NLR CR 95684 L

NATIONAAL LUCHT- EN RUIMTEVAARTLABORATORIUM

NATIONAL AEROSPACE LABORATORY NLR

THE NETHERLANDS

NLR CR 95684 L

NLR ROBOTICS DEVELOPMENT- AND TEST ENVIRONMENT
- REAL-TIME OPERATIONS SIMULATOR -

by

Z. Pronk and J. de Haas

NLR

NATIONAAL LUCHT- EN RUIMTEVAARTLABORATORIUM
NATIONAL AEROSPACE LABORATORY NLR

Anthony Fokkerweg 2, 1059 CM AMSTERDAM, The Netherlands
P.O. Box 90502 , 1006 BM AMSTERDAM, The Netherlands
Telephone : 31-(0)20-5 11 31 13
Fax : 31-(0)20-5 11 32 10

NLR CONTRACT REPORT

CR 95684 L

NLR ROBOTICS DEVELOPMENT- AND TEST ENVIRONMENT

- REAL-TIME OPERATIONS SIMULATOR -

by

Z. Pronk and J. de Haas

gebruik van ERA?

This investigation has been carried out under a contract awarded by the Netherlands Agency for Aerospace Programs, contract number 02501N.

Division : Space
Prepared : ZP/ JdH/
Approved : HFAR/

Completed : 960116
Order number : 102.502
Typ. : JS

Summary

This report contains a description of work performed under contract with the "Nederlands Instituut vor Vliegtuigontwikkeling en Ruimtevaart, NIVR".
The objective of the work was to implement and demonstrate a real-time simulation package in the development and test environment of the NLR Teleoperations and Robotics laboratory. Originally, EUROSIM was planned to be implemented, but could not be made available in time. Therefore, the NLR real-time simulation package PROSIM was used.
For the demonstration, the HSFP model software was used. This software consists of dynamics model software and onboard software models.
A new user interface software has been developed using the simulation control software interface package Tcl/Tk.
The architecture of the software is such that a distributed simulation can be realised. The simulator, the visualisation, and the simulation control and operations control interface can run on three different workstations.

Contents

(34 pages in total)

Acronyms

A&R	Automation & Robotics
AGAR	Advisory Group for space A&R
ARMADE	A&R in Microgravity Application Demonstration
CAT	Columbus Automation Testbed
COF	Columbus Orbital Facility
CPU	Central Processor Unit
ERA	European Robotic Arm
EUROSIM	European Real-time Operations Simulator
HERA	Hermes Robot Arm
HSFP	HERA Simulation Facility Pilot
ISS	International Space Station
IPU	Internal Processor Unit
LEDA	Lunar European Demonstration Approach
MMI	Man-Machine Interface
NIVR	Nederlands Instituut voor Vliegtuig- en Ruimtevaartontwikkeling
PC	Personal Computer
PROSIM	Programme and Real-time Operations SIMulation support.
ROTEX	Robotics Technology Experiment
TNO-FEL	TNO- Fysisch Electronisch Laboratorium
VR	Virtual Reality

VME

This page is intentionally left blank.

1 Introduction

This report contains results of work performed under contract with the "Nederlands Instituut voor Vliegtuig- en Ruimtevaartontwikkeling NIVR", contract 02501N "Ruimte Robot Technologie", workpackage 2 "Robot Ontwikkel- en Test-omgeving" (Ref. 8).

1.1 Background

The application of Automation & Robotics (A&R) in the European space operations infrastructure has been discussed since about 1980. Many concepts have been studied for both external and internal robotics, and various studies have been performed on issues related to A&R, artificial intelligence and advanced control. Examples of external robot concepts are the Service Manipulator System (SMS) and the Hermes Robot Arm (HERA), and examples of internal robotic concepts are the Experiment manipulator System (EMS), the Equipment Manipulation and Transformation System (EMATS), BIOROB, and the Automated Manipulation and Transportation System (AMTS). The latter system has been designed for the European part of the International Space Station (ISS), the Columbus Orbital Facility (COF).
An overview of initial concepts for A&R technology was already given in reference 7. From this study and a concurrent study on identification of projects (Ref. 3), a Dutch consortium started the project on "A&R in Microgravity Application Demonstration (ARMADE)".

The German Robotics Technology Experiment (ROTEX) is the only (experimental) internal robotic system that has been realized and has been demonstrated in Europe up till now. It flew on the German D-2 spacelab mission onboard Shuttle in April 1993.

Some examples of studies on automation (in the sense of control) are the Control Techniques Study, the study on Interactive Remote A&R Servicing (IRAS), the study on a ground based Flexible Automation Monitoring Operator User Station (FAMOUS), and the study on a Space Automation & Robotics Controller (SPARCO) (Ref. 2).

Presently, the external manipulator for the Russian segment of the ISS, the European Robotic Arm (ERA), is scheduled to be delivered in 1998, so it has reached the C/D development phase. Another external robotic system (JERICO) has been scheduled for a technical demonstration mission.
The above mentioned studies are focused on Extra Vehicular Activities (EVA) in Low Earth Orbits (LEO), but also external robotic concepts are prepared for geostationary orbits to support servicing of geostationary satellites (Geostationary Service Vehicle, GSV).

In the area of planetary missions a number of conceptual studies have been performed on robotic and rover concepts for both missions to Mars and Moon. Next to the many non-european developments, some prototype developments have been realized in Europe, mainly in the France and Italy. In the Netherlands only TNO-FEL has developed a rover, based on the Russian MARSHOKOD, for the purpose of demonstration of a telepresence concept.

An overview of present European robotics development activities was presented during the meeting of the Advisory Group for space Automation & Robotics (AGAR), may 1995 (Ref. 6).

The conclusion can be drawn that the realization of an A&R system for space application takes many years of research and preparation, caused by programmatic evaluations (due to cancelling of programmes or elements) and complexity of the system. For instance, the AMTS concept has not been accepted for budgetary reasons, while most people are convinced that internal robotics will be part of future space laboratories. So, to be prepared for the future the devlopment of such systems should be continued.
For this reason a number of European A&R development and test support facilities are needed. ESA/ESTEC has been setting up a facility for A&R development and test support, the Columbus Automation Testbed (CAT).
NLR started to set up such a facility in 1992 focussing on teleoperational aspects, and is still in the process of development and implementation of support tools. The reasons for having such a facility are:

- A&R study results need support from experimental studies and demonstrations.
- users of micro-gravity facilities could be made familiar with internal A&R in space laboratories by demonstration of robot to payload interaction.
- the influence of robotic operations on the micro-g level could be investigated by real system measurements.
- a number of existing study areas at NLR could be integrated.
- the problem of time delays in telescience experiments could be investigated using realistic simulations, based on a reconfigurable system set-up.
- NLR is interested in (participation of) development of ground support facilities (European Proximity Operations Simulation (EPOS), European Real-time Operations Simulation (EUROSIM), Dutch Utilisation Centre (DUC), ERA ground operations facility).
- NLR would gain experience with real A&R systems, to support theoretical studies.

