A PDA-based Research Platform for Cochlear Implants

Arthur P. Lobo, Philip C. Loizou, Nasser Kehtarnavaz, Murat Torlak, Hoi Lee, Anu Sharma, Phillip Gilley, Venkat Peddigari and Lakshmish Ramanna

***Abstract*—Currently researchers interested in developing new signal processing algorithms for commercially available cochlear implants must rely on coding these algorithms in low-level assembly language. We propose a Personal Digital Assistant (PDA) based research platform for developing and testing in real-time new signal processing strategies for cochlear implants. Software development can be done either in C or in LabVIEW. The C implementation can be further optimized using Intel's primitive routines. In this paper, we report on the real-time implementation of a 16-channel noise-band vocoder algorithm, which is a similar algorithm used in commercially available implant processors. We further report on EEG recordings on the PDA acquired through a compact-flash data acquisition card.**

I. INTRODUCTION

Cochlear implants (CI) are electronic devices which can restore partial hearing to profoundly deaf individuals [1]. The field of cochlear implants has experienced a considerable growth in the last few years. The number of cochlear implant users grew from 12,000 in 1995 to nearly 100,000 during the past decade.

The growth and progress in cochlear implant technology is largely stimulated and driven by research. Having access to a flexible research platform is critical for the advancement of cochlear implants. While most implant manufacturers provide research speech processors which allow researchers to develop and test new signal processing algorithms, most research labs are unable to use them due to limited technical resources. Of those research processors, only the one made by Cochlear Corporation (in collaboration with CRC/Hearworks) is portable and wearable. The research processors provided by the other manufacturers are constrained for use in the laboratory environment. The main drawback of laboratory research processors is that they do not allow for investigation of novel algorithms after long-term use. Given that the performance of CI users may improve or change within a period of a few months, it is necessary that novel algorithms are evaluated after long-term use of the device. Such evaluations would give us a more realistic assessment of the performance of new algorithms and new experimental methods.

In addition to being portable, the research processor needs to be flexible and easy to use. While the research processor made by Cochlear Corporation is portable, it is not easy to program as it requires a skilled software engineer to implement the algorithms in assembly language. The processor needs to be flexible so that it can be used by both clinicians and researchers without requiring advanced programming skills. Such a processor would also be valuable in animal studies assessing long-term effects of electrical stimulation, and it would therefore bridge the gap and speed up the transition from animal studies to clinical applications.

To achieve the above goals we sought for a research platform which required minimal investment in hardware development. For that reason, we opted for a research platform that is more software driven and that is nearly hardware independent. Changing the software to accommodate new technologies is much easier than replacing existing hardware.

In this paper, we propose the use of Personal Digital Assistants (PDAs) as a research platform for cochlear implants. PDAs nowadays are not only used for managing appointments and contact lists, but are also used as cellphones, GPS devices and for accessing the Internet. We propose a PDA-based signal processor on which the software development can be done in a high level language like C or in a LabVIEW environment. We have implemented a noise band vocoder which is frequently used in hearing research, to evaluate the processing power of the PDA. The processing steps in the noise band vocoder are the same as those involved in a common CI processing strategy called Continuous Interleaved Sampling (CIS) strategy [2, 3]. We report on the real-time performance of a 16-channel noise-band vocoder running on a commercially available PDA. Secondly we report on our experience in recording electroencephalogram (EEG) using a compact-flash data acquisition card plugged into the PDA.

II. NOISE-BAND VOCODER IMPLEMENTATION

Fig. 1 shows a block diagram of the noise-band vocoder. The signal from the microphone is sampled at 22,050 Hz and band pass filtered into 16 channels. Sixth order Butterworth bandpass filters were used in our implementation. The channel center frequencies are logarithmically spaced spanning a bandwidth of 300-5,500

A. Lobo, P. Loizou, N. Kehtarnavaz, M. Torlak, H. Lee, V. Peddigari and Lakshmish Ramanna are with the Department of Electrical Engineering, University of Texas at Dallas, Richardson, TX 75080 USA (phone: 972-883-4650; fax: 972-883-2710; e-mail: arthur.lobo@utdallas.edu; loizou@utdallas.edu).

A. Sharma and P. Gilley are with the Department of Speech Language and Hearing Science, University of Colorado, Boulder, CO 80309 USA (e-mail: anu.sharma@Colorado.EDU)

Hz.

