Direct Interfaces for Smart Skins based on FPGAs

Óscar Oballe-Peinado, Julián Castellanos-Ramos, José A. Hidalgo-López, and Fernando Vidal-Verdú
Dept. of Electronics, University of Malaga, Málaga, SPAIN, vidal@ctima.uma.es

ABSTRACT

Many artificial skins for robotics are based on piezoresistive films that cover an array of electrodes. Local preprocessing is a must in these systems to reduce errors and interferences and cope with the large amount of data provided by the sensor. This paper presents circuitry based on an FPGA to implement the interface to the artificial skin. The approach consists of a direct connection. The analog to digital conversion procedure is simple. It consists of measuring the discharging time of a capacitor through the resistance we want to read. This first proposed approach needs isolated tactels, so the raw sensor has to be fabricated in this way. If the tactile array is large, the strategy is not feasible. For instance, up to 288 pins are required to implement the interface with an array of 16x16 tactels. The proposal of this work for this case is to replace passive integrators by active ones. The result is a circuitry that allows the cancellation of interferences due to parasitic resistors and the sharing of the addressing tracks. Moreover, the FPGA allows the processing of data from the tactile sensor at a very high rate. This is because the high number of I/O pins of the device allows the conversion of many channels (in our case one per column) in parallel. The internal processing of the tactile image can also be done in parallel. This means we could be able to respond to very high demanding tasks in terms of dynamic requirements, like slippage detection. This also means we can run complex algorithms at real time, so a smart, programmable and powerful sensor is obtained.

Keywords: tactile sensors, local pre-processing, FPGA

1. INTRODUCTION

Although the work on tactile sensors or artificial skins is not new, the most interesting results are still to come [1][2]. The interest and research in this field have increased a lot in the last years. This is because more advanced technologies are able to provide complex and smart sensors. These sensors can face applications in unpredictable or unstructured environments which cannot be coped with simpler ones. There are several applications that fit this description, for instance in medicine, robotics of food processing industry, virtual reality and telepresence, or security [1].

Most tactile sensors are made of sheets of piezoresistive materials. The sheet covers an array of electrodes and we get an array of force sensing resistors or tactels. However, if a continuous sheet is used, parasitic resistors are present between tactels in the array (one of these resistors is depicted at Fig.1(a) to illustrate this). Moreover, even if these parasitic resistors do not exist because they have been cancelled in the fabrication process there are crosstalk if the tactels of the sensor are arranged in rows and columns to be addressed. The reason is the addressing tracks are shared by many tactels and they allow resistive paths that cause crosstalk. A few strategies to reduce the interferences due to these resistive paths have been proposed. The best one is shown at Fig.1(a) [3]. Its goal consists in having the same voltage at both sides of parasitic resistors, so they are virtually short-circuited.

In addition, large tactile sensors and real-time operation are often required. Some preprocessing at the sensory plane results in a reduction of the amount of information to be transmitted to the central decision unit [4]-[8]. Moreover, detection and processing circuitry should be located near the sensor to avoid problems caused by large distance wiring. It must also have a low number of integrated circuits and I/O connections. This reduces the number of cables and allows it to be housed in hands and grippers.

A quite direct strategy to reduce the area and power consumption consists of implementing an Application Specific Integrated Circuit (ASIC) that could be in charge of reducing errors, compensation of interferences and analog-to-digital conversion (see Fig.1(b))[7]. It is also the best choice in terms of area and power efficiency. Moreover, slippage detection in manipulative tasks with hands or grippers has to be done in the range of 2-4ms. This means the whole array of tactels has to be processed in this time (analog-to-digital conversion plus detection algorithm). The high dynamic performance of