In order to realise a flexible reconfigurable Teleoperations and Robotics laboratory, a number of development and operations support tools are required. A real-time simulator is one of the required tools.

1.2 Study objectives and scope

The main objective of present development activities in the NLR Teleoperations and Robotics laboratory is to have a flexible system set-up which can be used for operations analysis, for teleoperations development, for design and development of payload interfaces and control, and for development of user interfaces with various sensor-feedback configurations.
A description of the NLR laboratory is given in chapter 2.

In a space A&R laboratory it is very important to have a simulation/emulation environment to simulate future space robotic concepts and emulate operations on e.g. the level of an end effector of a dedicated hardware robot. In this way a remote operator will control and monitor, by graphics simulation, a robot, while the laboratory robot is actually operating in a similar simulated task environment. The most important feature of the simulation environment is that it has to perform in real-time.

The work reported encompassed the implementation of a real-time operations simulation package in the Teleoperations and Robotics laboratory environment. Initially, EUROSIM was suggested to be the appropriate package, but since EUROSIM could not be installed in time, the NLR simulation environment PROSIM has been used.

The scope of the work was determined by its application. In order to re-use existing software as much as possible, the model software of the HERA Simulation Facility Pilot (HSFP) was used as application software. The reason for re-use of HSFP-software is two-fold: NLR developed this software, so the knowledge of this software is available, and the model software had been satisfactory. The second reason is that the HSFP models are assumed to be easily converted to ERA models in a later stage. This is expected to be done on parameter level.
The HSFP included both dynamics models and visualisation models, interfaced to each other. The transfer of this software to the laboratory workstation is assumed to be possible at reasonable effort.

This report contains a results of the activities performed to implement a real-time operations simulator based on PROSIM, that will be used to support emulation of hardware and software elements of the HERA.

The report contains the conceptual design of the system including the development and implementation steps. In addition, the real-time simulation tool is described, the simulation of the robot, and the integration of the system.

2 NLR teleoperations and robotics laboratory

From 1990 NLR has been preparing for a laboratory for teleoperations and robotics. The objective of the facility is mainly to investigate operational concepts on automation and robotic systems, such as operations preparation, training, operational support, teleoperations, crew assisted operations, etc. These systems include both payload automation, laboratory robots, planetary robotics concepts, as well as external robotics, such as ERA. It is not the intention to design and develop robot hardware systems or automation products, but to implement present technologies in an operational concept where an optimization has to be found in the combination of technologies, interfaces, communication, and distribution of controller elements in ground and space segments.

For this purpose the NLR Teleoperations and Robotics laboratory needs a set of basic technologies and tools to allow for flexibility in configuration and detail of experiment set-ups. Until beginning of 1995 the laboratory comprised a number of uncoupled systems:

- Laboratory robot system (CRS Plus), including controller, and a standard Columbus payload double rack in front of the robot.
- Robotic operations preparation and planning system (ROBCAD)[1], installed on a Silicon Graphics Inc. (Indigo2) workstation.
- A stand-alone Virtual Reality (VR) system, that can be coupled to the workstation.
- Some teleoperation devices, such as cameras, video equipment and video data handling packages.
- A user support infrastructure (Dutch Utilisation Centre, DUC) for micro-gravity experimentation, focused on telescience and crew-supported operations.

The elements mentioned above are planned to be integrated such that a complete ground to space system can be configured. Since the laboratory includes a number of dedicated earth-based hardware devices, simulation is needed to emulate operations with flight hardware and software in order to prepare flight operations, to train operations, and to validate operations.
In figure 1 an overview of the NLR Teleoperations and Robotics laboratory is given. The main elements mentioned above can be recognised.

The robot system is a commercially available robot from the Canadian manufacturer CRS-Plus inc., type CRS A460. The six axes robot is built on a separate but controllable linear track, which enhances the reach of the robot gripper along the standard Columbus payload rack. The robot includes controller, torque-force sensor, and a small video camera has been attached to the

1 RABCAD is a 3D-graphic engineering software platform for designing manufacturing processes. Is is a product of Technomatix technologies Ltd.

standard gripper at the end-effector.
The robot system can be controlled from both a teach panel or a Personal Computer (PC). On the PC it is possible to prepare robot operations in a high-level robot control language.

The payload rack is a 'chinese copy' of a Columbus double rack, based on a 19" rack system. The rack will be equipped with realistic payload, simulated payload, and payload front ends, dependent on the purpose of the payload in the study to be performed. Provisionally, two payloads have been developed and integrated: an incubator, and a drawer used as storage locker.

The data handling system of the (simulated) "space segment" has been implemented in a VME system. Via this system it is possible to control integrated operations including robot, payload and sensors. The data handling system can be seen as the onboard data handling system. It is possible to control the robot from a remote workstation.

The teleoperations workstation consists of a number of devices that enables a (remote) operator to control the robot and sensors and to monitor the whole system. Presently, the teleoperations system is focused on controlling and monitoring using vision systems, including a video teleconferencing system.

For future preparation and training of robotics operations Virtual Reality techniques will be used in case of teleoperations, also called telerobotics. A VR system is part of the laboratory, but has up till now only been used as a stand-alone system. Within the scope of the NIVR-NRT activities the VR system will be integrated.

The operations preparation system is (presently) based on a commercially available robotics and workshell design package, ROBCAD, implemented on a Silicon Graphics workstation (Indigo2). ROBCAD enables the preparation of operations of dedicated robot systems, such as the CRS Plus system, based on a geometry model of robot and its environment.

The real-time simulator mentioned is to be implemented on the laboratory workstation.
Note: It should be clear that ROBCAD can not be used for this purpose, since ROBCAD is an off-line robot operations planning package, which is not capable of running in real-time.

3 Implementation approach

From the objective and scope of the work it can be concluded that most of the effort had to be spent on the transfer of HSFP-software to the Teleoperations and Robotics Laboratory workstation.
The HSFP-software was implemented in the HSFP on three computer systems (see Fig. 2). The dynamics and control software was implemented on the Gould/Sel computer, the Man-machine Interface software was implemented on a SUN workstation, and the visualisation software was implemented on a Silicon Graphics workstation.
In addition, hand controllers were used to control the HERA manipulator.

