territory, control utilization at 3.3 V, and most extreme operating rate. It ought to be noticed that this Gepard processor was a delicate center, yet the VS-DSP processors were actualized as hard centers.

In the future, the VS-DSP processors will be further improved. One of the main considerations is to improve program code density by replacing the relatively wide 32-bit instruction word with a dual-length instruction word. Lastly, a soft-core version of the VS-DSP2 processor is currently under development.

VI. CONCLUSION

This paper has been to a great extent applied technical research rather than basic research. The published results address a wide range of issues which are associated with the specification, design, and implementation of a commercially viable DSP processor architecture. Furthermore, the research work covers specification of wireless communication systems, an application area which clearly benefits the most from the raw computational power, low power consumption, and instruction-set specialization provided by modern DSP processors. In this chapter the main results are summarized, and the thesis is concluded with a discussion on future trends in wireless system design and DSP processors.

VII. REFERENCES

[1] B.-S. Ovadia, W. Gideon, and E. Briman, "Multiple and parallel execution units in digital signal processors," in Proc. Int. Conference on Signal Processing Applications and Technology, Toronto, Canada, Sep. 13-16 1998, pp. 1491–1497.

[2] B.-S. Ovadia and G. Wertheizer, "PalmDSPCore - Dual MAC and parallel modular architecture," in Proc. Int. Conference on Signal Processing Applications and Technology, Orlando, FL, U.S.A., Nov. 1–4 1999.

[3] R. E. Owen and S. Purcell, "An enhanced DSP architecture for the seven multimedia functions: the Mpact 2 media processor," Proc. IEEE Workshop on Signal Processing Systems, pp. 76–85, Nov. 3-5 1997.

[4] D. Parsons, The Mobile Radio Propagation Channel, Pentech Press Publishers, London, United Kingdom, 1992.

[5] T. Parkkinen, Digitaalisen audiodekooderin toteutus, University of Technology, Tampere, Finland, 1999, in Finnish.

[6] A. Peleg and U. Weiser, "MMX technology extension for the Intel architecture," IEEE Micro Magazine, vol. 16, no. 4, pp. 42–50, Aug. 1996.

[7] S. D. Pezaris, "A 40-ns 17-bit by 17-bit array multiplier," IEEE Trans. on Computers, vol. 20, pp. 442–447, Apr. 1971.

[8] Philips Electronics North America Corp., TriMedia TM-110 Data Book, July1999.

[9] J. G. Proakis, Digital Communications, McGraw-Hill Book Co., Singapore, 1995.

[10] S. Purcell, "The impact of Mpact 2," IEEE Micro Magazine, vol. 15, no. 2, pp.

[11] L. R. Rabiner, "Terminology in digital signal processing," IEEE Trans. on Audio and Electroacoustics, vol. 20, no. 1-5, pp. 322–337, Dec. 1972.

[12] J. M. Rabaey, "Reconfigurable processing: The programmable DSP," in Proc. IEEE Int. Conference on Acoustics, Speech, and Signal Processing, Munich, Germany, Apr. 21–24 1997, pp. 275–278.

[13] S. Rathnam and G. Slavenburg, "An architectural overview of the programmable multimedia processor, TM-1," in IEEE Compcon '96, Digest of Papers, Santa Clara, CA, U.S.A., Feb. 25–28 1996, pp. 319–326.

[14] Z. Rozenshein, M. Tarrab, Y. Adelman, A. Mordoh, Y. Salant, U. Dayan, O. Norman, K. L. Kloker, Y. Ronen, J. Gergen, B. Lindsley, P. D'Arcy, and M. Betker, "StarCore 100 - A scalable, compilable, high-performance architecture for DSP applications," in Proc. Int. Conference on Signal Processing Applications and Technology, Orlando, FL, U.S.A., Nov. 1–4 1999.

[15] U. Schmidt and K. Caesar, "Datawave: A single-chip multiprocessor for video applications," IEEE Micro Magazine, vol. 11, no. 3, pp. 22–94, June 1991.

[16] M. Schlett, "Trends in embedded-microprocessor design," IEEE Micro Magazine, vol. 31, no. 8, pp. 44–49, Aug. 1998.