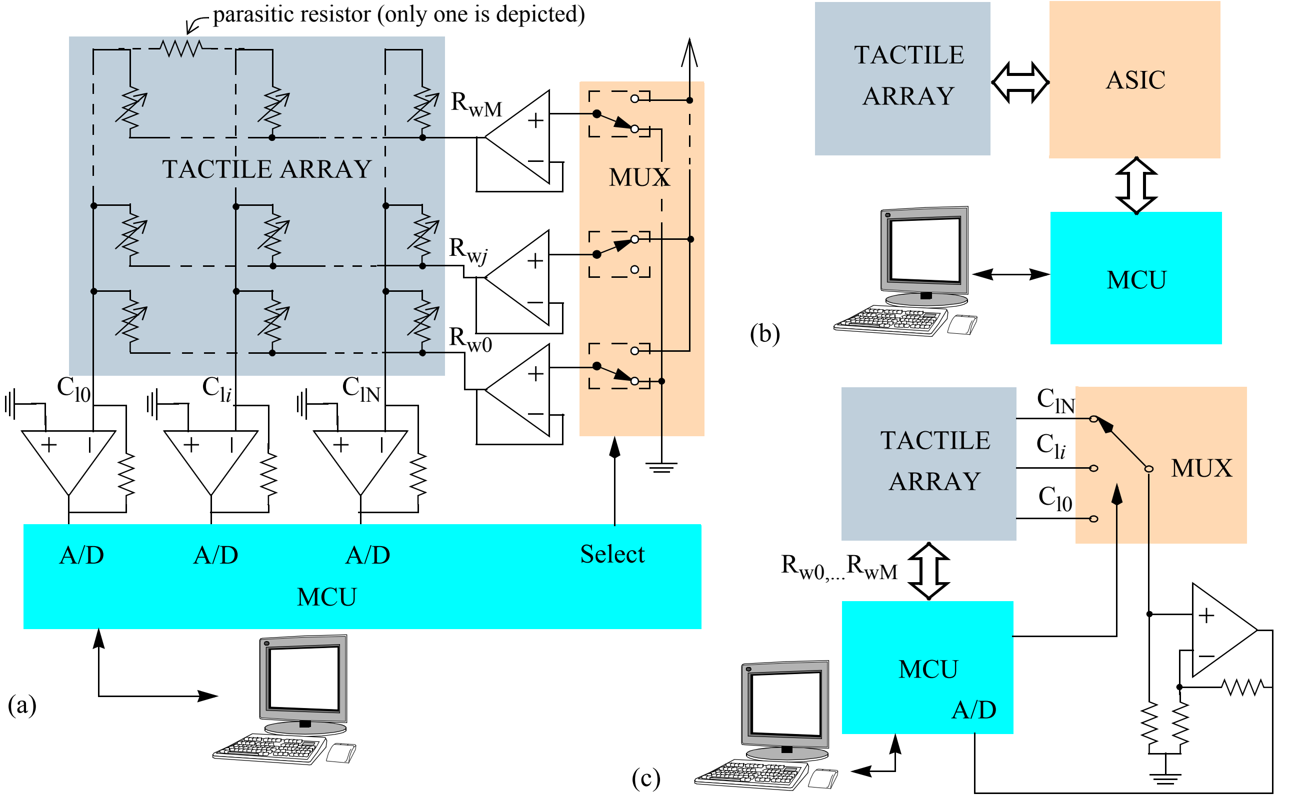

Fig.1. Tactile sensor interfaces: Common approach to cancel interferences from parasitic resistors (a), interface with an ASIC (b), interface with a microcontroller and sequential reading (c).

ASICs allows to detect slippage [8] (another strategy consists of the use of a more complex raw sensor that has a piezoelectric layer besides of the resistive one [4]). Unfortunately, ASICs are quite rigid and their programmability is low, so their functionality cannot be updated much once they are fabricated.

Other implementations are based on microcontrollers to implement the local circuitry (see Fig.1(c)) [4]-[6]. This strategy usually requires a higher number of devices in the PCB board. This means a large area because the room they consume as well as due to a more complex wiring. Furthermore, tactels are read sequentially in Fig.1(c), hence the dynamic performance is poor and slip detection is not feasible with a simple raw piezoresistive sensor. However, this approach allows the design to be updated, the tasks to be carried out by the microcontroller can be changed just programming it again.

The use of a Field Programmable Gate Array (FPGA) is between both previous strategies. They are flexible devices because they can be programmed, and at the same time they have a high dynamic performance due to the parallel processing they allow. The main advantage of this strategy is the possibility of performing a quite complex preprocessing at real time. As the system becomes more an more complex, many tactile sensors are used, for instance in fingers and palms, so the huge amount of data provided by these senors should be preprocessed for the main controller to be able to manage it in real time. On the other hand, FPGAs do not have commonly analog-to-digital converters. Therefore, the use of external converters could increase the complexity and cost of the circuitry. This paper proposes an implementation that does not need such external converters. It is based on the direct connection of sensors to microcontrollers [9]. Since the FPGAs have many I/O pins, they allow a very direct connection between the tactile sensor and the device. The obtained smart sensor is compact and powerful in terms of capability to real time processing. Passive intergrators are common to implement direct connection of sensors. In the case of tactile sensors, the tactels have to be addressed with separate tracks to avoid interferences due to parasitic resistive paths. This is feasible for low-medium size sensors. However, for sensors with a high number of tactels this approach is not valid. We propose the use of active integrators in this case because they allow the implementation of a common strategy to cancel the interferences.