The target configuration on which the HSFP model software had to be implemented consists of the Indigo2 workstation, including the NLR simulation environment PROSIM. The Indigo2 is a single processor system, but is fast enough to run the HERA models in real-time, including the visualisation of the HERA geometry models.

A number of development and implementation steps were performed in order to reach the objectives. Each of the steps will be explained briefly.

a. Installation of PROSIM on the Indigo 2 workstation

The "Programme and Real-time Operations Simulation support (PROSIM)" programme is a generic simulation tool that can be used for aerospace applications. PROSIM has been developed for the National Simulation Facility (NSF) and used in the NLR Flight Simulator (Ref. 1).
Based on the requirements for using a general simulation tool in the robotics development and test environment, PROSIM was selected because:

- The programme is running for some time now and NLR hands-on experience on is available.
- The programme has generic interfaces for different model software.
- The programme has the characteristics needed to perform real-time simulations.
- the programme is an NLR product that can be supported in-house.

PROSIM has been installed on the Indigo2 workstation, but also remote access is possible via the NLR network with "remote file access" and "remote login".

b. Familiarisation with PROSIM

For familiarisation with PROSIM, the PROSIM Acceptance Test Sequence for the Test and Verification Equipment Test Software (Ref. 4) was used.

c. Identification of the HSFP models

The original HSFP model sofware consisted of three packages:

- Dynamics models (including "Onboard Software" models).
- MMI (control interface) control software.
- Visualisation models.

This software structure was similar to the originally planned Hermes onboard computer software structure. The software packages were implemented on three different computers in the HSFP: a Gould/Sel computer, a SUN workstation, and an IRIS workstation (see Fig. 2).

Since the MMI software was dedicated SUN software, directly implemented on the target machine, this software could not be used in the target configuration. The PROSIM graphical user interface development tool, Tcl/Tk (public domain software), could ,however, be used.

The dynamics and control software and the visualisation software were transferred to the Indigo2 workstation. Therefore, a selection of model software was made.

d. Definition of implementation requirements of HSFP model software

Taking into account the HSFP structure and the model interfaces, model requirements were defined in order to reach a sensible implementation and demonstration. In particular, the use of the hand controllers was evaluated. The need for these controllers and the consequences on models and implementation of models were analysed.

e. Architecture design of the simulator

In setting up an architecture, the PROSIM requirements on models to be implemented, and the requirements from the integration structure of the existing models were taken into account. Initially, the structure was based on automatic setpoint generation (no external control). Later, control was implemented via the user interface. In addition, a distinction between control models and dynamic models was proposed. Interface models, such as ethernet coupling modules were handled separate, because these are machine dependent.
Also very important was to define the scheduling of the model software blocks.

f. Adaptation of of HSFP dynamics and MMI models to PROSIM

Given the proposed architecture and the requirements for interfacing with PROSIM, HSFP models were adapted, and ethernet modules were recoded separately.

g. Adaptation of HSFP visualisation models to the Indigo2

Given the proposed architecture and the requirements for interfacing to the Indigo2 workstation, the HSFP visualisation models were adapted.

h. Integration and test of HSFP models

Next to the implementation of model software, the scheduler and the simulation scenario were implemented. With the latter two elements the model reconfiguration could be tested. The demonstration scenario and the data of the scenario are determining the possibilities of the simulation.

i. Coupling of the simulator to the Teleoperations and Robotics lab

A first step towards integration with the hardware robot system was to investigate the possibilities for coupling the position and orientation data of the HERA end effector of the simulated robot to the controller of the hardware robot in the Teleoperations and Robotics laboratory.

This step could not yet be implemented within the framework of the reported activities, but will be dealt with later.

j. Generation of a User Manual

A User Manual was prepared. The real-time simulator should be seen as an additional tool to the laboratory facilities. The PROSIM User Manual will be part of it.

The most important steps of the implementation work are elaborated below.

4 Preparation of PROSIM

As mentioned before, EUROSIM was planned to be used as real-time simulation package for the NLR Teleoperations and Robotics laboratory. However, EUROSIM is not yet available for use in that environment and therefore the decision was made to use PROSIM.

PROSIM is a generic simulation tool that can be used for support in design, development, verification, test, training and operations planning of various applications (Ref. 1). PROSIM was developed as part of the National Simulation Facility (NSF) at the National Aerospace Laboratory NLR. The package is developed and used for flight simulations, but can also be used in other areas of simulation, such as space simulations.

PROSIM is available on the NLR network as part of the engineering work environment ISMus (Ref. 5), but this version is a non-real-time version. In order to use PROSIM for real-time applications, PROSIM needs to be installed on a dedicated workstation. Therefore, a PROSIM version has been installed on the Teleoperations and Robotics Laboratory workstation, the Silicon Graphics Inc. Indigo2.

For familiarisation with PROSIM the knowledge and experience within the development team for the Test and Verification Equipment (TVE) was used. The acceptance test procedure seemed to be a very good starting procedure to experience the capabilities of PROSIM (Ref. 4).

5 HSFP model preparation and conversion

HSFP-software had been stored earlier on magnetic tapes at NLR's computer centre of the Informatics Division. The software has been made accessable for re-use in the robotics lab. As explained before a distinction had been made between dynamics model software and visualisation model software. The man-machine interface software could not be reused. It was decided to implement the simulation scenario user interface in the PROSIM simulation control user interface (see ch.3, under c).
The development and adaptation of these three types of software will be described in the following sections.

5.1 Dynamics and onboard software models

In the HSFP the software for simulation was divided into so-called 'Building Blocks' or BBs. In order to copy the BBs to PROSIM, the code of the software had to be adapted to the new method of using global variables in the data pool of PROSIM. For this purpose a program (src2prosim) has been written, which replaces the #DATAPOOL statements with respective data definitions by the /COMMON/ definitions of PROSIM.

In appendix A a list of models and model software, transported to PROSIM, is given.

5.2 Visualisation models

For visualisation of the status and motion of the space structure model, the original HSFP geometry models were used. The model software has been adapted to the Indigo2 workstation environment. The model parameters have not been changed.

Except for some minor problems the visualisation software could easily be ported (small changes and recompilation) to the new environment. Only the communication interface software had to be changed.
The coupling between visualisation and dynamics model software is implemented via TCP/IP. For this reason the platform dependent routines VHINIT, VHSEND, and VHSTOP had to be replaced. For the dynamics interface software the visualisation interface routines from the TCP/IP library were slightly adapted and reused. The necessary data are assembled via a buffer and sent from the HSFP taskgroup module via TCP/IP to the visualisation module.

