

HARDWARE IMPLEMENTATION OF A CONTROLLER BASED ON NEUROBIOLOGICAL ADAPTIVE MODELS OF THE HUMAN MOTOR-CONTROL SYSTEM

J.J. Martínez-Álvarez ⁽¹⁾, A. Guerrero-González ⁽²⁾, J.L. Pedreño-Molina ⁽²⁾, A. Villaescusa-Fernández ⁽²⁾, J. Manuel Ferrandez ⁽²⁾ and J. López-Coronado ⁽²⁾

Jjavier.martinez@upct.es ⁽¹⁾, Jm.Ferrandez@upct.es ⁽²⁾

Dept. of Computer Architecture, Technical University of Cartagena
Campus Muralla del Mar, 30.202 Cartagena, Murcia, SPAIN

Antonio.Guerrero@upct.es ⁽²⁾, Juan.Pmolina@upct.es ⁽²⁾, Antonio.Villaescusa@upct.es ⁽²⁾,
Jl.Coronado@upct.es ⁽²⁾

Dept. of Systems Engineering and Automatic, Technical University of Cartagena
Campus Muralla del Mar, 30.202 Cartagena, Murcia, SPAIN

Abstract.-In this work, a neural structure has been implemented into a device based on the new tendencies on hardware integration, for motor-control in multisensorial anthropomorphic robotic systems. This implementation gives a solution to the problem of physic integration of biologically inspired control hierarchies in a robotic head-arm installation for robotic reaching tasks. The complete architecture has been implemented on a electronic board connected to a PC computer through a PCI interface. The hardware structure consists on two block, one for the working phase of the system, and other for the learning and supervision phase of the system. These two blocks have been implemented with different technologies based on DSP processors and FPGAs. The algorithms implemented on DSPs have the function of updating the neural network on the FPGA, supervise the well-working of the algorithm implemented on FPGA and introduce corrections when the neural network produces results with little errors. The neural network has been implemented on FPGA and implements the spatial - motor transformations of the robotic platform. It is programmed and updated by the supervisor implemented on the DSP processor.

Keywords: Neurocontroller, robotic, neural devices

The coordination of all senso-motor control processes of the architecture is established by means of the priorities of behaviours in BSM module. It is a dedicated to the selection and commutation of the different learning maps implemented into a FPGA device rely on the importance of sensorial (visual) or proprioceptive (joint positions) stimuli in each instant of the process.

Some important advantages are presented by this hardware architecture. First, the reduction of system complexity by means of dedicated devices for transformation spatial-motor from relative visual information between the target and the arm current position, and the position increments in robot arm joints. Second, the decrease of response time for computing the desired joint positions in comparison with the time to compute and transmit that information to joints controller. Third, the possibility to export this neural hardware device to other applications such as tactile-motor transformation in stable grasping tasks [3], visual-motor coordination for head controller in objects tracking operations or prediction for delay compensation in trajectories tracking for reaching tasks with objects in movement. Finally, the physical implementation of a neurobiological model [2] inspired on human system for spatial-motor transformations and behaviour selection.

I. INTRODUCTION

One of the aims in robotic reaching and grasping is to build a system that emulate the behaviour hierarchies based on neurobiological inspired systems. In this sense two are the performance modes in the human control system: the fast sensorial-motor neural projections and the supervision, neural updating and learning processes. So, in this work the advantages and possibilities that the semi-custom devices (FPGAs) offers, have been employed. These hardware devices allow to carry out implementations of conventional neural networks over small surfaces areas, such as the system acquires a certain degree of autonomy, which reduces the global complexity of the system.

The proposed system allows to build neural modules in hardware devices for learning of visual-motor transformation based on VAM (*Vector Associative Maps*) [1] models. The neural weights of the network are stores in FPGA (*Field-Programmable Gate Arrays*) devices and continually updated by the BSM (*Behaviour Selection Module*) implemented in DSP (*Device Signal Processor*).

II. HARDWARE STRUCTURE

The proposed structure has been divided into four main blocks, as figure 1 shows. The first one is formed by the platform "Master & Communication" (M&C) FPGA and the PCI bus. Visual information which is supply by the system, supposes a upper level control, by means of a PC, and the continuous interchange of information between the target and the computer (in both directions).

The employ and management of data flow in this level will be the main task of this communication platform. Indeed, this block synchronizes the data between the different devices of the target.

The second block is formed by DSP and the program memory. Its basic task is to execute the learning and supervision algorithms, which update the neural transformation maps. The executed program can be modified and loaded in any instant through the M&C FPGA. This process will be always started and supervised by the PC. When the learning process finishes the DSP will load through the Transceiver, the content of

the updated net map (512 cells) to the memory of the hardware FPGA neural network (NN FPGA).