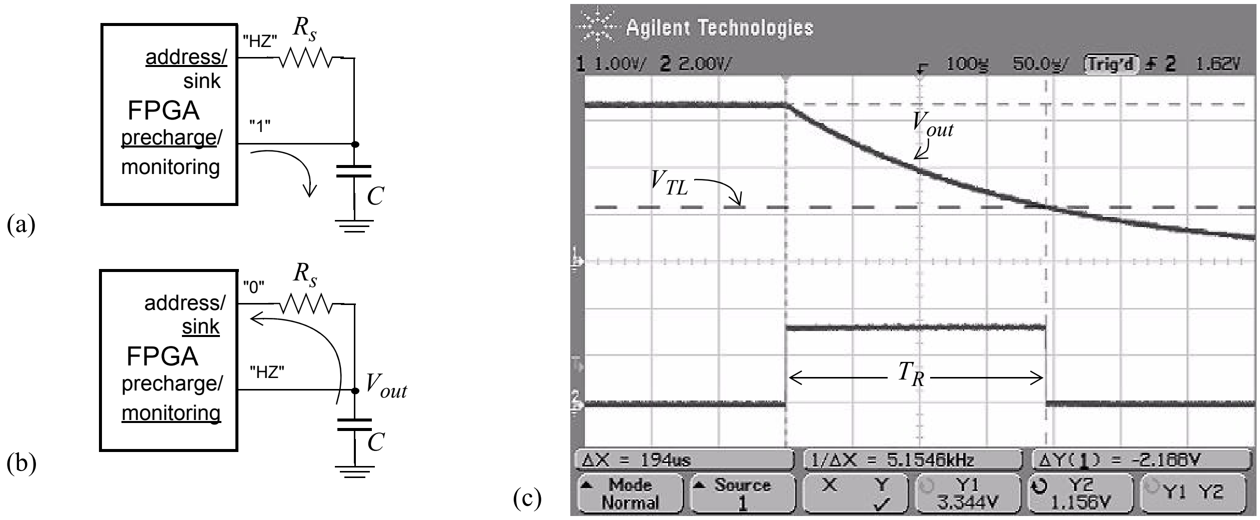

Fig.2. Direct interface between a resistive sensor and an FPGA with a passive integrator.

2. DIRECT CONNECTION WITH PASSIVE INTEGRATORS

The following procedure describes how to connect a resistive sensor to a device with digital interface, i.e. a device designed to interface with digital signals. A thorough study of this strategy for resistive sensor-to-microcontroller interfaces is reported in [9]. The approach is illustrated at Fig.2. In a first phase the capacitor is charged through the pin named "precharge/monitoring" at Fig.2(a). In the second phase it is discharged. To do that, the pin "address/sink" at Fig.2 is set to a digital low value and it sinks the current from the capacitor. A timer starts its count at this instant, and the voltage across the capacitor is monitorized by the input with label "precharge/monitoring" at Fig.2(b). When it takes a value corresponding to a digital "0" V_{TL}, the count stops. The measured time is

$$T_R \quad R_s C \ln\left(\frac{V_s}{V_{TL}}\right) \tag{1}$$

where V_s is the voltage across the capacitor at the beginning of the discharging phase. Note that the resistance can be obtained from (1) once T_R is measured. Fig.2(c) shows the voltage across the capacitor and the trigger signal to stop the count in the discharging phase measured with the scope for a given resistance value.

Slippage detection is the task with the highest dynamic requirements. Specifically, it is detected at frequencies around 250Hz [8]. Therefore, we should be able to carry out the A/D conversion of a whole array in the range of 2-4ms. Since we can perform many A/D conversions in parallel, this does not mean we should read the array into 2ms/(MxN) being MxN the number of tactels. Instead, we have to do it into 2ms/N, where N is the number of columns in the array, as described in the next section. This means the time constant is very small and trigger noise effects are negligible, so the resolution is given by [9]

$$\mathrm{ENOB} \approx lb\left\{ f_{CLK} C \ln\left(\frac{V_s}{V_{TL}}\right) \Delta R_s \right\} \tag{2}$$

where ENOB means Effective Number Of Bits, *lb* is the binary logarithm, and ΔR_s is the range of the resistance read from the tactile sensor. For a given ΔR_s and a required resolution we determine the value of C and f_{CLK}.

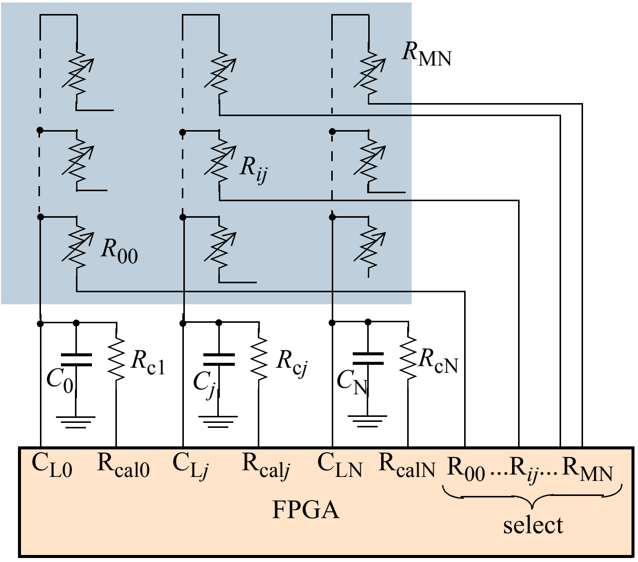

Fig.3. Proposed direct tactile sensor-to-FPGA interface with passive integrators.

