Experimental Device for In-Torus Handling
- EDITH Intermediate Report -

A. Suppan, R. Krieg, H.G. Krumm,
U. Kühnapfel, K. Leinemann, J. Reim, J. Woll

Hauptabteilung Ingenieurtechnik
Projekt Kernfusion

Kernforschungszentrum Karlsruhe

KERNFORSCHUNGSZENTRUM KARLSRUHE

Hauptabteilung Ingenieurtechnik

Projekt Kernfusion

KfK 5252

Experimental Device
for In-Torus Handling

- EDITH Intermediate Report -

A. Suppan, R. Krieg, H.G. Krumm,
U. Kühnapfel, K. Leinemann, J. Reim, J. Woll

The Experimental Device for In-Torus Handling (EDITH) is based on an articulated boom system (ABS), consisting of a support structure, the articulated boom transporter (ABT), the end-effector positioning unit (EEPU) and different end-effectors (EE's). It is the prototype of an in-vessel handling system for NET/ITER. In combination with a Full Scale Mock-up, EDITH is required to demonstrate that maintenance of plasma facing components can be carried out with the anticipated reliability and in time. A further aim of EDITH is to allow testing of the articulated boom components and subassemblies.

In this paper the testbed EDITH and the Full Scale Mock-up are described. In addition, the performance and results of the commissioning and the qualification are summarized and an outlook is given for future tasks.

Experimentiervorrichtung für In-vessel Hantierung - Zwischenbericht

Die Experimentiervorrichtung für In-Vessel Hantierung (EDITH) basiert auf einem Vielgelenkarm-System, bestehend aus einer Stützkonstruktion, dem Vielgelenkarm-Transporter (ABT), der Endeffektor-Positioniereinheit (EEPU) und verschiedenen Endeffektoren. Sie ist der Prototyp eines In-vessel Handhabungssystems für NET/ITER. In Verbindung mit einem Mock-up im Maßstab 1:1 wird EDITH benötigt, um zu demonstrieren, daß die Instandhaltung von Komponenten, die dem Plasma einer Fusionsmaschine ausgesetzt sind, mit der erwarteten Zuverlässigkeit und in angemessener Zeit möglich ist.

In diesem Bericht sind der Teststand EDITH und das Mock-up beschrieben. Darüberhinaus sind darin die Durchführung und Ergebnisse der Inbetriebnahme und der Abnahmetests zusammengefaßt und ein Ausblick auf zukünftige Aufgaben gegeben.

Preface

Work documented in this report was performed within the frame of the Nuclear Fusion Project of the Kernforschungszentrum Karlsruhe GmbH. and supported by the European Fusion Technology Programme. It is based on the investigations and justification for a full scale prototypic articulated boom as documented in /1/.

Contributions are from:

* H.Braun

* G.Dillmann

* B.Dolensky

* M.Englert (D.T.I. Dr.Trippe Ing. Gesellschaft mbH., Karlsruhe)

* A.Fiege

* B.Haferkamp

* J.Hübener

* H.E.Holler

* H.Knüppel

* W.Köhler

* R.Krieg (co-author)

* H.G.Krumm (co-author)

* U.Kühnapfel (co-author)

* K.Leinemann (co-author)

* W.Link

* A.Ludwig

* G.Müller

* W.Müller

* J.Reim (co-author)

* H.A.Rohrbacher

* M.Salaske

* E.G.Schlechtendahl

* U.Schygulla

* A.Suppan (editor)

* J.Woll (co-author)

Contents

Figures

Tables

List of Abbreviations

ABS	Articulated Boom System
ABT	Articulated Boom Transporter
BHU	Blanket Handling Unit
C, C++	C, C++ Programming Language
CAD*I	CAD Interface (Data Format for CAD Data Transfer)
CAT	Computer Aided Teleoperation
CCSS	Configurable Communication SubSystem
CCTV	Closed-Circuit TeleVision
CU	Control Unit
DH	Divertor Handling
DHU	Divertor Handling Unit
DOIF	Direct Operating InterFace
DOIF-C	DOIF for Camera system
DOIF-E	DOIF for EDITH control system
DOIF-SS	DOIF SubSystem
EAC	EDITH Area Controller
EDITH	Experimental Device for In-Torus Handling
EE	End-Effector
EEPU	End-Effector Positioning Unit
EMSM	Electric Master/Slave Manipulator
ERCR	EDITH Remote Control Room
ERWS	EDITH Remote Workstation
FDDI	Fibre Distributed Data Interface
FEMOS	Function Editing,siMulation, execution, and mOnitoring System
FPCS	Fusion Plant Control System
IPC	Unix Inter-Process Communication
IRDATA	Industrial Robot DATA transfer format
ISO	International Standardization Organisation
ISO/DIS	ISO Draft International Standard
ISO/OSI	ISO Open System Interconnection
ITER	International Tokamak Experimental Reactor
IVH	In-Vessel Handling
IVHS	In-Vessel Handling System
IVHU	In-Vessel Handling Unit
IVT	In-Vessel Transporter
IVTS	In-Vessel Transport System
IVVS	In-Vessel Vehicle System
JET	Joint European Torus
KISMET	KInematic Simulator, Monitor, and programming Environment for Tele-manipulation
MAP	Manufacturing Automation Protocol
MCS	Motion Control System
MHU	Manipulator Handling Unit
MMI	Man-Machine Interface
MMIM	Man-Machine Interface Management
MMS	Manufacturing Message Specification
MPU	Manipulator Positioning Unit
MSM	Master Slave Manipulator
MSSM	Master-Slave ServoManipulator
NET	Next European Torus
NRWS	NET Remote handling Work Station
PEXOS	Procedure Editing,simulation,eXecution, and mOnitoring System

POI	Point of Interest
POR	Point of Reference
PTP	Point To Point
RH	Remote Handling
RHCR	Remote Handling Control Room
RHCS	Remote Handling Control System
RHWS	Remote Handling WorkStation
TARM	Telescopic Articulated Remote Mast
TBD	to be defined
TBU	to be updated
TCP	Tool Center Point
TFTR	Tokamak Fusion Test Reactor at Princeton
TH	Tile Handling
THU	Tile Handling Unit
TIF	Test InterFace of controllers
UIMS	User Interface Management System
UNIX	Workstation Operating System from AT&T
VS	Viewing System
VV	Vacuum Vessel
WFRM	Work FRaMe
WPHS	Work Phase Support
WU	Work Unit (z.B. divertor handling unit)
2D	two-dimensional
3D	three-dimensional

1.1 Objectives of this Document

Objectives of this document are

* to describe EDITH , its components and subassemblies,

* to show the present status of the testbed EDITH and the mock-up,

* to summarize the preliminary insvestigations and the results of the qualification tests performed until now,

* to identify areas of further investigations and to outline the future test programme.

1.2 Requirements for EDITH

EDITH is a prototypic in-vessel transporter (IVT) for the demonstration of maintenance tasks and component replacement inside the torus of a fusion machine like NET/ITER as described in the "NET Predesign Report" /2/ and in "ITER Assembly and Maintenance" /3/.

The following maintenance tasks and operations will be carried out by the in-vessel transport system (IVTS):

Scheduled:

* Protective armour tiles replacement
* Radio frequency launchers replacement (TBD)
* Divertor modules replacement
* Inspection of first wall components and of vacuum vessel (VV)
* Dust vacuum cleaning (TBD)

Unscheduled:

* Active control coils replacement
* Leak detection
* Leak repairs (TBD)
* Debris recovery

It is the logical step between the proof of principal as it was already demonstrated at JET and TFTR, respectively, and a later operational equipment as it is shown in Figure 1 on page 2.

The requirements for EDITH are based on the requirements for the articulated boom transporter (ABT) which again is part of the basic equipment for in-vessel operations. The requirements differ only by the environmental conditions as shown in Table 1 on page 2.
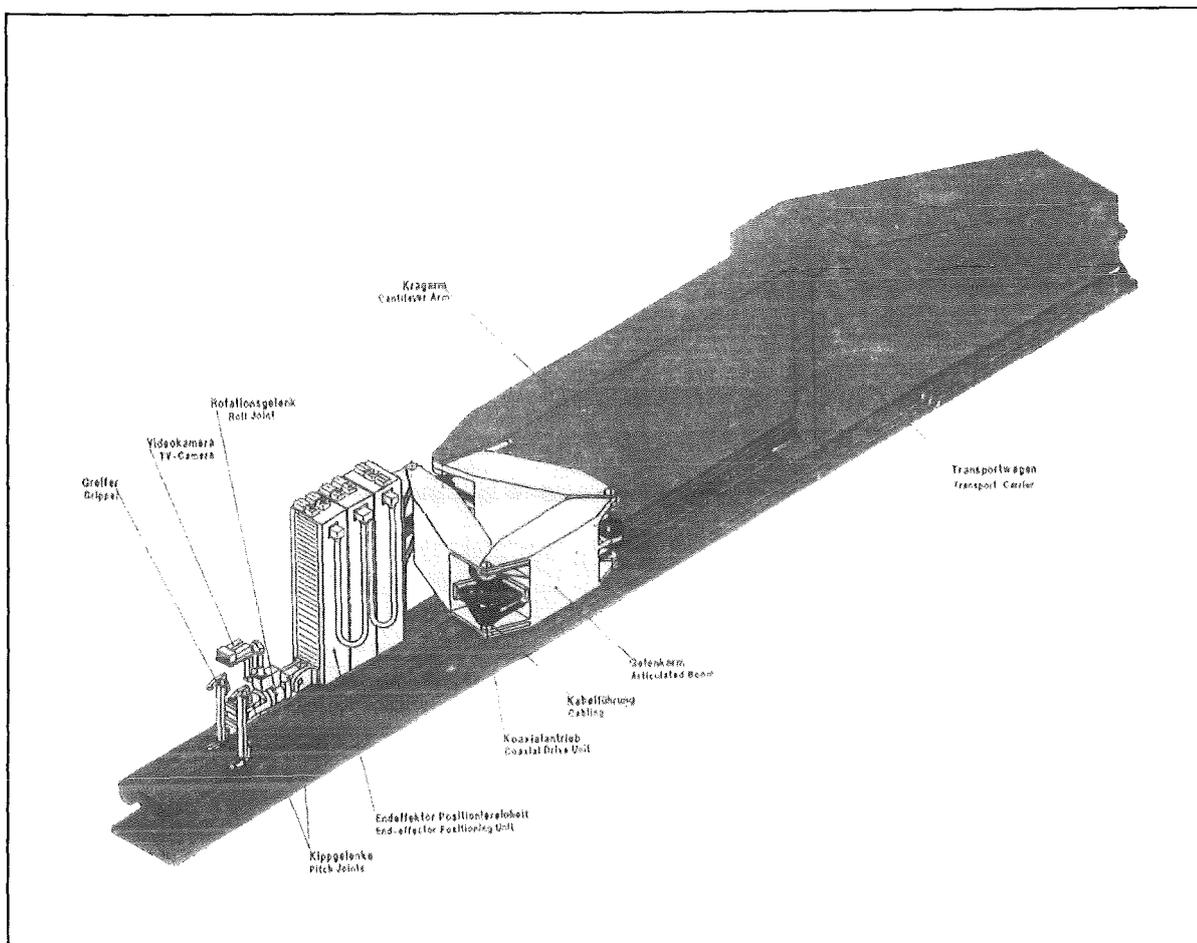
Figure 1. Concept of the In-Vessel Handling Unit (Operational Equipment)

	Environmental Conditions of Operational Equipment	Environmental Conditions of EDITH
Temperature	+ 20 to + 150°C	0 to + 30°C
Atmosphere	He (TBU)	Air
Pressure	approx. 1 atmosphere	approx. 1 atmosphere
Radiation	3×10^6 rad/h γ, negligible α and β	none
Condaminated dust	TBD	none
Magnetic field	negligible	none

Table 1. Environmental Conditions of Operational Equipment and EDITH

All the other requirements for the ABT and EDITH are summarized in Table 2 on page 3 and Table 3 on page 3. Their basis is the technical specification for NET/ITER in-vessel transporters /4/, the document "Divertor Maintenance Using IVHU" /5/ and the document "Definition of Requirements for the Design of Prototype Divertor Handling Equipment" /6/ as well as the tentative dimensions and arrangement of the NET device shown in Figure 2 on page 4 and Figure 3 on page 5.

Since the replacement of divertor modules is the most demanding operation due to the weight of the modules and the required dexterity of the handling operations this task was the driving factor and the basis for the ABT investigations and thus also for the

design of EDITH. As there is no decision from NET/ITER about a poloidal or toroidal segmentation of the modules both options were taken into account for the testbed.

Number and reach of ABT and EDITH, respectively	2 x 90° at full load capacity
Number of entry ports for ABT	2
Additional entry ports at NET/ITER for insertion and withdrawal of components and equipment.	2
Size of entry ports	H = 2952 mm, W = 1248 mm

Table 2. Geometrical Data for Articulated Boom Transporter and EDITH

	Toroidal Segmentation	Poloidal Segmentation
Number of divertor modules	32 upper/32 lower ones	32 upper/32 lower ones
Width of one module (mm)	745/1202	1861/2394
Length of one module (mm)	3462	1657 (3350 for the inboard module including coolant supply)
Height of one module (mm)	approx. 600	approx. 670
Weight of one module (kg)	approx. 1000	approx. 1000

Table 3. Maximum Dimensions and Weights of Divertor Modules
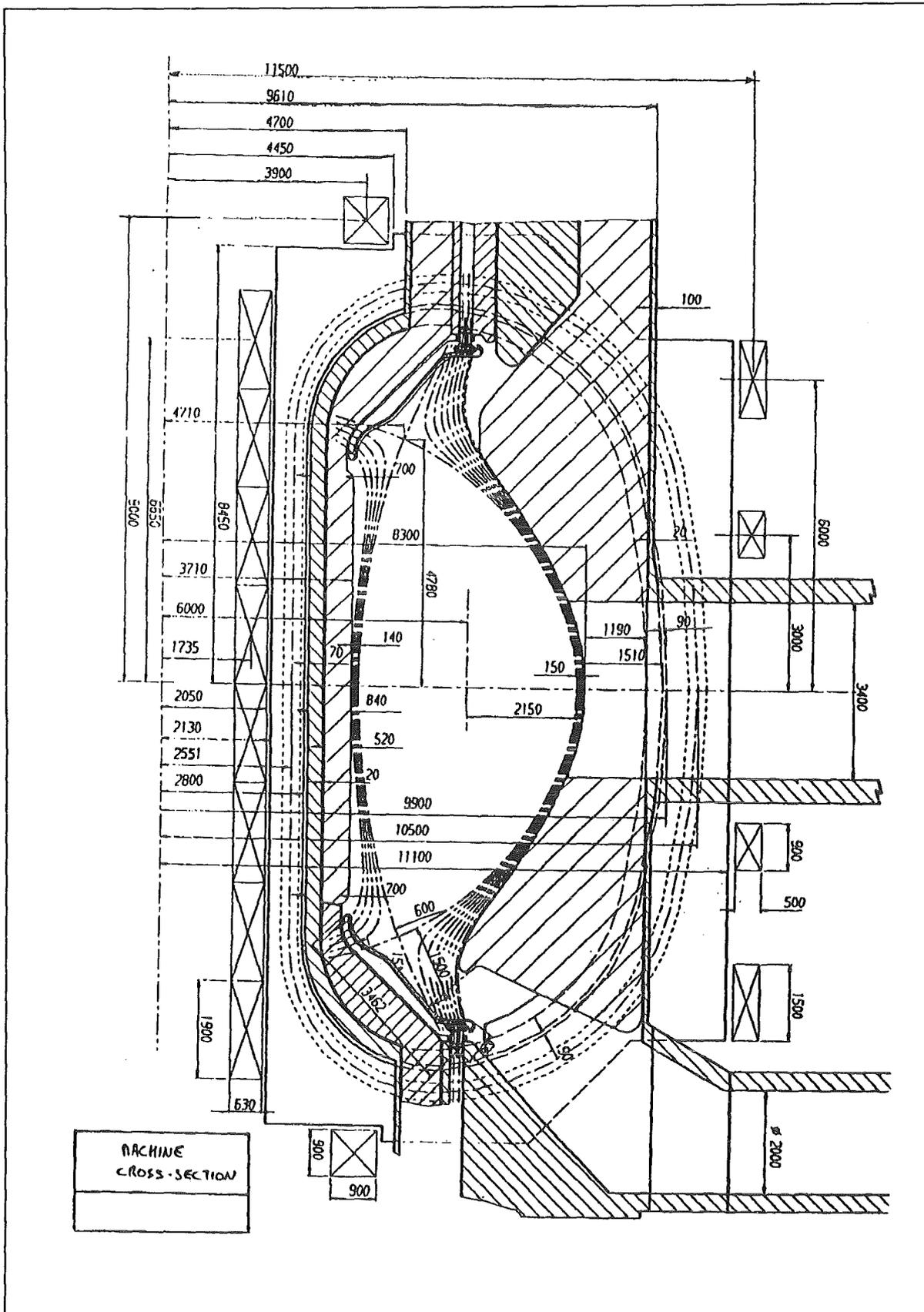
Figure 2. NET Machine Cross Section

Figure 3. NET Machine Equatorial Cross Section

1.3 Concept of EDITH and Full Scale Mock-up

Out of several design options for an IVTS, two solutions were identified which may have the capabilities to fulfill the NET/ITER requirements. One is the in-vessel vehicle system (IVVS) which is based on vehicles carrying a telescopic mast with exchangeable end-effectors and moving on rails extendable into the torus. Another one is a teleoperation system based on an articulated boom which is favoured for short term or unexpected interventions. It was investigated by KfK. Its prototype is EDITH /7/.

Figure 4 gives an overview of the testbed EDITH and Figure 5 on page 7 shows the block diagram of EDITH including the mock-up. The main sub-assemblies of EDITH are the support structure, the articulated boom linked up with the structure, the attached end-effector positioning unit with exchangeable end-effectors as divertor handling unit, manipulator handling unit and others which are not yet defined, and the control system including the viewing system.

