An Open and Configurable Embedded System for EMG Pattern Recognition Implementation for Artificial Arms

Jun Liu-*IEEE Member,* Fan Zhang-*IEEE Student Member*, He (Helen) Huang-*IEEE Senior Member*

Abstract— **Pattern recognition (PR) based on electromyographic (EMG) signals has been developed for multifunctional artificial arms for decades. However, assessment of EMG PR control for daily prosthesis use is still limited. One of the major barriers is the lack of a portable and configurable embedded system to implement the EMG PR control. This paper aimed to design an open and configurable embedded system for EMG PR implementation so that researchers can easily modify and optimize the control algorithms upon our designed platform and test the EMG PR control outside of the lab environments. The open platform was built on an open source embedded Linux Operating System running a high-performance Gumstix board. Both the hardware and software system framework were openly designed. The system was highly flexible in terms of number of inputs/outputs and calibration interfaces used. Such flexibility enabled easy integration of our embedded system with different types of commercialized or prototypic artificial arms. Thus far, our system was portable for take-home use. Additionally, compared with previously reported embedded systems for EMG PR implementation, our system demonstrated improved processing efficiency and high system precision. Our long-term goals are (1) to develop a wearable and practical EMG PR-based control for multifunctional artificial arms, and (2) to quantify the benefits of EMG PR-based control over conventional myoelectric prosthesis control in a home setting.**

Index Terms—**EMG pattern recognition, open and configurable design, embedded system, artificial arms**

I. Introduction

Pattern recognition (PR) based on electromyographic (EMG) signals for control of multifunctional artificial arms has been discussed for decades [1-8]. However, evaluation of such control has been mainly carried out in the lab environments. To demonstrate the practical potential of such prosthesis control, evaluation of the device in realistic environments (such as home) to support activities of daily living is needed; however, the related study is very limited.

One of the reasons is the lack of an open, configurable embedded system to implement the EMG PR algorithm for artificial arms. Most real-time implementations of EMG PR have been on the PC platforms [9-11], which are unfortunately not portable. Acquisition and Control Environment (ACE) software has been developed based on MATLAB environment (Mathworks, MA, USA) to implement EMG PR control [9]. This software allows real-time EMG signal acquisition and configuration, visualization, and control of a virtual arm or a real artificial arm. Based on ACE and Musculoskeletal Modeling Software (MSMS), the Johns Hopkins University has developed a Virtual Integration Environment (VIE) supported by the Revolutionizing Prosthetics project program, which is capable of real-time prosthesis control and interaction in a virtual environment [10]. Recently, a stand-alone PC software package, called Control Algorithms for Prosthetics System (CAPS), with a refined user interface and intuitive control configuration has been developed by a research group at the Rehabilitation Institute of Chicago [11].

The number of embedded implementations of EMG PR for artificial arms has been relatively limited. One of the hardware implementations was called Biointerface Board (BIB) [12]. This system was designed to record and decode EMG signals to control a 7 degree-of-freedom artificial arm. The core of the system consisted of an embedded fixed-point digital signal processor (DSP) and an analog-to-digital converter (ADC). The system executed real-time data acquisition, signal pre-processing, online EMG feature extraction, and numerical operation for classification decisions. However, it did not provide on-board PR training function, due to the lack of volatile memory needed for training data storage. The PR parameters still needed to be calculated on the aforementioned PC-based VIE and loaded via Bluetooth link. A recent study reported an embedded EMG PR controller for artificial arms [13]. A high speed Gumstix CPU board (Verdex Pro XM4-BT COM) was utilized as a main board to pre-process EMG signals, extract features, and execute pattern recognition. A Robostix microcomputer with an ADC (3 fixed channels, 8-bits) was connected with the main board to collect EMG signals. In addition, a set of push buttons were designed in the embedded system to provide an on-board PR training interface. In our group, a prototypical embedded system was also designed [14]. The system consisted of two parts: a microcontroller on a RoboVero expansion board for EMG signal sampling and dispatching, and an ARM Cortex-A8 processor on a Gumstix Overo COM for EMG pattern recognition. Exciting system developments have resulted in one commercialized EMG PR controller for artificial arms [15] (Complete Control, Coapt, LLC). Unfortunately, the product is closed for researchers to modify, configure, and optimize.

This paper presented an open and configurable embedded EMG PR system for artificial arms. The system was open so that researchers can easily modify and optimize the control algorithms upon our designed platform; the system was highly flexible in terms of number of inputs/outputs and calibration interfaces used so that other researchers can

This work is partly supported by DOED/NIDRR #H133G120165/ H133G130308, NSF#1406750, NSF#1361549, and HPSFY#Q20111504.