In figure 3 a picture of the whole "space" configuration is presented.

5.3 User interface

As already mentioned above, the HSFP hand controllers could not yet be made available. These hand controllers directly interface to the Onboard software model (see Fig. 2).
Also the MMI software was not available, and re-development of the HSFP MMI software is much work.

In order to solve the problem of the lack of a man-machine interface or user interface to the HSFP-software, a new user interface was developed based on the PROSIM user interface tool, the Tcl/Tk software package. This package is also used in EUROSIM for preparation and control of the simulation.

All model software interfaces were checked on relevance in manipulator control and related to the limited implementation configuration. Some of the variables were set, others were connceted to the user interface of the simulation control interface. For instance, most of the hand controller inputs were represented by activation buttons on the new user interface.

In figure 4 the initial user interface of simulation control is presented. All control buttons can be recognised.

6 Integration and demonstration

6.1 Overall architecture

The architecture of the real-time simulator software (which includes the HSFP model software) includes the following main components:

- Real-time simulator, including dynamics and onboard software models of the HSFP.
- Visualisation models.
- Simulator control user interface.

These components run under a multi-tasking system, since they are running independently. The advantage of such a structure is that each of the processes can run on a different machine. The three processes can also run on one machine, what has been demonstrated on the workstation (Indigo2) of the teleoperations and Robotics laboratory.
In figure 5 an example is given of a distributed implementation of the software configuration.

6.2 Scheduling of dynamics and onboard software models

The scheduling of the HSFP model software was based on a two-processor system, the IPU and the CPU. All dynamics high speed (50 Hz) routines run on the IPU. On the CPU different onboard software model routines were running on lower frequences.
These separate scheduling of model software on the HSFP had to be implemented on the single processor based Indigo2 with PROSIM. The scheduling mechanism of PROSIM is different from the original HSFP scheduling system. For this mechanism the so-called Sub Frames and Activation Frames of the HSFP processors have been identified (see appendix B). On the basis of the Activation Frames, the scheduling frequencies of the various subframes are determined. Next, a number of Sub Frames has been adapted due to the absence of some Building Blocks.

In appendix C the composition of the software groups is given based on Sub Frames. These groups of software modules are defined in the TaskGroup_HSFP.c module. This module defines the groups that can be scheduled independently in PROSIM.
In addition, the way how to schedule the groups can be identified and implemented in the SCHEDULE-FILE of PROSIM. It should be noted that the dynamics group has the highest priority in the detailed scheduling procedure.

7 Conclusion and future activities

The NLR real-time simulation package PROSIM has been included in the NLR Teleoperations and Robotics laboratory as an additional tool to support research and development in the area of tele-operations and telerobotics. EUROSIM was initially proposed as the appropriate real-time simulation environment, but this nationally developed and funded simulation package could not yet be installed.

The HSFP model sofware has been re-used as the the application software. The HSFP-software was chosen because of the NLR experience with the software and because of the relation to future conversion to the ERA model sofware, which is expected to be performed on parameter level. Not all HSFP model software (dynamics and onboard software) has been transferred, but enough to enable a complete demonstration.

The transfer of visualisation software from the original HSFP IRIS workstation to the robotics laboratory workstation, the Indigo2, appeared to be a rather simple.

Since the HSFP MMI software was implemented on a dedicated workstation, a new user interface was developed making use of the PROSIM simulation control interface software, Tcl/Tk. With this package it is possible to simulate both the HSFP handcontroller and the MMI controller input.

The visual output of the system was coupled to the Virtual Reality system of the laboratory. so, it has been used to create a virtual environment for a user in an artificial viewpoint in space, in the environment of the HERA/Hermes system.

One of the tasks originally planned could not yet be executed. The intention was to couple the simulation output to the controller of the NLR Teleoperations and Robotics laboratory. The goal was to control a simulated space robot, HERA, and to simulate (scaled) in hardware the motion of its end effector. It was not possible to implement this concept. This task is planned to be performed in 1996.
Another future application in this area can be found in the LEDA programme. The laboratory robot at NLR can be used very well for hardware simulation of the manipulator onboard the Lunar lander. This manipulator will be used for sample replacement and storage.

8 References

1. Dam, A.A. ten; Schrap, P.; Brouwer, W., Programme and Real-time Operations Simulation Support Tool PROSIM, National Aerospace Laboratory NLR, NLR TP 94524 L, 941118.
2. Peuter, W. de; Putz. P.; Visentin. G., Space A&R Controller Development, ESA/ESTEC, 1995 Plenary Meeting of the ESA Advisory Group on A&R (AGAR), Noordwijk, May 17, 1995.
3. Dieleman, P.; Pronk, C.N.A., Space Automation and Robotics for Microgravity Experiment Applications; Preliminary Identification of Research and Development Topics, National Aerospace Laboratory NLR, NLR CR 90400 L, August 1990.
4. Elburg, H.L. van, Test and Verification Equipment Test Software, PROSIM Acceptance Test, TVE-NL-95-044, National Aerospace Laboratory NLR, NLR CR 95086 L, 950216
5. Cazemier, R.J.; Couwenberg, M.J.H., Software Description, Installation, Maintenance, and User's Manual of ISMuS v1.0, National Aerospace Laboratory NLR, NLR TR 95146 L, 950720.
6. Guerin, M.; Peuter, W. de, Advisory Group for space Automation & Robotics, ESA/ESTEC, Third Meeting of ESA Advisory Group on A&R (AGAR), Noordwijk, May 17, 1995.
7. Pronk, C.N.A.; Dieleman, P., Space Automation and Robotics in Support of Microgravity Experiment Applications; Research and Activities, National Aerospace Laboratory NLR, NLR CR 91003 L, 901219.
8. Pronk, Z., Robotica, Ruimte Robot Technologie, NIVR-NRT Projectvoorstel 1995, Contract 02501N, Versie 2, National Aerospace Laboratory NLR, NLR Memorandum RS-95-021, 950518.
9. Welman, C., HERA Simulation Facility Pilot, HSFP Simulation Data Document, National Aerospace Laboratory NLR, NLR TR 88118 L, 890720.