The mock-up forms the upper or the lower half of a NET/ITER torus sector and includes typical plasma facing components. EDITH allows in combination with the mock-up the demonstration of maintenance tasks including the replacement of components as protective armour tiles and divertor plates (Figure 6 on page 8).
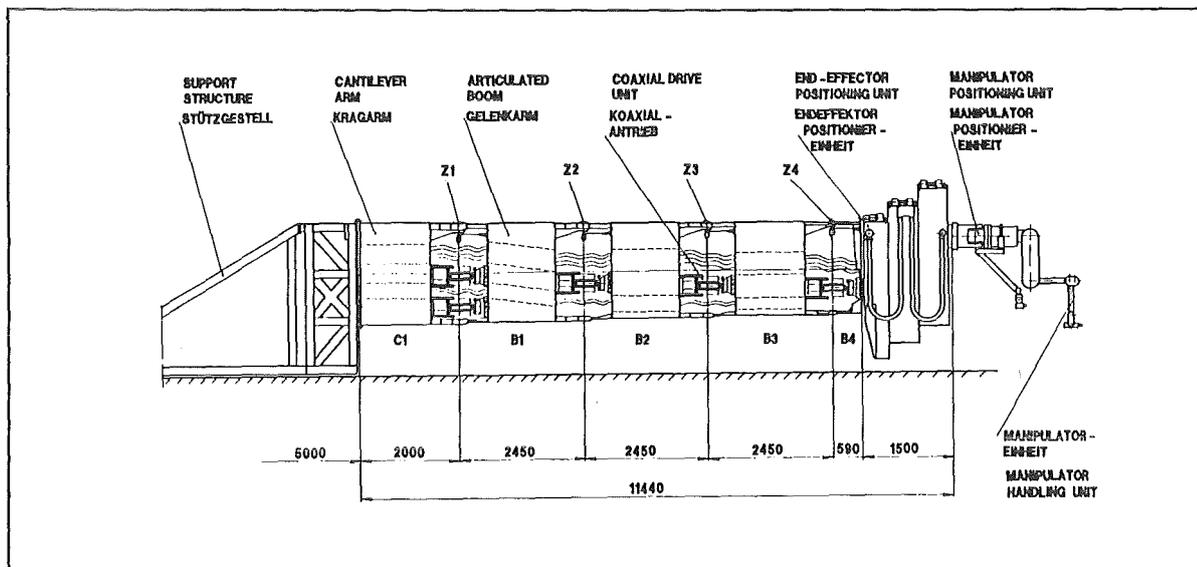
Figure 4. Testbed EDITH

2.0 Mechanics and Kinematics

As mentioned in chapter 1.3, "Concept of EDITH and Full Scale Mock-up" and according to the block diagram Figure 5 on page 7, completed by the drawing IT-OUT-08-446, the main subassemblies of EDITH are the support structure, the articulated boom transporter with the permanently attached end-effector positioning unit and the end-effector. In the present drawing and in Figure 7 on page 9 a manipulator handling unit is shown as end-effector. The total length of the ABT, including the cantilever arm which is rigidly connected with the support structure and including also the manipulator positioning unit is 12480 mm.
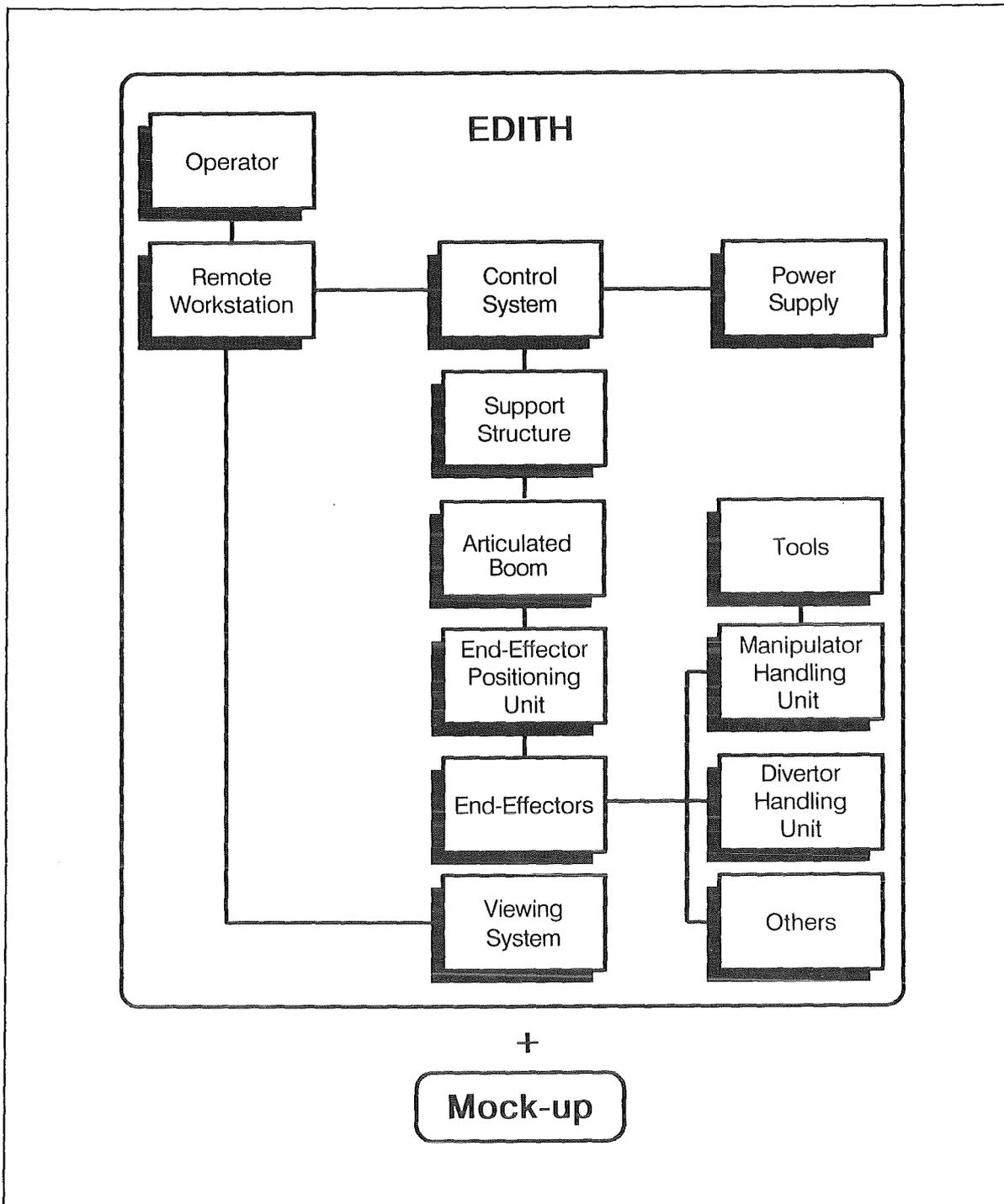
Figure 5. Block Diagram of EDITH and Mock-up

2.1 Support Structure

The support structure, shown at Drwg. IT-OUT-08-423B, is the basis for the ABT. It substitutes a radially to the NET/ITER torus movable transport carrier of the later operational articulated boom system (ABS). It consists of a girder construction which is fixed at the floor. The structure and its fixation are calculated for maximum bending moments and maximum torques of 1200 kNm. The structure material is RST.37-2.
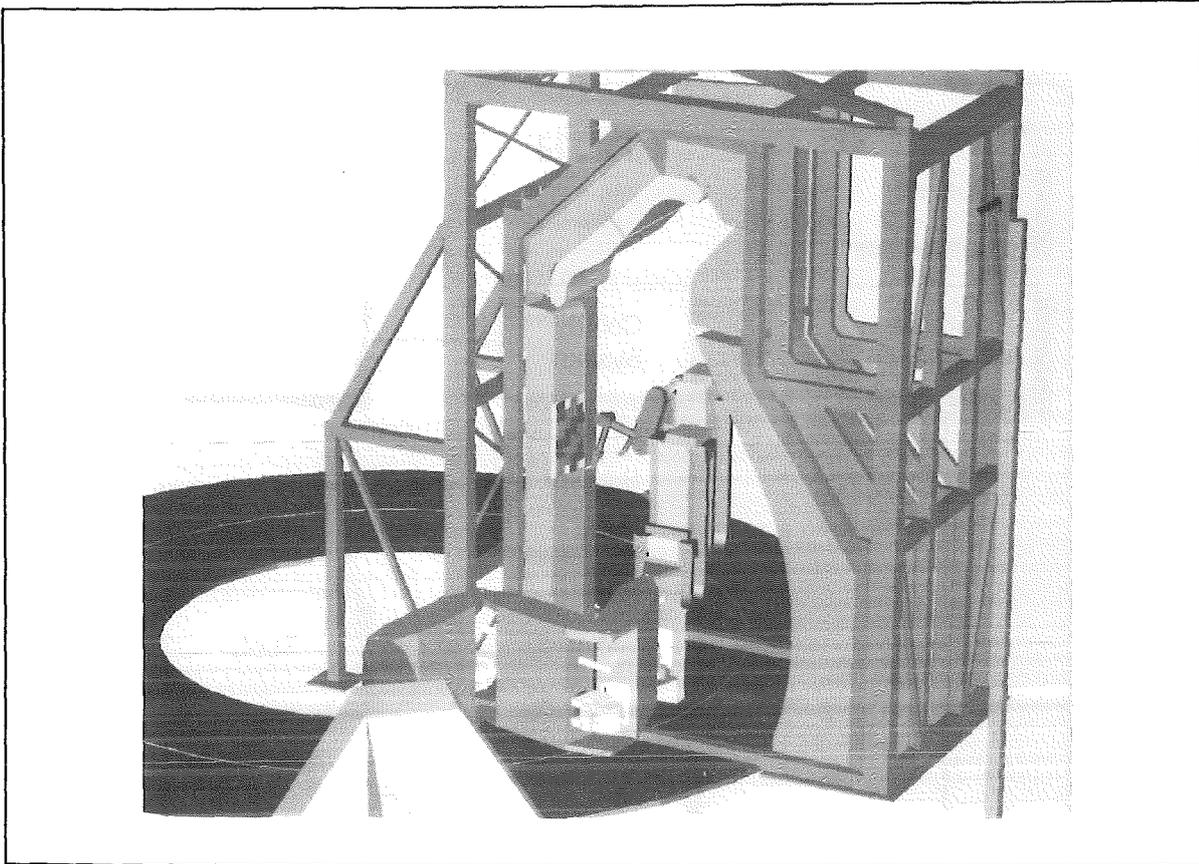
Figure 6. Articulated Boom Operating in the Upper Torus' Region. Replacement of protective armour tiles which are arranged at the inner side of the torus' mock-up.

2.2 Articulated Boom Transporter (ABT)

The EDITH ABT is shown in Drwg. IT-OUT-08-446. Its four links B1 to B4 plus the additional cantilever arm C1 are connected via the link joints Z1 to Z4. Cantilever arm and the links B1 to B3 have a box cross section, the ABT front link B4 is different from the other ones as it consists of the attachment plate with the upper and lower hinges for the link joint Z4. The height of the links is stepwise reduced from the cantilever arm (link C1) to the link B4. The dimensions of the links are listed in Table 4 on page 8. The link boxes contain cable ducts for power supply and signal transmission lines.

Link Nr. according Figure 8 on page 10	Link length (mm)	Width (mm) x Height (mm) x Wall thickness (mm)
C1	2000	600 x 2000 x 12
B1	2450	600 x 1930 x 8
B2	2450	600 x 1871 x 6
B3	2450	600 x 1825 x 5
B4	590	600 x 1858 x 15

Table 4. Articulated Boom Transporter - Link Dimensions

The links are connected by means of yaw joints which are equipped with spherical roller bearings. Arrangement and performance of a typical yaw joint are shown in Drwg. IT-OUT-08-428
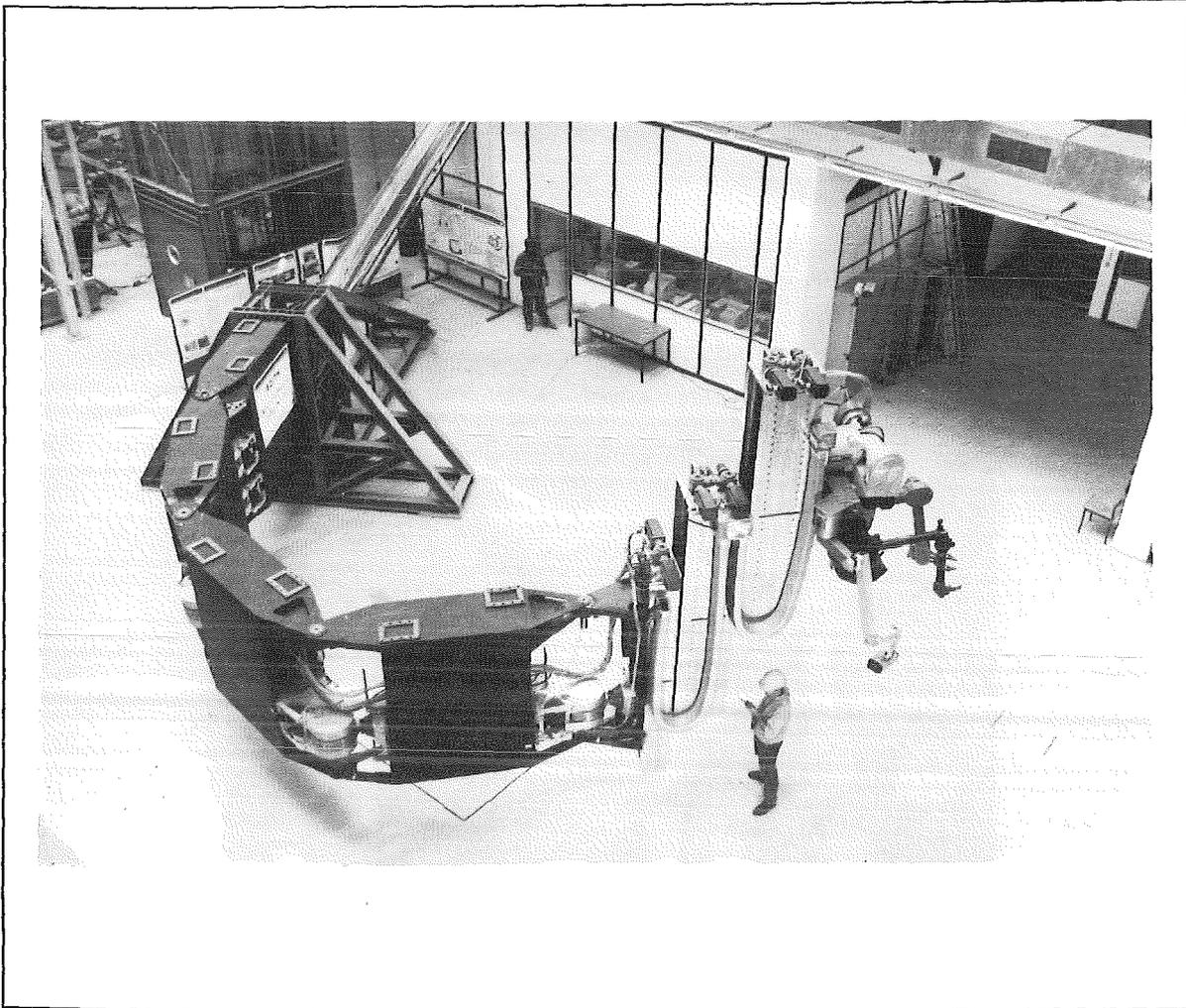
Figure 7. Testbed EDITH with Attached Manipulator Handling Unit

As shown in Figure 8 on page 10 and indicated in Drwg. IT-OUT-08-446 the yaw joints Z2 to Z4 are actuated by one coaxial drive unit each while the joint Z1 is driven by two coaxial drive units taking into consideration the required higher torques.

The link joints Z1 to Z3 have a range of operation of $\pm 115°$, link joint Z4 of ± 105 °. Their movements are controlled by control components which give either position signals of the link joints to the position control system or stop the movements of the joints via the drive control system or mechanically if the other systems which have priority fail. The sequence of priority for the control components is the following:

> At normal circumstances absolute multi-turn resolvers arranged at the upper joint axles measure the revolutions of the planetary roller spindles and signal the data to the position control system.

> Limit switches mounted at the lower link hinges will be actuated by stoppers, arranged at the complementary link hinges, if one of the resolvers fails. They stop the according drive.

> The third level are mechanical stops if both resolvers and limit switches fail.

Each coaxial drive unit (Drwg. IT-OUT-12-214) consists of identical trains of drive, composed of AC-servo motor, spur wheel gear, CYCLO drive and torque arm, acting upon the torque arm restraint. The duplication of the trains of drive results almost in a duplication of the torque and makes optionally possible a mechanical or electrical backlash removal. The main data of the drive units are given in Table 5 on page 10.
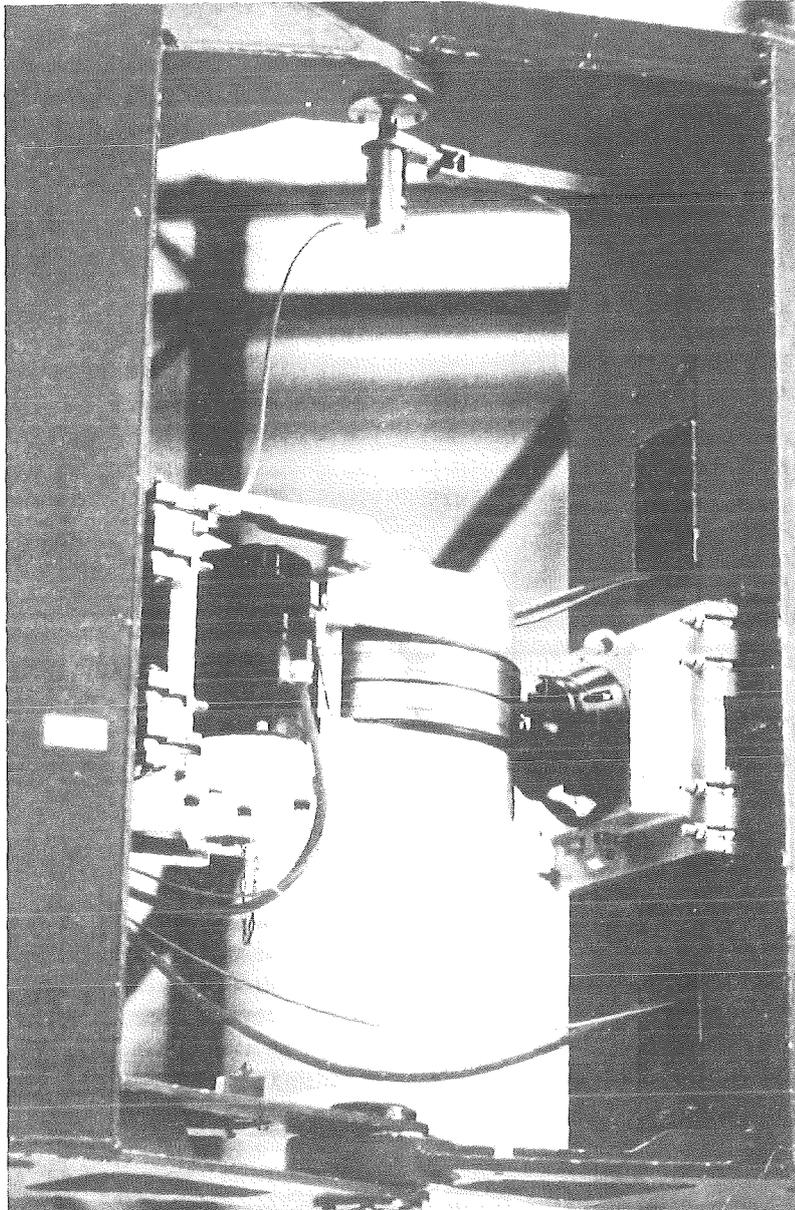
Figure 8. Yaw Joint Z2 with Installed Coaxial Drive Unit

Parameter	Value
Maximum angular joint speed (rad/sec)	0.15
Maximum acceleration (rad/sec²)	0.085
Number of gear trains (motor with failsafe brake, spur wheel gear, CYCLO drive)	2
Transmission ratio of CYCLO	89:1
Transmission ratio of spur wheel gear	5.99:1
Total transmission ratio	532.9:1

Table 5. Main Data of Coaxial Drive Units for Yaw Joints Z1 to Z4

For controlling the speed and accuracy of the ABT the torque rating, stiffness and efficiency of the drive units are of primary importance. Therefore, extensive qualification tests were performed with the drive units prior to their installation at the joints. The main results of the tests are summarized in Table 6 on page 11 and Table 7 on page 12. Figure 9 on page 13 shows the characteristic curves (motor torque vs. joint torque and efficiency vs. joint torque) of the drive unit (serial nr. 2). The curves are also typical for the other drive units. The unpublished report /8/ contains details of the test performance and results.

For the ABT a stress analysis was carried out. Its performance and validation is described in chapter 2.6, "Stress Analysis of the Articulated Boom Transporter and its Validation" on page 22.