3. INTERFACE FOR LOW-MEDIUM SIZE ARRAYS

Fig.3 shows a direct interface to a tactile sensor. The high number of I/O pins of the device is exploited to address the tactels. The array is read as follows. First, the capacitors $C_0 \ldots C_j \ldots C_{\mathrm{N}}$ are precharged by setting pins $\mathrm{C_{L0}} \ldots \mathrm{C}_{\mathrm{L}j} \ldots \mathrm{C_{LN}}$ to'1' and the remaining I/O pins to HZ. Then, a whole row is selected by setting its corresponding I/O pins to'0'. For instance, pins $\mathrm{R}_{i0} \ldots \mathrm{R}_{ij} \ldots \mathrm{R}_{i\mathrm{N}}$ are set to'0' while the remaining "select" I/O pins are set to HZ. The capacitors are discharged and the voltages across them are monitorized by pins $\mathrm{C_{L0}} \ldots \mathrm{C}_{\mathrm{L}j} \ldots \mathrm{C_{LN}}$, which are set to HZ. A set of timers are started in the FPGA at the beginning of the discharging phase and their counts are stopped when the low threshold $\mathrm{V_{TL}}$ is reached at the related column pins. Therefore, a whole row is read in parallel. Note that there is a dedicated pin per tactel in the array. It is not possible to address a whole row with just a pin because the tactels become connected to each other and the charge in the capacitors is redistributed among them by many different resistive paths. An implementation with isolated tactels is possible, for instance [10] reports a sensor with 272 tactels that are addressed with a track per tactel plus a common electrode.

To get the resistance R_{ij} from (1) we have to know the values of C, $\mathrm{V_s}$ and $\mathrm{V_{TL}}$. However, we ignore their exact value, and they can drift with time, power voltage supply or temperature. A calibration procedure is usually implemented to compensate such lack of knowledge and/or interferences. The simplest one consists in using calibration resistors, like those labeled $R_{\mathrm{c1}}, \ldots R_{\mathrm{c}j} \ldots R_{\mathrm{cN}}$ at Fig.3. The whole set of calibration resistors is read like a row of the tactile array. Then the resistance after calibration is computed as

$$R_{ij} \quad \frac{t_{Dj}}{t_{Dcj}} R_{cj} \tag{3}$$

A two-point calibration is better but it requires another set of resistors for calibration. The calibration procedure also consumes time, however we can do it once the time to read the whole array, or even at a lower rate to increase the dynamic performance.

The total number of I/O pins in Fig.3 dedicated to address the tactile sensor is MxN+2xN. This limits the size of the array that can be addressed in this way. For instance, an array of 8x8 tactels requires 96 pins of the FPGA to implement its interface. To obtain the results of this paper we have used a Spartan 3E-500 with 250 I/O pins, thus it is possible to implement this strategy. Its main advantage is that the number of integrated circuits is just one, so the interface is very compact. However, there are a high number of tracks and pins to connect and the PCB is more complex and expensive.

4. INTERFACE FOR LARGE ARRAYS

If the tactile array is large, the strategy in Fig.3 is not feasible. For instance, up to 288 pins are required to implement the interface with an array of 16x16 tactels. We propose for this case the use of active instead of passive integrators to implement the direct connection. Fig.4 shows the implementation for a single resistive sensor. The concept is the same that for the use of passive integrators, in Fig.4(a) the capacitor is charged, and it is discharged in Fig.4(b) with a constant current given by $i_D \quad \frac{V_s}{R_s}$ (note the linear discharge at Fig.4(c)). However there are some differences. First, we need to'turn off' the operational amplifier in the charging phase of the capacitor because otherwise its output interferes in the charging and it is not completed. So we need an amplifier with 'shutdown' input and a dedicated pin of the FPGA to address it. Second, another output of the FPGA is devoted to clamp the non inverting intput of the amplifier to a voltage close to ground in the charging phase. The charge would be completed without this clamp but the time to charge the capacitor would depend on the value of the resistor. Therefore, to reduce the time for the analog to digital conversion it is recommended the use of this clamp. However, it can be removed in the case of low dynamic requirements to reduce the number of pins of the FPGA dedicated to the analog to digital conversion.

Fig.5 shows the proposal to implement the interface with the tactile sensor in this case. Passive integrators are replaced by active ones and we obtain a circuitry with meaningful similarities to Fig.1(a). Note that the pin to shutdown the amplifiers is shared by them. Note also that columns in the array are virtually grounded due to the negative feedback loop implemented by the active integrators. This means we can follow the usual strategy in Fig.1(a) to short circuit the resistors that are not selected and avoid they contribute with parasitic currents to the output. This can be done as follows. In a first phase, the selection pins $R_{w0}...R_{wi}...R_{wM}$ are set to'0'. The tactile array and the FPGA share the ground, therefore a '0' at these pins means this voltage is almost 0 and the resistors of the whole array are short circuited. At the same time pins $C_{L0}...C_{Lj}...C_{LN}$ are set to '1' (voltage V_s), $C_{P0}...C_{Pj}...C_{PN}$ are set to '0', shutdown is set to '1' and the capacitors $C_0...C_j...C_N$ are charged to a voltage V_s across them. In the second phase, the set of column timers start their counts, and a row is selected. For instance R_{wi} is selected and there is a voltage drop V_s across tactels $R_{i0}...R_{ij}...R_{iN}$. The amplifiers are turned on by setting shutdown to '0'. Pins $C_{L0}...C_{Lj}...C_{LN}$ and $C_{P0}...C_{Pj}...C_{PN}$ are now at HZ. Therefore, currents $i_{Dj} \quad \frac{V_s}{R_{ij}}$ where $j \quad 1, \ldots N$ flow into the integrators, and the voltages at $C_{L0}...C_{Lj}...C_{LN}$ decrease. They decrease until threshold V_{TL} is reached at every pin C_{Lj}, then the count of the corresponding timer stops. At this time C_{Pj} is set to '0' to avoid that the

