In Proceedings of the 8th ESA Workshop on Advanced Space Technologies for Robotics and Automation
'ASTRA 2004' ESTEC, Noordwijk, The Netherlands, November 2 - 4, 2004

# CONTEXT – SPACE A&R CONTROLLER CAPABILITY EXTENSION

**P. Bologna(1), C. Mondellini(1), E. Crudo(1), L. Foresti (1), F. Didot(2)**

***(1)Galileo Avionica S.p.A. - Via Montefeltro 8, 20156 Milano (MI), Italy***
***(2)ESA/ESTEC, Noordwijk, The Netherlands***

***Email: patrizia.bologna@galileoavionica.it, carola.mondellini@galileoavionica.it, emily.crudo@galileoavionica.it, luca.foresti@galileoavionica.it, Frederic.Didot@esa.int***

## INTRODUCTION

The International Space Station (ISS) requires Extra-Vehicular Activity (EVA) for maintenance, servicing, repair and other mission tasks. In order to minimise astronauts' EVA, these tasks can be performed by robotic systems such as EUROBOT, the three-arms ESA system currently under development.
Purpose of the CONTEXT (Space A&R Controller Capabilities Extension) ESA Project is to develop the controller for the ESTEC EUROBOT ground testbed. The development activities exploit the results reached during the previous ESA Projects SPARCO, CESAR SW, CESAR HW and CIRCUS.
CONTEXT is able to control both in position and in force and to tele-manipulate three 7 degrees of freedom (DoFs) arms, at the same time, in free motion, in free coordinated motion and by applying the impedance control.
This paper presents the CONTEXT Controller, under development finalisation at Galileo Avionica.

## SYSTEM CONFIGURATION

The EUROBOT testbed (shown in Fig. 1) consists of three industrial PA-10 arms, of Mitsubishi Heavy Industries, mounted on a central body mechanic subsystem. The central body, integrated with the three arms, is located at the ESTEC Robotic Laboratory in Noordwijk. The three 7 DoFs are symmetrically arranged at 120° on the central body mounting triangle for maximum operational flexibility. Each PA-10 arm is equipped with a force / torque sensor and a parallel gripper End Effector. CONTEXT, mounted inside the body structure, performs all the control tasks of this system and is derived as an extension of the CIRCUS (Compact Integrated Robot Controller Unit and Servo amplifier) developed for ESA.

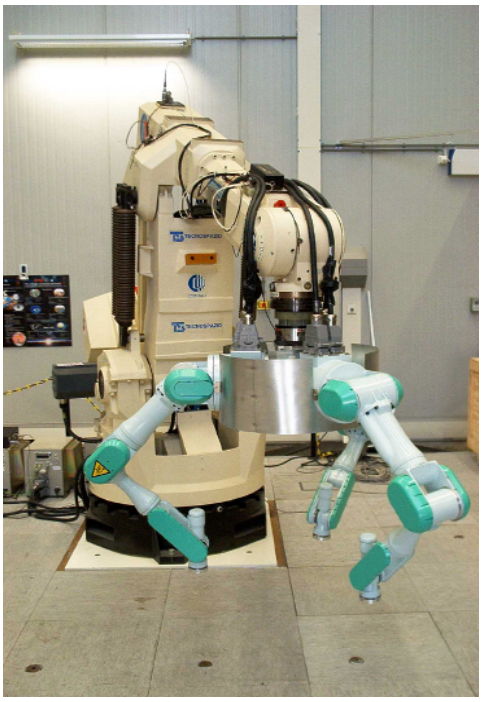


Fig. 1 – The EUROBOT testbed

## CONTEXT HW

The CONTEXT hardware (see Fig. 2) consists of one Robot Control Unit (RCU) based on a PC/104 plus [1] CPU interfaced, via SERCOS Bus, to three Arm Control Electronics (ACEs), one dedicated to an arm. Each ACE consists of seven Servo Control Unit (SCU) based on a TI (Texas Instruments) Digital Signal Processor (DSP) ([2]) and is supplied by a dedicated internal Power Supply Unit (PSU).