Measurements and remarks	Parameters	Serial Nr. 0	Serial Nr. 1	Serial Nr. 2	Serial Nr. 3	Serial Nr. 4
Load tests: average values, measurements at 100% of rated speed and load	Joint speed (rad/sec)	0.153	0.156	0.153	0.152	0.154
	Total motor current (Ampere)	22.948	25.245	24.167	27.199	25.435
	Total motor torque (Nm)	32.69	35.51	34.43	38.53	35.61
	Joint torque (kNm)	9.891	10.164	10.801	12.774	10.566
	Joint efficiency (%)	56.8	54.1	59	62.3	55.6
Maximum pulse torque: average values, measurements at 100% of rated speed and at 140% of the rated load	Joint speed (rad/sec)	0.153	0.14	0.139	0.151	0.124
	Total motor current (Ampere)	31.471	33.371	30.956	34.786	30.554
	Total motor torque (Nm)	47.76	51.17	46.75	52.3	46.17
	Joint torque (kNm)	15.354	14.895	14.674	15.798	14.104
	Joint efficiency (%)	60.6	54.68	59.13	57.17	57.3

Table 6. Results of Drive Unit Qualification Tests

Measurements and remarks	Parameters	Serial Nr. 0	Serial Nr. 1	Serial Nr. 2	Serial Nr. 3	Serial Nr. 4
Back driven efficiency: average values, measurements at 100% of rated speed and load	Joint speed (rad/sec)	0.151	0.149	0.152	0.151	0.154
	Total motor current (Ampere)	5.342	5.631	5.842	5.606	6.104
	Total motor torque (Nm)	6.12	5.95	6.52	6.21	6.08
	Joint torque (kNm)	10.267	10.443	10.679	9.966	10.401
	Joint efficiency (%)	31.75	30.35	32.55	32.75	31.65
	Total drive unit stiffness (Nm/rad)	1.74E6	1.82E6	1.73E6	1.70E6	1.70E6
	Total backlash at 300 Nm (rad)	3.18E-4	3.10E-4	3.09E-4	3.04E-4	3.12E-4

Table 7. Results of Drive Unit Qualification Tests (continued)
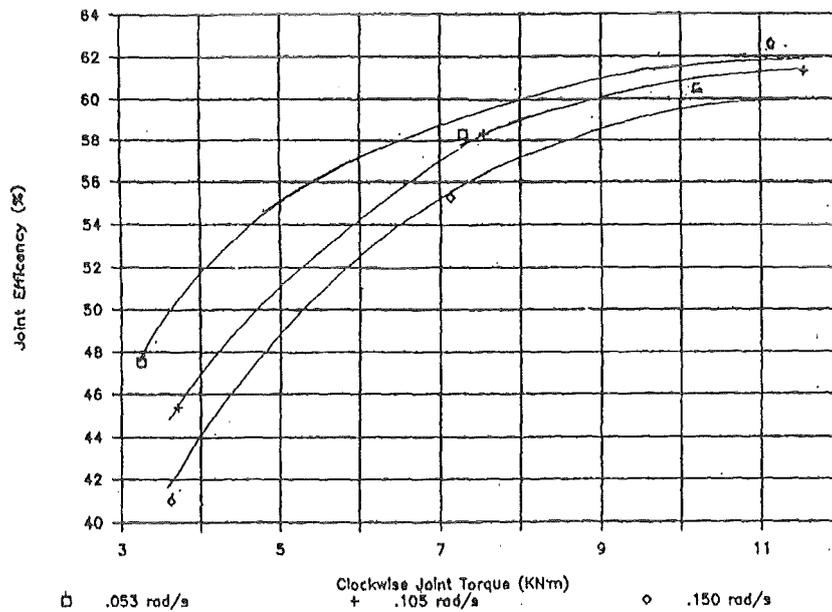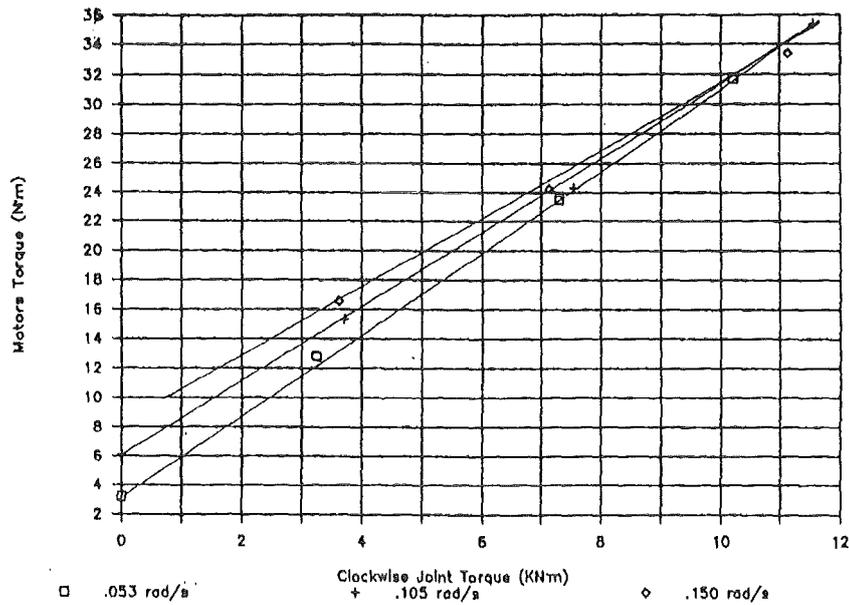
Figure 9. Characteristic Curves of Drive Unit Serial Nr. 2. Motor torque and joint efficiency vs. joint torque

2.3 End-Effector Positioning Unit (EEPU)

The EEPU forms the permanently attached connection between the ABT and the different end-effectors (see Figure 10 on page 14 and Drwg. IT-OUT-08-446). It is required to reach the upper and lower torus regions. The main technical data are given in Table 8 on page 14.
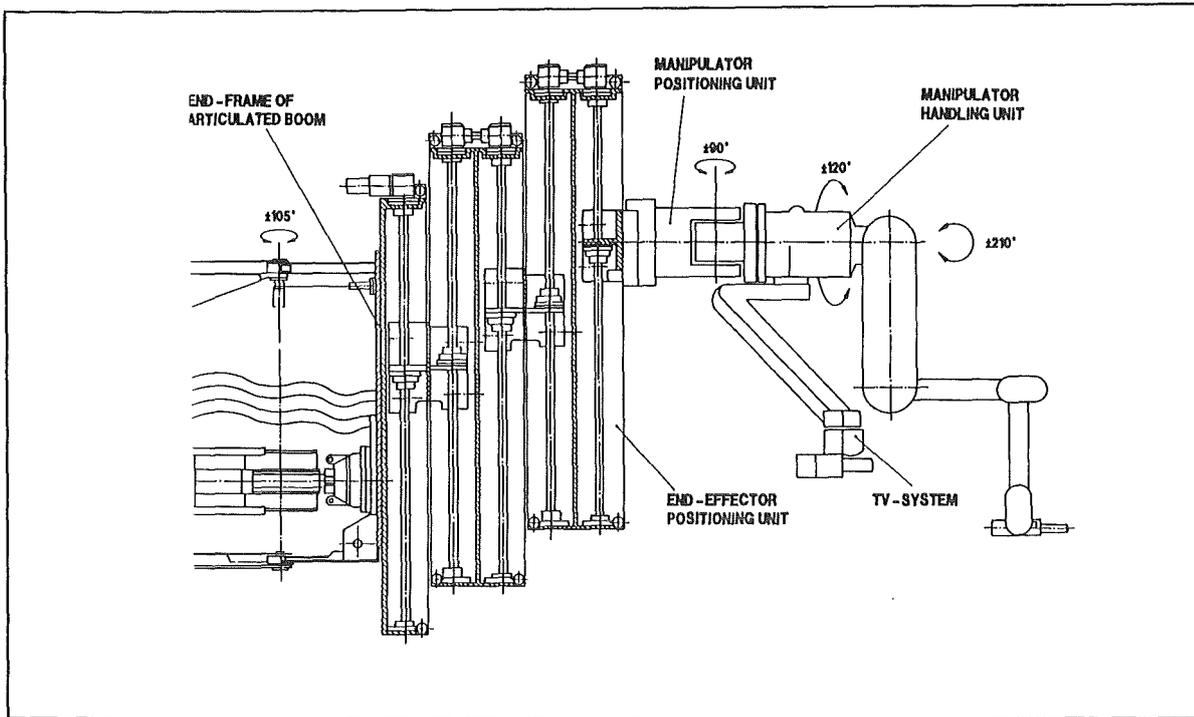
Figure 10. End-Effector Positioning Unit. The EEPU which is attached to the end-frame of the ABT is equipped with the manipulator positioning unit and the manipulator handling unit.

Parameter	Value
Length x width x height (mm)	1499 x 974 x 2758
Total stroke (mm)	±5075
Maximum speed under load -combined motion of all slides plus trolley- (m/sec)	0.1
Maximum acceleration under load -combined motion of all slides plus trolley- (m/sec²)	0.1
Positioning accuracy -not taken into account the deflection of the EEPU- (mm)	±2
Number of slides	2
Number of trolleys	1
Number / size of planetary roller spindles -diameter x lead- (mm)	5 / 39 x 5
Number / Transmission ratio of bevel gears	5 / 3:1
Number of motors with failsafe brake	6

Table 8. Technical Data of the End-Effector Positioning Unit

The EEPU is based on a slide and trolley system, as shown in Drwg. IT-OUT-21-40. To achieve high position accuracy and repetitive accuracy, size and weight of the EEPU were kept as small and low as reasonably possible by fabricating the EEPU as compact as possible and using for the large components an Al-alloy (AlZn4,5Mg1(F34)) welded structure.

The EEPU was calculated to withstand the maximum peak loads in vertical direction and fully extended. Peak loads may be caused, for instance by bumps during taking over a

divertor plate. The external loads and forces as well as the main dimensions are shown in Drwg. IT-OUT-21-08A. Bending moments are possible in X and Y directions. The calculation is summarized in an unpublished report /9/; characteristic values of the material and values used for the calculation are given in Table 9.

Parameter	Value (N/mm^2)
Tensile strength	340
Yield strength	270
Yield strength for welding seams	125
Admissible stresses	
Tensile, pressure, bending	178.2
Stability against pressure	156.6
Shearing	102.6
Surface pressure	315.9
Local stresses	270

Table 9. Characteristic Values for the Material AlZn4,5Mg1 (F34)

The EEPU as shown at the Drwg. IT-OUT-21-40 is composed of the following subassemblies:

* Support frame
* Two slides
* One trolley
* Drive units
* Positioning control components
* Cabling

The support frame forms the mechanical connection between the ABT link B4 and the EEPU. It is equipped with a drive unit which actuates a carrier which again is part of the support frame as well as of the first slide. The carrier is guided by four roller pads. The guiding railes are fixed at the frame.

The cables for the EEPU and for the end-effectors are distributed into two looms of cables and are then routed in two cable ducts arranged at both sides of the EEPU.

As to be seen in Figure 11 on page 16 there are two similar slides. As mentioned, the first slide is connected via the carrier with the support frame but also via the second carrier with the second slide which again is on the other hand connected with the third carrier, the trolley.

The trolley forms the connection between the second slide and the exchangeable end-effectors as manipulator handling unit and divertor handling unit.

To prevent dust deposition at the planetary roller spindles, the roller pads as well as at the bearings, the inner part of the slide frame is separated from the environment by roller cover bands which are on the one hand fixed at the trolley and on the other hand at reels mounted in bearings at the frame. The reels are rotated by flat spiral springs in one direction and by the trolley via the cover bands in the other direction. The covers are guided in grooves.

GEAR FOR COUPLING OF
BOTH SPINDLES
THE SPINDLES OF ONE SLIDE
ARE WORKING IN OPPOSITE

SLIDES

TROLLEY
WITH NUT

PLANETARY
ROLLER
SPINDLES

INTERFACE FOR
END-EFFECTORS

CARRIERS
WITH NUTS

INTERFACE FOR ABT
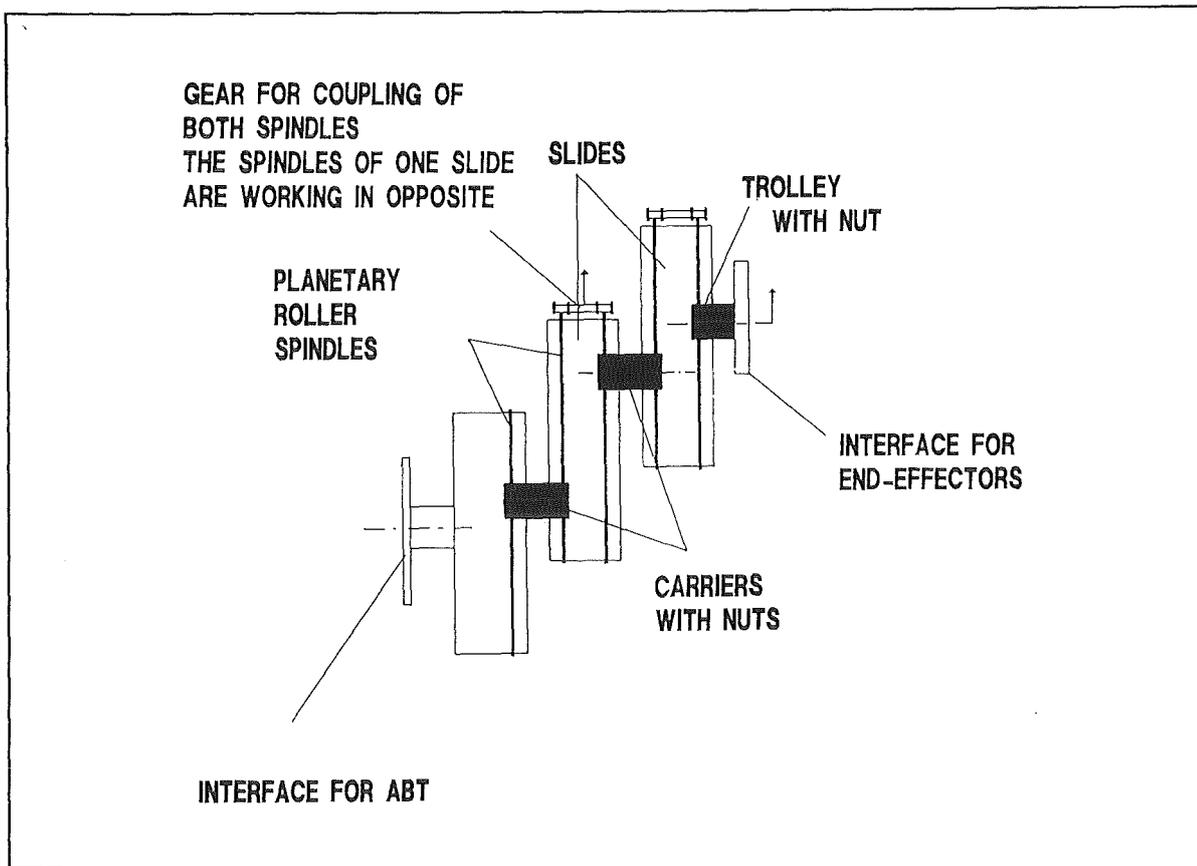
Figure 11. Scheme of the End-Effector Positioning Unit

The drives are actuating the carriers. Each slide is equipped with two drives which actuate different carriers. Drives of one slide are connected via an axis and intermediate flanges plus clutches. Thus obtaining redundancy in case of a failure at one drive. In this case the other drive will be able to keep the EEPU in position and bring it back with reduced speed into starting position.

Each drive is composed of redundant AC-servo motors with failsafe brakes, a bevel gear and a planetary roller spindle. Motors and motor plus bevel gear are remotely replaceable as one unit from the top.

For safety reasons the planetary roller spindles are loaded in tension only. Therefore, fast and loose bearing arrangements are alternating from the upper to the lower part of the frames and are replaceable, if needed, from outside. The angular ball bearings are sealed by Nilos rings and covers, respectively. A required double sealing was achieved by the roller cover bands of the slides.

The positioning control components give either position signals of the EEPU to the position control system or stop the movement of the EEPU via the drive control system or mechanically if the other systems which have priority fail. The sequence of priority for the control components is the following:

At normal circumstances absolute multi-turn resolvers arranged at the bevel gears measure the revolutions of the planetary roller spindles and signal the data to the position control system.

Limit switches mounted at the frames will be actuated by the carriers if one of the resolvers fails. They stop the according drive.

The third level is the mechanical stop if both resolver and limit switch fail.

2.4 End-Effectors (EE)

Based on the most in-vessel handling tasks, listed in chapter 1.2, "Requirements for EDITH," which have to be demonstrated at the moment two types of end-effectors were considered:

- The manipulator handling unit (MHU) which represents a multi-purpose work unit can handle different tools, e.g. tile positioning device and tile detachment device to replace protective armour tiles at the first wall of the vacuum vessel. The final manipulator handling unit will be constructed by combining two electric master-slave manipulators. The unit may be based either on Mascot IV manipulators or on a combination of one Mascot IV manipulator and one manipulator RD 500.

- The divertor handling unit (DHU) as a special handling device is foreseen to replace only divertor plates arranged inside the vacuum vessel.

In a first step, an interim solution, based on available manipulators, was constructed which is manually attachable to the EEPU. Design and construction of the DHU were postponed and will be continued when the ITER design will have been frozen. Additional EE's might be possible but are not yet defined. It will be common to all EE's that later on they will be remotely exchangeable at the EEPU.

2.4.1 Manipulator Handling Unit (MHU)

The MHU, constructed as an interim solution for the demonstration of maintenance tasks in connection with EDITH in the full scale mock-up, is shown in Figure 12 on page 19 and in the Drwg. IT-OUT-24-52. It is composed of one electric master-slave manipulator -KfK-type EMSM3-, one electric master-slave manipulator -KfK-type EMSM2-B- and one adaptation unit for connecting the slave arms of both manipulators and for the attachment to the manipulator positioning unit.

The slave arm of the EMSM3 has a load capacity of 100 kg. It has seven force reflecting movements. Its drive housing is formed like an arm in order to increase its range of operation. The slave arm is mounted at the upper part of the drive housing to the adaptation unit. There is a further movement -tilting of the housing- implemented which is switch-controlled.

The slave arm of the EMSM2-B has a load capacity of 45 kg. It has also seven force reflecting movements. It is mounted via the housing to the adaptation unit. A further switch-controlled movement is not available.

It is the purpose of the adaptation unit to connect both slave arms with the MPU. On the other hand, the adaptation unit has to compensate the geometrical differences of both slave arms. Thus it is possible to operate with both arms almost within the same reach of envelope.

Both manipulators have a high working speed and a great dexterity due to the ability to coordinate the motions and the feeling of the operator. Furthermore, they have force reflection and a free programmable path control. Their technical data are shown in Table 10 on page 18, at Figure 13 on page 20, Figure 14 on page 21 and Figure 15 on page 22

Parameter	EMSM2-B	EMSM3
Load capacity of tong during continuous operation (kg)	24	60
Load capacity of tong for 15 min at EMSM2-B and for 30 min at EMSM3 (kg)	45	100
Power amplification master:slave	1:2, 1:6, 1:20	1:2, 1:20, 1:50
Maximum speed of tonge (in X-Y-plane) (m/sec)	> 1	> 1
Drives	3-phase induction motors with frequency and current controls	Asynchron motors with 3-phase amplifiers and field oriented frequency control

Table 10. Technical Data of Electric Master-Slave Manipulators

According Drwg. IT-OUT-24-52, the main subsystems of the adaptation unit are:

* Extension arm
* Shoulder piece
* Rotation unit
* Attachment flange

It is equipped with three electric drives. In the initial position the extension arm is at the right side of the shoulder piece. The slave arm EMSM2-B is mounted at the tip of the extension arm. Its housing is inclinable by an electric drive (slave pitch). The extension arm is inclinable versus the shoulder piece by a second drive (arm pitch). The slave arm EMSM3 is mounted at the left side of the shoulder piece. There is an own drive arranged in its drive housing for tilting the drive housing versus the shoulder piece. By means of a third drive the shoulder piece is rotatable versus the rotation unit (unit rotation). The fixation of the adaptation unit at the MPU is performed by means of the attachment flange.

Dimensions, ranges of operation and the interfaces are shown in Drwg. IT-OUT-24-52. The maximum speed of the three movements is 3 revolutions/min.

Each drive unit is composed of a AC-servo motor with failsafe brake, one gear and one resolver.

2.4.2 Manipulator Positioning Unit (MPU)

The MPU (Figure 10 on page 14 and Drwg. IT-OUT-24-53 serves as a link between the EEPU and the MHU. Its design phase is finished and the technical specification for its manufacture prepared.

The MPU is made up of two links combined to the horizontal joint, the drive, and the control system. The MPU data are listed in Table 11 on page 19.
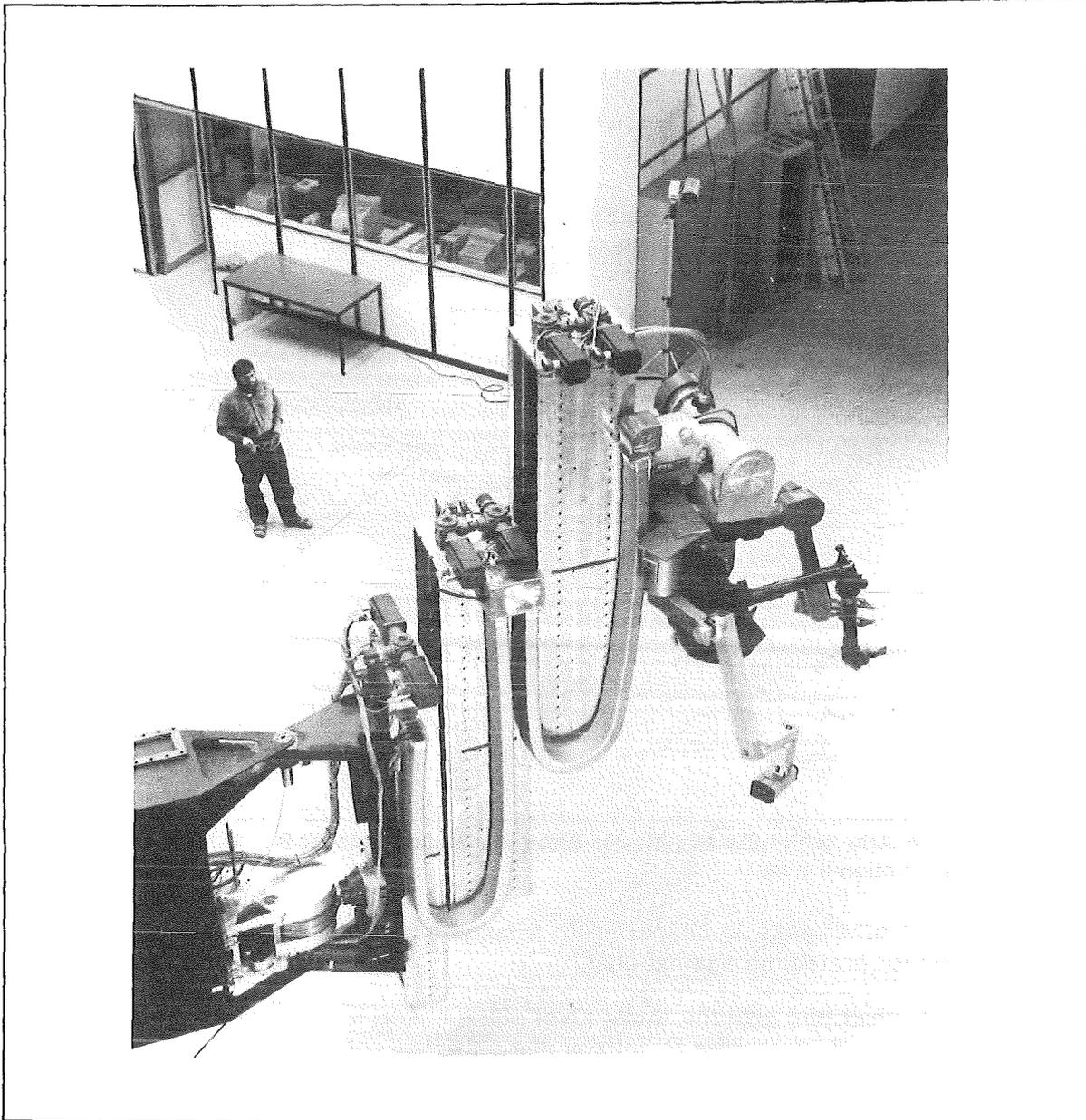
Figure 12. Manipulator Handling Unit. The MHU is directly attached to the EEPU without the MPU