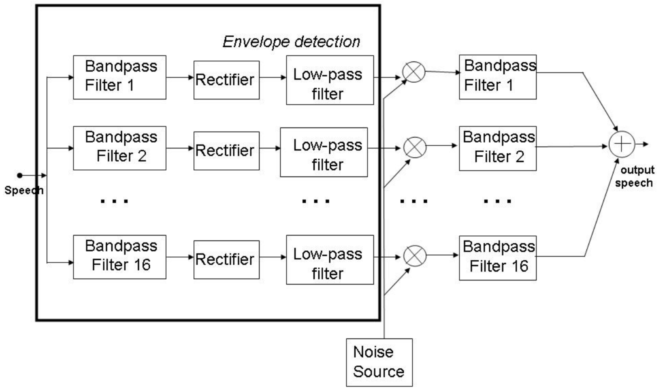

Fig. 1. Block diagram of the 16-channel noise band vocoder. Blocks enclosed in the square show the processing involved in commercial cochlear implant processors.

The output of each band pass filter is full-wave rectified and low pass filtered at 400 Hz to get the smoothed envelope. The smoothed envelope for each channel is modulated by a noise source and filtered through synthesis filters with identical frequency and phase response characteristics as the analysis filters. The synthesized outputs of each channel are then summed to form the output of the noise band vocoder which is played through the PDA speaker.

As shown in Fig. 1, the signal processing involved in commercially available implant processors is only a subset of the overall vocoder processing. The blocks enclosed in the outer square show the processing involved in cochlear implants. In addition, the computed envelopes are compressed, using a log-type function, to the electrical dynamic range and modulated by a train of biphasic pulses rather than a noise source.

The noise-band vocoder was implemented in C language using Microsoft Visual Studio 2005 on a 624-MHz iPAQ hx2790 PDA running Windows Mobile 5.0. The speech signal was acquired and played in 46.4 ms blocks in Full Duplex mode. The Microsoft Wave API was used for receiving and sending samples from/to the codec.

The C implementation was further optimized using the Intel’s Integrated Performance Primitives (IPP) library [4]. The signal processing IPP library in particular was used to implement the noise-band vocoder. The modular nature of the IPP routines allowed for a compact implementation of the noise-band vocoder. An example usage of IPP functions to compute *N* samples of the envelope in channel 1 is shown below:

```
ippsIIR_BiQuadDirect_16s(ch1_in, signal, N, ch1_filter,
    BQ_SECTIONS, DelayLineCH1);        // Bandpass filter
ippsAbs_16s_I(signal,  N);             // Rectification
ippsIIR_BiQuadDirect_16s_I(signal, N, lpf,LPF_BQ_SE
    CTIONS,DelayLineCH1_LPF);          // Low pass filter
```

The above IPP filtering routine (ippsIIR_BiQuadDirect_16s) implements IIR filtering using bi-quad sections (3 sections were used in our implementation). We found that the IPP 16-bit filtering function did not provide sufficiently high precision to match the floating-point output. For one, the intermediate precision between biquad sections was 16 bits, which was found to be inadequate. We therefore implemented the filtering in assembly language (ARM) using 32-bit precision for the intermediate samples resulting in an output that matched the floating-point output very closely.

Time profiling was performed to assess the timing involved in the various signal processing functions operating on a 46.4 ms block. The timing distribution for a 46.4 ms block of data is shown in Table I.

TABLE I
PROFILING OF NOISE BAND VOCODER FUNCTIONS ON PDA

Function	Time (ms)	% processor load
Analysis filtering & envelope detection	6.0	48
Noise modulation	2.3	18
Synthesis filtering	4.3	34
Total	12.6	100

From the above Table we can conclude that the PDA implementation of the 16-channel noise band vocoder is on the order of 3.7 times faster than the real-time limit. As mentioned earlier, in an actual speech-processing strategy for cochlear implants the synthesis filter stage is not present. Consequently, the timing involved to implement say the CIS strategy will be considerably lower. For 16 channels, it was estimated to be approximately 7 times the real-time limit, after including the compression stage.

The noise band vocoder was also implemented in LabVIEW running on the PDA [5]. The initial LabVIEW implementation required a processing time of nearly 2 seconds for a 100ms frame. Three optimization steps were taken in order to bring the processing time below 100ms, thus enabling the system to run in real-time. It is worth emphasizing that these steps are general purpose in that they can be deployed to run other signal processing algorithms in real-time on PDA platforms. The three optimization steps include:

1. Use of Dynamic Link Libraries (DLLs)

LabVIEW’s virtual instruments (VIs) for filtering use floating-point math. There is significant overhead, however, associated with these filtering VIs due to floating point emulation. Thus, we used optimized C code for the bandpass and lowpass filtering sub-blocks and incorporated them as Dynamic Link Libraries (DLLs) within the LabVIEW programming environment. This minimized the amount of floating-point overheads.

