A FPGA Spike-Based Robot Controlled with Neuro-inspired VITE

Fernando Perez-Peña[1], Arturo Morgado-Estevez[1], Alejandro Linares-Barranco[2], Angel Jiménez-Fernández[2], Juan Lopez-Coronado[3], and Jose Luis Muñoz-Lozano[3]

[1] Applied Robotics Research Lab, University of Cadiz, Spain
[2] Robotic and Technology of Computers Lab, University of Seville, Spain
[3] Automation and System Engineering Department,
University Polytechnics of Cartagena, Spain
fernandoperez.pena@uca.es

Abstract. This paper presents a spike-based control system applied to a fixed robotic platform. Our aim is to take a step forward to a future complete spikes processing architecture, from vision to direct motor actuation. This paper covers the processing and actuation layer over an anthropomorphic robot. In this way, the processing layer uses the neuro-inspired VITE algorithm, for reaching a target, based on PFM taking advantage of spike system information: its frequency. Thus, all the blocks of the system are based on spikes. Each layer is implemented within a FPGA board and spikes communication is codified under the AER protocol. The results show an accurate behavior of the robotic platform with 6-bit resolution for a 130º range per joint, and an automatic speed control of the algorithm. Up to 96 motor controllers could be integrated in the same FPGA, allowing the positioning and object grasping by more complex anthropomorphic robots.

Keywords: Spike systems, Motor control, VITE, Address Event Representation, Neuro-inspired, Neuromorphic engineering, Anthropomorphic robots.

1 Introduction

Movement generation is one of the most studied topics at science and engineering. The community known as neuro-engineers has a look into biological movement, which is supposed to have nearly the perfect behavior, with the aim to mimic the process [1]. The nervous system is the driver for movement generation in humans.

From the very beginning it is known that the nervous system uses spikes or action-potentials to carry the information across the organism [2]. The excellent behavior of those systems leads us to mimic them into electronic devices based on interconnected neuron systems; they are called neuromorphic systems. Therefore, the challenge of

* This work was supported by the Spanish grant (with support from the European Regional Development Fund) VULCANO (TEC2009-10639-C04-02) and BIOSENSE (TEC2012-37868-C04-02).

the neuroengineering community is to create architectures of neuromorphic chips with the same properties of human neural system: low power consumption, compact size and scalability.

In our aim of generating intended movements towards a target in a biological, neural way with electronic devices, we have to deal with several problems when implementing the spike processing blocks:

- How to consider the information: In these systems each neuron fires a spike when it reaches a specific threshold in a completely asynchronous way. There are several ways to encode these spikes; for example, the rate coded [3]: when the excitation is low, the spike rate is low and thus the time between spikes is high; however, when the signal excitation increases, the inter-spikes interval time (ISI) decreases, while the spike rate increases. Consequently, the information is codified as the firing rate or frequency.
- The way to implement this architecture: We have implemented it into a FPGA and apparently it is not an asynchronous system but the clock frequency of these digital systems is high enough to allow us to consider an asynchronous behavior for the neurons.
- The other problem is related to the manner of holding communication between different neuromorphic devices. Since neurons communicate in a point-to-point manner and it is possible to integrate several thousands of artificial neurons into the same electronic device (VLSI chip or FPGA), new communication strategies have been adopted, such as the Address-Event-Representation (AER) protocol [4]. AER maps each neuron with a fixed address which is transmitted through the interconnected neuron system.

In this way, with these three considerations, a neuromorphic chip is continuously sending information about its excitation level to the system [5]. Thus, connecting several of them with a parallel AER bus, all the information is available for real time processing. Just by adding chips to the bus, it is possible to enlarge the system. That is one of the most important reasons for using AER, i.e. the scalability allowed by parallel connections. Since each chip has an internal arbiter to access the AER bus [5], real time is limited by the digital clock.

Previous works show that the spikes paradigm in conjunction with AER technology is a suitable join. There are VLSI chips for sensors [5-6], extended systems like the spike-based PID motor controller [7], neuro-inspired robotics [8] and bio-inspired systems for processing, filtering or learning [9-10].

At previous works we can find an approach to a spike processing architecture but not entirely [11] and in [12] a complete one for real-time objects tracking with an AER retina.

Our motivation for the entire research in progress is to succeed in integrating the visual information from an AER retina to a bio-inspired robot by using just spikes for the whole process. That is, to set up a complete neuro-inspired architecture to generate intended movements.