Parameters	Values
Maximum swiveling angle (degr.)	±111
Swiveling angle restraint by limit switch (degr.)	±105
Maximum speed (rad/sec)	0.15
Transmission ratio of bevel gear	1:1
Number of CYCLO-gears	2
Transmission ratio of each CYCLO-gear	29:1

Table 11. Technical Data of the Manipulator Positioning Unit

The joints, an upper and a lower joint, connect the yokes, one being part of the EEPU and the other one being part of the MHU. The upper pivot pin is bolted with the yoke of
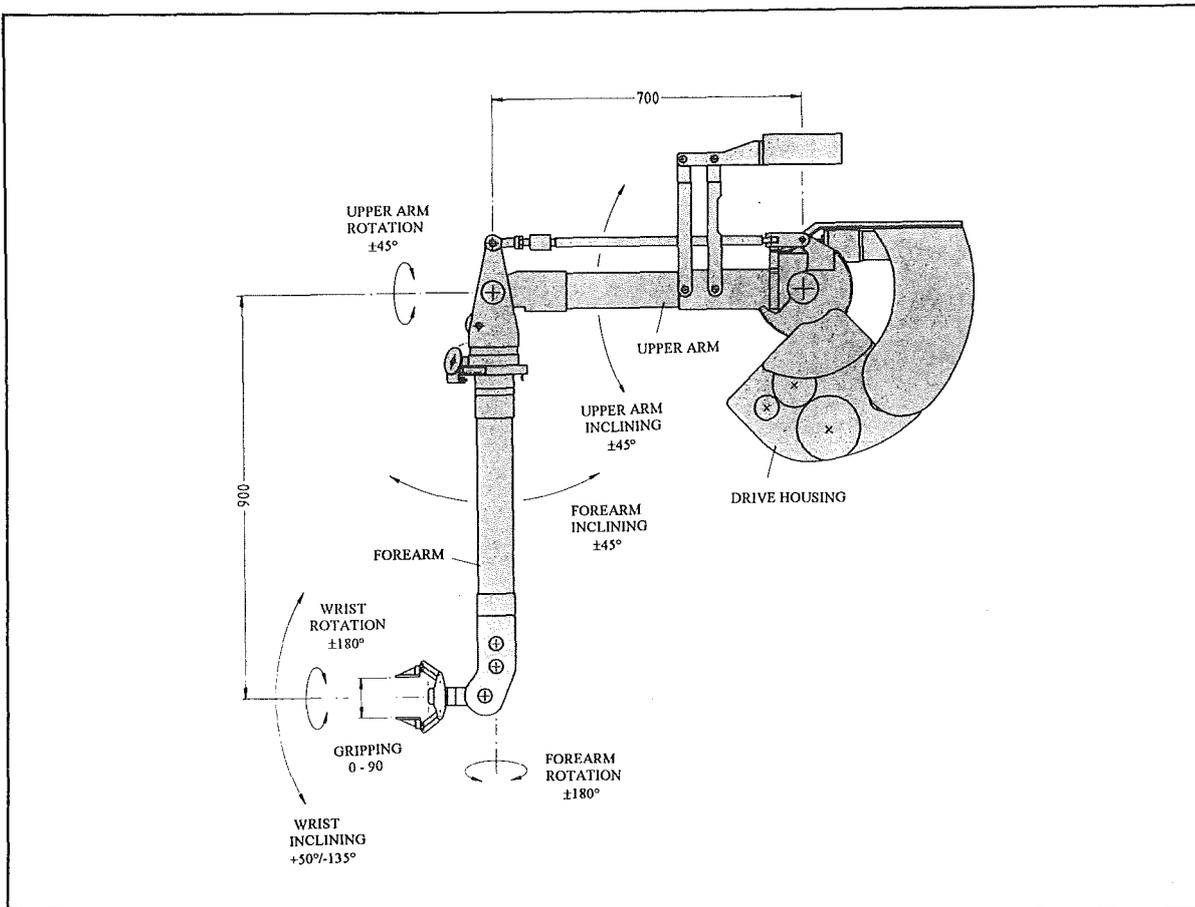
Figure 13. Slave Arm of the Electric Master-Slave ManipulatorEMSM2-B:. Dimensions, motions and motion travels

the EEPU. The other yoke is supported in a spherical roller bearing . The bearing is double sealed for protection against dust.

Like the upper pivot pin, also the lower pivot pin is bolted to the yoke of the EEPU. And also the other yoke is supported by means of a spherical roller bearing. The two brackets of the lower yoke are mounted with the pivot pin, the plugwasher, and the nut in such a way that the brackets of the lower yoke will not be deformed under pressure. Like the upper spherical roller bearing, also the lower bearing is protected against dust.

The MPU is moved by a coaxial drive made up of two redundant AC-servo motors driving the in-line CYCLO-gears via a common bevel gear unit. The bevel gear unit has a transmission ratio of 1:1, while the two CYCLO-gears have transmission ratios of 29:1 each. The force is transmitted to the turntable via the two bolts and bushes. Two alignment pins connect the turntable and the insert, which in turn, is pinned and bolted to the yoke. The bolts in combination with the alignment pins and the pin allow the turntable and the insert to be decoupled in case a gear unit were to lock. The two CYCLO-gears are bolted to the yoke at the EEPU via the large lantern and the supporting crossbar. The bevel gear unit is flanged to the CYCLO-gear via the small lantern.

The drive control system is made up of the control electronics of the motors, the multiturn resolver, the limit switch and the connecting cables.
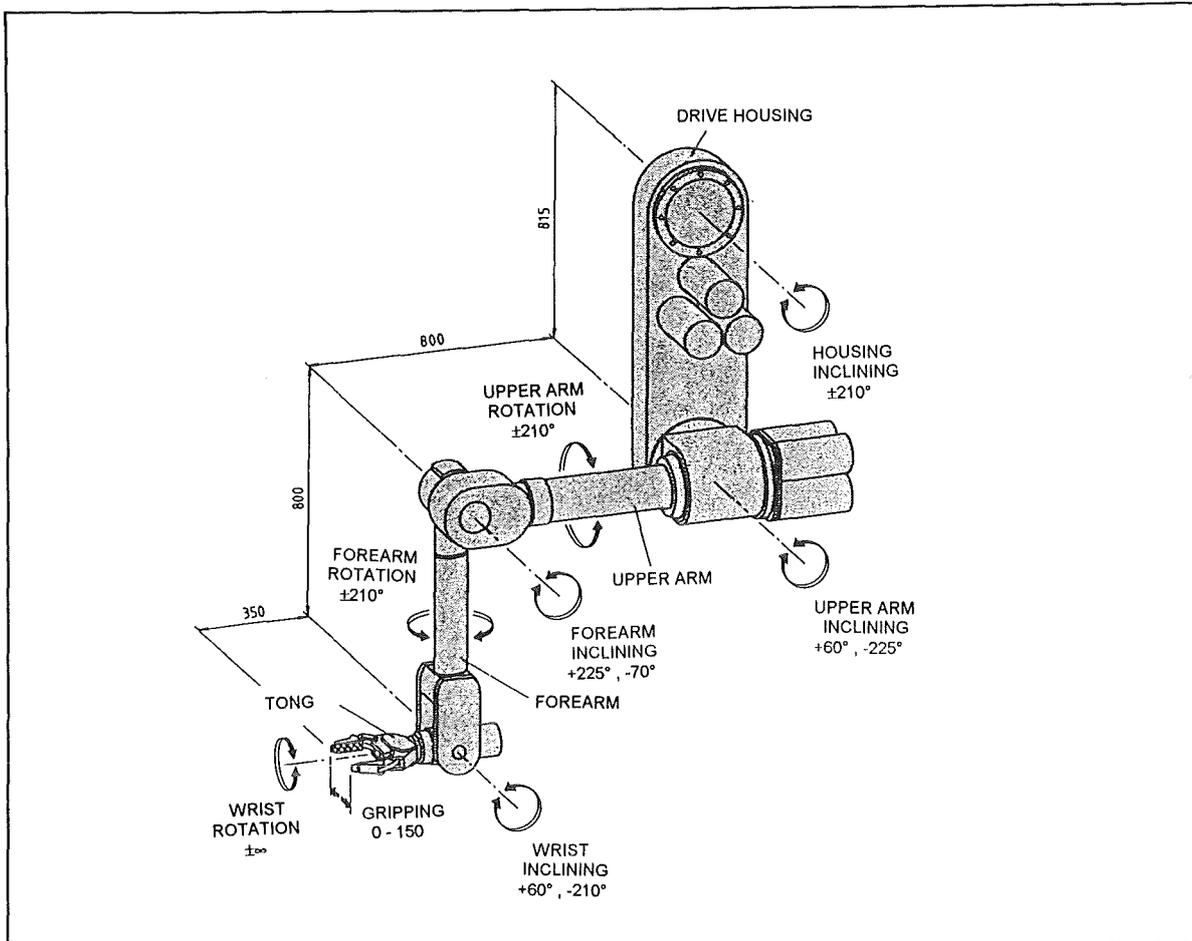
Figure 14. Slave Arm of the Electric Master-Slave ManipulatorEMSM3:. Dimensions, motions and motion travels shown in the basic position 1.

2.5 Viewing System (VS)

The EDITH viewing system is foreseen to be composed of three camera systems, one overview camera and two cameras (Figure 16 on page 23) installed at the EEPU of the ABT (Figure 17 on page 24). For overview a stereo camera is foreseen while the other two mono cameras give a detailed view of the working site. The stereo camera is mounted at the front of the MPU adaptor. Also the camera booms are installed at the MPU adaptor. This arrangement allows that the cameras make the same rotations as the manipulator slave arms and give therefore always adequate camera views with respect to the manipulator slave arms.

The camera booms are made up of the subassemblies rotation unit, upper arm, forearm and swivel-pitch head (Figure 18 on page 25). They have four driven movement options. By means of joint 3 which has an additional passive movement option the swivel-pitch unit can be kept in vertical position independently of the upper arm pitch. This is performed via a fixed pinion at the upper arm and another pinion which is connected with the first one.

Type and arrangement of the movement options, the range of operation, the lengths of the arm links and the design of the camera boom is to be seen in Figure 18 on page 25.

The drives of the camera booms are based on AC-servo motors. For position control and operation in an automatic mode all axes are equipped with potentiometers and limit switches.
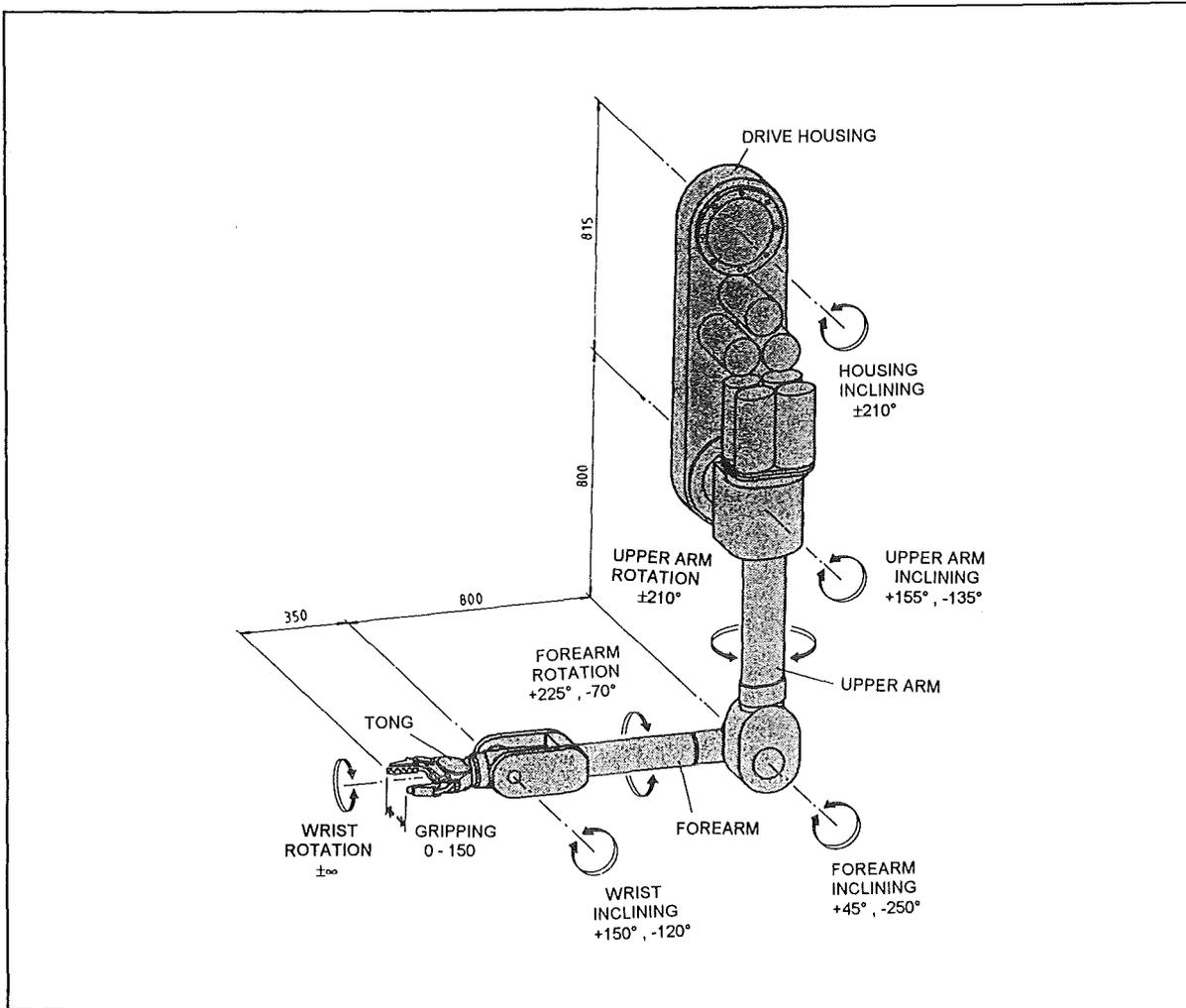
Figure 15. Slave Arm of the Electric Master-Slave ManipulatorEMSM3:. Dimensions, motions and motion travels shown in the basic position 2.

2.6 Stress Analysis of the Articulated Boom Transporter and its Validation

2.6.1 Describing the Problems

The stress analysis of the ABT /10/ was performed based on the requirement to handle at the boom tip payloads up to 3900 kg without any support inside of the vacuum vessel. In order to allow operating the boom with extremely high accuracy the deformation of the boom caused by its dead weight and the payload should have been kept to a minimum. Therefore, the two requirements mentioned above had the highest priority. Furthermore, considering high speed operation (high acceleration and lag of each link), a small dead weight was desirable. But as will be shown the two requirements, small tip deflection and small dead weight, exclude each other. Based on the port dimensions and the required reach of the boom as listed in Table 2 on page 3 and with respect to maintain a particular distance and to avoid horizontal collisions, the number and dimensions of the links were determined as shown in Table 4 on page 8. As Figure 4 on page 6 shows EDITH has one front link B4 and three intermediate swiveling links B1 to B3. On the other hand the fixing link C1 is cantilevered attached to the support structure as it will have been accordingly installed to the transport carrier of the later operational in-vessel handling unit shown in Figure 1 on page 2.
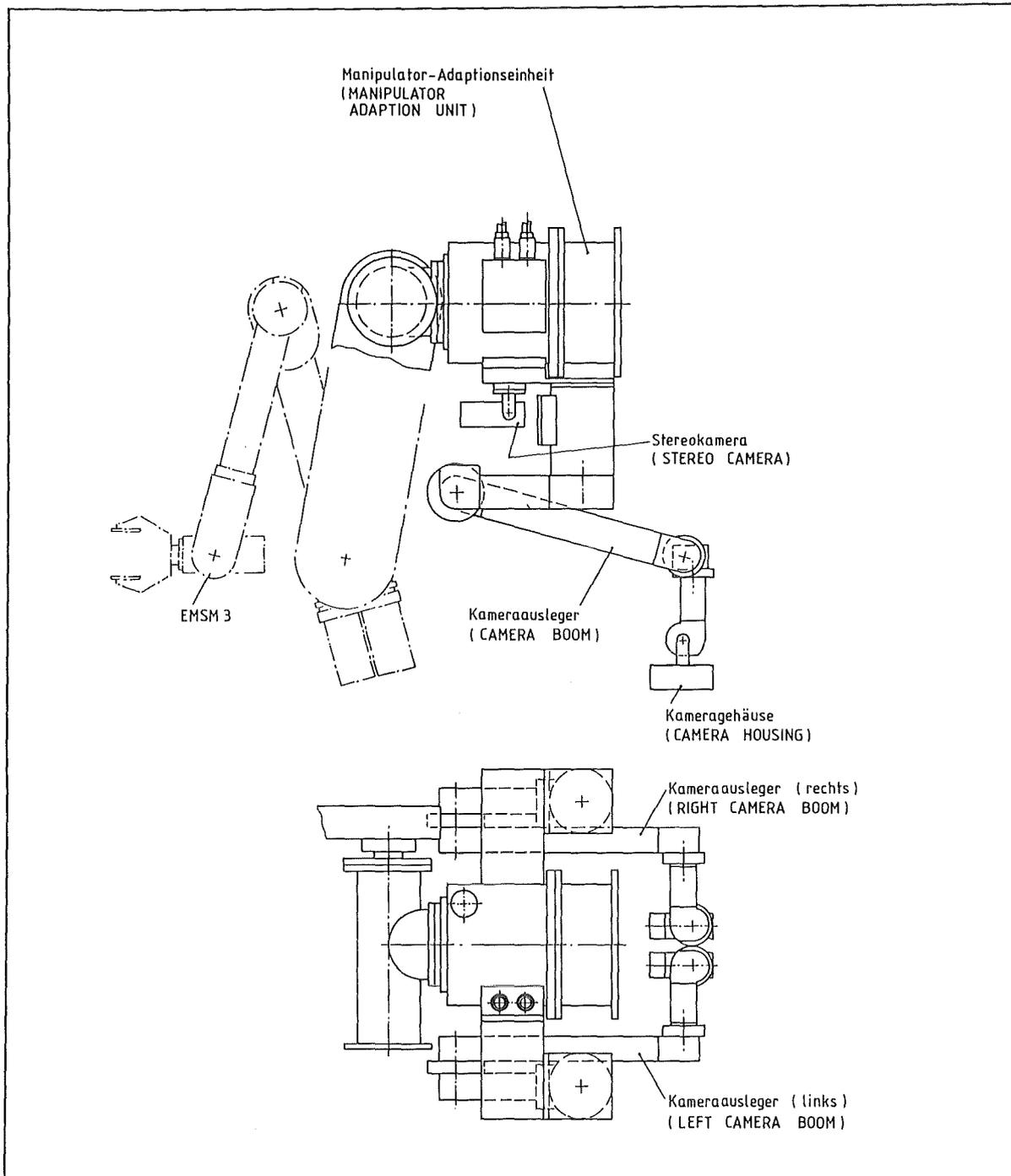
Figure 16. Viewing System Arrangement at the Manipulator Handling Unit

In former designs smaller ports and a working space including the whole torus were assumed. These assumptions contributed to an unacceptable high boom flexibility.