2. Better memory allocation practices

For handheld devices such as PDAs, memory management plays an important role in time-critical applications due to the limited memory resources. In general, since dynamic memory allocation requires more processing time than static memory allocation, we avoided using dynamic memory allocation within the real-time loops. Instead, a pre-allocated array was initialized with a constant outside of the loops. Furthermore, we de-allocated the memory that had been allocated for temporary variables within the functions of the LabVIEW DLLs. Finally, we passed a pointer to the address of the pre-allocated array instead of copying the data.

3. Use of fixed-point arithmetic

Since most PDAs lack floating-point arithmetic capabilities, we considered re-writing the code based on fixed-point arithmetic operations. More specifically, the following modifications were made: (1) converted the floating-point parameters to fixed-point Q-format, (2) computed the output in Q-format to avoid overflows, and (3) converted the Q-format output back to higher precision integers. The floating point coefficients for the bandpass and lowpass filters were converted to Q-20 format during the design phase. This format ensured the stability of the filters by placing all the poles within the unit circle. Furthermore, it corresponded to the lowest Q-format that avoided overflows, and cause of any noticeable loss in accuracy.

The real-time LabVIEW implementation of the noise-band vocoder incorporated all the above optimization steps and reduced the time to process a 100ms frame to 77 ms. If we exclude the synthesis stage (Fig. 1), the timing required to process a 100 ms frame is 52 ms. More details regarding the LabVIEW implementation can be found in [6].

III. EEG RECORDINGS ON THE PDA

In this section we report on recordings of electroencephalogram (EEG) signals on the PDA. As a first step, we started collecting EEG data using a basic amplifier setup. This setup included a single, differential amplifier (Grass model P511K) with an analog band-pass filter (1-100 Hz) that had a 12 dB/octave rolloff. This is a standard frequency range for recording EEG and cortical auditory evoked potentials (CAEPs). The amplifier gain was set at x500. The active electrode was placed at the vertex location (CZ) of the subject's scalp, with a reference electrode placed on the right mastoid. The common/ground electrode was placed on the forehead. The electrodes were connected directly to the amplifier. The single-ended signal from the amplifier was sent to a compact flash data acquisition card (Dataq CF2, C-Cubed Ltd, UK), which sampled the incoming signal at 1000 Hz. Data was stored in PDA memory for later analysis. The Dataq-CF2 data acquisition card plugs into the compact flash slot of the PDA (see Figure 2). It has four 24-bit analog input lines, two 12-bit analog output lines and four digital I/O channels. It supports sampling rates up to 40 kHz.

Fig. 2. Data acquisition card (Dataq CF2) used for recording EEG on PDA.

The EEG (and evoked potential) data acquisition set-up is shown in Figure 3. At this point, it was desired to simply test the ability of the A/D card to sample a raw EEG signal on the PDA. We chose to record EEG at rest with eyes closed for a period of one minute. The purpose of this recording was to: 1) test the sampling capacity of the A/D card, 2) to verify portability of the stored data for further analysis, and 3) to analyze the sampled signal in the frequency domain in terms of examining the presence of typical EEG characteristics.

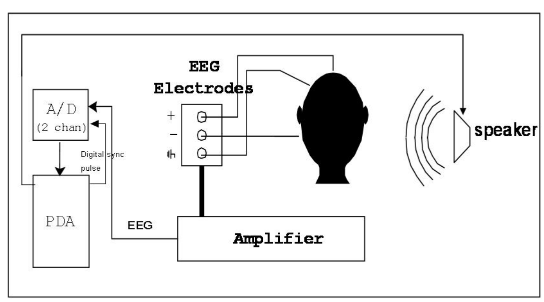

Fig. 3. Setup for recording auditory evoked potentials and EEG on the PDA.

Frequency domain analysis was performed on the raw EEG data using the Fast Fourier transform (FFT). The long-term average magnitude spectrum was computed by averaging the FFT magnitude spectra over 130 epochs of the raw EEG signal. This was done to test for activity that might be attributed to non-physiological signals, as well as to ensure that physiological activity appeared to contain standard low-frequency components. Figure 4 shows the computed long-term magnitude spectrum. . As seen in this figure, most of the activity is between the 10 and 20 Hz frequency range. The 10-Hz activity, called Alpha activity, is the dominant activity in adults during a relaxed state (with eyes closed). The Alpha activity is typically accentuated over the posterior portion of the scalp. The 20-Hz activity, called Beta activity, is the dominant activity in adults during an active or anxious state. The Beta activity is typically accentuated over the frontal portion of the scalp. These results are reasonable in that during a short recording session (i.e., one minute) it is likely that the subject had some transition activity from Beta to Alpha.