The overall electronics and arms are powered by an Auxiliary Power Supply Unit (APSU) in charge of providing the +28V and +100V power supply lines and to manage the auxiliary inputs/outputs as well as the safety circuitry and other external safety devices (barriers, controlled doors, etc.). Safety circuitry is based on the implementation of three parallel safety chains, each one devoted to a single arm and implemented as a double fail-safe (main/redundant) closed current loop circuit shared among all system components (i.e. APSU, RCU, ACE1, ACE2, ACE3).

The CONTEXT system is designed according to safety machinery rules [3].

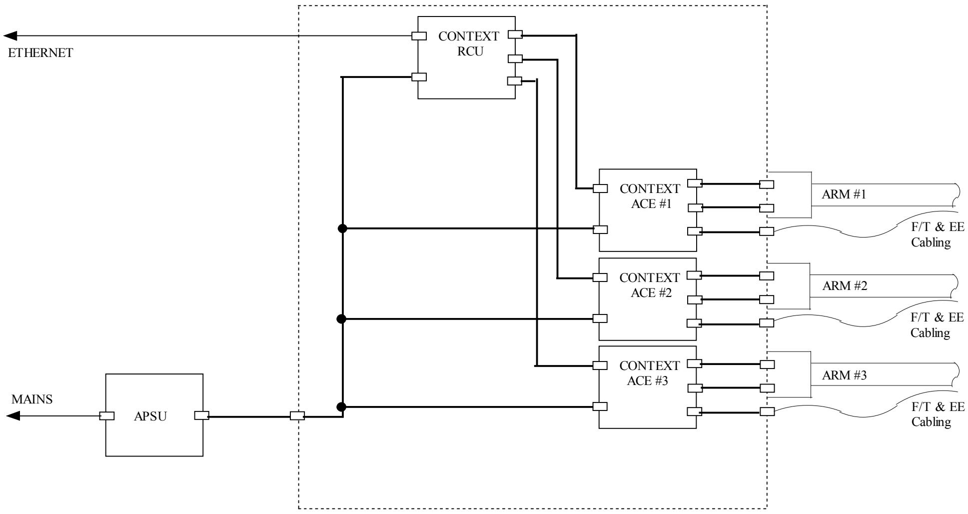


Fig. 2 – CONTEXT HW Block Diagram

## Robot Control Unit

The RCU is the “smart core” of the system: it provides the operator with the possibility of featuring remote control through a command interface by means of an Ethernet link. The RCU gets all the commands from the command interface and sends back telemetry to the remote terminal, it interprets and executes commands, performs robot motion planning and interpolation and sends servo level commands to the ACE(s).

The communication between RCU and ACEs is realised via three dedicated SERCOS bus implemented as a part of the Electrical Distribution / Power Supply Unit.

The RCU is composed of the following main parts (see Fig. 3):

- RCU PC104 CPU;
- three SERCOS interface boards (devoted to implement SERCOS communication logic between ACEs and RCU);
- three Field I/F boards (devoted to I/O handling);
- two Switch Boards (for controlling the 3 End Effectors as well as other ancillary I/O lines);
- one DAQ board;
- RCU Power Supply Unit to feed the PC104 stacked boards;
- Fan Unit for forced air cooling;

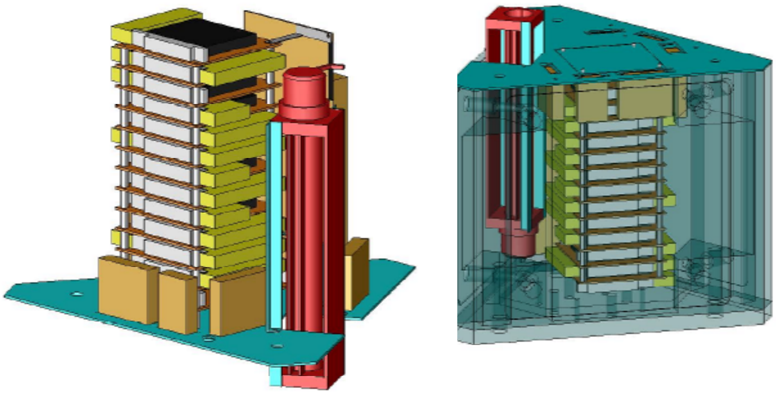


Fig. 3 – RCU Layout

The internal layout of the RCU was carefully studied in order to meet the volume constraints and allocate the RCU inside the mounting triangle of the CONTEXT central body (Fig. 4).