The necessary considerations and calculations for the new design of EDITH are scheduled in the chapters 2.6.2, "Designing the Cross Sections of the Links" to 2.6.5, "Detailed Computations Using Finite Element Codes and Determination of the Wall Thicknesses." Load experiments are discussed in the chapter 2.6.6, "Comparison of Actually Occuring and Computed Deformations." Especially the actually occurring deformations are compared to those obtained by calculations. This way one can see up to which degree the mechanical behaviour of complicated structures may be predicted by concurrent computational methods.
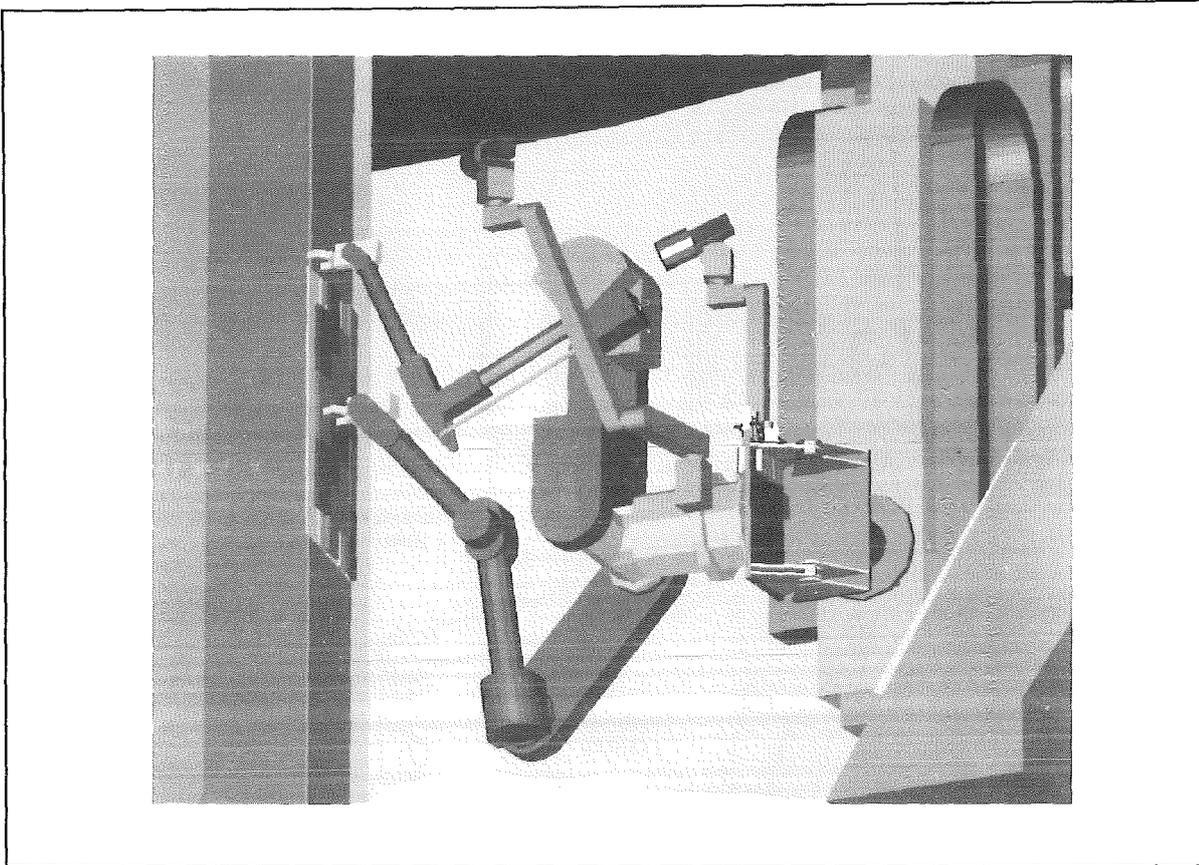
Figure 17. Manipulator Handling Unit with Viewing System

2.6.2 Designing the Cross Sections of the Links

The cross sections of the links should be determined in a way that the boom's tip deflects as little as possible because of the boom's dead weight. Depending on the configuration of the boom the links are loaded with bending and torsional moments. That's why one is looking for cross sections with lowest possible bending and torsional flexibility. For this selection the interesting cross sections should be comparable in the remaining important qualities. Here the dead weight is an important issue. It is proportional to the cross sectional area (the part of the link's cross section consisting of the walls) and therefore all the cross sections under consideration should have the same cross sectional areas. Furthermore, in the context of stiffness equal distributions of stress (no peaks!) are asked for.

2.6.2.1 Determination of the Cross Section with Smallest Bending Flexibility under the Condition of Given Surface

Bending flexibility is proportional to the curvature d^2y/dx^2. This is approximately derived by the beam bending equation:

$$\frac{d^2y}{dx^2} = \frac{M_b}{E\,I_\eta} \tag{1}$$

Figure 19 on page 26 shows the bending moment M_b, Young's modulus E of the employed material and the moment of inertia I_η.
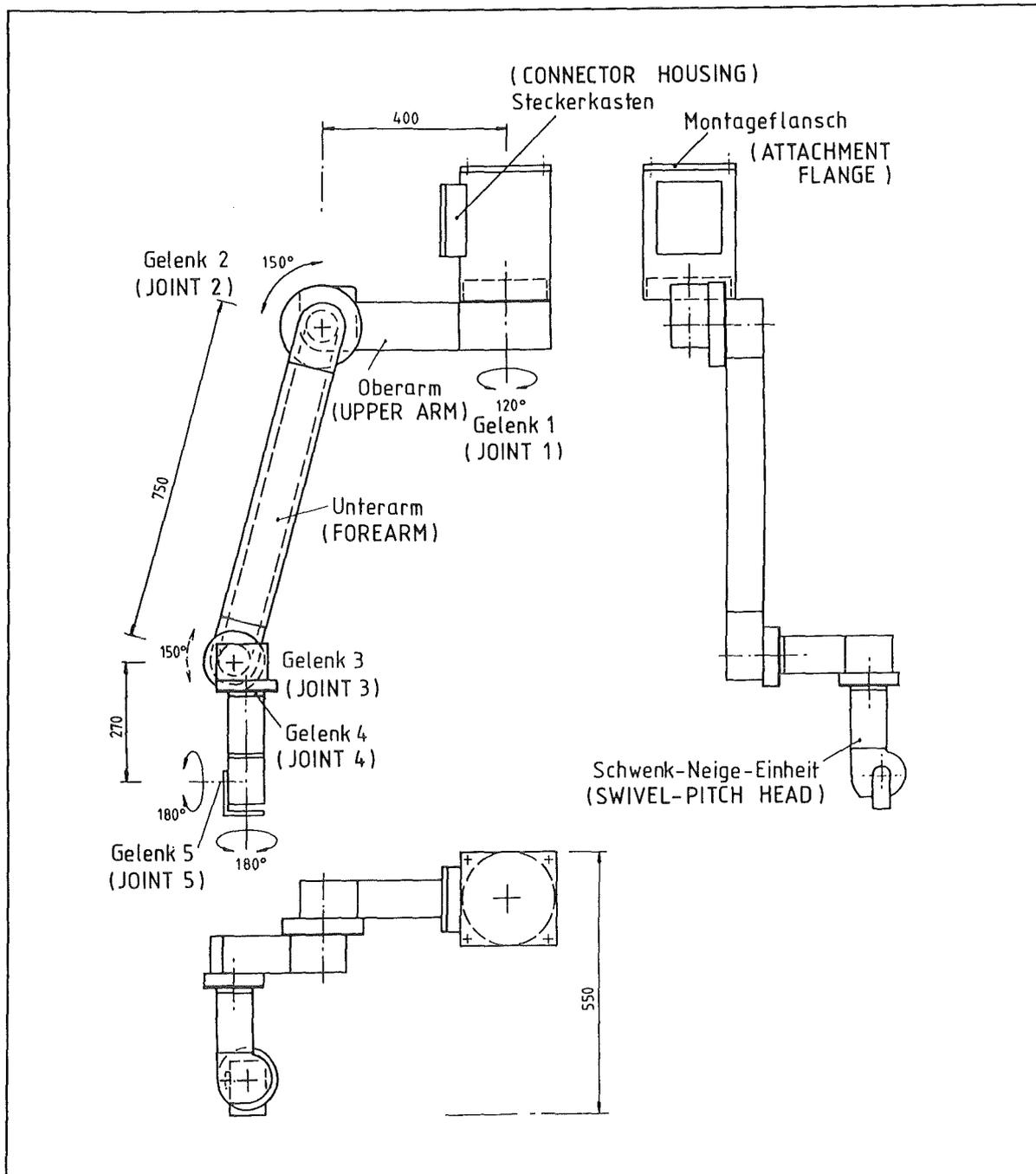
Figure 18. Manipulator Handling Unit - Camera System Based on Telescopic Arm

$$I_{\eta} = \int \eta^2 dF \tag{2}$$

The determination of the cross section with lowest bending flexibility, e.g. minimum curvature d^2y/dx^2, under a given cross sectional area of the walls results in the determination of the maximum moment of inertia I_n. It can be easily shown that this leads to the cross section depicted in Figure 20 on page 27. It mainly consists of an upper and a lower chord with a maximum distance, e.g. a maximum height of the cross section. That is why the highest possible value of 2.0 m should be realized. In this case the moment of inertia I_n is approximately calculated by
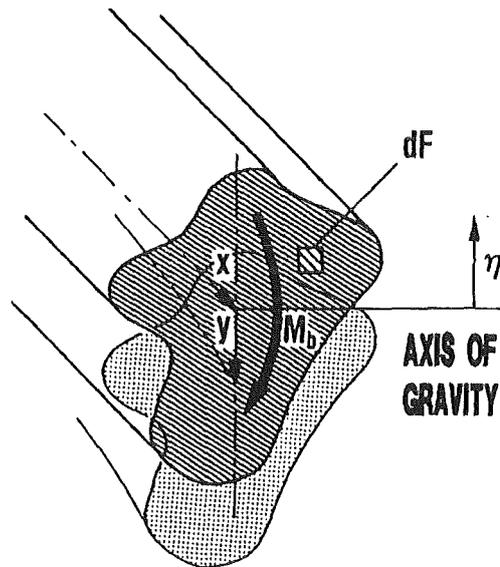
Figure 19. Arbitrary Cross Section of a Beam

$$I_\eta = \frac{1}{2} a^2 b s \tag{3}$$

The width of the cross section is arbitrary because with a constant cross-sectional surface assumed

$$F = 2 b s \tag{4}$$

there is no influence on the moment of inertia. With a pure bending load $M_b > 0$ in the upper chord there is only equally distributed tensile stress and in the lower chord one only find equally distributed compressive stress. Theoretically the connecting plates between the two chords shown in Figure 20 on page 27 are not necessary. So their influence is not taken into account. However, if there are also shear forces the connecting plates have to transmit shear stress. Usually this shear stress is so small that there may be break-throughs in these plates. The shear stress acting vertically leads to tensile and compressive stresses in directions inclined at an angle of 45° So the break-throughs should be arranged in a way that the remaining struts have a sloping position of these 45° (see Figure 21 on page 28). Thus the shear stress loading leads to cross sections with side walls well known from lattice structures.

2.6.2.2 Determination of the Cross Section with Smallest Torsional Flexibility under the Condition of a Given Surface.

On condition that the cross-sectional area is kept constant hollow sections have relatively small torsional flexibility. This may be illustrated by the torsion of a rolled sheet of paper according to Figure 22 on page 29. As long as the overlapping parts may slip in axial direction the torsional flexibility is rather high. If the overlapping parts are agglutinated a closed hollow section is generated and the torsional flexibility is reduced distinctively.

Shear stress in thin walls are inversely proportional to the wall's thickness, e.g. the product of shear stress and wall's thickness is constant at every circumferential location. This leads to the cross section depicted in Figure 23 on page 30.

However, the question remains whether the cross section with a maximum possible height of 2.0 m and width of 0.6 m also leads to a minimum torsional flexibility among
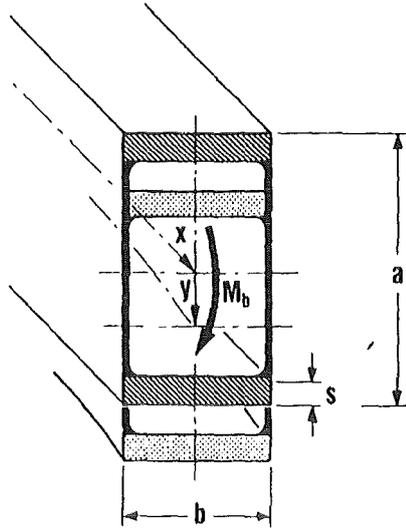
Figure 20. Link Cross Section with Minimum Bending Flexibility

all the other cross sections with the same cross-sectional area. According to Bredt's equation /11/ the torsional flexibility (torsion angle per length of the link) is approximated by:

$$\Phi = \frac{M_t}{G} \frac{a + b}{2a^2 b^2 s} \tag{5}$$

where M_t is the torsional moment and G the modulus of torsional shear of the applied material. The meaning of the remaining quantities is explained by Figure 23 on page 30. The cross-sectional area becomes:

$$F = 2(a + b)s \tag{6}$$

With a small increase of the cross-sectional height Δa and a small change in the wall's thickness Δs a small change in torsional flexibility is expected:

$$\Phi + \Delta \Phi = \frac{M_t}{G} \frac{a + \Delta a + b}{2(a + \Delta a)^2 b^2 (s + \Delta s)} \tag{7}$$

The cross-sectional area is assumed to be constant:

$$2(a + b)s = 2(a + \Delta a + b)(s + \Delta s) \tag{8}$$

Neglecting the products of small values, equation (8) gives:

$$\frac{\Delta s}{s} = - \frac{\Delta a}{a + b} \tag{9}$$

and inserting this into equation (7):

$$\frac{\Delta \Phi}{\Phi} = - 2 \frac{b}{a + b} \frac{\Delta a}{a} \tag{10}$$

A similar relation may be derived for a change in the width of the cross section Δb.
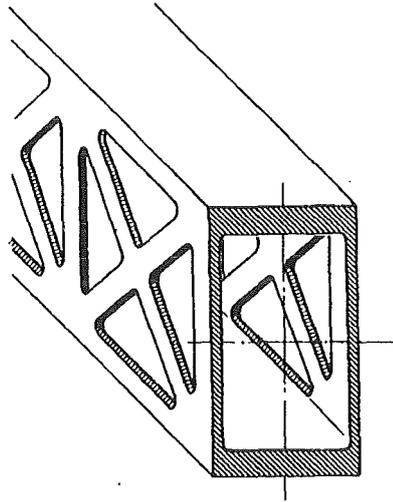
Figure 21. Link Cross Section with Lattice Structure, Suitable at Bending Stresses

Equation (10) indicates that keeping the cross-sectional area constant each rise in the cross-sectional height Δa or width Δb reduces the torsional flexibility.

If the height of the cross section is essentially higher than the width a further raise of the height reduces the torsional flexibility only very little. In the case of a quadratic cross section the torsional flexibility increases to the same ratio as the cross-sectional height. Finally assuming a cross section with an essentially smaller height than width a rise in the height results in a double reduction of the torsional flexibility.

2.6.2.3 Determination of the Cross Section Combining Relatively Small Bending and Torsional Flexibility with the Cross-Sectional Area Kept Constant

The cross sections determined so far optimize only the bending or the torsional flexibility. But in reality cross sections are needed which comply with both requirements. The investigations above suppose that the cross section found last with constant wall's thickness and minimum torsional flexibility, appears to be suitable here. This is supported by the results of subsequently conducted FEM computations which exhibit the crucial influence of the torsional flexibility on the deflections of the boom's tip.

It follows that frame work side walls are not taken into consideration any longer. They would not be suitable to deal with the considerable shear stress caused by torsion. But still great importance is attached to exhausting the full possible cross-sectional height. This is not only meant to minimize the bending flexibility but also to minimize the considerable joint forces.

On the other hand constructive aspects have to be considered, too. That means that the design of the links B3, B2, B1 and C1 is very easy if one increases their heights -in the given order- to the measure of the upper and lower joint hinges. So the heights of the links B3, B2, B1 and C1 have been determined to 1825 mm, 1871 mm, 1930 mm and 2000 mm.

Optimal grading of the wall's thickness starting with rather thin walls for the front link B3 and ending with a rather thick wall for the fixing link C1 is not investigated here. In principal this could be done using the analogy to the preceding procedure. However, some important features cannot be taken into account, as there are manufactural problems of buckling due to thin walls or problems concerning the junction with the joints. Therefore, the grading of the wall's thickness were determined empirically.
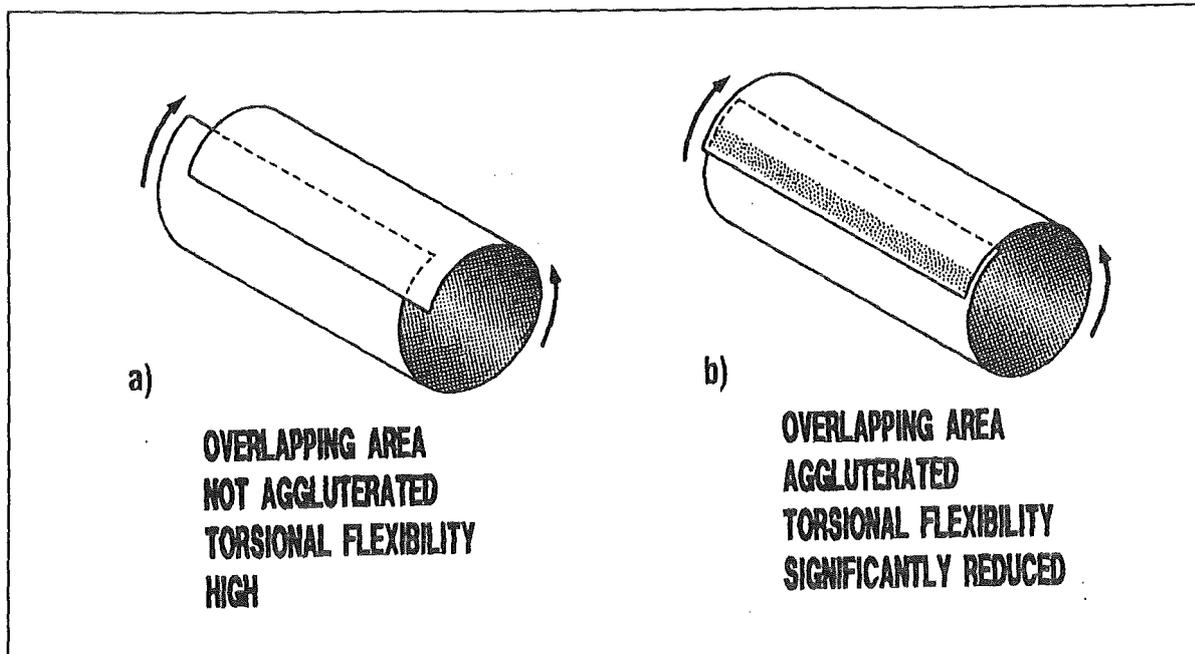
Figure 22. Torsional Flexibility of a Hollow Structure, Discussed on Hand of a Rolled Sheet of Paper. a) If the overlapping area is not agglutinated the torsional flexibility is high. b) At agglutinated overlapping area the torsional flexibility is significantly reduced.

On the other hand a determination of an absolute value for a wall's thickness is not possible according to the above considerations. For example increasing the wall's thickness not only reduces the flexibility but also the dead weight. Therefore, at a first approach the deflection of the tip does not change.

2.6.3 Material Selection

Again the decisive criterion is the deflection ot the tip caused by the dead weight.

The starting point is again equation (1) describing the bending flexibility. Taking into account that the bending moment M_b is proportional to the density ρ and the wall's thickness and, that according to equation (3) the moment of inertia I_n is proportional only to the wall's thickness, the wall's thickness is cancelled out and it turns out:

$$\frac{d^2y}{dx^2} \sim \frac{\rho}{E} \tag{11}$$

Another starting point is equation (5) describing the torsional flexibility. Noticing that the torsional moment M_t is proportional to the density ρ and the wall's thickness, too, the wall's thickness is cancelled out again and it remains:

$$\Phi \sim \frac{\rho}{G} \tag{12}$$

With basic relations the modulus of shear G may be expressed by Young's modulus E and Poison's ratio v, $G = E / 2 (1 + v)$. For the materials in discussion v is about 0.3 and so it follows:

$$\Phi \sim \frac{\rho}{E} \tag{13}$$
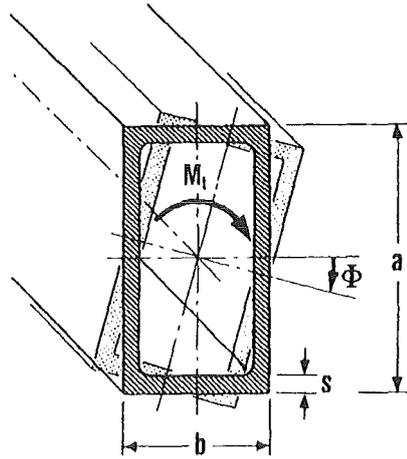
Figure 23. Link Cross Section with Minimum Torsional Flexibility

It does not matter if the deflection under dead weight is due to the bending flexibility (curvature d^2y/dx^2) or whether it is due to the torsional flexibility (torsional angle Φ); in both cases the minimization of the deflection requires the selection of the material with the lowest ratio ρ/E.

For the most common or available materials this ratio ρ/E does not differ very much, for instance:

Steel:

$$\frac{\rho}{E} = \frac{7.8 \times 10^3 kg/m^3}{2.1 \times 10^5 N/mm^2} = 3.7 \times 10^{-8} kg/Nm$$

Aluminium:

$$\frac{\rho}{E} = \frac{2.7 \times 10^3 kg/m^3}{0.675 \times 10^5 N/mm^2} = 4.0 \times 10^{-8} kg/Nm$$

Titanium:

$$\frac{\rho}{E} = \frac{4.54 \times 10^3 kg/m^3}{1.05 \times 10^5 N/mm^2} = 4.32 \times 10^{-8} kg/Nm$$

Steels appears to be best qualified. Furthermore among these materials most experience has been gained with steel. But if there were some important reasons to use aluminium or titanium these materials would be acceptable in the context of the tip's deflection under dead weight, too.