In summary, we successfully recorded a valid EEG signal on the PDA. Further tests are needed to examine EEG recordings with different settings of the amplifier.

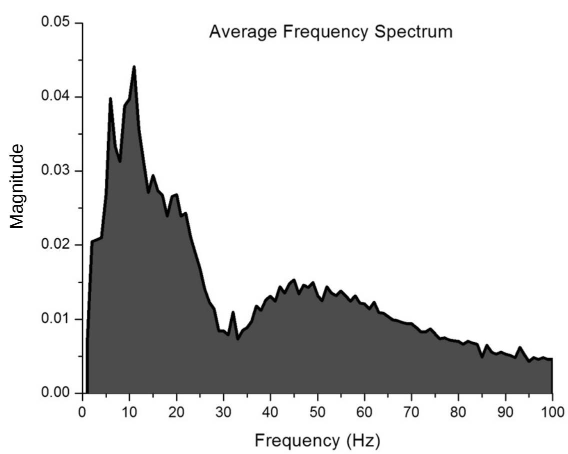

Fig. 4. Average magnitude spectrum for resting EEG recorded on the PDA. This spectrum was computed by averaging the FFT spectra of 130 epochs of 512 samples from the continuous EEG waveform. The standard EEG signature is present with most of the energy in the 0-30 Hz range, and with peak activity in the Alpha band (7-13 Hz).

IV. CONCLUSIONS AND FUTURE WORK

In this paper, we demonstrated that the PDA platform has the computing capability to process speech in real-time via a vocoder algorithm, similar to that used in commercial implant processors. The algorithm was implemented in C and LabVIEW, with the C implementation found to be considerably faster with the use of Intel's IPP libraries. We also demonstrated a successful recording of EEG on the PDA.

Ongoing work of the present NIH contract includes:

1. Establish communication with a commercial implant via the Secure Digital (SD) slot of the PDA. We are currently developing an SDIO card to interface the PDA to the implantable cochlear stimulator. The card will convert from the SDIO 4-bit protocol of the SD Host controller on the PDA to the protocols used by the commercial implantable cochlear stimulators.
2. Extend our EEG work to record acoustically and electrically evoked potentials on the PDA.
3. Build a stimulator circuit, which will interface with the PDA, and provide controlled electrical stimulation suitable for animal studies.
4. Establish wireless communication between the PDA and the implant headpiece. In this approach, sound would be picked up by the microphone contained in the headpiece, and transmitted wirelessly to the PDA. The incoming sound would be processed by the PDA, and then transmitted wirelessly back to the implant headpiece. This approach capitalizes on the fact that most PDAs are already equipped with wireless technology (e.g., Wi-Fi, Bluetooth).

Updates and progress reports for this project can be found at: http://www.utdallas.edu/~loizou/cimplants/PDA/. Code for both C and LabVIEW implementations will be made available to the research community.

ACKNOWLEDGMENT

This research is supported by NIDCD/NIH under contract No. N01-DC-6-0002.

REFERENCES

[1] P. C. Loizou, "Mimicking the human ear," *IEEE Signal Processing Magazine*, vol. 15, no. 5, pp. 101-130. September 1998.
[2] B. Wilson, C. Finley, D. Lawson, R. Wolford, D. Eddington, and W. Rabinowitz, "Better speech recognition with cochlear implants," *Nature,* vol. 352, July 1991, pp. 236-238
[3] P. Loizou, "Speech processing in vocoder-centric cochlear implants," *Cochlear and Brainstem Implants* (ed. Moller, A.), Adv. Otorhinolaryngol. Basel, Karger, vol. 64, pp. 109–143, 2006.
[4] Intel Integrated Performance Primitives on Intel Personal Internet Client Architecture processors Reference Manual, Version 5.0, Intel Corporation., Document Number 278378-025, 2005.
[5] National Instruments, LabVIEW PDA 8.0 PDA Module, http://www.ni.com/labview/
[6] V. Peddigari, N. Kehtarnavaz, and P. Loizou, "Real-time LabVIEW implementation of cochlear implant signal processing on PDA platforms," *Proc. IEEE Intern. Conf. Acoust. Speech, Signal. Processing*, Hawaii, 2007.