In this paper we have developed the processing layer which generates the trajectory and the actuation layer that applies the commands to reach a target by the motors that mimic the biological muscles (Fig. 1). Both layers use the spikes processing blocks presented in [13]. The processing stage is implemented in a Spartan-6 board with a micro controller plug-in to send configuration parameters to the spike blocks. This layer uses the neuro-inpired VITE (Vector Integration To Endpoint) algorithm developed by Daniel Bullock and Stephen Grossberg [14], although reformulated into the spikes paradigm. The target position, at this moment, is fed manually and the speed of the movement produced can be adjusted by a signal called GO also implemented as spike streams [15].

The second layer is the actuation and it has been implemented in a Spartan-3 board [7]. It has two different parts, the control and the power stage. At the first one we adequate the signal (expanding the spikes) to feed the motors and the power stage transforms the signal to the motors. The motors are controlled with PFM taking advantage of the way we have chosen for codifying the information: the frequency. It operates in an open loop until we integrate the proprioceptive sensors to close the loop. This integration will be carried out with other algorithm also purposed by Daniel Bullock and Stephen Grossberg [14], the Factorization of Length and Tension (FLETE) algorithm.

The robotic platform used is a fixed stereo vision head with two arms with two degrees of freedom for vision sensors holding (Fig. 2).

In section two the first layer is presented: details of VITE algorithm transformed into spikes processing blocks and hardware details in the FPGA. Then in section three we describe the second layer and the advantages of using PFM modulation. In section four a block diagram shows and explains the real hardware used. Then, the characterization of the robotic platform and their limits will show the range of configurable parameters for the first layer. Finally the results of different movements are presented with the main conclusions.

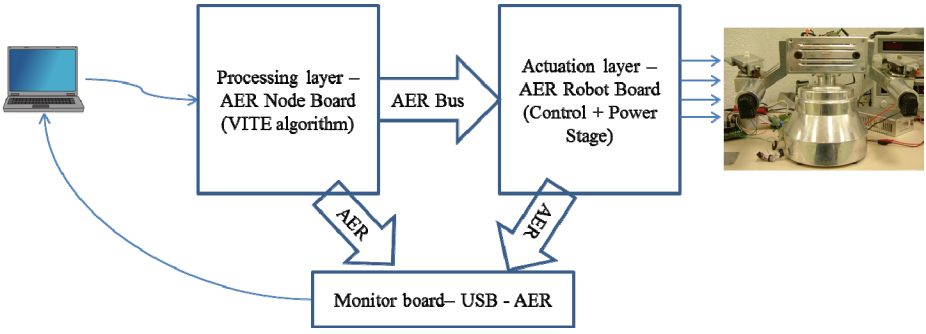

Fig. 1. Layer diagram for the system. The PC sends several configuration parameters to the processing layer across a microcontroller who communicates with the FPGA. Also, the PC sets the target, shoots the movement and receives the spikes to monitor the complete architecture.

2 Processing Layer

This layer is responsible for planning the movement. It receives the target position and generates a spike stream for the actuation layer. The VITE algorithm implemented with spikes ensures a synchronized movement of several joints in order to reach the target position. AER has been used for the communication with previous and next layers.

The hardware used consists of a Xilinx Spartan-6 FXT 1500 FPGA platform developed by RTC lab under the VULCANO project (called AER-node board), that allows high speed serial AER communications over RocketIO transceivers, and adaptation to particular scenarios through plug-in hardware PCBs connected on the top. It includes a plug-in with a USB microcontroller that communicates with the FPGA using SPI (Serial Peripheral Interface). This USB interface has been used for configuring the spike-based blocks of the VITE algorithm.

2.1 Spike-Based VITE Algorithm

The VITE algorithm [14] calculates a non-planned trajectory by computing the difference between the target and the present position. It also introduces the problem to deal with different frames of reference, one for the visual sensor, another one for the central processing (typically the head) and the last one for the actuator. It solves the problem by using the motor frame for all the system.

This algorithm introduced a non-specific control signal called GO. This signal allows separating the spatial pattern characteristics, such as distance and direction, from the energy of the movement. Thus, this manages the movement rate. This signal is introduced as a gate for the movement.

References [16] and [17] justify this algorithm. They show by means of electromyogram how the activity is present at any area of the motor cortex before the muscles initiate the movement. More specifically, the activity is present at the premotor cortex area.

The algorithm will be replicated for each motor present at the robotic platform. Consequently it is very important to analyze the consumption of hardware resources for the algorithm.