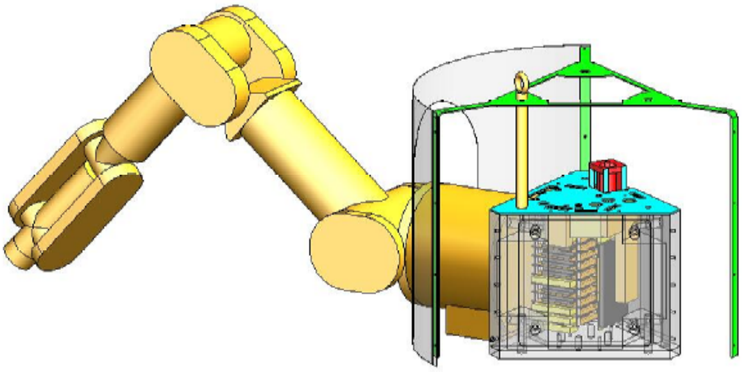


Fig. 4 – RCU mechanical accommodation

**Arm Controller Electronics**

The three ACEs are composed of the following items (see Fig. 5):

- seven SCU (Servo Control Unit);
- Power Supply Unit;
- Motherboard;
- End Effector Controller;
- two Fan Units for forced air cooling;
- Base and upper heat sink.

The ACEs receive +28Vdc and +100Vdc power lines from APSU and position set-points and operation commands (on a joint by joint basis) from the RCU, with which ACEs exchange both data and control / status lines.

Every SCU is in charge of driving one robot joint equipped with 3-phase brushless motor, motor shaft resolver, output shaft position resolvers (coarse and fine), failsafe brake and force / torque sensor.

The SCU motor control is based on an industrial DSP; the adopted single-chip solution has allowed to realise the best function integration with the minimum layout implementation effort.

Each SCU implements the functions of μ-interpolator, position / velocity / current control, motor commutation (based on resolver reading), PWM power amplifier and on / off brake control.

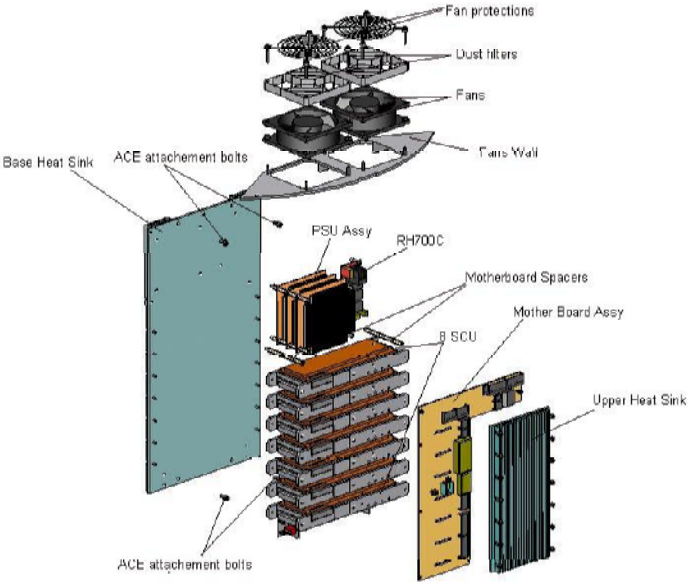


Fig. 5 – ACE layout

ACEs are accommodated in the free spaces between the external cater and the central mounting triangle, as shown in Fig. 6, with two fans mounted on the External Cover to produce a "wind tunnel" effect to cool the ACE itself.
Between the RCU Baseplate and the External Cover a spacer is used to leave a gap where cables from RCU to ACE are accomodated.