The authors are with the UNC/NCSU Joint Department of Biomedical Engineering, NC State University, Raleigh, NC 27606 USA

Jun Liu is also with Hubei Province Key Laboratory of Intelligent Robot, School of Computer Science and Engineering, Wuhan Institute of Technology, Wuhan 430073, China. Corresponding author: He (Helen) Huang (phone: 919-515-5218; fax: 919-513-3814; e-mail: hhuang11@ncsu.edu).

978-1-4244-7929-0/14/$26.00 ©2014 IEEE

customize their design upon our platform. Our developed embedded system can enable the research community to explore the function and performance of EMG PR in realistic scenarios for prosthesis control, which in turn will speed up the translation of EMG PR for artificial limbs to the amputee users.

II. METHODS

A. Hardware System Architecture

The hardware architecture of the designed embedded system is shown in Fig. 1. The whole system consisted of four components: an ADC subsystem for EMG signal acquisition, a Micro Controller Unit (MCU) core board for fast data processing and pattern recognition, a digital-to-analog converter (DAC) subsystem for prosthesis control signal output, and a multifunctional I/O interface for on-board EMG PR training. The ADC and DAC were connected with MCU via serial peripheral interface (SPI).

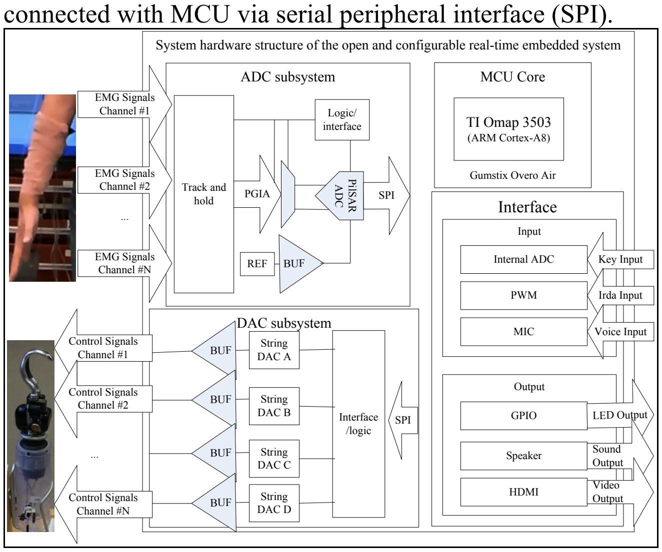

Fig. 1. Hardware system architecture

1) *ADC subsystem*: an evaluation board (EVAL-ADAS3023EDZ, ADI) with a 16-bit ADC chip (ADAS3023, ADI) was used to design the ADC subsystem. This ADC device can simultaneously convert 8 channel analog signals with a sampling rate up to 125K samples per second (SPS), which was sufficient to meet the sampling requirement for EMG signal acquisition. In addition, this device provided configurable options for a variety of system applications, which allowed researchers to choose different numbers of signal inputs and different input voltage ranges.

2) *MCU module*: the embedded system was implemented on a Gumstix Overo Air (ARM Cortex-A8 OMAP3503) board based computer-on-module (COM), which made the entire system open and highly configurable. An expansion TOBI board was selected to provide two SPI interfaces for the ADC and DAC chips and several other I/O interfaces in other modules. The Gumstix Overo Air communicated with the expansion board via two 70-pin connectors. The ARM Cortex-A8 processor on the Overo COM can receive and dispatch data from SPI interfaces, continuously refresh the FIFO ring buffer of the system, and execute EMG pattern recognition and fast online classifier retraining.

3) *DAC subsystem*: an evaluation board (EVAL-AD5664REB, ADI) with a 16-bit buffered voltage-out DAC chip (AD5664, ADI) was utilized to design the DAC subsystem. This subsystem incorporated a power-on reset circuit that ensured the DAC output powers up to 0 V and remained until a valid write took place, which can effectively prevent incorrect operation of artificial arms. In addition, the system contained a per-channel power-down feature that reduced the current consumption of the device to 480nA at 5V and provided software-selectable output loads while in power-down mode. Such a design can allow developers easily configure each individual channel's output loads by software. Also, the low power consumption made the system practical for portable embedded system design. The device provided a versatile 3-wire SPI that operated at clock rates up to 50 MHz, which satisfied the requirements of prosthesis control signal output rate.

