A REAL-TIME SPEECH RECOGNITION ARCHITECTURE FOR A MULTI-CHANNEL INTERACTIVE VOICE RESPONSE SYSTEM

Agus Trihandoyo, Adam Belloum and Kun-Mean Hou

Heudiasyc CNRS URA 817, Université de Technologie de Compiègne
B.P. 649, 60206 Compiègne Cedex, FRANCE
Email: atrihand@hds.univ-compiegne.fr

ABSTRACT

Achieving reliable implementation of a real-time speaker independent speech recognizer through the telephone network is a challenging research problem. Besides requiring the selection of a suitable algorithm, which takes into account both real-time and accuracy constrains, it requires also adequate hardware architecture and optimized software.

This paper presents a dedicated multiprocessor DSP architecture for telecom applications. Based on ADSP-21060 SHARC DSPs, it is the kernel of a digital interactive voice response (IVR) system that is connected to a digital switch through the primary CCITT standard time division multiplexing line of 2.048 Mbps. We attempt to show how a multi-DSP based hardware can be designed for a specific problem in telecommunication, along with the implementation of automatic speech recognition (ASR) to the digital IVR system.

1. INTRODUCTION

Currently, speaker-independent large vocabulary continuous speech recognition is a challenging task for many researchers. On important area is concerned with the telecommunications, where there are a lot of applications of automatic speech recognition (ASR) technology. Voice response systems is one of the most interesting [1]. Most of the currently developed systems are based on the use of the push-button telephone as a data entry console in order to retrieve information or to process voice messages [2]. An alternative approach to provide a data entry to those systems is via ASR. However, for ASR technology to be of value, the voice interface to the machine has to be a natural one in which voice input is a reasonable way of requesting information, and the interface performs reliably (with high accuracy) and robustly for all users and in all environments.

The prospective applications for the IVR architecture partly determined the functional design of the material and level of performance to be achieved. Our implementation is aimed to be a "system initiative" prototype, which means that only the system may direct the interaction. The interaction begins with the system prompting the customer in order to gain enough information to perform a database query. Examples of applications include :

- voice interactive systems to support telephone services like collect calls, third-party billing, or rate inquiries,
- database information retrieval services that might provide schedule or availability information in a limited semantic domain,
- systems for ordering theater or concert tickets, or for making medical or other appointments,
- for manipulating bank accounts or other financial resources.

This contribution extends a recently published work [2] where a robust push-button based IVR is proposed and a multi-DSP board has been developed. The present contribution introduces the use of speech recognition technology. For the first attempt, the recognizer is an isolated, speaker independent speech recognition system that uses Hidden Markov Models (HMM) to recognize some command keywords and digits.

2. SYSTEM OVERVIEW

A number of parallel architectures for speech recognition systems have been proposed ([3], [4] or [5]). To obtain the best performance, algorithms and hardware need to be adapted to each other, thus it is very important and vital for a system envisioned for supporting our implementations in speech recognition algorithms to be reconfigurable and flexible.

The interactive voice response system used in our prototype is completely described in [2] and its main

0-7803-2431-5/95 $4.00

features will be summarized for completeness. Our objective in this project is to provide a set of hardware and software suitable for interactive telephone based applications. It is designed to provide a flexible, real-time implementation of various classes of ASR. The system is composed of the following major blocks, as summarized in figure 1 :

- the acoustical speech feature extraction,
- the recognition processing,
- the dialogue manager, with access to the information database,
- the message generation module, with the text to speech synthesizer,
- the interface to the E1 phone network link.

This general block decomposition simplifies the evolution of different blocks in order to increase the overall system accuracy and response system.

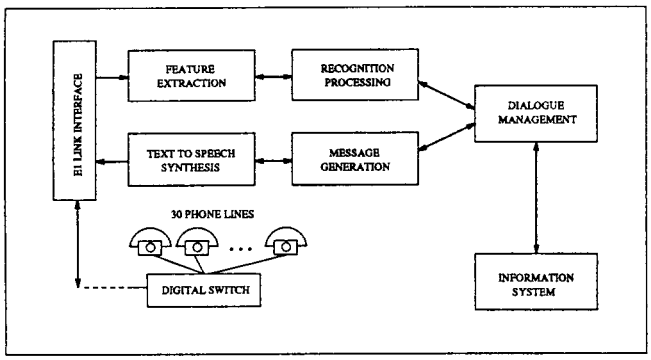

Figure 1: System Block Digram

3. MATERIAL DESIGN

The key concept of our IVR system is the dedicated programmable and reconfigurable hardware for speech and signal processing. It is suitable to use the system as a platform to evaluate and experiment new architectures and algorithms. The system contains four main elements :

- telecom E1 link interface board
- audio module for speech compression and storage
- speech and signal processing board based on ADSP-21060 DSP
- supervisor PC