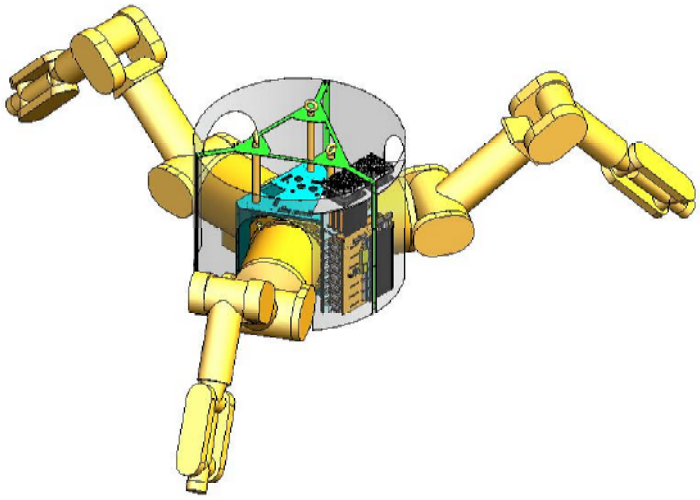


Fig. 6 – ACE accommodation

## CONTEXT SW

CONTEXT software (SW) consists of the RCU SW performing the most intensive computation high level tasks, and the SCU SW devoted to control servo drivers and sensors. To allow the communication among RCU and SCUs, a SW driver of SERCOS is integrated at both sides: the master SERCOS driver at RCU side, the slave SERCOS driver at SCUs side.
CONTEXT RCU SW is based on real-time multi-tasking operating system VxWorks.
CONTEXT SW issues in two versions: the version able to control the actual three PA-10 arms and the "emulator" version, running on the RCU Emulator and stubbing the external devices (SCUs, devices, sensors, SERCOS interface, input and output signals).

### Robot Control Unit software

RCU SW is an extension of the Controller for European Space Automation and Robotics (CESAR) software developed in the frame of ESA projects [4].
RCU SW inherits from CESAR SW the well-proven and mature industrial product COMAU C3G SW controller, and extends the CESAR SW capability for controlling three arms.
It is able to control in position three arms in such a way that the commanded arms can move concurrently (in the same time frame) or sequentially (one arm moves when the others have completed their motion).
The motion control (trajectory planning, trajectory generation, set-point generation) software architecture is based on the "master-slave" structure: the master properly sets-up the multi-arm motion, delegates the three

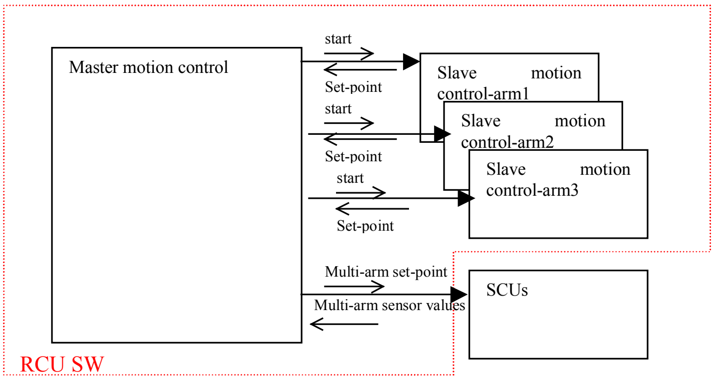


Fig. 7 - Motion control architecture

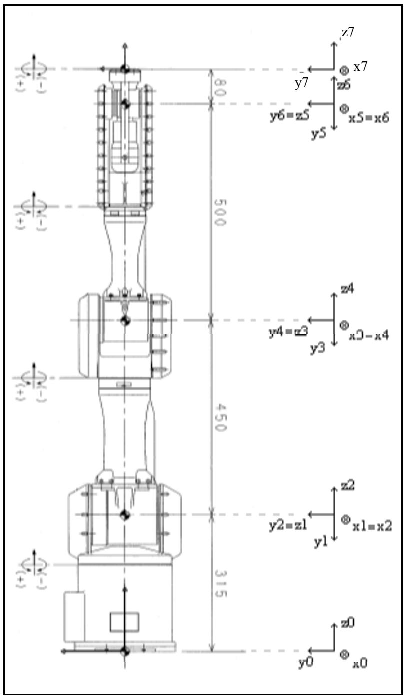
slaves (one for each arm) to generate the motion of the proper arm and then, at fixed frequency, collects from the three slaves the set-points to feed-up the SCUs (see Fig.7).
The PDL2 programming language [5] capability is fully integrated in the CONTEXT RCU SW, so that the PDL2 can be exploited to program the three arms and to control the input and output signals acquired by the CONTEXT I/O boards.