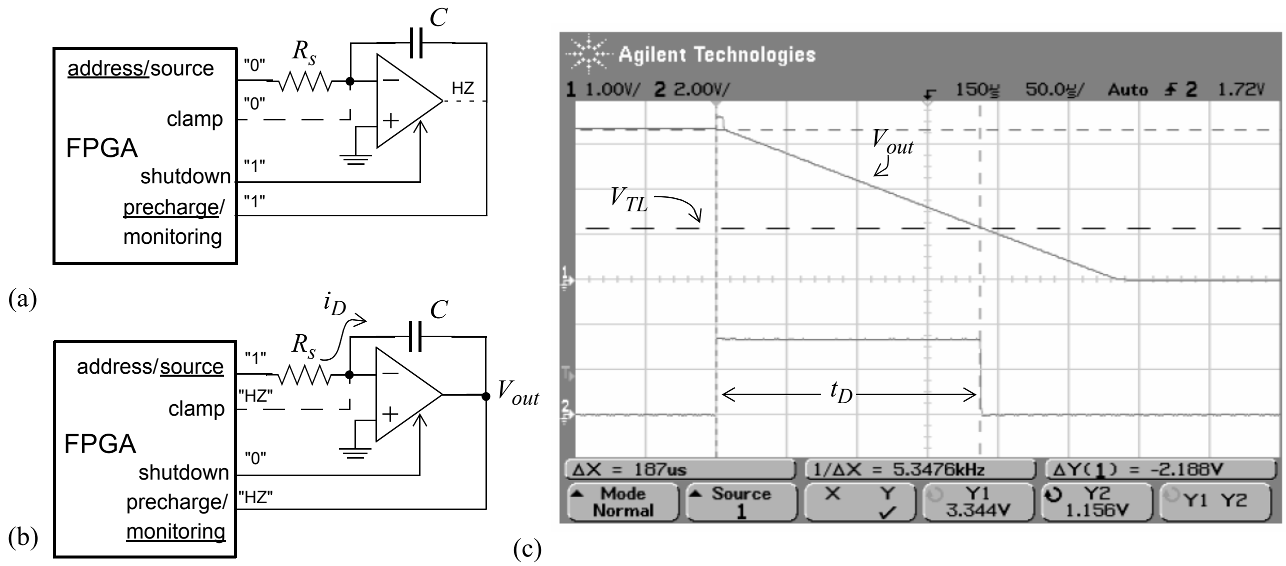

Fig.4. Direct interface between a resistive sensor and a FPGA with an active integrator.

voltage at the inverting input of the amplifier grows and generates interferences in tactels of the same row. We can get the value of the resistance from

$$R_{ij} = \frac{V_s}{(V_s - V_{TL})C_j} t_{Dj} \tag{4}$$

where t_{Dj} is the time measured by the timer. If this time is short enough, i.e. the time constant $R_{ij}C_j$ is small enough, we can neglect the trigger noise at threshold V_{TL} and take into account just quantization noise to obtain the resolution given by

$$\text{ENOB} \approx lb\left\{ f_{CLK} C_j \left(\frac{V_s - V_{TL}}{V_s} \right) \Delta R_{ij} \right\} \tag{5}$$

Note that a high current flows now from pin R_{wi}. Therefore, if the resistance of the tactile array is not high enough, additional resistors could be added to limit this current. This protects the device and also reduces the error caused by the output impedance of the pin R_{wi}. This impedance should be zero ideally, but it is determined by the output impedance of the MOS transistor that sources the current in the output stage of the pin R_{wi}.

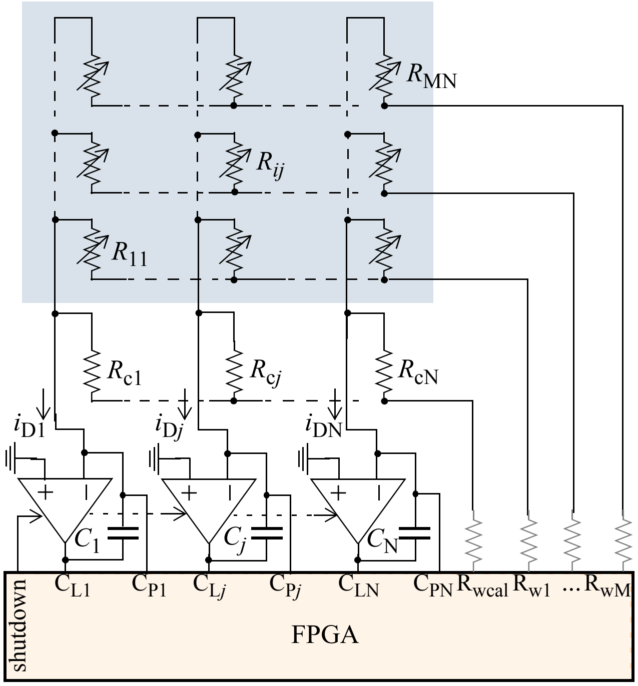

Fig.5. Proposed direct tactile sensor-to-FPGA interface for large arrays.