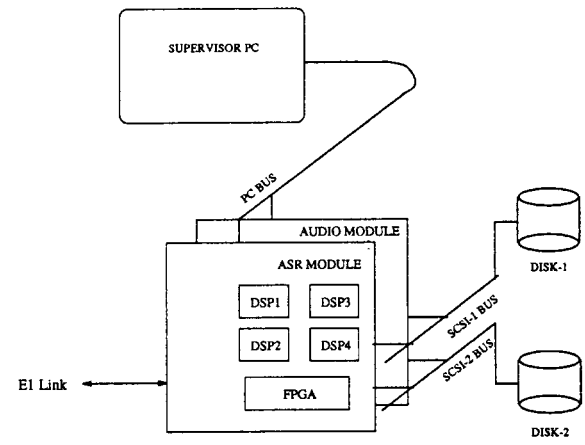

Figure 2: Structure of the IVR system

Figure 2 shows the schematic diagram of flexible IVR system. This type of architecture, implying the connection to a double SCSI bus, allows the number of basic modules to easily be changed by inserting or removing boards. In the same time, if a module contains a defective element it may be bypassed by desselection.

To make the ASR system work properly, it has two possess two main capabilities:

- Computation power. A large number of MIPS have to be executed in many of the basic processes that make up the system.
- Continuous monitoring and control are needed to provide the system the capability of reacting to any new external event as soon as it happens.

The latter requirement should achieve a better user-oriented operation of the system; user can establish the communication without performing any manual operation that might be a source of psychological difficulties and utilization errors.

3.1. Multi-DSP board

The ASR subsystem consists of a PC bus and several ASR cards that plug into the PC bus. Each card has four ADSP-21060 floating point processors, memory, and an FPGA circuit. A reasonable maximum expected system configuration is 4 boards, giving a total of 16 DSPs. The DSPs perform the high-bandwidth computations. The host processor who takes the role of the master module, assigns jobs to the DSPs and controls the DSPs' access to the serial telecommunication links (ST-BUS) by means of a digital switch device. Thus many aspects of the system functional configuration on the DSPs can be modified by software.

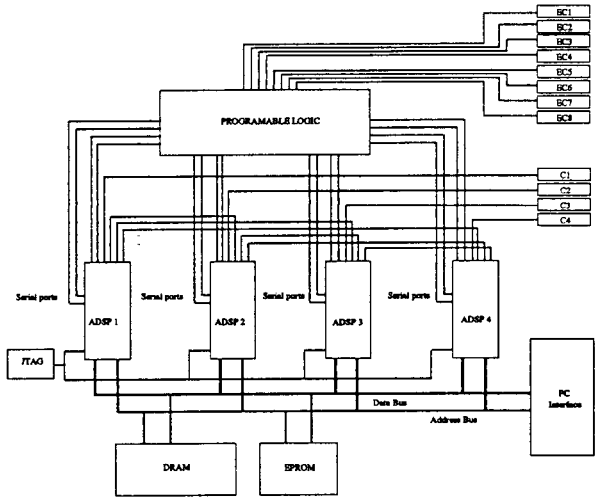

Figure 3: Organization of the architecture

The four ADSP-21060 processors are organized in a full crossbar topology as depicted in figure 3. Two communication links and two serial ports from each processor are connected to an FPGA enabling to have different additional connections between processors and a completely reconfigurable external connections. The two serial ports connections permit to have a redundant feature to improve viability. To provide for user expansion, the communication ports are brought to connectors at the edge of each board. The connectors allow for either attaching a cable or plugging another board parallel to the main board. The JTAG test loop is also routed to all processors. The board is controlled by a host processor operating on the PC bus. Host communication uses the ADSP-21060 parallel I/O by reading and writing areas in the address space that map each unit.

Analog Device ADSP-21060 includes functionality and features that allow the design of multiprocessing DSP systems, such as: distributed arbitration for bus master-ship and multiprocessor access to the internal memory. Arbitration logic is included on-chip to allow the connection to up to six ADSP-21060 and a host processor. The communication link bus is used as interprocessor communication. Master processor changeover incurs only one cycle of overhead. Interprocessor commands are carried out by a vector interrupt. An interesting feature of this new generation of Analog Device DSPs is its architecture that meets the need of high level language programming. An important advantage of using floating-point DSPs is their addressing capabilities. This provides a more efficient solution, since data memory addressing is performed in a single step, without using a page register.

Using floating-point DSPs also allows to shorten the development time cycle, since a substantial amount of algorithms can be easily ported from the simulation to the final DSP if both environments have a similar arithmetic precision, whilst suitable normalizations and scaling are required when porting floating-point simulated algorithms to fixed-point DSPs : these additional computations require additional programming efforts.

The ADSP-21060 offers two blocks of on-chip SRAM (2 Mbits each), which can be configured for different combinations of code and data storage. In addition, the board provides also an external DRAM, organized in words of 48 bits and includes a total of 16 Mwords, extendable 32 Mwords.

Even the DSP supports various booting modes (host boot, EPROM boot and link port boot), the EPROM booting has been chosen. This is the simplest and therefore the safest way. This technique needs only 2 KBytes EPROM for booting.