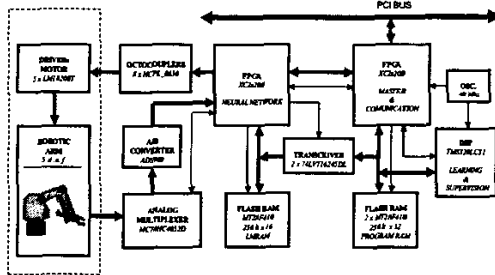


Fig. 1. Complete structure of hardware

The third block, named WNNB (*Working Neural Network Block*) is formed by the NN FPGA and the Learning RAM Map. Their functions will be the execution in real time of the NN algorithm and the implementation of the five neurobiological joint controllers, by means of a GO modulation function.

The last block is formed by the in/out interface with the robotic arm. This block is composed by an optical isolating level, for the power level of the joints motors, and an A/D conversion and multiplexing level. It extracts the information of the absolute position of each of the five motors and is supplied to the interior of the NN FPGA. Main characteristics of the target, are:

- Capability of processing and neural control of five robotics joints.
- Parallel processing of the five joints with times in control loop minor than 10 μ sec.
- Possibility of exporting this hardware to others applications, which require spatial transformations by means of neural maps.
- Distributed architecture with three main tasks: to establish the communications with the PC and the robotic arm; the learning and supervision of the NN and finally, the performance of the WNNB.
- Two FPGAs (Spartan II), a processor DSP (in floating mode) and a PCI bus interface.
- All the devices employ Surface Mount Devices (SMD) technology, LVT compatible.

III. DESCRIPTION OF THE ARCHITECTURE

In this section the aspects considered more important related with the internal architectures of the main devices of the electronic system will be treated. These internal structures are strongly related with the global requirements of the system structure.

M&C FPGA

The internal architecture developed in the device M&C FPGA (XC2S200) with the PCI bus, constitutes the main part of the communication block which was described in the other section. The architecture of this FPGA is constituted on one hand by the interface elements with a PCI Master/Slave configuration, this interface verifies the PCI V2.1 specifications. This interface contains five main blocks: the configuration

space PCI block, the interface I/O block, the master sequence machine, the target sequence machine and the parity generator-checker, which manage and configure the bus whose function is to bridge with the rest of devices which constitute the electronic board.

Other important part of the architecture of the FPGA is constituted by the load module of the program of the DSP with a new modified code. Finally, other block is the Arbitrer State Machine which control and manage the priority between communications of the different devices which constituted the board, as for example the load of the update neural map by the DSP processor to the LMRAM memory. Complete architecture scheme of this FPGA is depicted in figure 2.

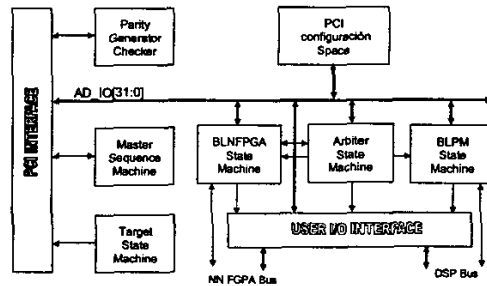


Fig 2. Master/Slave PCI Interface

NN FPGA.

In figure 3 a part of the internal architecture of the NN FPGA (XC2S200) has been represented. In concrete, this part has as objective to support the execution of the NN once the learning phase of the net as the weight update process has been concluded. This operation is carried out by the DSP processor. The working phase of this module one the LMRAM updating process has concluded is the following:

1. The PC from the M&C FPGA transfers its contain to the *Buffer Position Target*; The position of the objective (P_T), the velocity of the trajectory (V_T) and the objective cell ($Cell_T$).
2. Using this information, the state machine RCB (RAM Cell Boot) loads in the *Buffer RAM Cell* the data corresponding to the objective cell, that is, the 15 weights associated to the objective cell (W_{Cij}), the 3 spatial co-ordinates of the center of the cell ($P_{C,x,y,z}$) and the 5 motor co-ordinates ($\theta_1, \theta_2, \theta_3, \theta_4, \theta_5$), also referred to the center of the cell.
3. The following phase consists in process the data obtained in the before phases and calculate the increments in motor coordinates under the mathematical law of the neural algorithm (in this case the VAN algorithm) of the equation (1). This phase will be supervised by the *NN State Machine*.
4. Finally, once the 3 phase has been concluded and once the results has been obtained, these results are loaded to the biologically inspired control module of the joints (GO Function).