4) *Multifunctional I/O interface*: the expansion TOBI board contained six 10-bits ADC interfaces, two Pulse-width modulation (PWM), four PWM/GPIO interfaces, two SPI/GPIO interfaces, ten GPIO mix interfaces, one Line-in for microphone, one Line-out for speaker, one cable networking, one wireless networking, Bluetooth compatible with the Overo Earth module, one HDMI port, one USB slave and one USB host (12 megabit/second). With these multifunctional interfaces, different on-board PR training interfaces (e.g. push button guided training, prosthesis guided training, voice guided training, etc.) can be easily implemented on the embedded system.

B. Software System Architecture

The software architecture of the designed embedded system is shown in Fig. 2. The software was built based on an open source embedded Linux Operating System (OS). The ADC and DAC drivers were programmed in the Industrial Input/ Output (IIO) framework, which is an open driver structure specifically designed for ADC and DAC chips. These two drivers were called by the Input/ Output processing module of the EMG PR system (Fig. 2) in application layer of Linux. Other I/O device drivers (i.e. LED-GPIO, TWL4030, PWM, and SOC audio) were programmed in the driver layer of the Linux kernel source tree that provided an interface for user space to access all other devices. The embedded Linux operating system also provided many interfaces for application program, such as epoll system calls, math library and multi-threading support library. The epoll system calls is an open scalable I/O event notification mechanism that can achieve better performance than the older POSIX select and poll system calls. Therefore, the epoll system calls was used in this system to enable users to access the ADC and DAC drivers' buffers of Linux kernel layer operate on a configurable kernel object in user space. In addition, the math library was utilized in the EMG PR system. The multi-threading support library was used to support the several kernel threads (the EMG PR system and the system control module) and their communication.

The EMG PR system consisted of four main modules: (1) the EMG PR module, which executed the EMG PR algorithm, (2) the calibration module, which processed on-board training function, (3) the I/O processing module, which handled the system buffer with a real-time scheme, and (4) the system EMG control state machine, which received and sent out the command messages. This open event message control system in the software framework was designed to increase the maintainability and extensibility of the system, and handle the input and output devices of the multifunctional interface.

The system control module was designed to handle the communications of input and output devices. It consisted of two main modules: the input processing module and the output processing module. The system control module communicated command messages with the EMG PR system through a two-thread program. The input processing module had three typical input methods, including GPIO-key, Voice input and IrDA remote input. Accordingly, three typical outputs were selected: GPIO-Led, Speaker, and digital screen. The EMG PR system sent command messages to the output processing module to notify users the current state of the machine. It is noteworthy that each module can be easily customized and configured to meet the requirement of researchers.

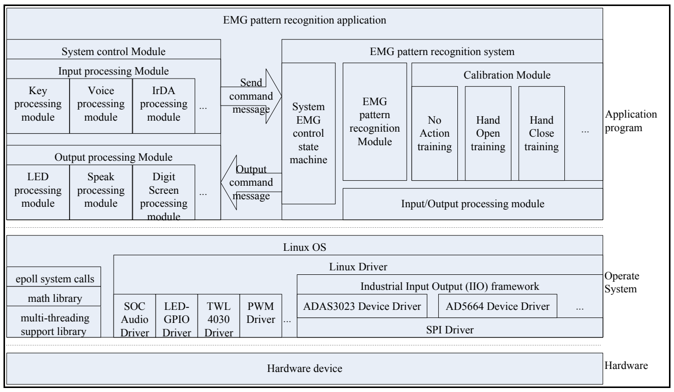

Fig. 2 Software system architecture

C. System Real-time Implementation

To implement the system in real-time, a circular buffer was used in IIO framework to stream the buffered EMG signal data, as shown in Fig.3. This design used a single, fixed-size buffer as if it were connected end-to-end and provided a data structure well suited for buffering data streams. The buffer size was carefully selected to provide adequate space to speed the data sampling process and prevent processing resources consuming. The IIO trigger interface was connected to an IIO device to determine uniform or non-uniform sampling for ADC. A hrtimer trigger, which was a high-resolution Linux kernel timer, was implemented. The User Space Application used the epoll system to read the buffered data, convert the 16-bit data from integer to float, and restore the data in another ring buffer in User Space, as shown in Fig. 3. In addition, a zero-copy mechanism was used to enhance the system performance and save the processing power and memory usage. In such a mechanism, the MCU does not perform the task of copying data from one memory area to another. Therefore, the MCU can move on to other tasks while data copies proceed in parallel in another part of the machine and reduce the number of time-consuming mode switches between user space and kernel space. To implement the zero-copy mechanism, a sliding windowing scheme was used to process buffered data, reuse the old data, maximally utilize computing capacity, and produce a decision stream in a real-time manner.