3.2. Interconnection network board

The architecture consists of a reconfigurable network of ADSP-21060. The reconfigurable routing network can be implemented in various way. Depending on the required connectivity, possible routing alternatives range from switched multiple bus, time division multiplexed buses or fully connected multi-stage routing networks and crossbar switches.

Our main preoccupations for the design of the system are the fault-tolerance, configurability and flexibility. For this purpose, the ADSP-21060 is used because it provides six high speed interprocessor communications links (up to 40 Mbytes/s per communication link). Each processing element is connected locally to its neighbors (three other processors on the same board) in full crossbar by using three communication links. To permit the reconfiguration of the system to adapt to different algorithms to get the optimal performance, an interconnection network is being developed by using programmable device, such as FPGA [6].

4. SOFTWARE

If all processes could be implemented on a single ASR module, that would reduce to a single program on each processor that handles a number of PCM channels. If, on the other hand, even one of the processes cannot be implemented on a single processor (for real-time or for memory constrains or for design convenient), that process will be distributed on the adequate number of processors, with parallel or serial organization that best

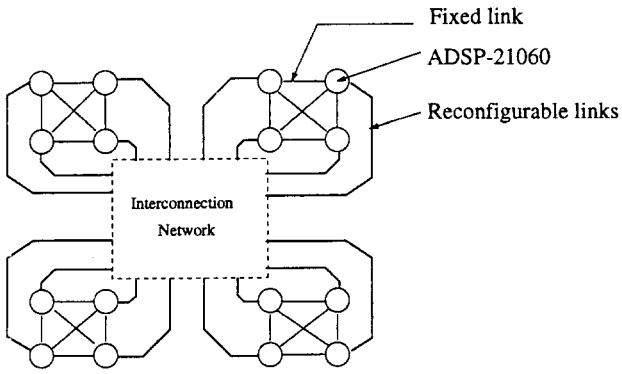

Figure 4: Interconnection network

suit its needs. This is especially true for a class of ASR that is complex enough to require more than one processor to operate in real-time.

In such cases, the process that has to be expanded is replaced by a group of processors, which, in that level, are represented by a supervisor process. It has the responsibility of coordinating the subordinate processes taking care of their synchronization and of the mutual exchange of data.

One of the major problems in such distributed architectures is the system software facilities for developing and debugging applications. Within most of the kernel, interprocess communication and synchronization use semaphores, messages, mailboxes or shared memory. Despite the efficiency of those models, they require that the programmer of the application has a specific knowledge of the hardware architecture. LINDA concept is communication mechanism and a parallel programming model but real-time constraint is not included in the implementation[7]. For this purpose, a real-time multiprocessor kernel using a communication model based on a variation of the LINDA concept: hierarchical LINDA has also been developed [8].

5. CONCLUSION

A multiprocessor architecture for the implementation of speech recognition has been described, that is applicable to accelerating speech recognition problem on IVR system. Thus, this architecture promises to allow real-time execution of the state-of-the-art speech recognition algorithm. As important, however, is the opportunity to develop the next generation of real-time large vocabulary recognizers in a high-level language. The multiprocessor approach permits to better distribute the computation load among processors and to operate in a degraded mode.

The system speed, the accuracy, the capability of handling multi-channel, and the flexibility of the system are the main advantages of the implemented system. The system provides the bridge which is needed between the multiprocessing architectures and the programming environment.

The future work will include analysis of lower level circuit implementations and software optimization.

6. REFERENCES

[1] L. R. Rabiner, "Applications of voice processing to telecommunications," *Proc. IEEE*, vol. 82, no. 2, pp. 199-228, 1994.

[2] A. Trihandoyo et al., "A Multi-DSP Architecture: Application to Detection and Generation of MF/DTMF Signaling for Digital Voice Server," *IEEE International Workshop on ISPACS*, 1994.

[3] A. Ciaramella, D. Clementino and R. Pacifici, "Real Time Speaker Independent Large Vocabulary CDHMM-Based Continuous Telephonic Speech Recognizer," *ICSLP*, 1992.

[4] N. Morgan and H. Bourlard, "The RAP: a Ring Array Processor for Layered Network Calculations," *International Conference on Application Specific Array Processors*, pp. 296-308, 1990.

[5] C. Vicenzi and C. Scagliola., "Multiprocessor Architecture for Real-Time Speech Recognition Systems," *IEEE International Conference on ASSP*, pp. 903-906, 1982.

[6] K. M. Hou et al., "A reconfigurable and Flexible Parallel 3D Vision System for Mobile Robot,", *Workshop on Computer Architectures for Machine Perception*, 1993.

[7] S. Ahuja et al., "Linda and Friends," *Computer*, vol. 19, no. 8, pp. 26-34, 1986.

[8] E. Yao et al., "Real-time Multiprocessor Kernel: Hierarchical LINDA,", *International Conference on Signal Processing Applications and Technology*, 1993

[9] Analog Devices. "ADSP-2106x SHARC Preliminary User's Manual", 1994.