*Impedance and Proximity control*

The sensor-based control capabilities (impedance and proximity control) present in CESAR SW has been extended to three arms. In CONTEXT, the impedance control can be applied to one, two or three arms working on different objects. It is also possible to control two arms working on the same object, being one arm controlled in position and the other one controlled in impedance, making it possible the execution of the hand-over task among several arms.

*PA-10 arm kinematics*

The Cartesian and Impedance control algorithms have been tailored for the redundant 7 DoFs PA-10 arms (see Fig.8).



**PA 10 Denavit-Hartenberg Parameters**

| Joint *i* | $\theta_i$ | $\alpha_i$ | $a_i$ | $d_i$ | Range |
|---|---|---|---|---|---|
| 1 | $\theta_1$ | -90 | 0 | 315 mm | [ -177 ° , +177 ° ] |
| 2 | $\theta_2$ | 90 | 0 | 0 | [ -91 ° , +91 ° ] |
| 3 | $\theta_3$ | -90 | 0 | 450 mm | [ -174 ° , +174 ° ] |
| 4 | $\theta_4$ | 90 | 0 | 0 | [ -137 ° , +137 ° ] |
| 5 | $\theta_5$ | -90 | 0 | 500 mm | [ -180 ° , +180 ° ] |
| 6 | $\theta_6$ | 90 | 0 | 0 | [ -165 ° , +165 ° ] |
| 7 | $\theta_7$ | 0 | 0 | 80 mm | [ -180 ° , +180 ° ] |

Fig. 8 - Denavit-Hartenberg Parameters of PA-10 Arm

Two inverse kinematics algorithms for controlling PA-10 arms have been added to CESAR SW architecture, the former is iterative and the latter is the exact one. The iterative algorithm is based on the Jacobian transpose and exploits the arm redundancy by implementing the constraint “wrist orientation far away from wrist singularity” according to the task priority strategy [6]. The exact method is parametrical in the joint 3 (J3), so that the robot is handled as a 6+1 arm being the J3 value fixed to that one specified by the user. By means of a PDL2 program [5], the user can select the kinematics algorithm to be applied. In order to improve absolute positioning and motion accuracy, the RODYM system will be used to identify the kinematics parameters of each arm of CONTEXT. Since these parameters are stored into a configuration file, their updating will be performed without SW re-building.
CONTEXT SW is equipped with software tools allowing the user to substitute the kinematics modules devoted to forward and inverse kinematics computation with third parties kinematics module.

*“Extended” Cartesian Control*

The Cartesian control has been extended to allow the user the explicit control of J3 value during a Cartesian motion of Tool Center Point (TCP). In the “extended” Cartesian control mode, the user specifies the TCP target POSE and the J3 target value; the motion of J3 and the Cartesian motion of TCP are planned in such a way that the TCP Cartesian motion and the J3 motion start at the same time and terminate at the same time. When the user commands the “extended” Cartesian control and commands to keep the TCP POSE fixed, the arm motion in the Null Space of Jacobian is obtained.

The user can command one, two or three arms in “extended” Cartesian control mode, either concurrently or sequentially.

*Tele-manipulation*

The three PA-10 arms can be controlled also by tele-manipulation.

The tele-manipulation consists of tele-commands and telemetry exchanged from / to an external MMI via Ethernet link. Four different channels are foreseen: three channels are dedicated to tele-commands, one channel for each arm; one channel is dedicated to telemetry. In this way, if communication problems raise in some tele-command channel, that channel is closed while maintaining the telemetry channel still open. The communication is based on the TCP-IP protocol.

Tele-manipulation tele-command is an “extended” Cartesian set-points: TCP POSE and J3 value. Each arm has its own tele-command frequency which can be different from the other arms.

Tele-manipulation telemetry contains the data of the three arms: tele-manipulation status, joint values, force and torque values. Telemetry frequency is fixed, and is equal to the “master motion control” frequency.