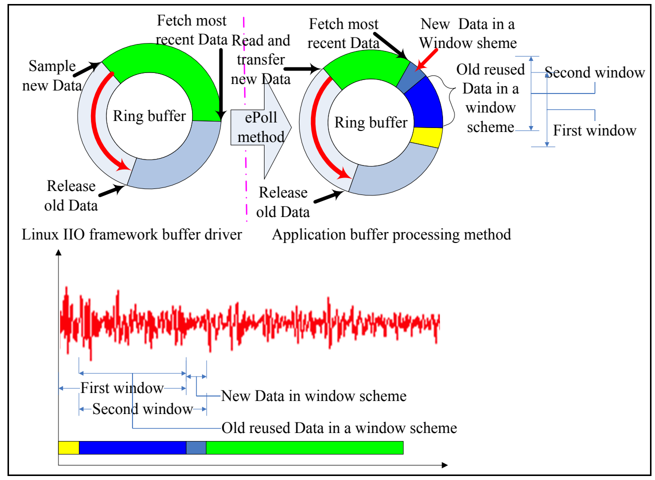

Fig. 3 Circular buffer design and sliding windowing scheme

III. Results & Discussion

Three EMG PR embedded systems were compared: the mobile controller developed in [11], our previously designed system [12], and the system designed in this study. These three systems were designed based on an open-source operating system and open CPU board (Table I). However, only the newly designed embedded system in this study provided open hardware and software frameworks.

Open design of embedded systems holds a great value for the researchers, who are developing EMG PR-based prosthesis control. First, with an open-source operating system, the researchers can concentrate more on development and optimization of EMG PR algorithms rather than the system platform development. Secondly, the open system allows software to be easily developed on different types of CPU, and provides a functionality for cross-platform evaluation. Thirdly, the building of an open cross-compilation environment is free and easy to implement.

TABLE I. Open Design Comparison of Three Embedded Systems

	Our new system	***Our previous system [12]***	***Mobile controller [11]***
Open source operating system	Yes	Yes	Yes
Open CPU board	Yes	Yes	Yes
Open hardware framework	Yes	No	No
Open software framework	Yes	No	No

Table II lists the multifunctional I/O interfaces, compatible prosthesis calibration interfaces, and configurable sensor inputs in the three embedded systems. Our new design provided much richer multifunctional I/O interfaces compared to the other two. More important, these I/O interfaces can be used to make the system compatible with different commercialized prostheses and calibration interfaces. For example, microphone line-in and speaker line-out can be explored to provide voice-guided calibration interface. This feature can be useful for prosthesis users, especially bilateral upper limb amputees. In addition, our system supported up to 8 EMG sensor input channels with 16-bit precision, which can provide the researchers with great flexibility to select the input channels based on their own application needs.

TABLE II. COMPARISON OF CONFIGURABLE DESIGN

	Our new system	***Our previous system* [12]**	***Mobile controller* [11]**
Multifunctional I/O interface	6 input push buttons, IrDA interface, Microphone Line-in, Speaker Line-out, 5 GPIO LED Lights, HDMI port	2 GPIO LED Lights	8 input push buttons, 2 GPIO LED Lights
Compatibility with different prosthesis calibration methods	Button-guided training, Remote control unit guided training, Voice-guided training, prosthesis-guided training	No	Button-guided training
Configurable EMG sensor inputs	Up to 8 flexible channels, 16-bit	4 fixed channels, 12-bit	3 fixed channels, 8-bit

The average processing time of these embedded systems for processing EMG PR algorithms were compared in Table III. The newly designed system was capable of processing all eight 16-bit channels of inputs, computing pattern recognition decisions, and produce four 16-bits outputs by only one microcontroller with low-power consumption. More impressively, less than 1 ms was needed to extract EMG features and classify the movement intent in real-time, which can meet real-time operation constraints[8].

TABLE III. COMPARISON OF REAL-TIME DESIGN

	Our new system	***Our previous system* [12]**	***Mobile controller* [11]**
Microcontroller type	ARM cortex-A8 OMAP3503	1.ARM cortex-A8 OMAP3503 2.ARM cortex-M3	1.PXA270, Gumstix Verdex Pro XM4-BT COM
Average processing time	0.85ms(2307 windows, 5 class, 4 channel)	0.55ms(2307 windows, 3 class, 4 channel)	2ms

Our designed embedded system can provide an effective and efficient platform for other researchers in the field to develop their own EMG PR controller, because our system is open and configurable. Based on this platform, researchers can easily modify and optimize each individual module (e.g. PR algorithm, prosthesis calibration interface, EMG sensor inputs, control outputs, etc.) in the embedded system. This can also significantly reduce the time and cost in the development cycles. So far, our system is portable, but is still in prototyping phase. Our future efforts will focus on development of a wearable and practical EMG PR-based controller for multifunctional upper limb prostheses, and systematic assessment of the benefit of EMG PR-based control over conventional myoelectric prosthesis control in a home setting.