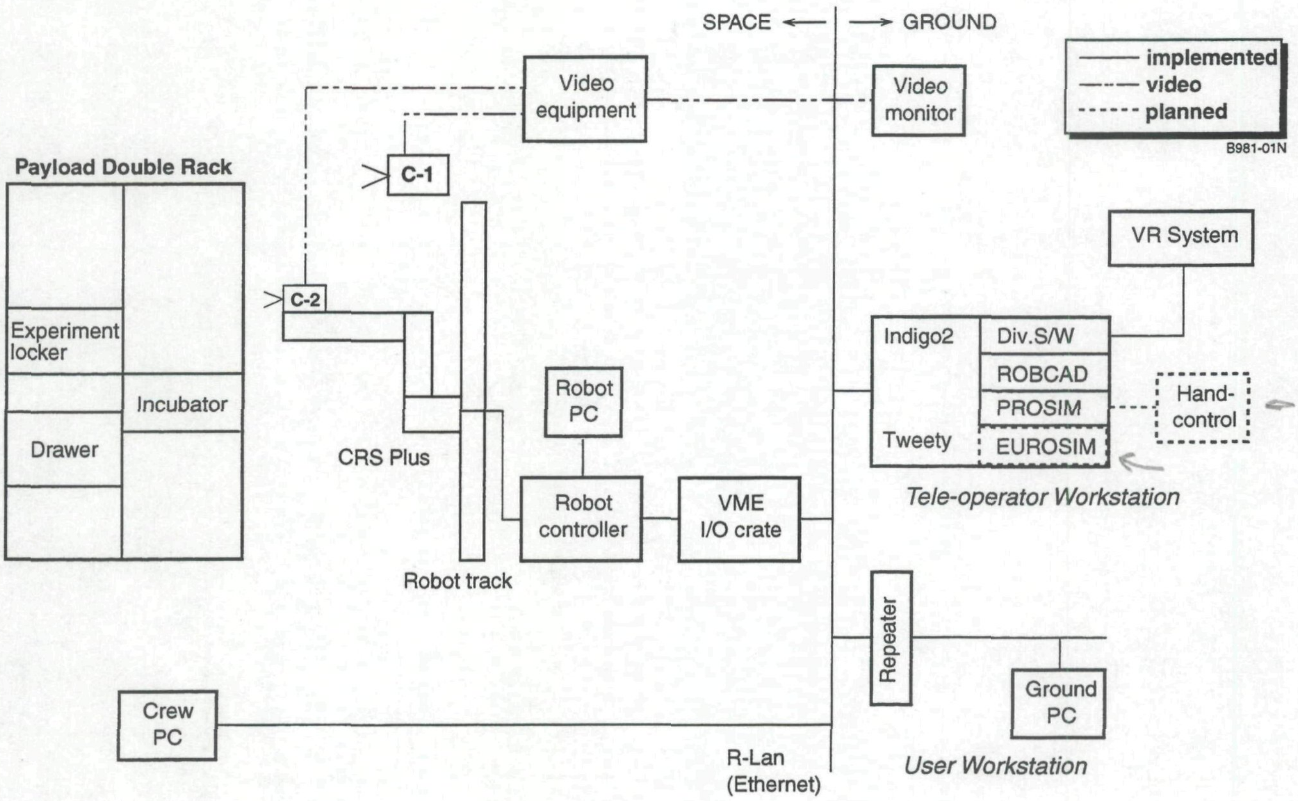

Fig. 1 Basic elements of the NLR Teleoperations and Robotic laboratory

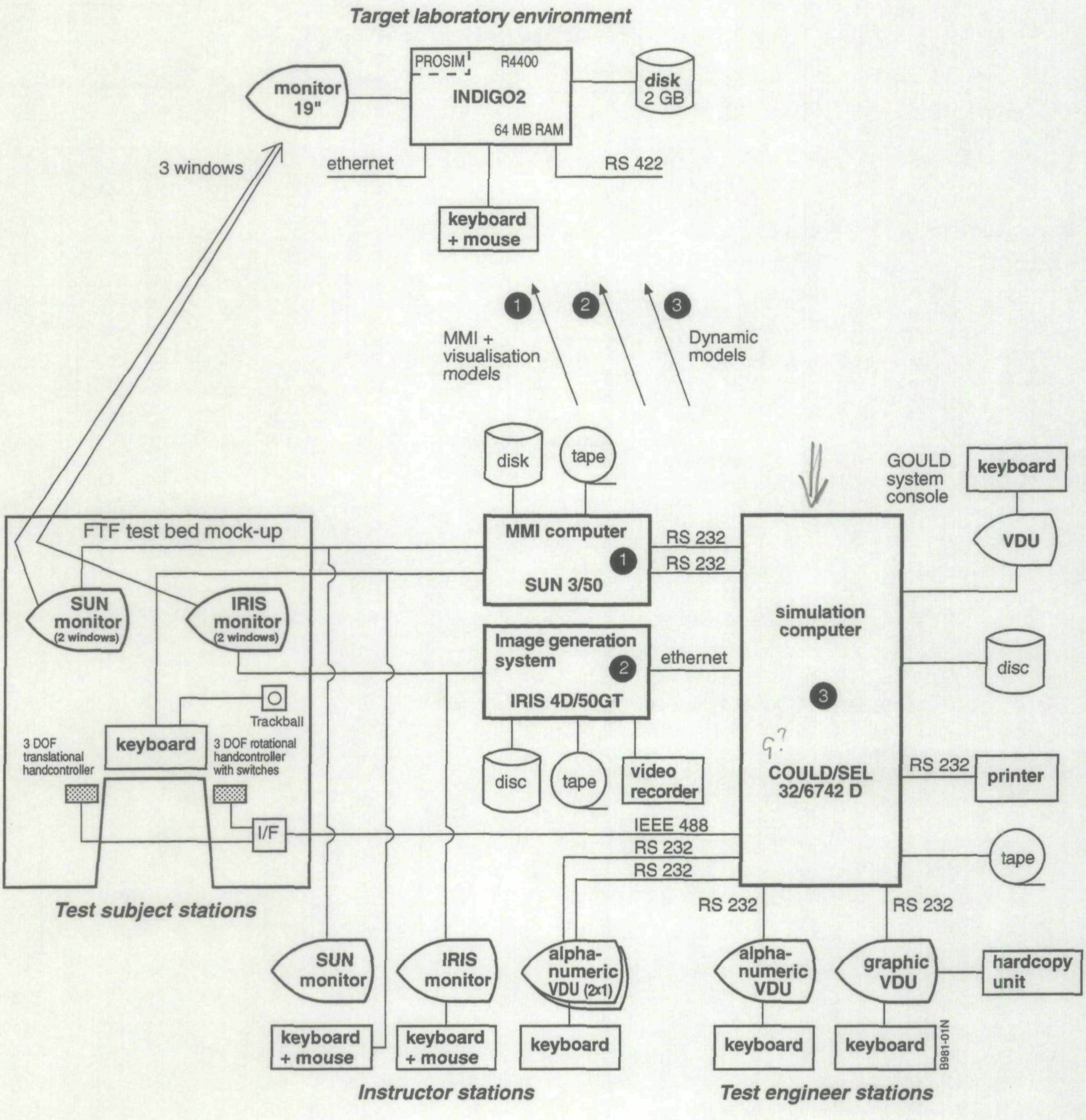

Fig. 2 Transfer paths of software from the original HSFP configuration to the target laboratory environment