5. RESULTS

To show the feasibility of the proposal we have used a NEXYS 2 [11] development board with the Xillinx Spartan 3E-500. We have made two sets of measurements, and we have applied the calibration procedure commented at previous section. First, we have implemented the direct connection of Fig.2 (a calibration resistor was added to compute (3))and we have measured the resistance with the FPGA as well as with an Agilent 34401A multimeter. A timer of 16 bits was used to count the discharging time and a clock frequency of 50MHz. Table I and Fig.6 show the results obtained. Both data agree very well. Moreover, the standard deviation of measurements made with the FPGA is very small (26 samples were taken for every resistance value). This confirms that the trigger noise is negligible and our assumption is valid.

TABLE I: Resistance measured with a multimeter and with the FPGA and a passive integrator

Resistance measured with the multimeter (ohms)	Resistance measured with the FPGA (mean value) (ohms)	standard deviation (ohms)
1033	1025,820	1,408
1125	1121,080	1,265
1240	1237,944	1,186
1319	1319,828	1,205
1405	1406,617	1,610
1530	1533,445	2,000
1696	1702,498	1,630
1890	1900,553	1,794
1973	1983,565	1,904
2109	2122,529	3,127
2506	2526,245	3,364
3000	3029,017	3,239
3513	3551,164	3,810
4050	4097,210	4,417

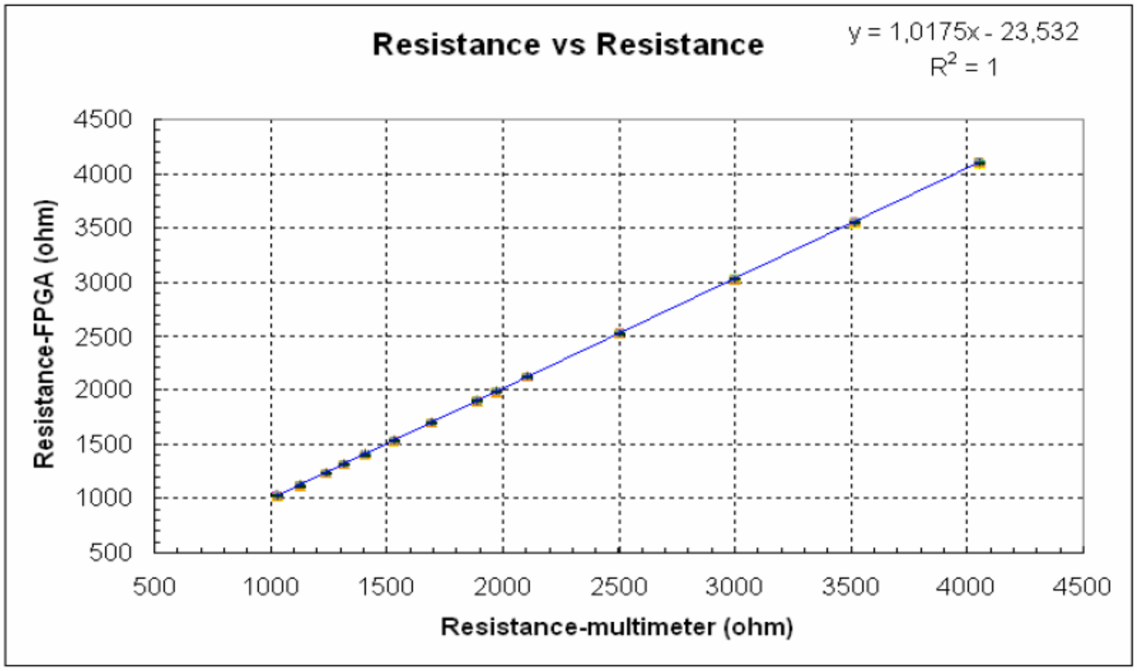

Fig.6. Resistance measured with the FPGA vs. resistance measured with the multimeter with a direct connection beased on a passive integrator.

Another set of measurements were done with the active integrator to test its feasibility. Table II and Fig.7 show the results. We observe a larger standard deviation in this case. This is due to the higher number of error sources this circuit has, for instance the commutation of the OPAMP. This paper is intended to show the proposal is valid, but a deeper study of the error sources and their impact on the final resolution has to be done. Nevertheless, this strategy is valid to cancel the interferences. In order to show it, the circuitry in Fig.5 is built for a tactile sensor with 16 x 16 tactels. A simple test is done that consists in pressing the surface of the sensor with an arrow made of rubber (see Fig.8(a)). Fig.8 shows the results. It is worth to highlight that just the tactels that are stimulated give a response, and the shape of the arrow is recognizable.