IV. CONCLUSIONS

This paper presented an open and configurable embedded EMG PR system for artificial arms. The system was open so that researchers can easily modify and optimize the control algorithms upon our designed platform. The system was highly flexible in terms of number of inputs/outputs and calibration interfaces used. So far our designed system was portable for take-home use. The outcome of this study may propel the translation of EMG-PR-controlled multifunctional prosthetic arm to upper limb amputees to improves their movement function and quality of life.

ACKNOWLEDGMENT

The authors thank Ming Liu, Aaron Fleming, and Lin Du at North Carolina State University, and Xiaorong Zhang and Diqing Hu at the University of Rhode Island for their suggestions and assistance in this study.

REFERENCES

[1] T. W. Williams, 3rd, "Practical methods for controlling powered upper-extremity prostheses," *Assist Technol,* vol. 2, pp. 3-18, 1990.

[2] S. H. Park and S. P. Lee, "EMG pattern recognition based on artificial intelligence techniques," *IEEE Transactions on Rehabilitation Engineering,* vol. 6, pp. 400-405, 1998.

[3] D. Graupe, J. Salahi, and K. H. Kohn, "Multifunctional Prosthesis and Orthosis Control Via Microcomputer Identification of Temporal Pattern Differences in Single-Site Myoelectric Signals," *Journal of Biomedical Engineering,* vol. 4, pp. 17-22, 1982.

[4] G. Li, A. E. Schultz, and T. A. Kuiken, "Quantifying pattern recognition-based myoelectric control of multifunctional transradial prostheses," *IEEE Trans Neural Syst Rehabil Eng,* vol. 18, pp. 185-92, Apr 2010.

[5] D. Tkach, H. Huang, and T. A. Kuiken, "Study of stability of time-domain features for electromyographic pattern recognition," *J Neuroeng Rehabil,* vol. 7, p. 21, 2010.

[6] H. Huang, P. Zhou, G. Li, and T. A. Kuiken, "An analysis of EMG electrode configuration for targeted muscle reinnervation based neural machine interface," *IEEE Trans Neural Syst Rehabil Eng,* vol. 16, pp. 37-45, Feb 2008.

[7] H. Huang, P. Zhou, G. Li, and T. Kuiken, "Spatial filtering improves EMG classification accuracy following targeted muscle reinnervation," *Ann Biomed Eng,* vol. 37, pp. 1849-57, Sep 2009.

[8] K. Englehart and B. Hudgins, "A robust, real-time control scheme for multifunction myoelectric control," *IEEE Trans Biomed Eng,* vol. 50, pp. 848-54, Jul 2003.

[9] E. Scheme and K. Englehart, "A flexible user interface for rapid prototyping of advanced real-time myoelectric control schemes," 2008.

[10] W. Bishop, R. Armiger, J. Burck, M. Bridges, M. Hauschild, K. Englehart*, et al.*, "A real-time virtual integration environment for the design and development of neural prosthetic systems," in *Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE*, 2008, pp. 615-619.

[11] T. A. Kuiken, G. Li, B. A. Lock, R. D. Lipschutz, L. A. Miller, K. A. Stubblefield*, et al.*, "Targeted muscle reinnervation for real-time myoelectric control of multifunction artificial arms," *Jama,* vol. 301, pp. 619-628, 2009.

[12] F. Tenore, R. S. Armiger, R. Jacob Vogelstein, D. S. Wenstrand, S. D. Harshbarger, and K. Englehart, "An embedded controller for a 7-degree of freedom prosthetic arm," in *Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE*, 2008, pp. 185-188.

[13] T. Hirata, T. Nakamura, R. Kato, S. Morishita, and H. Yokoi, "Development of mobile controller for EMG prosthetic hand with tactile feedback," in *Advanced Intelligent Mechatronics (AIM), 2011 IEEE/ASME International Conference on*, 2011, pp. 110-115.

[14] X. Zhang, H. Huang, and Q. Yang, "Real-time implementation of a self-recovery EMG pattern recognition interface for artificial arms," in *Engineering in Medicine and Biology Society (EMBC), 2013 35th Annual International Conference of the IEEE*, 2013, pp. 5926-5929.

[15] http://www.coaptengineering.com/complete-control.html.