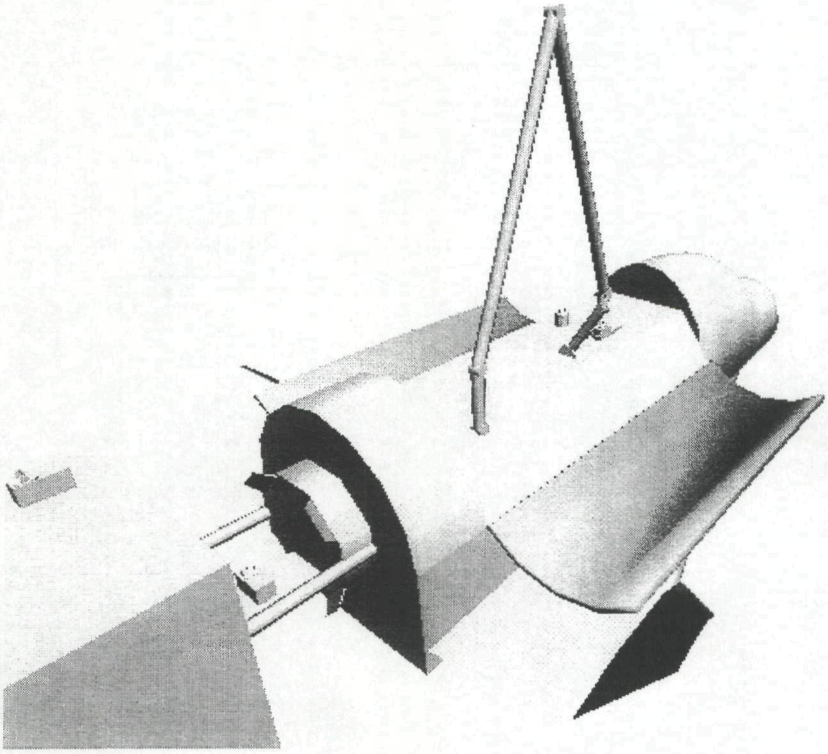

Fig. 3 Graphics simulation of the Hermes/HERA configuration

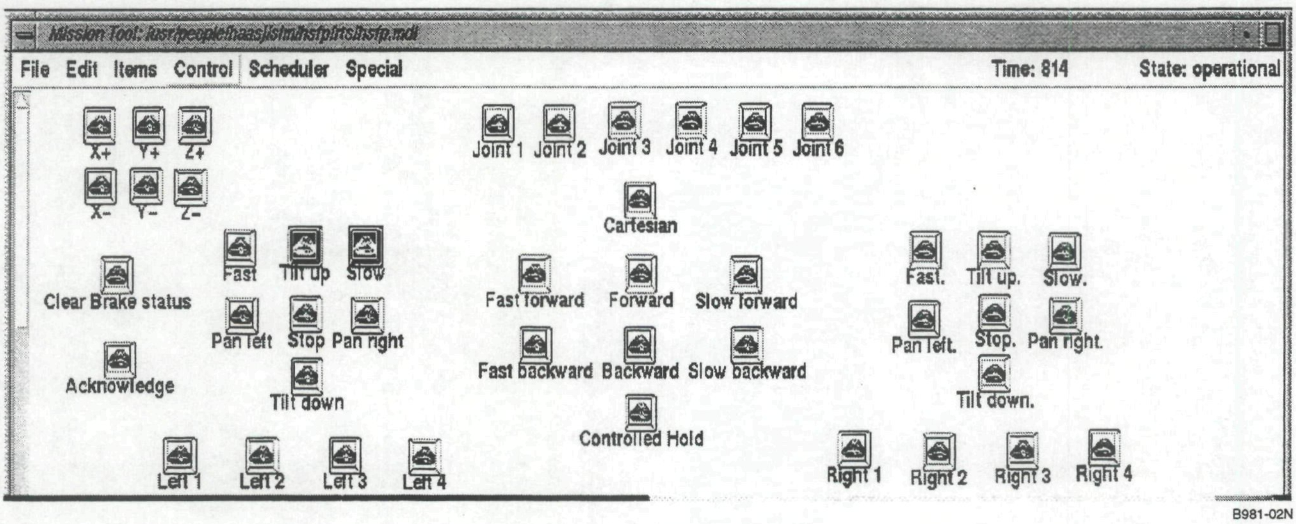

Fig. 4 The user interface for simulation and operations control

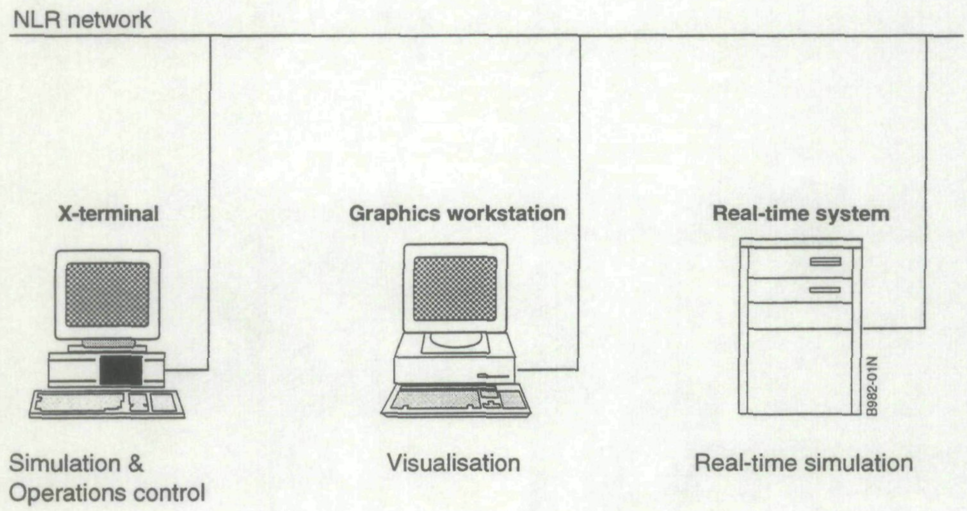

Fig. 5 Distributed implementation of the simulator

Appendix A Selected HSFP model software

A.1 Dynamics models

In the HSFP-software the following BBs were used for HERA dynamics simulation:

DYNDUM	Dymmy dynamics subroutine (empty)
INTEGR	Integration dynamic state
INTEGZ	INTEGR without joint stiction or backdrive
LOWCAL	Low-frequency calculations
MADMCC	Desired motor currents calculation
MAJVLC	Joint velocity calculation (MARS)
MAMOTA	Motor tachometer output
MAPSNC	Position encoder output
POSENC	Position encoder model
TACHOM	Tachometer model
TRADAT	Buffer switching dynamics
TSTIDL	Empty building block

These BBs run on the IPU processor of the Gould/Sel machine, while the other BBs (for the Onboard Software (OBS)) run on the CPU processor.