2.2 Hardware Resources Consumption

In general, in order to measure the hardware consumption in a FPGA, two points should be considered: the dedicated resources included to build up complex devices such as multipliers and the configurable logic blocks (CLBs) for general purposes.

The presented block does not use any multiplier or memory available. It just needs counters and simple arithmetic operations. Therefore the measurements are focused into the available slices at the FPGA.

The Spartan-6 FPGA present in the AER node board is the XC6SLX150T. It has two slices per CLB, reaching a total of 23,038 slices. We have implemented the system also in a Xilinx Virtex-5 prototyping board (XC5VFX30T FPGA) which has 5120 slices because this board was the first option for the whole architecture.

The VITE algorithm requires around 240 slices (VITE and AER bus interface) and 533 slices with a spikes-monitor block. Therefore, the AER node board is able to implement up to 95 and 43 spike-based VITE in parallel respectively in front of 21 and 9 for the Virtex-5 prototyping board.

The results obtained let us control complex robotic structures with up to 240 degrees of freedom just with one board.

3 Actuation Layer

This layer will adapt the spike-based input signal in order to feed the motors of the robot. It receives the AER output of the processing layer and adapts these addresses to produce the output frequency signal (PFM) for the corresponding motor.

We propose to use PFM to drive the motors becasuse it is intrinsically a spike-based solution almost identical to the solution that animals and humans use in their nervous systems for controlling the muscles.

If we make a comparison between the common used modulation PWM (Pulse Wide Modulation) and the PFM being proposed, we can find some advantages: a typical use of microcontrollers with PWM output generators limits the performance by the hardware timers and its bit resolution. But if PFM is used instead, the system frequency is only limited by the input signal frequency, and the duty-cycle would be limited by the motor driver (optical isolator and H bridges) which implies a low pass filter. Thereby, the use of microcontrollers implies resource sharing, which is not desirable for multi-motor controllers.

Moreover, the use of PFM instead of PWM considerably improves the power consumption when driving the motors because PFM, in average, will produce a lower commutation rate on the power stages. This is because PWM has a constant commutation rate while with PFM the commutation depends on the input of the system, thus it can be adjusted for low power.

Besides, there are more advantages of using PFM instead of PWM for motor control. Resource consumption is half for PFM than PWM when using spike-based controllers, and the power consumption is also much lower for PFM, as expressed in [7].

For the control, right now, it operates in an open loop until we integrate the proprioceptive robot sensor information to close the loop.

The hardware used to implement the actuation layer is a Spartan-3 family FPGA by Xilinx. The board also includes a power stage that consists of optical isolators and H bridges to feed up to four DC motors.

The board is called AER Robot [7].

4 Experimental Section

In this section we present the hardware scenario to develop the tests of the architecture designed, the characteristics that fix the functionality of the system and the techniques for the test carried out.

The boards, robotic platform and power supply are shown in Fig. 2.

To carry out the test, first of all it is necessary to characterize the architecture: the power stage limits the actuation layer, the dc motors used and the relation between the targets fixed and the movement of the robotic platform, that is, the resolution that can reach our system:

- The power stage uses an optical isolator that limits the frequency up to 48.8 KHz and the H bridge can reach this level. Thus this data give us the operation region.
- The DC motors need at least 15.4 μs of pulse width to start-up. They include an encoder with a resolution of 25K pulses/rev.
- Experimental findings with the saturation value (48.8 KHz) fixed as the input for the system, the global resolution can be calculated as:

$$\text{Resolution} = 65 \times 2^{10} / 2^{\text{NBITS}-1} \text{ (degrees / generator step)} \quad (1)$$

Where the parameter NBITS is the number of bits selected to implement the spike generator that supply the target. For example, if we consider 16 bits the resolution will be 2.031 degrees / step.