Telemetry generated by CONTEXT is the signal synchronising CONTEXT SW and the external MMI: as soon as the telemetry is received, the MMI sends the “extended” set-points to be applied in the current time slot.

In the same time frame, multi-arms can be tele-manipulated.

During tele-manipulation, the CONTEXT impedance control can be active.

The user can adapt to his own need tele-command frequency and arms to be tele-manipulated by means of PDL2 [5] program.

*Collision Detection*

RCU SW capability has been augmented by including also a collision detection (CD) algorithm, which predict collisions of each arm against its parts, of each arm against the other arms, of each arm against the central support body.

For CD purpose, the CONTEXT arms and central body shapes have been modelled as a collection of spheres. This sphere-based model, according to [7], minimises computational complexity and time, is highly scalable and do not require hypotheses about the shapes of the objects to model.

Each arm can be modelled by 10 spheres: 7 spheres are rigidly connected to joint reference frames; 3 spheres are rigidly connected to the 7th-frame in order to model the End Effector (EE) and the payload. Radius and position of each sphere can be selected at user convenience, in order to satisfy the desired degree of approximation.

Considering the CONTEXT layout and kinematics and being supported by simulation results, CONTEXT has been modelled as a collection of 21 spheres, 5 spheres for each arm and 6 spheres for modelling the central support and the main body (see Fig. 9).

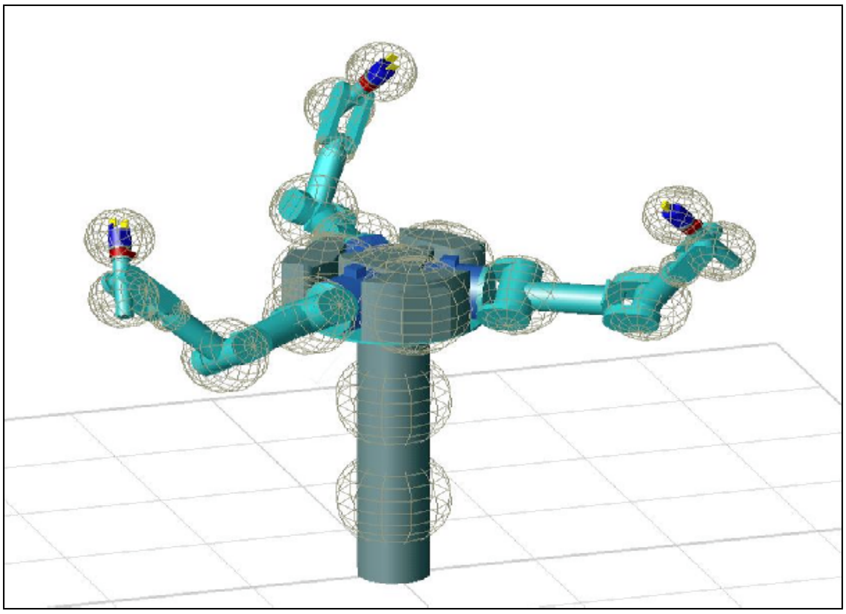


Fig. 9 - CONTEXT model overview

CD algorithm is able to detect collisions raised in three regions: caution, warning and emergency (see Fig. 10).

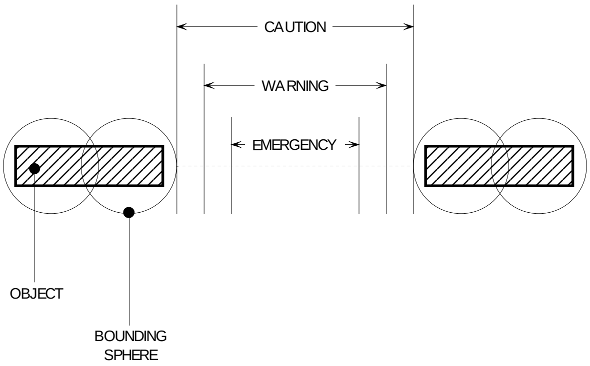


Fig. 10 – Caution / Warning / Emergency regions for collision detection

The user can modify the dimensions of the three regions as well as the kind of action to be performed when a collision is detected. During arm motion, the CD algorithm can be enabled or disabled at all.