TABLE II: Resistance measured with a multimeter and with the FPGA and an active integrator

Resistance measured with the multimeter (ohms)	Resistance measured with the FPGA (mean value) (ohms)	standard deviation (ohms)
456	448,571	0,975
663	653,584	1,297
785	775,326	1,637
991	981,337	2,704
1202	1192,167	2,603
1650	1649,883	3,059
2200	2205,828	3,715
3886	3959,806	9,792
5663	5781,567	6,295
6796	6942,782	11,401

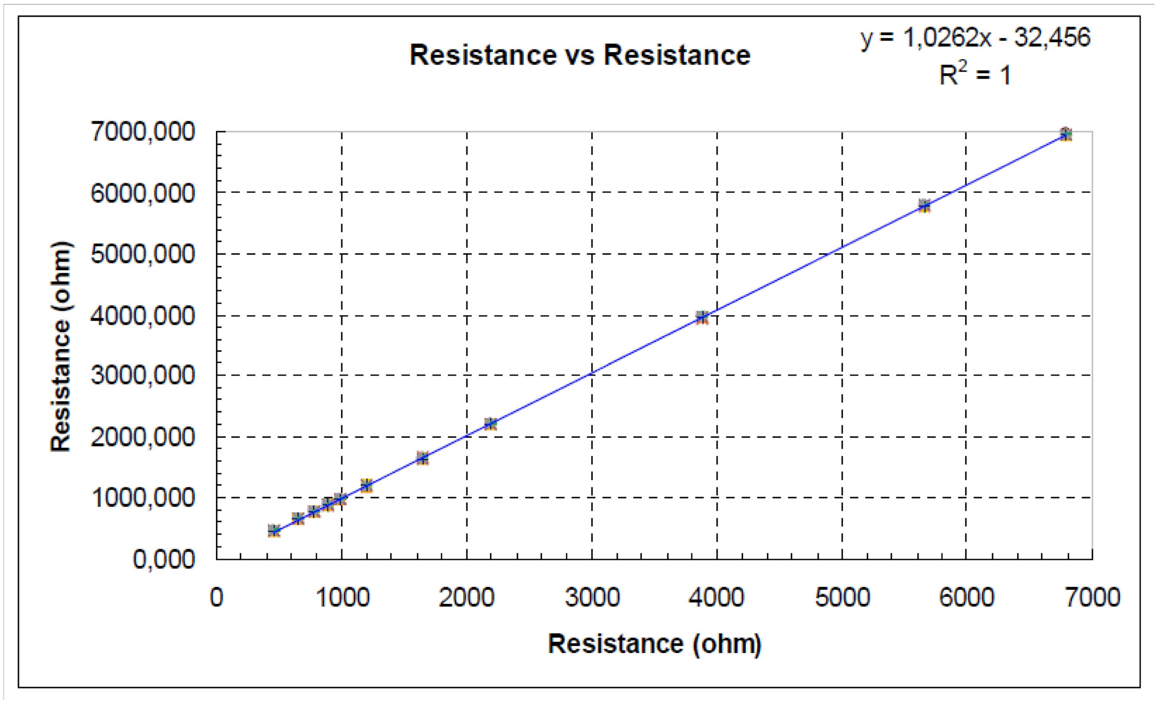

Fig.7. Resistance measured with the FPGA vs. resistance measured with the multimeter.

VI. CONCLUSIONS

This paper proposes a new way to implement a direct tactile sensor-to FPGA interface. The interface is implemented with the board NEXYS 2 [11] for the work of this paper. The analog to digital conversion procedure is simple. It consists of measuring the discharging time of a capacitor through the resistance we want to read. Results show the strategy works and it is able to read out the tactile sensor. For the case of low-medium size tactile sensors (in terms of number of tactels) the use of direct connection with passive integrators is feasible. It requires just the FPGA and a few passive components. However, tactels have to be addressed separately to avoid crosstalk. This means the PCB board is more complex and the number of tactels to be addressed is limited by the number of I/O pins of the FPGA. To overcome this limitation we propose the implementation of direct connection with active integrators. This circuitry cancels the crosstalk and allows the addressing tracks to be shared among tactels, so they can be arranged in rows and columns and the number of I/O pins of the FPGA required for the interface with the sensor is reduced a lot ($2N + M + 2$ instead of $N \times M + 2N$) as well as the addressing tracks. The price to pay is we have to add the operational amplifiers to the PCB board. Moreover, the implementation based on an FPGA allows the processing of data from the tactile sensor at a very high rate. This means we could be able to respond to very high demanding tasks in terms of dynamic requirements, like slippage detection. This also means we can run complex algorithms at real time, so a smart and powerful sensor is obtained.

ACKNOWLEDGMENTS

This work has been partially funded by the spanish government under contract TEC2006-12376-C02.