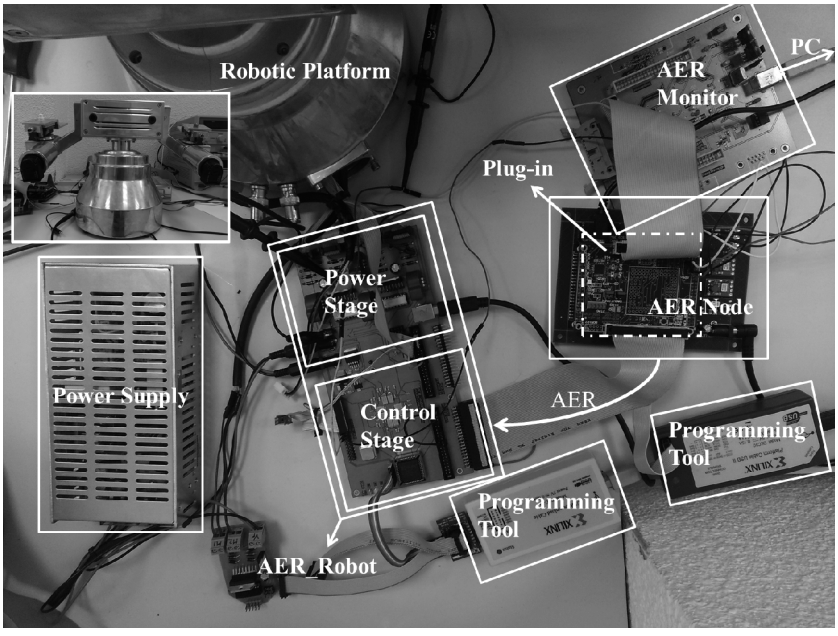

Fig. 2. Scenario with all the elements to carry out the tests

We have carried out several tests changing the speed profile and filling the complete movement range of the robotic platform. As it has been commented, the achieved results are with open loop control. Consequently, the signal sent to the motors is the position instead of the speed. The next section shows the results.

5 Results

The tests are restricted by the monitor board [18] and its maximum spikes firing rate, set to 5Mevps (Mega events per second).

Fig. 3, 4 and 5 show the results achieved. Both VITE behaviors are shown: spike-based on real application (solid lines) and non-spike-based simulation with MATLAB® (dotted lines). The speed of the movement is controlled by a slope profile signal called GO which multiplies the error inside the VITE algorithm [15]. This multiplication provides the speed at which it could provoke instability for the system.

We have plotted the results for one motor. The input signal is shared by them as the target to reach. In the graphs, the red line represents the input generated with the synthetic spikes generator, the purple line shows the speed profile and the green line is the output delivered to the motor. The higher the slope, the faster the fixed target is reached.

The figures show three different slope profiles that confirm the bell shape profiles predicted by [17].

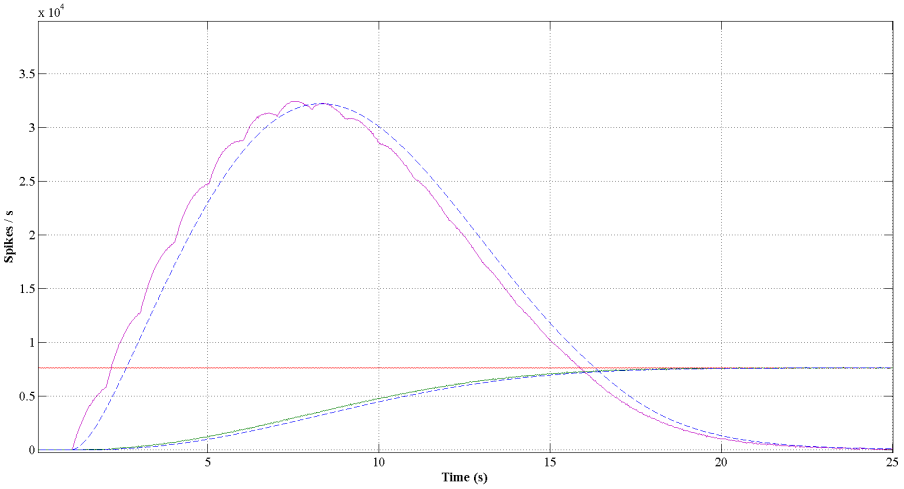

Fig. 3. Performance achieved corresponding to 1 % GO signal slope. The bell shape profile signals represent the speed. The ripple in the spike-base behavior is due to the function that transforms the spikes into a continuous signal. It takes a total of 17 seconds to reach the target if we look through the position.