*Input and Output signals*

The user can monitor the input signals acquired by the CONTEXT I/O boards and can control the End Effectors by means of the "user defined port I/O" [5] provided by PDL2.

**CONTEXT Emulator SW**

CONTEXT emulator SW consists of the same software modules of CONTEXT RCU SW with the exception of the modules handling the external devices. So, in CONTEXT emulator SW, the servo control loop is modelled as an ideal one, being the evaluated set-point equal to the desired one; the actual three arms are "emulated" by RobCad 3D draw; the set-points are sent to a RobCad application which uses them for displaying the 3D drawing. The sensors are simulated as ideal ones, being their values equal to the desired ones. The input/output signals are received / sent from / to "Robot Monitoring and Command" external workstation via Ethernet. The CONTEXT scheduler is activated by an internal timer.
CONTEXT emulator SW runs on the RCU emulator, having the same CPU of the actual RCU.
CONTEXT emulator equipped with CONTEXT emulator SW is a powerful tool for the analysis, programming and mission preparation of the three PA-10 arms.

**Servo Control Unit SW**

SCU SW is in charge to perform the function of micro-interpolation, position / velocity / current control / motor field oriented control [7] based on resolver reading [8], on / off brake control.
It performs health checks on drive, sensor and, at power on, on flash data persistency. Every time a set-point is received, a plausibility check is performed so that if two consecutive set-points are not "plausible" an extrapolated value is used. Check failures are notified to RCU.
Configuration parameters of the drive under control as well as the SW to be burned in flash memory are loaded via proper RCU commands.

**Robot Control Unit and Servo Control Unit Interface**

Communication between RCU and SCUs is via SERCOS.
The interrupt raised by SERCOS board is used as synchronisation signal between RCU and SCU and it is used as clock for the RCU SW scheduler: when the interrupt is raised, the set-point evaluated by RCU SW are read by SCU; the sensor values acquired by SCUs are read by RCU; the RCU scheduler is waked up so that the "master motion control"

starts its loop by sending, first of all, the tele-manipulation telemetry to the external MMI. In this way, SERCOS interrupt is the synchro signal among RCU, SCUs and external MMI performing the tele-manipulation.
The message exchange through SERCOS is via Dual Port RAM (DPR); messages have fixed length; alive protocol has been implemented to detect problems in the RCU/SCU communication.

## CONCLUSIONS

CONTEXT RCU software architecture is based on a master-slave-type structure: this significant characteristic allows to have one master SW module which can properly coordinate a number of different slave SW modules, each one devoted to one arm control.
Furthermore, the modular and configurable HW and SW architecture of CONTEXT allows a further extension to systems with more than three arms (e.g. locomotion control for planetary exploration).
Finally, a possible evolution of this controller is the inclusion of a relocation control capability into an integrated test environment.

## REFERENCES

[1] *PC/104 Specification*, Version 2.4 – August 2001 – PC/104 Embedded Consortium, hppt://www.pc104.org.
[2] *TMS320C240 TMS320F240 DSP Controllers*, Texas Instruments, November 1998, http://www.ti.com.
[3] EN 60204-1: 1998 - *Safety of machinery. Electrical equipment of machines. General requirements.*
[4] G Visentin et al., *Common European Space Controller (CESAR),* ASTRA '98
[5] *PDL2 Programming Language Manual*
[6] P.Chiacchio, S.Chiaverini, L.Sciavicco, B.Siciliano, *Task-Space Augmentation and Task-Priority Strategy in Closed-Loop Inverse Kinematic Schemes for Constrained Redundant Manipulators,* International Journal Of Robotics Research, Feb 1989
[7] Martinez-Salvador, del Pobil, Pérez-Francisco, *Very Fast Collision Detection for Practical Motion Planning*, Proceedings of the 1998 IEEE International Conference on Robotics & Automation, Leuven, Belgium, 1 May 1998G.
[8] Application Report SPRA588, *Implementation of a speed field oriented control of three-phase PMSM motor using TMS320F240* , Texas Instruments, September 1999, http://www.ti.com
[9] Application Report SPRA588, *TMS320F240 DSP solution for obteining resolver angular position and speed* , Texas Instruments, February 2000, http://www.ti.com