Next to these BBs a number of stand-alone programs were developed. During simulations these models were standby and upon particular events for which the simulator had to be stopped in order to reconfigure the model sofware, the routines were activated. These events represent mechanical changes and reconfigurations in the operations by e.g. grappling, latching, payload releasing, etc. These routines are:

CAPTUR	Simulates capture procedures
GRAPPL	Simulates procedures for Grapple/Release
ITBOOM	Inspection tool boom status setting
LATREL	Simulates stsuts of Latch-Release procedures
RELEAS	Simulates Release procedures

From all above mentioned BBs most have been copied to PROSIM, except for TSTIDL and DYNDUM. The latter routines are empty routines that are not relevant for PROSIM. In addition, the routines for grappling, latching, etc. have not been copied, since reconfiguration of software is not possible within PROSIM without problems.

A.2 Onboard software models

For modelling of the HERA onboard software the following BB's were used in the HSFP-software:

ACSCTC	Hermes ACS control
CAMCON	Camera control
CAMERA	Camera simulation
CHINIT	Start recording condensed results
CHSEND	Condensed results data recording
CHSTOP	Stop recording condensed results
CRTJNT	Joint velocty calculation (TEC)
DATOBU	Double buffer data access
EHEXEC	Event Handler
EHINIT	Start event handler
HCCART	Handcontroller command interpretation
HERSEN	Hermes sensor simulation
JOISET	Initial joint settings
MMHCIP	Handcontroller input processing
MMIRDE	Stop read task MMI interface
MMIRDI	Start read task MMI interface
MMIWRE	Stop write task MMI interface
MMIWRI	Start write task MMI interface
OBJECT	Object simulation (dynamics)
OBJINI	Object initialisation (dynamics)
OBSINI	Initialization OBS
OHIDAT	Output handler
PRECDC	Calculation end-effector dep.vars.
PREPRO	Dynamics preprocessin
PRESEN	Pressence sensor model
PRFGFF	Processing external velocity files
TBINIT	Initialise handcontrollers IEEE
TBSTOP	Stop handcontrollers IEEE
TECOPY	Buffer switching (OBS)
TEFDKM	Forward kinematics calculation
TEGIMC	Generalized inertia matrix calculation
TEJCGD	Joint control gains determination
TEPRCL	Proximity information calculation
TSMEPR	Message processing
TSSEIP	Sensor information processing (TSS)
VHINIT	Start visualisation handler

VHSEND Send data to visualisation system

VHSTOP Stop visualisation handler

From these BBs the following are not copied:

CHINIT Start recording condensed results

CHSEND Condensed results data recording

CHSTOP Stop recording condensed results

Condensed data recording and storage are not executed, because this program makes use of dedicated HSFP platform dependent implementations.

EHEXEC Event Handler

EHINIT Start event handler

The event handler is an integrated part of PROSIM. Events can be specified with the so-called Mission Definition Language (MDL).

MMHCIP Handcontroller input processing

Since there are no handcontrollers, the input signals will be simulated.

MMIRDE Stop read task MMI interface

MMIRDI Start read task MMI interface

MMIWRE Stop write task MMI interface

MMIWRI Start write task MMI interface

These BBs will start tasks similar to CAPTUR etc. These tasks read and write MMI information. At this moment they are not used.

OHIDAT Output handler

PROSIM has an integrated output handler, in a similar way as the event handler, on the basis of MDL.

TBINIT Initialise handcontrollers IEEE

TBSTOP Stop handcontrollers IEEE

There are no handcontrollers active.

VHINIT Start visualisation handler

VHSEND Send data to visualisation system

VHSTOP Stop visualisation handler

These BBs are platform dependent and are therefore replaced by new BBs for the SG Indigo2.

Appendix B Activation and Sub Frames for HSFP scheduling

```
BUILD,SUB,TSKINI,FIX
SEL,PREPRO
SEL,JOISET
SEL,TRADAT
SEL,LOWCAL
SEL,TRADAT
SEL,OBJINI
SEL,EHINIT
SEL,PRECDC
SEL,TEFDKM
SEL,TECOPY
SEL,TEGIMC
SEL,TEJCGD
SEL,MMIRDI
SEL,MMIWRI
END
BUILD,SUB,TSKCMP,FIX
SEL,EHEXIT
SEL,MMIRDE
SEL,MMIWRE
END
BUILD,SUB,RTCINR,FIX
SEL,TBINIT
SEL,OBSINI
SEL,PRFGFF
SEL,OHIDAT
SEL,VHINIT
SEL,CHINIT
END
BUILD,SUB,RTCENR,FIX
SEL,TBSTOP
SEL,VHSTOP
SEL,CHSTOP
END
BUILD,SUB,RTIINR,FIX
SEL,DYNDUM
END
BUILD,SUB,RTIENR,FIX
SEL,DYNDUM
END
BUILD,SUB,HVCAL,FLO
SEL,LOWCAL
END
BUILD,SUB,OBSLO,FLO
SEL,TEGIMC
SEL,TEJCGD
END
BUILD,SUB,CPUSF1,FIX
SEL,MMHCIP
SEL,TSMEPR
SEL,HCCART
SEL,CRTJNT
SEL,EHEXEC
END
BUILD,SUB,CPUSF2,FIX
SEL,DATOBU
SEL,CHSEND
SEL,EHEXEC
END
BUILD,SUB,CPUSF3,FIX
SEL,CAMCON
SEL,CAMERA
SEL,OBJECT
SEL,DATOBU
SEL,VHSEND
SEL,EHEXEC
END
BUILD,SUB,CPUSF4,FIX
SEL,CAMCON
SEL,CAMERA
```

```
SEL,OBJECT
SEL,EHEXEC
END
BUILD,SUB,CPUSF5,FIX
SEL,TEFDKM
SEL,EHEXEC
END
BUILD,SUB,CPUSF6,FIX
SEL,TEFDKM
SEL,TECOPY
SEL,EHEXEC
END
BUILD,SUB,CPUSF7,FIX
SEL,EHEXEC
END
BUILD,SUB,CPUSF8,FIX
SEL,TEPRCL
SEL,TSSEIP
SEL,EHEXEC
END
BUILD,SUB,CPUSF9,FIX
SEL,PRESEN
SEL,HERSEN
SEL,ACSCTC
SEL,EHEXEC
END
BUILD,SUB,DUMMY,FIX
SEL,DYNDUM
END
BUILD,SUB,IPUSF1,FIX
SEL,POSENC
SEL,TACHOM
SEL,MAPSNC
SEL,MAJVLC
SEL,MAMOTA
SEL,MADMCC
SEL,INTEGZ
SEL,TACHOM
SEL,MAMOTA
SEL,MADMCC
SEL,INTEGR
SEL,TSTIDL
END
BUILD,SUB,IPUSF2,FIX
SEL,POSENC
SEL,TACHOM
SEL,MAPSNC
SEL,MAJVLC
SEL,MAMOTA
SEL,MADMCC
SEL,INTEGZ
SEL,TACHOM
SEL,MAMOTA
SEL,MADMCC
SEL,INTEGR
SEL,TRADAT
SEL,TSTIDL
END
BUILD,ACT,TSKINI
SEL,TSKINI
END
BUILD,ACT,TSKCMP
SEL,TSKCMP
END
BUILD,ACT,CPU
SEL,RTCINR
SEL,CPUSF1
SEL,CPUSF2
SEL,CPUSF3
SEL,CPUSF6
SEL,OBSLO
SEL,CPUSF8
SEL,CPUSF1
SEL,CPUSF2
SEL,CPUSF4
SEL,CPUSF5
SEL,CPUSF9
SEL,CPUSF1
```