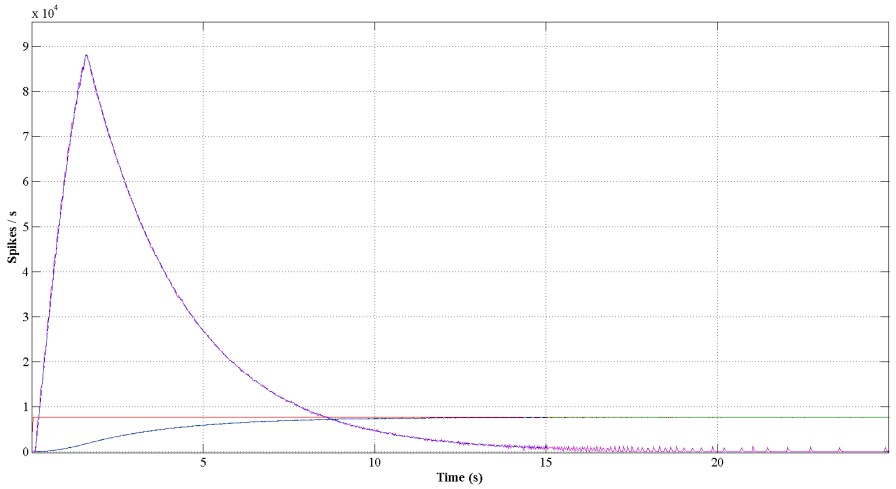

Fig. 4. Performance achieved corresponding to 10 % GO signal slope. The bell shape profile signals represent the speed. The ripple in the spike-base behavior is due to the function that transforms the spikes into a continuous signal. It takes a total of 11 seconds to reach the target if we look through the position.

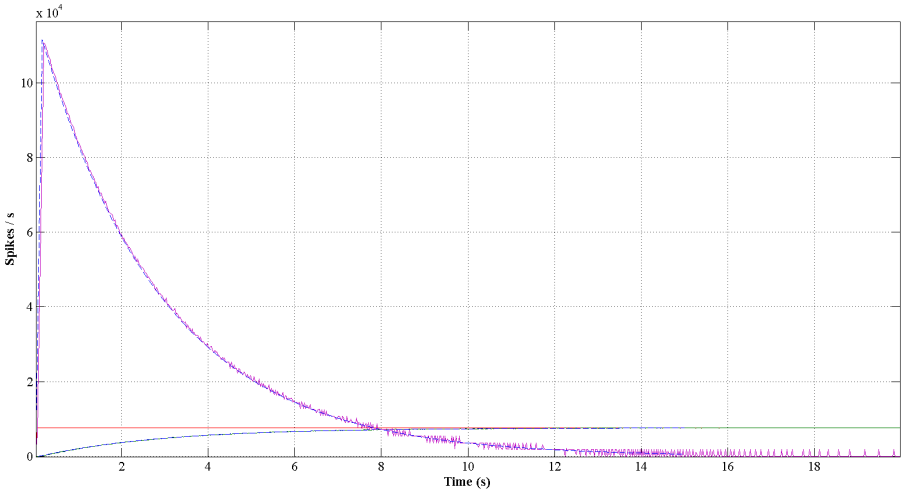

Fig. 5. Performance achieved corresponding to 100 % GO signal slope. The bell shape profile signals represent the speed. With this high slope, the ripple in the spike-base behavior is more significant than in the others. It takes a total of 9 seconds to reach the target if we look through the position.

6 Discussion and Conclusions

A complete spike processing architecture is proposed with excellent results in a fixed robotic platform. The bell shape profiles achieved with the spikes VITE algorithm implemented confirm the studies in [17] about the asymmetric speed profiles for higher speeds. The internal signal "GO" of the VITE algorithm is responsible for putting the movement on speed. Moreover, the signal has a temporal effect. Consequently, if its effect is not enough, the target could not be reached. But the temporal effect avoids instability.

The latency present in all results at the beginning can be interpreted as the previous activity detected at the premotor cortex in humans. In the hardware architecture, this latency is due to the counter that generates the slope profile signal. It can also be understood as the motor priming.

It is possible to reach a high resolution (0.25 degrees by step of the generator with 19 bits) but it always depends on the robot used.

We have fed the motors with the position according to an open loop control and the results fit with the expected behavior. If we change the robotic platform or include some control, it is possible to use the speed profile (available due to an integrator at the final processing blocks chain of the algorithm) for the motors. Nevertheless, the next step is to close the loop, although from a biological point of view, which means using the proprioceptive sensors of length and tension at the joints of the robot. The algorithm FLETE considers both sensors called neurotendinous spindle and muscles spindles and also the gamma neurons. The last step will be to include the feedback of the retina for fine tuning and passive movements updates.