Actually a second similar criterion may be derived from minimizing the dead weight. If σ_{zul} stands for the maximum acceptable stress this criterion requires that the material with minimum ρ/σ_{zul} is to be selected and that σ_{zul} is exhausted.

Taking into account the following data

Steel:

$$\frac{\rho}{\sigma_{zul}} = \frac{7.8 \times 10^3 kg/m^3}{200N/mm^2} = 39 \times 10^{-6} kg/Nm$$

Aluminium:

$$\frac{\rho}{\sigma_{zul}} = \frac{2.7 \times 10^3 kg/m^3}{150N/mm^2} = 18 \times 10^{-6} kg/Nm$$

Titanium:

$$\frac{\rho}{\sigma_{zul}} = \frac{4.54 \times 10^3 kg/m^3}{400N/mm^2} = 11.35 \times 10^{-6} kg/Nm$$

titanium should be given preference. Now the wall's thicknesses which have not been determined yet may be chosen. But in doing so the tip's deflection under dead weight has to be accepted without any possibility to reduce it. Taking into consideration the requirements presented in chapter 2.6.1, "Describing the Problems" on page 22 a rather little deflection under dead weight and payload is more important than minimizing the dead weight itself. Therefore, it was decided to choose steel as structural material. This way the wall's thickness will be determined later to make sure that the deflection of the tip under payload is sufficiently small. The idea of minimizing the tip's deflection is abandoned in order to allow a compromise solution considering the demand for a small dead weight. It shows that for most cases the occuring stesses lie far below acceptable limits. Only a few peak values occuring in complicated structural parts should be paid attention to.

2.6.4 Design of the Joints and Specific Problems of Leading the Joint Forces into the Links

The design of the joint and the joint plates 1 to 6 is scheduled in Figure 24 on page 32 (a). The distance from the front surfaces of the links to the joint's axis is determined by the demand for free space for the driving units. To avoid extreme bending forces the constructively more complicated V-shaped joints have been selected. The joint's forces acting in the horizontal plane and especially the force component P_t are also shown in Figure 24 on page 32 (a).

The vertical forces are transmitted only by the plates 2 and 3. That is shown in Figure 24 on page 32 (b).

2.6.4.1 Loading of the Front Area of the Links

The joint forces P_t lead to the torsional moment M_t into the link:

$$P_t = \frac{M_t}{a_2} \tag{14}$$

This yields a shear stress τ_1 in the upper and lower chord of the link (Figure 25 on page 33 on the left):
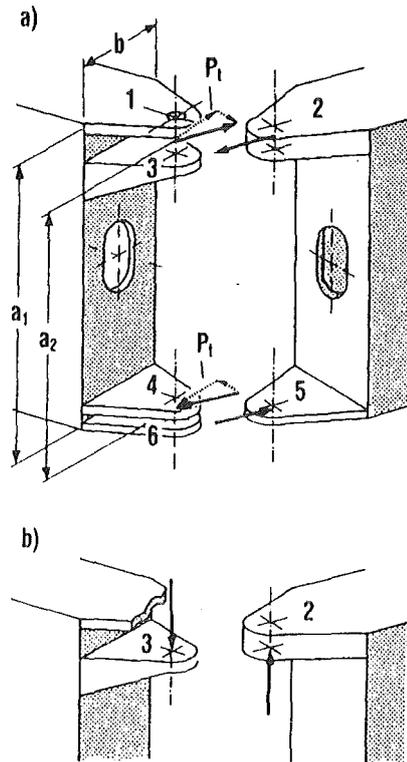
Figure 24. Design of the Link Joints. Horizontal forces will be transmitted by all joint hinges, vertical forces only via the joint plates 2 and 3.

$$\tau_1 = \frac{P_t}{s \times b} \frac{a_2}{a_1} \tag{15}$$

The factor a_2/a_1 accounts for the somewhat different lever arms of P_t and τ_1.

The shear stress τ_2 loading the torsional moment into the link is not only acting in the upper and lower chord but is distributed equally along the link's circumference (steady shear flux; middle of Figure 25 on page 33). Additionally the distribution of the shear stress τ_3 has to be superposed (on the right of Figure 25 on page 33). Because of its definition no reaction forces may be generated and the horizontal and vertical components of the associated distribution of shear stress have to be equal. From Figure 24 follows:

$$\tau_1 = \tau_2 + \tau_3 \quad (upper + lowerchords)$$

$$\tau_2 = \tau_3 \qquad (side\ walls) \tag{16}$$

This leads to

$$\tau_2 = \tau_3 = \tau_1 / 2 \tag{17}$$

The distribution of shear stress τ_2 transmitting the torsional moment M_t in the link may be obtained directly from M_t. It may be confirmed that this result complies with the results of equations (14) - (&sv17).

Practically a thin walled hollow section, built up by the upper and lower chord and the side walls, may not bear the shear stress component τ_3. Therefore, the links must have
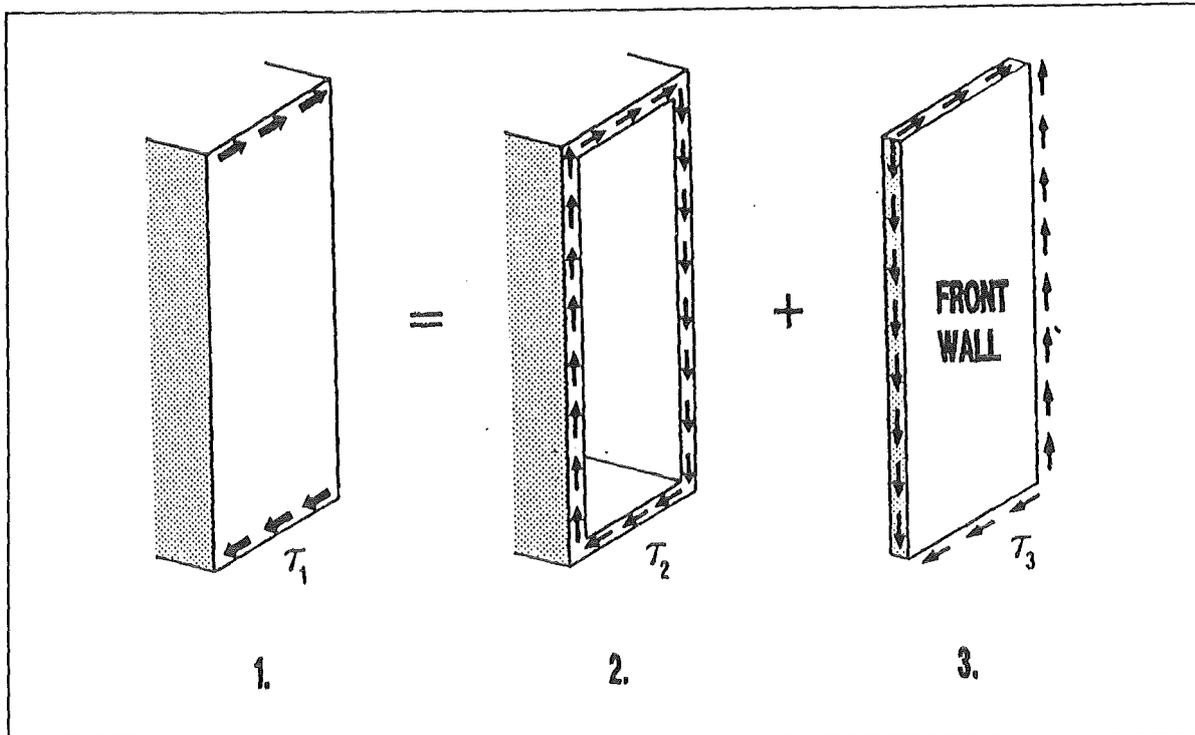
Figure 25. Transmitting the Torsional Moment into a Link. The Resulting Shear Stress Requires a Sufficiently Dimensioned Front Wall. Left: Shear stress transmitted from the link hinges. Middle: Shear stress transmitting the torsional moment in the link. Right: Shear stress loading the link cross section by shear.

sufficiently dimensioned front walls. The necessity of such walls can also be demonstrated by another consideration. According to Figure 26 on page 34 the forces acting at the joint plates not only may be interpreted as force couples P_1/P_2 and P_3/P_4 which distort the link along its longitudinal axis (torsional moment M_t) but they may also be understood as force couples P_1/P_3 and P_2/P_4 which distort the link along its vertical axis. A structure which is able to deal with such loadings is in fact a hollow section coposed of the two side walls and the front walls. Each of these walls should have similar thicknesses.

2.6.4.2 Loading of the Joint Plates

Particular regard has to be given to the vertical forces acting at the joint plates 2 and 3 (Figure 24 on page 32 (b)). To bear the resulting bending stresses, respectively to limit the bending deformation to an acceptable degree, thicker joint plates with a sufficient section modulus respectively moment of inertia are necessary. To avoid a strong increase of the dead weight these joint plates are actually designed as box-frame constructions. For the joint plate 2 follows that the upper and lower plate are rather thick. For the joint plate 3 a variable thickness and thus a better adaptation to the demands of mechanical resistance is possible. This leads to slightly thinner walls.

2.6.4.3 Connection of the Joint Plates and the Links

Some particular constructive measures have to be taken to make this connection bear strong forces. The rather weak front wall of the simple design (Figure 27 on page 35 (a)) can hardly bear the tensile forces of the joint plates. Only the small surfaces contacting the side walls have to transmit the tensile forces (lumped contact). Therefore, at these spots there is an immense concentration of stress. Even moderate tensile forces (welds) would make the structure collapse.
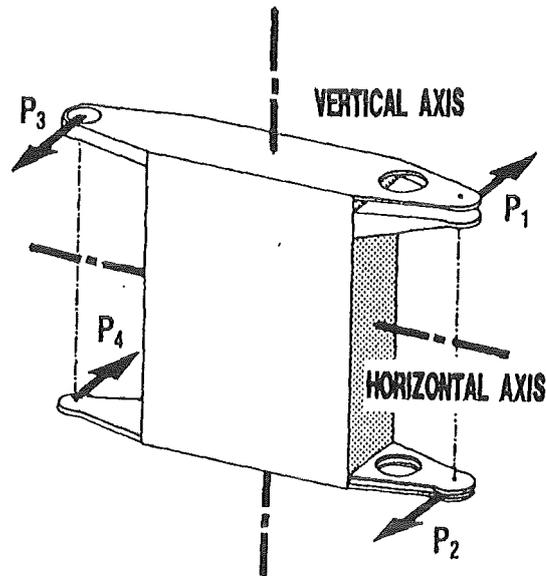
Figure 26. Forces Acting on the Link Hinges Distort the Link also along is Vertical Axis. This is the Second Reason for the Necessity of Sufficiently Dimensioned Front Walls.

The improved design (Figure 27 on page 35 (b)) produces an extended contact area with the force transmitting side walls by prolonging the joint plate into the interior of the link (line shaped contact area) . Thus hot spots of stress can almost be avoided. These prolongations of the joint plates are designed for the connection of the joints to the links.

2.6.5 Detailed Computations Using Finite Element Codes and Determination of the Wall Thicknesses

2.6.5.1 Building up a Finite Element Model

For detailed analyses the finite element model shown in Figure 28 on page 36 has been applied. It consists of 785 elements (8 node elements). The joint plates 2 and 3 are modelled as hollow boxes. Furthermore it was assumed that the joint plates 1 and 3 as well as 4 and 6 are connected by rigid joint pins.

The wall's thicknesses for each case are listed in Table 12 on page 35. It was assumed that due to constructive reasons the smallest wall's thickness should not be less than 5 mm.

The material behaviour was supposed to be linear elastic with a Young's modulus of $E = 2.1 \times 10^5 N/mm^2$. The front link B4 is not designed as a box but in a very simple way. Because of its clearly composed loading no finite element model was employed.
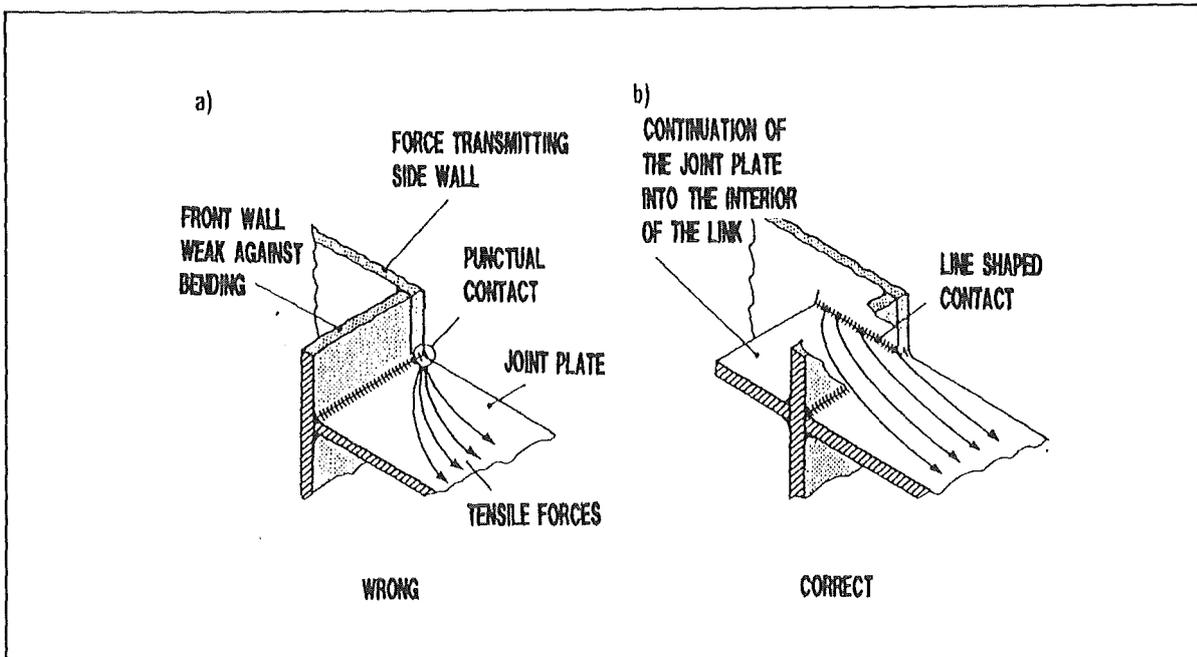
Figure 27. Connection of the Joint Plate with the Force Bearing Side Wall of the Link. By Continuing of the Plate into the Interior of the Link the Unfavourable Stress Concentrations are Avoided.

	B1	B2	B3
Links (upper and lower chord, side- and front-walls)	8 mm	6 mm	5 mm
Joint plate 1	15 mm	10 mm	10 mm
Joint plate 2 - horizontal walls	15 mm	15 mm	10 mm
Joint plate 2 - vertical walls	10 mm	10 mm	10 mm
Joint plate 3 - horizontal walls	8 mm	6 mm	5 mm
Joint plate 3 - vertical walls	8 mm	6 mm	5 mm
Joint plate 4	15 mm	10 mm	10 mm
Joint plate 5	25 mm	20 mm	15 mm
Joint plate 6	15 mm	10 mm	10 mm

Table 12. Wall Thicknesses of the Links B1, B2, B3 and their Joint Plates 1 to 6

2.6.5.2 Calculation of the Tip Deflection Due to the Payload with Different Configurations of the Links

To limit the computational effort for each link and each unit force, respectively unit force couple, shown in Figure 28 on page 36, the corresponding unit deformation was calculated in a finite element analysis. Because the actual forces or force couples are easily determined in a simple calculation. Subsequently a weighted superpositioning of the unit deformations yields the deflection of the tip. Only five interesting configurations are given in Table 13 on page 36 (reference configuration). It shows that the deflections of the wedged configurations are higher than the deflection of the straight configuration. In many cases the torsion contributes more to the deflection than the bending. Parallel to this, further analyses have been conducted which vary the wall's thicknesses of the links. The analysis with the smallest increase in the wall's thicknesses from B3 to B1 is of particular interest (variation 1). The resulting deflections are also given in Table 13
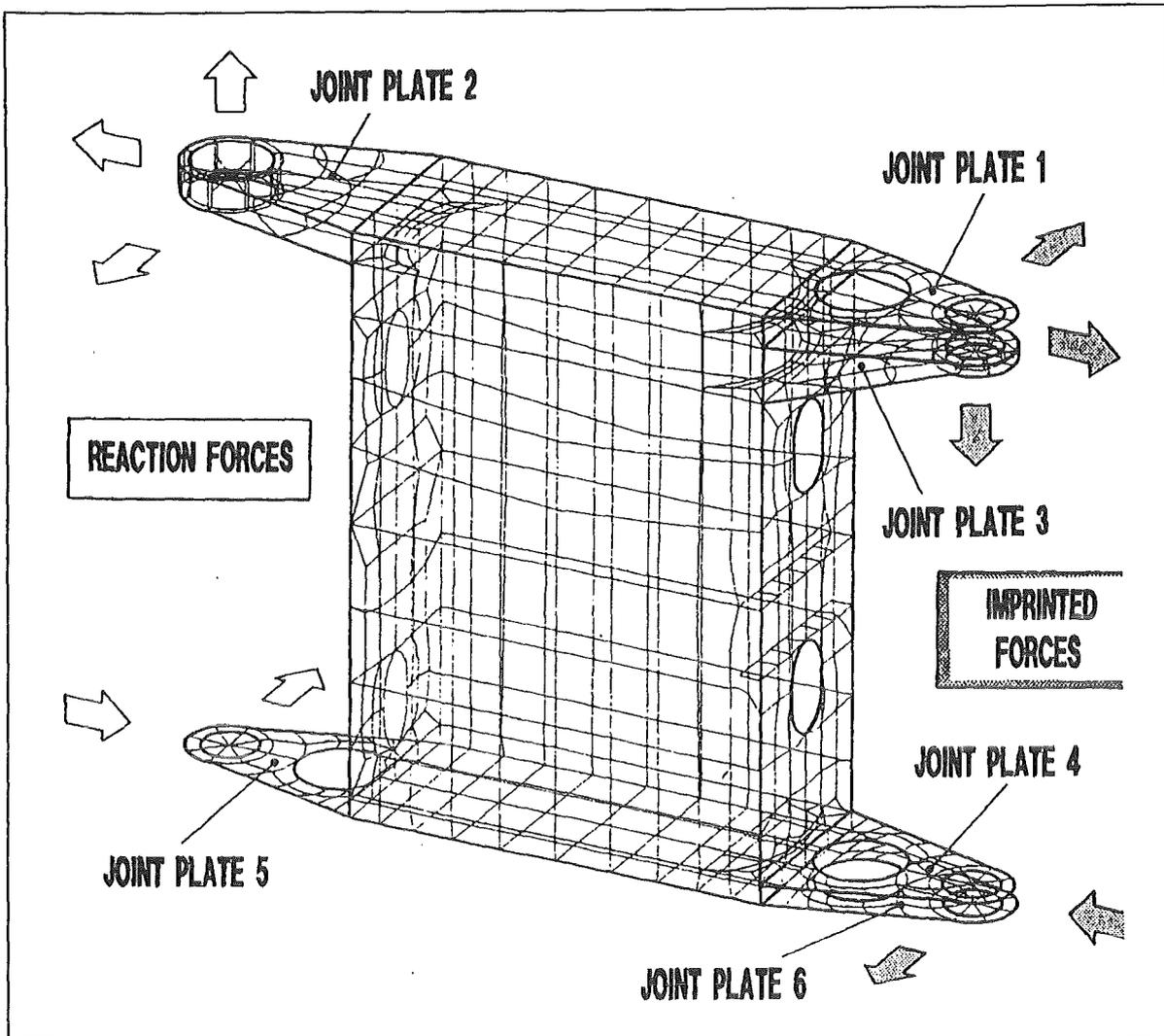
Figure 28. Finite Element Model for the Links B1, B2, B3

on page 36. They are up to 17% higher than these of the first analysis. The reduction of the dead weight on the other hand is only about 2%. All in all a certain increase in the wall's thicknesses is useful.

ABT Configuration	Reference case	Variation 1
Figure 29 on page 37, a)	12.3 mm	13.1 mm
Figure 29 on page 37, b)	17.9 mm	19.3 mm
Figure 29 on page 37, c)	32.2 mm	36.7 mm
Figure 29 on page 37, d)	35.8 mm	42.1 mm
Figure 29 on page 37, e)	28.8 mm	33.1 mm
Reference case: The links B3, B2, B1 have wall thicknesses of 5, 6, 8 mm. Variation 1: The links B3, B2, B1 have wall thicknesses of 5, 5, 6 mm.		