$$\Delta\theta_i = \sum_{j=1}^3 W_{ij} \Delta x_j \quad (1)$$

$$(i=1, \dots, 5)$$

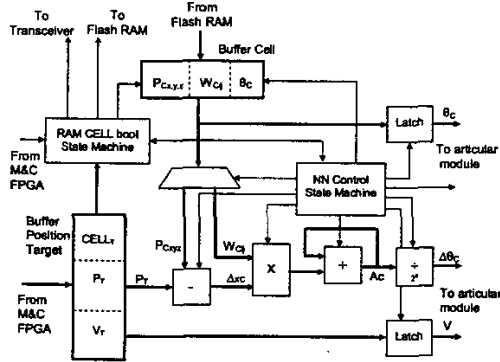


Fig 3. Neural Network FPGA Architecture

The joint control module constitutes the rest of the internal architecture of *NN FPGA*. This part has as function, to establish in a parallel way, the control loop of the five motors of the robot arm, according to the control algorithm (Function "GO") expressed by the equation (2) and depicted by figure 4

$$\theta_T = \theta_0 + \frac{(\Delta\theta_r)}{1 + e^{-t(v)}} \quad (2)$$

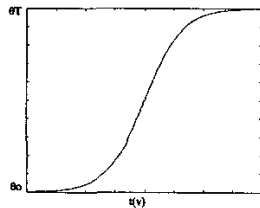


Fig 4. Ideal output of joint controller

This joint controller, represented in motor coordinates, is corresponded essentially with the sigmoid function lagged θ_0 angle with respect to a reference system or initial value of the curve and where, θ_T is the absolute position of the objective (desired position), θ_0 is the absolute position of the origin (initial position of the trajectory), $\Delta\theta_r$ ($\theta_T - \theta_0$) is the relative increment which is produced to reach the objective, and $t(v)$ is a time function which depends of V_T and which is used to establish the rate of the curve and therefore the velocity with the objective is reached. In the figure 5, it is shown the schematic which represents this part of the architecture, and over this it is possible to see as the joint controller is executed inside the *NN FPGA*.

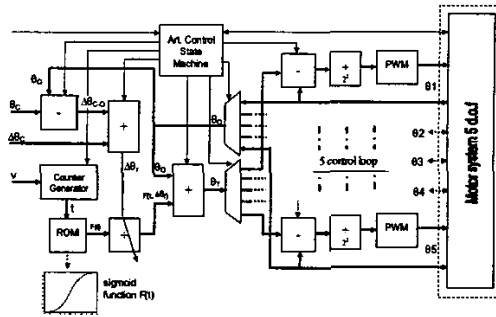


Fig 5. Articular module in NN FPGA Architecture

The *Articular Control State Machine* block synchronise all the procedure and permit to calculate in real time the evolution (sigmoid) of the temporal consigns in each one of the five proportional controllers whose constitute the system and whose main characterise is a sample rate of 10 us.

IV. SUPERVISION AND LEARNING PROCESS

One of the main function that is developed by the DSP module, is to achieve the learning and supervision process for sensorial-motor coordination tasks. Learning process updates the value of neuron weights into the neural map, by means of the LMS (*Least Mean Squared*) adaptive algorithm. Supervision process is included into a upper hierarchy level (BSM).

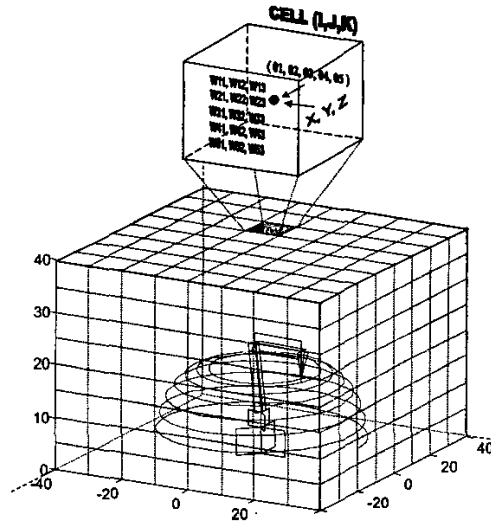


Fig. 6. Neural structure of learning process

For first process, see figure 6, the workspace for robot movement has been divided into 512 ($8 \times 8 \times 8$) cells. Each of one stores 15 neuron weights W_{ij} (3×5). Each three dimensional spatial position of the centre of cell P_{ijk} is projected by means of VAM algorithm over the five motor positions g_i .