References

1. Bullock, D., Grossberg, S.: Neural dynamics of planned arm movements: Emergent invariants and speed-accuracy properties during trajectory formation. Psychological Review 95, 49–90 (1988)
2. Sherrington, C.E.: Notes and Records of the Royal Society of London 30(1), 45–63 (1975)
3. Linares-Barranco, A., Jimenez-Moreno, G., Linares-Barranco, B., Civit-Balcells, A.: On algorithmic rate-coded AER generation. IEEE Transactions on Neural Networks 17(3), 771–788 (2006)
4. Sivilotti, M.: Wiring Considerations in Analog VLSI Systems with Application to Field-Programmable Networks, Ph.D. Thesis, California Institute of Technology, Pasadena CA (1991)
5. Lichtsteiner, P., Posch, C., Delbruck, T.: A 128 × 128 120 dB 15 μs latency asynchronous temporal contrast vision sensor. IEEE J. Solid-State Circuits 43, 566–576 (2008)
6. Chan, V., Liu, S.C., van Schaik, A.: AER EAR: A matched silicon cochlea pair with address event representation interface. IEEE Trans. Circuits Syst. 54, 48–59 (2007)
7. Jimenez-Fernandez, A., Jimenez-Moreno, G., Linares-Barranco, A., Dominguez-Morales, M., Paz-Vicente, R., Civit-Balcells, A.: A Neuro-Inspired Spike-Based PID Motor Controller for Multi-Motor Robots with Low Cost FPGAs. Sensors 12(4), 3831–3856 (2012)

8. Linares-Barranco, A., Gomez-Rodriguez, F., Jimenez-Fernandez, A., Delbruck, T., Lichtensteiner, P.: Using FPGA for visuo-motor control with a silicon retina and a humanoid robot. In: Proceedings of ISCAS 2007 IEEE International Symposium on Circuits and Systems, New Orleans, LA, USA, pp. 1192–1195 (2007)
9. Barranco, F., Diaz, J., Ros, E., del Pino, B.: Visual system based on artificial retina for motion detection. IEEE Trans. Syst. Man Cybern. Part B: Cybern. 39, 752–762 (2009)
10. Indiveri, G., Chicca, E., Douglas, R.: A VLSI array of low-power spiking neurons and bistable synapses with spike-timing dependent plasticity. IEEE Trans. Neural Netw. 17, 211–221 (2006)
11. Linares-Barranco, A., Paz-Vicente, R., Jimenez, G., Pedreno-Molina, J.L., Molina-Vilaplana, J., Lopez-Coronado, J.: AER neuro-inspired interface to anthropomorphic robotic hand. In: Proceedings of International Joint Conference on Neural Networks, Vancouver, Canada, pp. 1497–1504 (2006)
12. Gómez-Rodríguez, F., Miró-Amarante, L., Rivas, M., Jimenez, G., Diaz-del-Rio, F.: Neuromorphic Real-Time Objects Tracking using Address Event Representation and Silicon Retina. In: Cabestany, J., Rojas, I., Joya, G., et al. (eds.) IWANN 2011, Part I. LNCS, vol. 6691, pp. 133–140. Springer, Heidelberg (2011)
13. Jimenez-Fernandez, A., Domínguez-Morales, M., Cerezuela-Escudero, E., Paz-Vicente, R., Linares-Barranco, A., Jimenez, G.: Simulating building blocks for spikes signals processing. In: Cabestany, J., Rojas, I., Joya, G. (eds.) IWANN 2011, Part II. LNCS, vol. 6692, pp. 548–556. Springer, Heidelberg (2011)
14. Bullock, D., Grossberg, S.: The VITE model: A neural command circuit for generating arm and articulator trajectories. In: Kelso, J.A.S., Mandell, A.J., Shlesinger, M.F. (eds.) Dynamic Patterns in Complex Systems, pp. 305–326. World Scientific Publishers, Singapore (1988)
15. Perez-Peña, F., Morgado-Estevez, A., Linares-Barranco, A., Jimenez-Fernandez, A., Lopez-Coronado, J., Muñoz-Lozano, J.L.: Towards AER VITE: building spike gate signal. In: 19th IEEE International Conference on Electronics, Circuits, and Systems, Seville, Spain, pp. 881–884 (2012)
16. Georgopoulos, A.P.: Neural integration of movement: role of motor cortex in reaching. The FASEB Journal 2(13), 2849–2857 (1988)
17. Nagasaki, H.: Asymmetric velocity and acceleration profiles of human arm movements. Experimental Brain Research 74(2), 319–326 (1989)
18. Berner, R., Delbruck, T., Civit-Balcells, A., et al.: A 5 Meps $100 USB2.0 Address-Event Monitor-Sequencer Interface. In: IEEE International Symposium on Circuits and Systems, ISCAS, New Orleans, LA, pp. 2451–2454 (2007)