SEL,CPUSF2
SEL,CPUSF3
SEL,CPUSF5
SEL,CPUSF7
SEL,CPUSF1
SEL,CPUSF2
SEL,CPUSF4
SEL,CPUSF5
SEL,CPUSF7
SEL,CPUSF1
SEL,CPUSF2
SEL,CPUSF3
SEL,CPUSF5
SEL,CPUSF7
SEL,CPUSF1
SEL,CPUSF2
SEL,CPUSF4
SEL,CPUSF5
SEL,CPUSF8
SEL,CPUSF1
SEL,CPUSF2
SEL,CPUSF3
SEL,CPUSF5
SEL,CPUSF7
SEL,CPUSF1
SEL,CPUSF2
SEL,CPUSF4
SEL,CPUSF5
SEL,CPUSF7
SEL,CPUSF1
SEL,CPUSF2
SEL,CPUSF3
SEL,CPUSF5
SEL,CPUSF7
SEL,CPUSF1
SEL,CPUSF2
SEL,CPUSF4
SEL,CPUSF5
SEL,CPUSF7

SEL,RTCENR
END
BUILD,ACT,IPU
SEL,RTIINR
SEL,IPUSF1
SEL,IPUSF1
SEL,HVCAL
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1

```
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF1
SEL,IPUSF2
SEL,RTIENR
END
GENTASK
SEL,ALL
END
QUIT
```

Appendix C Scheduling of software modules

For scheduling the following groups have been composed on the basis of Sub Frames:

HSFPINIT:
- *PREPRO*
- *JOISET*
- *TRADAT*
- *LOWCAL*
- *TRADAT*
- *OBJINI*
- *PRECDC*
- *OBSINI*
- *PRFGFF*
- *TEFDKM*
- *TECOPY*
- *TEGIMC*
- *TEJGCD*
- *VHINIT*

DYNAM:
- *POSENC*
- *TACHOM*
- *MAPSNC*
- *MAJVLC*
- *MAMOTA*
- *MADMCC*
- *INTEGZ*
- *TACHOM*
- *MAMOTA*
- *MADMCC*
- *INTEGR*

HVCAL:
- *TRADAT*
- *LOWCAL*

HCIN:
- *TSMEPR*
- *HCCART*
- *CRTJNT*

OUTPUT
- *DATOBU*

CAMERA:
- *CAMCON*
- *CAMERA*
- *OBJECT*
- *DATOBU*
- *VHSEND*

FWK:
- *TEFDKM*

GAIN:
- *TECOPY*
- *TEGIMC*
- *TEJCGD*

PROX:
- *TEPRCL*
- *TSSEIP*

PRES:
- *PRESEN*
- *HERSEN*
- *ACSCTC*

The above mentioned groups of software modules are defined in the TaskGroup_HSFP.c module. This module defines the groups that can be scheduled independently in PROSIM.
In addition, the way how to schedule these groups can be identified and implemented in the SCHEDULE-FILE of PROSIM. It should be noted that the dynamics group has the highest priority in the detailed scheduling procedure.

```
BASIC_FREQUENCY   50

RT_PROCESSORS     1

STATE CONFIGURED
  STRONGLY_PERIODIC    EventManager PERIOD 1
  CYCLIC                    SetData

STATE RESET_CONDITION
  STRONGLY_PERIODIC    EventManager PERIOD 1
  CYCLIC                    SetData

STATE INITIAL_CONDITION
  STRONGLY_PERIODIC    EventManager PERIOD 1
  CYCLIC                    SetData
  CYCLIC                    HSFPINIT

STATE REAL_TIME
  STRONGLY_PERIODIC    EventManager     PERIOD 1
  STRONGLY_PERIODIC    DYNAM            PERIOD 1
  PERIODIC             HVCAL     PERIOD 50 OFFSET 2
  PERIODIC             HCIN      PERIOD 5
  PERIODIC             OUTPUT           PERIOD 5 OFFSET 1
  PERIODIC                  CAMERA            PERIOD 5 OFFSET 2
  PERIODIC                  FWK               PERIOD 5 OFFSET 3
  PERIODIC                  GAIN              PERIOD 50 OFFSET 4
  PERIODIC                  PROX              PERIOD 50 OFFSET 4
  PERIODIC                  PRES        PERIOD 50 OFFSET 8

STATE STOPPED
  STRONGLY_PERIODIC    EventManager PERIOD 1
```

STATE EMERGENCY
STRONGLY_PERIODIC EventManager PERIOD 1

STATE FROZEN
STRONGLY_PERIODIC EventManager PERIOD 1

The OFFSET definitions in the schedule file are meant to implement a sequence in the scheduling order.