Table 13. Computed Deformation in Dependence on the Wall Thickness
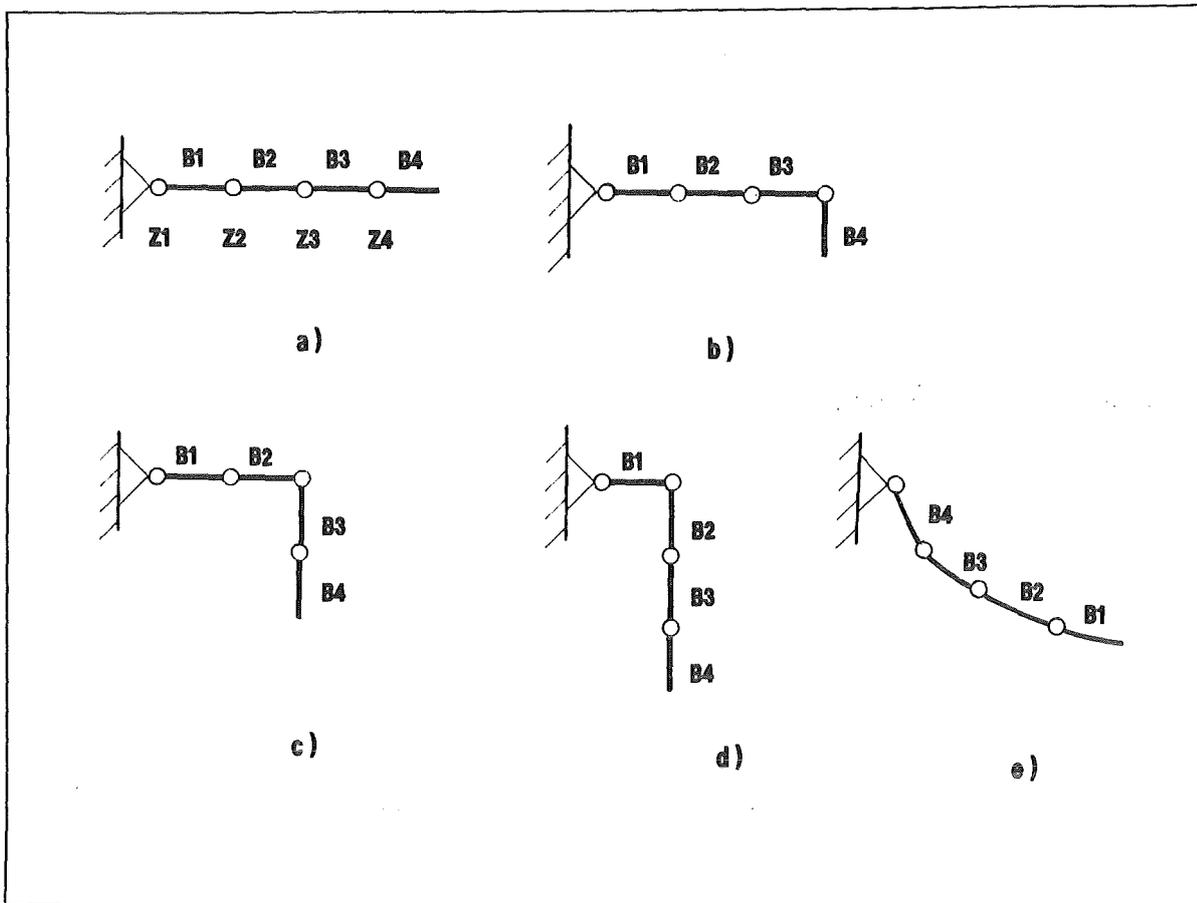
Figure 29. Boom Configurations for Stress Analysis

2.6.5.3 Computation of the Stresses Caused by Dead Weight and Payload under Unfavourable Configurations of the Links

The membrane and bending stresses in the wall computed with the help of a finite element analysis (in particular the von-Mises stress and if necessary also the corresponding stress components) have been displayed in many isoline plots. In most cases the numerical values lie far below the allowable stresses. Nearly all over the link B3 only 20% of the material's bearing capacity is used. However, as mentioned above, a reduction of the wall's thickness below 5 mm is not practicable. In the two backward links B2 and B1 the materials's bearing capacity is used to an higher extent but not totally exhausted. Reducing the wall's thicknesses of these links leads to an undesired increase of the tip's deflection. Not acceptable stress peaks may be avoided by, for instance, the constructive measures dicussed in chapter 2.6.4, "Design of the Joints and Specific Problems of Leading the Joint Forces into the Links" on page 31.

2.6.5.4 Buckling Analysis

Such kind of analysis have mainly been conducted for the huge side walls of the links, The critical buckling stess lies partly only 40% above the actual values. But this was accepted, because exceeding the buckling stress in plane shells does not lead to excessive deformations - contrary to cylindrical shells. This has been proved by analyses of not exactly plane side walls. The bulgy deformations are increased by dead weight and payload. For most unfavourable deformations (first eigenmode) according Figure 30 on page 39 the maximum deflection increases up to 40%. The shear stresses cause a diagonal orientation of the bulge. The negative bending stress in the lower part of the side walls which propagates the buckling moves the maximum of the bulge downward.

An important result is that the reinforcing rips are not necessary for the side walls. For a wall thickness below 5 mm these rips would be needed.

2.6.6 Comparison of Actually Occuring and Computed Deformations

After the assembly the deflections under payload and at the different boom's configurations have been measured at the boom's first joint (Z4 in Figure 4 on page 6). The results of this measurement are listed in Table 14, first row. For the last two configurations measurement was only possible up to the deflected link. In these cases the results for the deflection of the first joint could be calculated using the data of the straight configuration, bacause the wedged part of the boom is loaded like the corresponding part of the straight boom.

ABT-Configu-ration	Measured de-flection at the front link	After sub-traction of in-fluences from attachment flexibility	After extrapo-lation onto the boom tip	Computed de-flection of the boom tip
Figure 29 on page 37, a)	19.1 mm	11.7 mm	15.9 mm	12.3 mm
Figure 29 on page 37, b)	18.3 mm	13.0 mm	20.3 mm	17.9 mm
Figure 29 on page 37, c)	26.1 mm	21.3 mm	34.3 mm	32.2 mm
Figure 29 on page 37, d)	33.5 mm	26.1 mm	38.0 mm	35.8 mm

Table 14. Comparison of the Measured Deflection of the Boom with the Calculated Values

It should be noticed that the fixing link is also displaced in axial direction and has a sloping position. Both effects have been measured and their influence on the deflection of the first joint has been determined (Table 14, second row). This influence has to be taken into account if one compares the results of measurement and the results of computation, which postulates an inflexible fixing point. Furthermore, it has to be considered that the results of measurement are related to the first joint. For the comparison with the results of the computations which relate to the boom's tip, the results of the measurement have to be extrapolated to the boom's tip, too (Table 14, third row). The comparison of these extrapolated data to the calculated deflections shows that the latter are generally smaller. These deviations are known and are probably caused by the following reasons: Finite element computations do not consider details of deformation occuring in domains smaller than a finite element, deformations and free spaces in particular components are neglected totally (hinge pins, roller bearings, etc.), non linear behaviour of the material with reduced Young's modulus under high stresses which may be a result of the superpositioning with high initial stresses. Considering all these effects, the yielded accuracy of the computations is satisfactory.
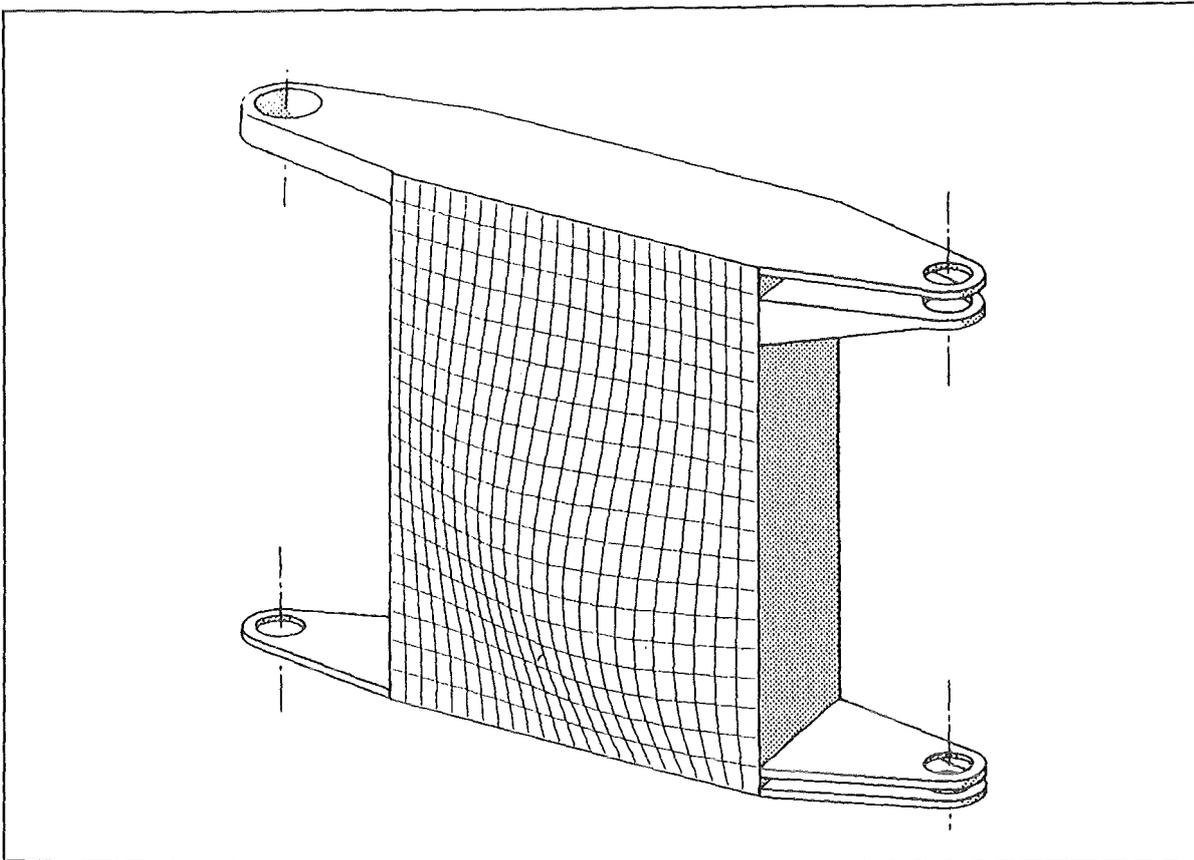
Figure 30. Buckling Analysis. Under load the deformation of the side wall increases up to 40%.

2.6.7 Conclusions of the Stress Analysis

Minimum deflection of the boom's tip under dead weight and under payload has been the essential requirements for the design of EDITH. The dead weight of the boom itself should have been in an acceptable range.

These requirements made necessary a more profound discussion of the basic mechanic effects. Finite element computations have been conducted not before a course design has been chosen and the associated hot spots have been determined.

The result has been a filigree construction. This is directly indicated by the joint plates. But also the walls of the box-shaped links are rather thin. So careful buckling analyses have become necessary.

After assembly the calculated deflections have been compared to those of the corresponding experiments. As it is known the mechanical properties of the articulated boom are satisfactory.

3.0 EDITH Guidance and Control System

3.1 General Concept

The work in the control system field has to be seen in a broader context: the control system architecture was designed with respect to the overall remote handling task, that means with respect to other remote handling areas as for example blanket handling and not only for EDITH or in-vessel handling. The reasons for this procedure were:

*   A uniform control system architecture for all areas makes the operation and maintenance of the whole RH control system easier and the development cheaper.

*   A clear separation of different abstraction levels in the control systems will result in a separation of generalized subsystems making the development cheaper and management more easy.

*   The concept of a remote handling workstation as a generalized subsystem on the highest level of abstraction will especially lead to a homogeneous man-machine interface and powerful operator support helping to minimize operator errors and maintenance time.

*   A uniform architecture and uniform interfaces of the RH area control systems make the cooperation between RH areas (e.g. in-vessel handling, ex-vessel handling) and the maintenance of the control system easier.

In the following subchapters the general concept and visions of an overall RH control system are shortly presented to show under which aspects the EDITH control system has to be seen as well. In this context the EDITH control system represents a prototype of a control system for a remote handling area and the prototype of a general purpose remote handling workstation.

3.1.1 Remote Handling Control Sytem Architecture

The **Remote Handling Control System** (RHCS) is a subsystem of the overall Fusion Plant Control System (FPCS) which consist mainly of the /12/ /13/(1) machine control and protection system, (2) the machine diagnostics, (3) the experimental diagnostics, and (4) the Remote Handling Control System (Figure 31 on page 41). The RHCS itself is partitioned into **RH-areas**, representing a set of RH devices grouped around a major device and normally dedicated to a specific group of tasks (e.g. in-vessel handling is done using a specific transporter as for example a boom and various specific devices for example the divertor handling unit, the antenna handling unit, general purpose master-slave manipulators, and the camera system). Each RH-area has its own **RH-area Control System** organized as a hierarchy of controllers (Figure 32 on page 42). To run a RH-area CS in the operational phase a **Remote Handling Workstation** (RHWS) has to be attached. For the commissioning and test phases each controller has its own special purpose **Test Interface** (TIF) which may also be used as a back-up interface. The only interface in the operational phase is the RHWS. Figure 32 on page 42 shows the hierarchical communication architecture of the RHCS. The RHWS has access to other RHWS controlling other RH-areas and to central RH services.

A central demand for the RHCS is to allow for two operational modes: manual performance of tasks and automation of tasks where possible, whereby the operator may select
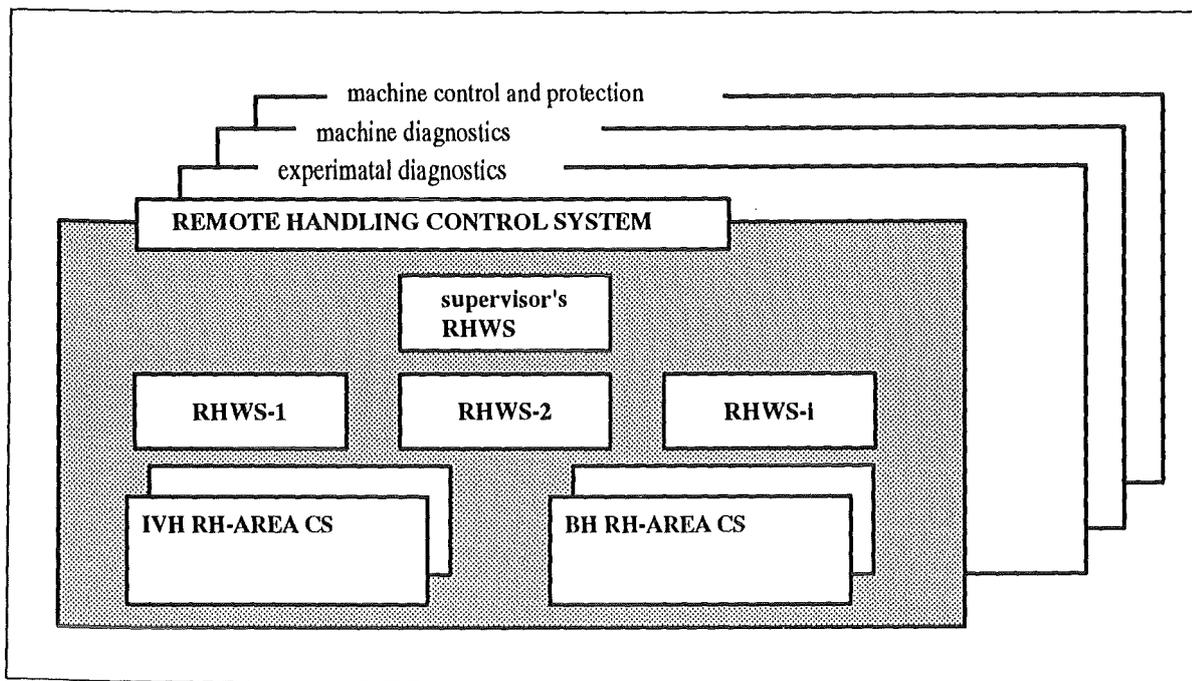
Figure 31. Fusion Plant Control System (FPCS). The RHCS is a specific part of the FPCS dedicated to controlling the remote handling devices. The RHCS is partitioned into RH-areas runnable via a RHWS. RH-areas are groups of RH devices used for a specific task as for example in-vessel handling.

which mode of operation may be most advantageous in a special situation. Having started an automated subtask the operator must have the possibility to intervene and to correct. The **mixed-mode control** is needed in such complex and partly unplannable situations as maintenance and repair of fusion plants. The manual work might be supported by computer means, the automatic work is supervised by the operator. Such a close cooperation between man and machine is characterized by mutual help and mutual supervison and optimal use of the cognitive and manipulative abilities of the operator. Figure 33 on page 43 shows the integration of the operator into the control flow. The operator is provided with a complete state presentation of the working scene and the devices. To guarantee highest operational flexibility the operator has access to all levels of control. He may control the motion of a device directly e.g. by a master device, he may start elemental or more complex functions, or he may act on the task level. This control system architecture is in accordance with other control system architectures used in related fields (e.g. EC-TELEMAN, space telerobotics). As shown in Figure 33 on page 43 the RHCS is based on a three level hierarchy:

1. The basic level of control is the **device/drive control level**. It is dedicated to the physical parameters and deals with geometry, motor currents, motion, system dynamics, and similar values. This control level is mainly realized in the so-called local device controllers shown in Figure 32 on page 42.

2. The **functional control level** deals with functions provided by one or a group of devices. Functions in this sense are: transport, drilling, flange assembly, welding etc.. Function control provides the basic features of working manually or automatically. In the latter case automated subtasks are started manually by the operator and supervised with the possibility of intervention. Functions are realized on a local device controller or in the case of more complex functions together with a coordinating area controller.

3. The highest level is the **task/mission control level** (also called telelogical or supervisory level). This control system level is dealing with networks of elemental functions (actions) representing a subtask as for example the flange unbolting. This
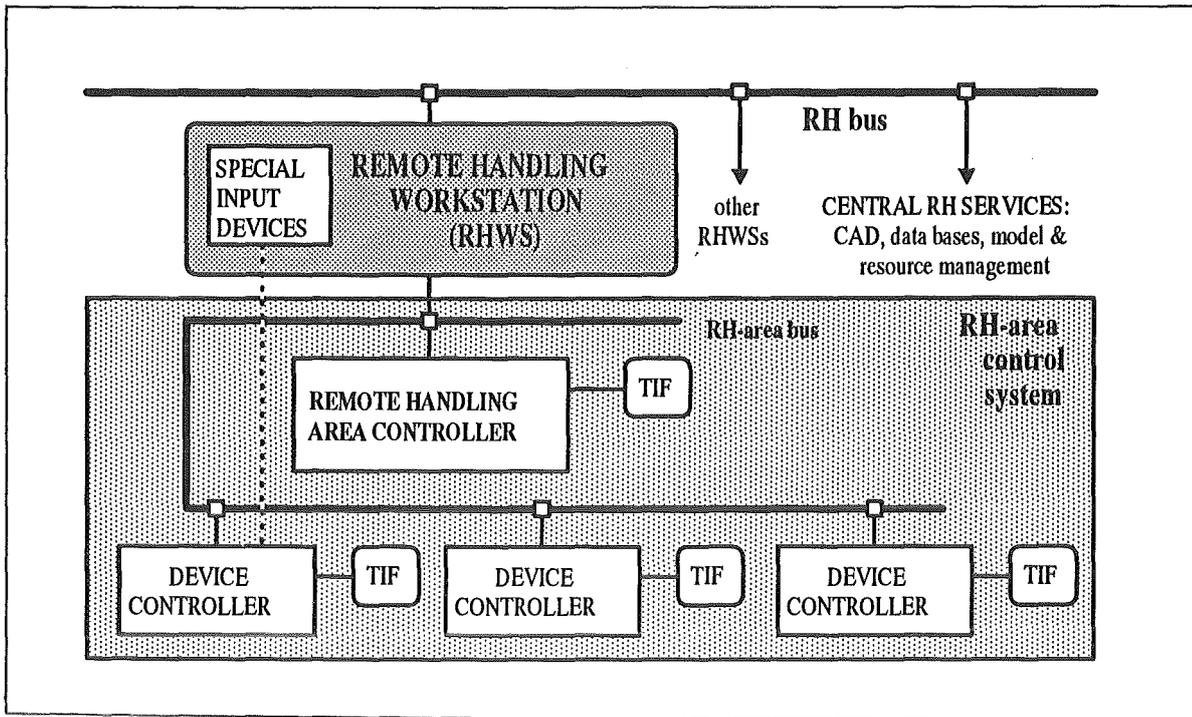
Figure 32. Architecture of the Remote Handling Control System. The RH-area control system integrates the devices in an RH-area to an operatable unit. In the execution phase a Remote Handling Workstation (RHWS) has to be attached to the RH-area control system. The RHWS may also communicate with central services and other RHWSs attached to other RH-areas. Dependent on the type of local devices special input devices e.g. for direct motion control may be needed. This input devices have to be directly connected to the RHWS but also to the related local device controler if response time is critical. Mechanically the input device is integrated in the RHWS working place. For test, commissioning , and back-up purposes all controllers have their own interface (TIF).

level provides the operational support needed for a fast and safe execution of the task. The level is also called the guidance level, it is represented by the RHWS.