REFERENCES

1. M.H. Lee, "Tactile Sensing, New Directions, New Challenges", *The international Journal of Robotics Research*, Vol. 19, Nº 2, pp. 636-643, Sage Publications, 7 July 2000.
2. J. Tegin and J. Wikander, "Tactile Sensing in Intelligent Robotic Manipulation - a review", *Industrial Robot*, Vol. 32, Nº 1, pp. 64-70, 2005.
3. T. D'Alessio, "Measurement errors in the scanning of piezoresistive sensors arrays", *Sensors and Actuators*, A72, pp. 71-76, 1999.
4. B. Choi, H.R. Choi and S. Kang," Development of Tactile Sensors for Detecting Contact Force and Slip", *Proceedings of the IEE International Conference on Intelligent Robots and Systems2005 (IROS 2005)*, pp. 1677-1983, August 2005.
5. D.Göger, K.Weiss, C.Burghart, H.Wörn "Sensitive skin for a humanoid robot", *Proceedings of the 2006 International Conference on Human-Centered Robotic Systems* (HCRS´06), Munich, October 2006
6. G. Cannata and M. Maggliali, "Processing of Tactile/Force Measurements for a Fully Embedded Sensor", *Proceedings of the 2006 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pp. 160-166, September 2006.
7. A. Iwashita and M. Shimojo, "Development of a mixed Signal LSI for Tactile Data Processing", *Proceedings of the 2004 IEEE International Conference on Systems, Man and Cybernetics*, Vol. 5, pp. 4408-4413, Oct. 2004.
8. R. Maldonado-López, F. Vidal-Verdú, G. Liñán, E. Roca, Á. Rodríguez-Vázquez, "Early slip detection with a tactile sensor based on retina", *Analog Integr Circ Sig Process.*, December 2007
9. F. Reverter, J. Jordana, M. Gasulla, R. Pallàs-Areny, "Accuracy and resolution of direct resistive sensor-to-microcontroller interfaces", *Sensors and Actuators A* Vol. 121, pp. 78–87, 2005.
10. D. Göger and H. Wörn, "A Highly Versatile and Robust Tactile Sensing System", *Proceedings of the 2007 IEEE International Conference on Sensors,* pp. 1056-1059, October 2007.
11. http://www.digilentinc.com/Products Detail.cfm?Prod=NEXYS2

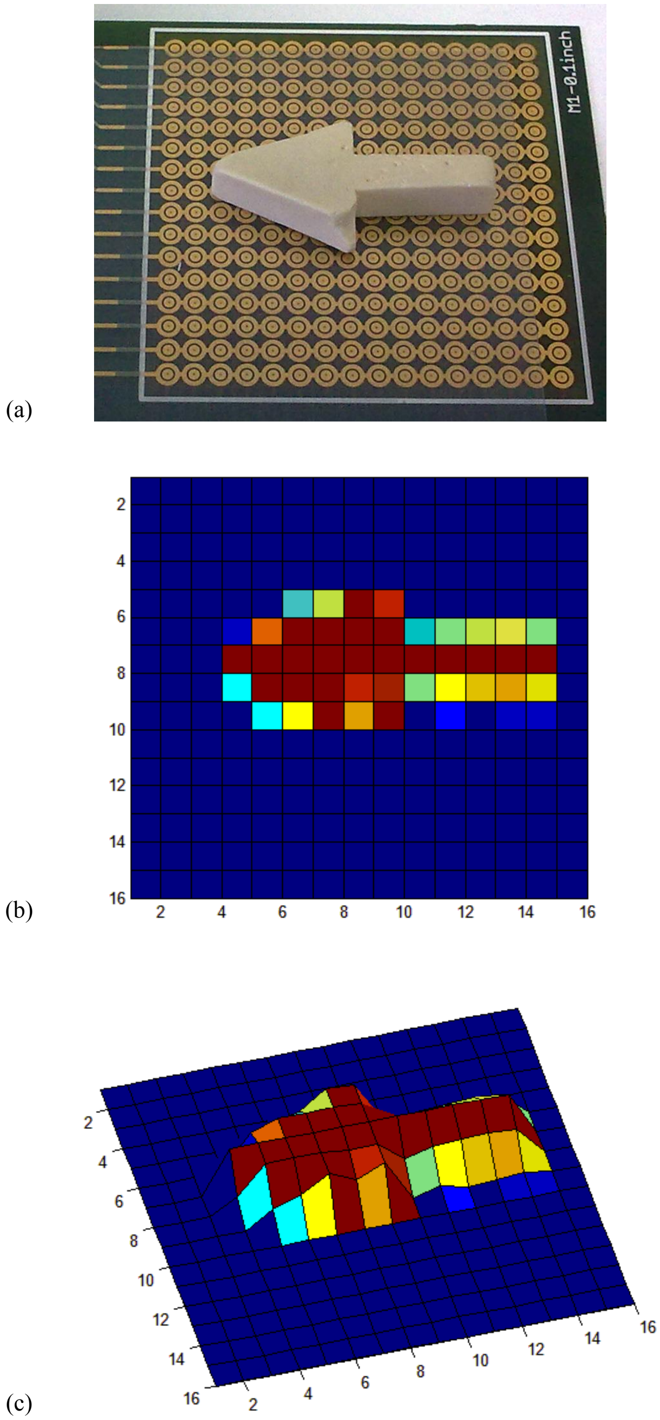

Fig.8. 2D (b) and 3D (c) output of the tactile sensor with the circuitry in Fig.5 when its surface is pressed with an arrow made of rubber (a)