3.1.2 Remote Handling Control Room

A special boundary condition for the design of a Remote Handling Workstation is its spatial and organizational environment: the scenario of a Remote Handling Control Room (RHCR), the requirements of cooperation between several tools, and the relation of the control via the RHWS or via generic device (tool) interfaces. The purpose of this chapter is to line out the RHWS working environment on which the design is based. As shown in Figure 34 on page 44 the remote handling control room may be devided into three areas:

1. The executive area with at least as many workstations as RH-areas have to be run in parallel. Because the RHWS concept claims for general purpose workstations further RHWS will enhance the availability of the RHCS. Depending on the specific requirements of a RH-area the RHWS will be equipped with specific input devices e.g. a master arm of a master-slave manipulator. In this case special manipulating skills of the operator are needed and the workstation may be run by two operators, one for manual motion control work, the other for the higher level tasks.

2. In the supervisory area one workstation may be used to coordinate and supervise the work of different RH-areas.
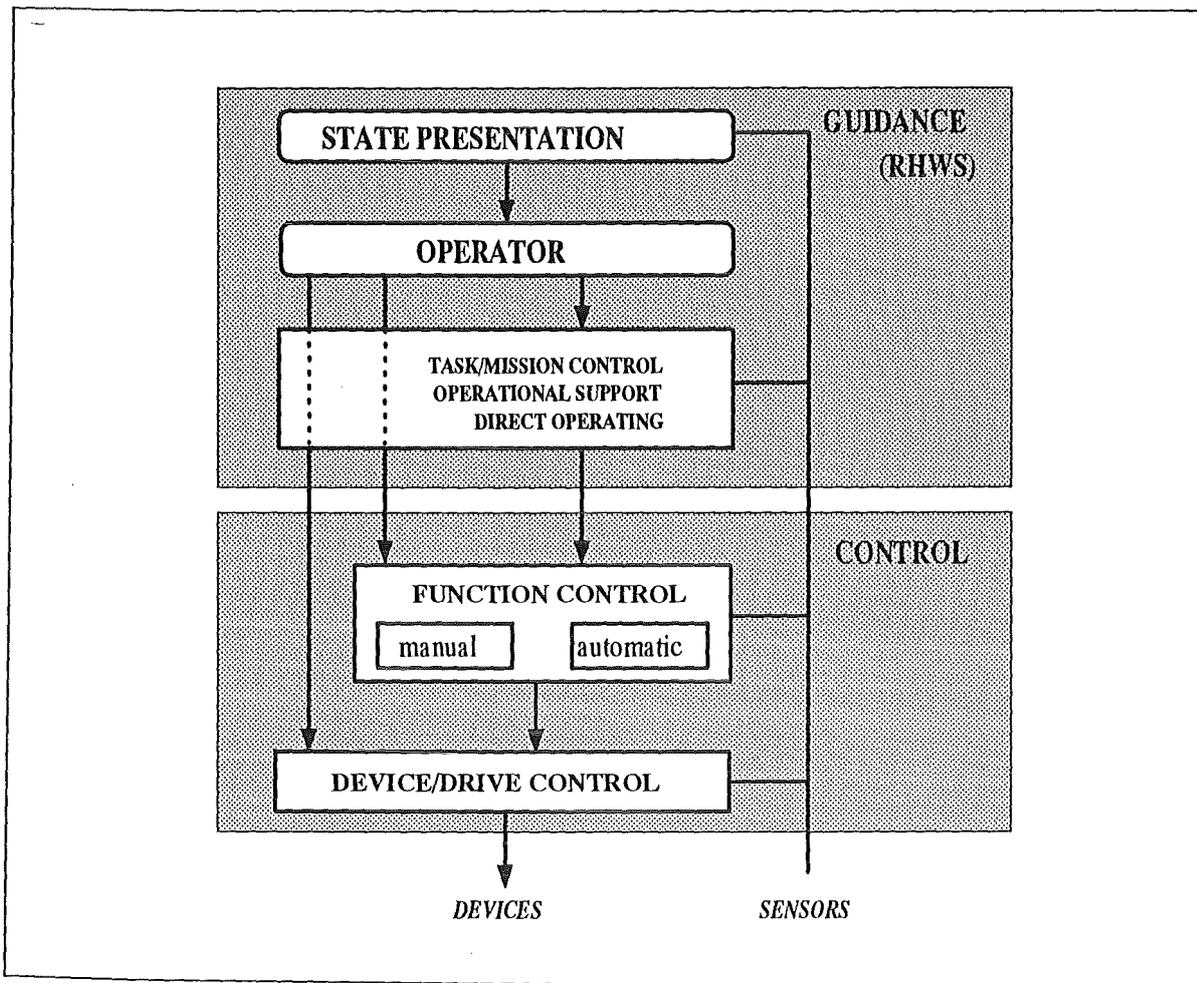
Figure 33. Man-Machine Cooperation and Devision of Labour. The operator is integrated into the control loop by providing him with all status informations and giving him the possibility of acting on the task control level as well as on the function/motion control and drive/device control level. The functions of the guidance or task level support subsystem are implemented as a RHWS.

3. In exceptional cases it may be necessary to run devices by their generic test and commissioning interfaces or by special back-up interfaces. Those interfaces are available in a special section of the RHCR or better in a separate room nearby.

A special problem in control room design is to find a solution for communication between the operators. Of course the executing operators will communicate directly and by pointing at the screens. The communication between different teams and/or the supervisor may be done, for important facts, formally through the RHWSs by a message transfer service. For this kind of communication the RHWS will provide a special operator-operator communication subsystem with predefined messages and additional form-free messages.

An additional communications medium may be a large common screen, composed of video projection areas and a display (e.g. liquid crystal) for status and message display. This type of communication is important and helpful in cases where overall pictorial information is relevant for all teams in parallel to their specific informations on their own RHWS. For working with these large screens each team has a pointer or cursor available for pointing at the screens.
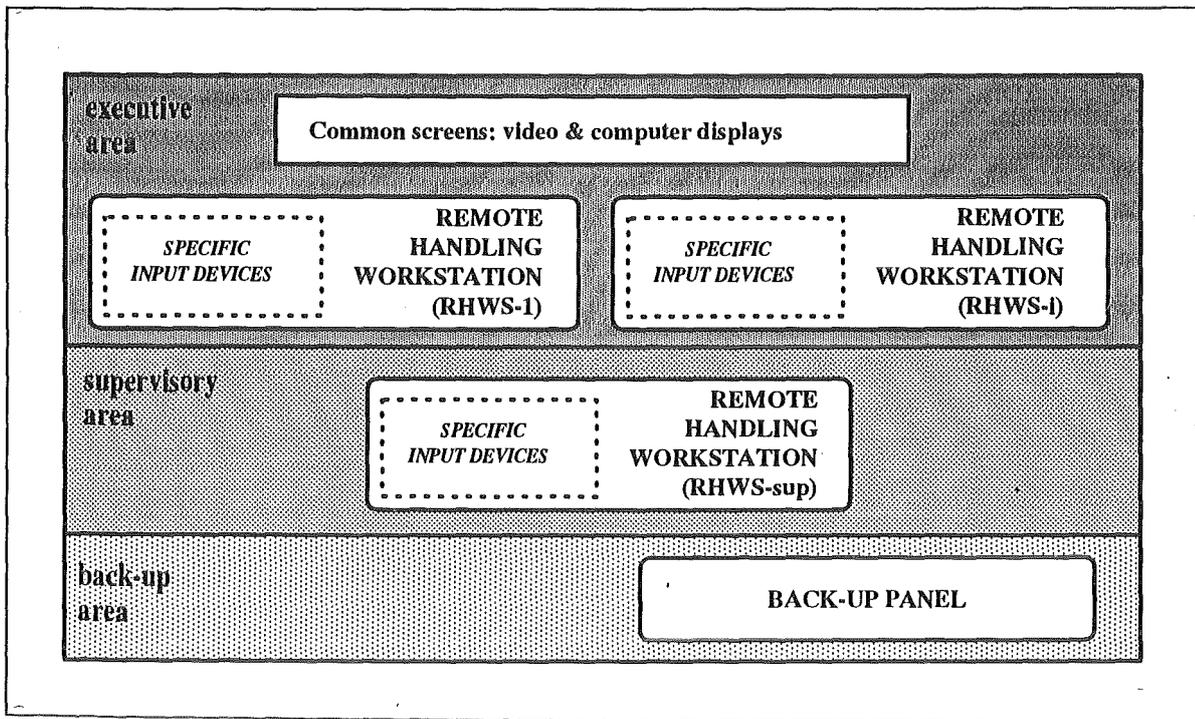
Figure 34. Remote Handling Control Room Layout (functional). This functional layout only shows the different components and working areas which have to be considered in a spatial layout.

3.1.3 Remote Handling Workstation

According to the state-of-the-art in related fields of operating for example power plants or space stations, the working place for the operators in fusion plant maintenance should be a computer based workstation. This provides a high flexibility in adapting the system to new working situations, to new technological situations, or to new requirements resulting from experiences. In addition, the hardware and basic software can be standard components of the computer market with all their benefits. The major point of using computer based techniques is the possibility of providing **operational support in all phases of work and on all levels of work abstractions** and allowing for a comprehensive **man-machine cooperation** based on mutual help and mutual supervision. This will help to fulfill the general requirements for maintenance:

* Minimize time for maintenance and repair
* Minimize time losses due to operator errors
* Ensure safe operation
* Minimize time for introducing enhancements or special solutions to unexpected situations (extendability, integration)

To meet these requirements especially operational and task oriented support is provided in addition to low level support of the operator for example during manual motion control. The EDITH CS is a semi-autonomuous system that means, that some subtasks are automated. The operator has to start those subtasks and has to supervise them. In case of problems the operator may intervene, overcome special situations or sequences which are not yet automized manually, and then restart the automatisme or start another automatic subtask. To be able to manage unforeseen situations the operator is supported in **planning or replanning and simulating working sequences** ad-hoc and at-once based on the actual state. The workstation supports this scenario of mixed-mode control and copes with the evolution of getting the operator step by step out of the inner, low level, high bandwidth control loops onto the tasking level of control.

High level or task level support in the execution phase of work means to provide at least information written down in handbooks. The ERWS therefore on this level provides an **interactive mutimedial handbook/notebook** which is available off-line in the preparation and training phase but also on-line to the maintenance equipment during execution to guide the operator synchronously to the task execution step by step through the procedures and to help him in unforeseen situations according to the actual state of the work.

The ERWS provides an **integrated working environment** supporting the operator not only on the tasking level but also on the functional and device level. The complexity of the maintenance task demands for a wide variety of different devices which have to be controlled manually in many situations. The RHWS should therefore provide a **compact homogeneous interface** integrating these functions and devices to an operatable system. Comprehensive **training** possibilities are important because the remote maintenence tasks are performed relatively seldom.

Summarizing, a Remote Handling Workstation

1. represents the **interface** to the remote handling devices and the working scene and

2. provides **software support tools** for the operator during all phases of the maintenance work: planning, training, execution and monitoring, and in error recovery.

More details about the general concept and applications can be found in /14/ /15//16/ /17/ /18/. The EDITH Remote Workstation and their components are described and discussed in 3.2, "EDITH Remote Workstation: A Prototypic Guidance System" on page 46, 3.5, "PEXOS: Procedural Simulation for ERWS" on page 80, 3.3, "KISMET: Spatial Simulation for ERWS" on page 60, and 3.4, "Control Panel of the Direct Operating Interface for the EDITH Control System" on page 78.

3.1.4 RH-Area Control System

The RH-area control system (the control system) realizes the device control by a hierachy of device (motion) controllers and an integrating area (cell) controller. As shown in Figure 32 on page 42 the area controller has access to all device controllers via the RH-area bus through which also the RHWS may access the area components. The task of the area controller is:

* Device coordination (e.g. safety functions as allowing for motion of device x if device y is not moving)
* Management of area data (e.g.parameters, programs)
* Communication with RHWS, direct communication between device controllers and RHWS is also possible but should be restricted to special cases.

All device controllers and the RH-area controller have its own test interface (TIF) which also can be used as back-up IF. But in this control system concept it is of high importance that all devices have two further access channels providing all functions available on the controller. These channels are needed for the connection to the RH-area controller and the RHWS.

The general functions of the RH-area controller and/or the device controllers available to the RHWS are the following:

* Program and data list management
* Start, stop, and interruption of programs
* Reading and writing of state variables

- Triggering of functions by events
- Alarm management, safety functions
- State description information: sending state infos on request and as a constant flow (asynchronously, synchronously) for monitoring.
- Switching between manual mode (single command execution, immediate mode) and automatic mode (program execution with possibility of interventions)
- Single command execution of devices, providing all functions of the device including program execution
- On-line teaching of devices
- Logging and backtracking

In principle these list shall express, that all functions and data relevant for using the device and preparing the device have to be accessible through messages for the RHWS and/or the RH-area controller.

The **EDITH control system** as a prototypic implementation of a RH-area control system is described in 3.6, "Control System" on page 91.

3.2 EDITH Remote Workstation: A Prototypic Guidance System

The EDITH Remote Workstation (ERWS) is a prototypic implementation of the KfK proposal of a future Remote Handling Workstation (RHWS) for fusion plant maintenance. Its design is based on experiences especially at JET /19/. The implementation is based on standard workstation hardware and software following accepted software design principles. The implementation of the ERWS has two goals:

1. To provide a working environment for the operator of the EDITH experimental system.

2. To evaluate which components and functions are needed in a future ITER general purpose Remote Handling Workstation.

3.2.1 General Features

The remote handling equipment (e.g. EDITH arm with end-effector positioning unit EEPU, manipulators, cameras, lights) will be controlled from the EDITH Remote Control Room (ERCR). The working place for the remote handling operators will be the ERWS, which represent the only interface to the remote handling process during normal maintenance operations. The ERWS provides access to all software or hardware tools which the operators need directly for performing the maintenance work. During the work the ERWS is attached to the control system of EDITH (IVH-area).

General features:

1. **Multi-level support and state presentation**: The operator has to execute *procedures* under *spatial restrictions* in a well defined working space using remote handling devices which provide him *remote handling functions*. This three level model is quite natural for an operator and therefore the ERWS provides a related multi-level support and state presentation. The ERWS opens basically three windows onto the scene representing three different views:

   - A procedural view dealing with the description of work which has to be done, representing a network of system states and operator actions
   - A spatial view, showing where the procedures have to be executed
   - A functional view onto the devices, which are available for task execution

2. **Task decomposition support:** Tools to support breaking down of tasks to subtasks and elemental functions.

3. **General purpose application:** The ERWS is designd as a general purpose workstation adaptable to all scenarios mainly by data exchange. This provides or enables:

   * Multiple usage of RHWS as far as possible
   * Identical hardware and functional interface arrangement for all tasks
   * High availability by redundancy (one RHWS can replace another)
   * High performance hardware because of multiple usage (e.g. high performance graphic workstations)

4. **Support of man-machine cooperation:** To enhance efficiency and safety by using special abilities of both partners.

5. **Multi-phase support:** Operator support is given in all phases of the maintenance work.

6. **Integration:** The ERWS integrates the various software and hardware support tools needed by the operator to one system such that data exchange between subsystems is possible.

7. **Operatability:** All remote handling devices (e.g. in-vessel handling unit, blanket handling unit, divertor handling unit) have, in principal and as far as possible, a very similar interface for the operator. The various device interfaces are as homogeneous as possible. This is important because the complexity of the maintenance tasks demands for a large variety of special devices and subsystems. The ERWS provides means to hold the system operatable.

8. **Concentration of general purpose tools:** General purpose support tools of the RH control system (RHCS) e.g. geometric/kinematic information processing for various purposes, camera control, are concentrated into a special system, the RHWS, for making them as efficient and convenient as possible. They are not distributed over the various controllers. Task-level support functions (operational support) for the operator such as a simulation, programming aids, or the operator guidance are implemented on specialized hardware and in a special software environment and is separated from low level control system functions, that means they are implemented on the ERWS.

9. **Openness to automation:** The ERWS as an operational support tool can cope with the stepwise introduction of more and more automatic subtasks into the remote handling toolbox by providing the possibility of planning the usage of those automated subprocesses and their supervision during execution. The semi-autonomuous maintenance is the final goal. The operator has the choice of doing the work manually or by automated subsystems.

3.2.2 Task Related Features

The ERWS has in general to enable the operator to perform the specific work as fast and as safe as possible.

Therefore the following task related features are available:

1. The ERWS provides a **representation of all devices and functions** that are available for maintenance and repair in a task suited manner.

2. The ERWS provides a **representation of the environment** (the machine, the plant) involved in the maintenance and repair work.

3. The ERWS provides a **representation of the procedures** to be executed

4. The ERWS supports **coordinated collaboration** of different devices used for a task.

3.2.3 Operator Related Features

The following list describes the additional subtasks that an operator may or has to perform during the execution of a maintenance procedure. The ERWS provides means to support the operator in doing these subtasks:

* **Preparation of a the ERWS for a special task**
* **Studying procedures**: This task is usually done using handbooks.
* **Training of complete procedures**
* **Preparation of site**: The workplace at the workstation and the remote site have to be prepared for the work.
* **Allocation of resources**: The operator has to find out whether the hardware and software tools he needs are available for him and he has to allocate them.
* **Simulation of procedures, adaption to actual state of the site**: Before execution the operator should simulate the whole procedure to become familiar with it and to see whether it is in accordance with the actual situation.
* **Start executive work**
* **Do manual work**: Direct device control (for example with a joystick and special device suited windows, similar to hardware control panels of devices).
* **Supervise automatic work**
* **Intervene**: Correction of malfunctions in a preprogrammed procedure.
* **Plan unexpected work**: In case of unexpected events the operator has to plan new procedures (work sequences).
* **Simulate new work sequences**: Newly planned work sequences should be simulated to guarantee their executability.
* **Search in computer lists or libraries for available special purpose tools (in a general sense) to cope with new situation**: Software support tools like special preprogrammed procedures and hardware tools like for example special grippers.
* **Search in computer lists or libraries for comparable preprogrammed procedures**
* **Execution monitoring**: The operator is supported in checking that procedural steps and motion steps have been completed as planned.
* **Failure diagnosis**: In case of malfunction the operator is supported in exploring the actual state of the RH system and in finding failures.
* **Logging**: For documentation, review, and learning but also for simply stepping back in complicated situations.
* **Helping**: In each operational situation a help facility is available supporting the operator in all aspects of his work.
* **Annotating**: The operator has the possibility to note specific experiences in relation to the actual state of the task so that the notices are easily available in case of repetition. This is a first step in the direction of acquiring heuristic knowledge to make the system more intelligent.
* **Programming**: The operator is supported in bundling (programming, teaching) approved working sequences to new "commands" (macros) such that they are usable later on. This programming feature is available on all levels of abstraction: for motion and device control, for function control, and for procedure control.

3.2.4 Specific Support Features

The ERWS provides

1. the interface to the remote handling devices and to the working scene,

2. operational support in form of tools for planning, training, execution, monitoring, and error recovery

to increase the operator capabilities, his effectiveness and efficiency, and to minimize the failure rate. Table 15 on page 50 shows in an overview which functions the ERWS provides basically in the different phases of work and related to the different views onto the work. These different views are introduced and used for classifying the support functions. The classification is quiet natural in representing the three levels of abstraction: **device/physical level** describing the geometry and physical parameters of the devices (e.g. move a transporter in a cell, motor current, transporter dynamics), the **functional level** of abstraction focusing onto the functions which may be performed using a device or a device group (e.g. transporting a device, drilling, welding), and the **procedural level** dealing with the task and their subtasks (e.g. divertor exchange).

In particular, the ERWS provides the following functions and aids:

1. **Planning support:** In the basic planning phase of the complete maintenance and repair work as well as in the replanning or additional ad-hoc planning phases during error recovery the workstation provides planning aids. These aids support

General RH goals	ERWS features to reach the goal
minimize time for maintenance	- automation of subtasks (elemental functions) - computer support for manual operation - facilitate close man-machine cooperation - provide mutual help and mutual supervision
minimize time losses due to operator errors	- enhanced viewing (synthetic viewing) - automation, semi-automation - training for operator - computer guidance for operator - complete integrated documentation of procedures
safe operation	- motion monitoring - procedure execution monitoring - function execution monitoring - simulation in planning and programming
minimize time for introducing enhancements or special solutions to unexpected situations (extendability, integration)	- modular architecture - software implementation - standard development & maintenance tools - thoroughly defined internal interfaces (hardware & software)

Table 16. EDITH Remote Workstation Features in Relation to General Remote Handling Goals. The table shows by which ERWS features the operator is supported in performing the maintenance task in an effective way.

WORK PHASES	MODEL TYPES / LEVELS OF ABSTRACTION / VIEWS		
	PROCEDURE	SPATIAL	FUNCTIONAL
PLANNING	action/state net generation, task simulation	path generation, motion simulation	parmeter set generation, device simulation
TRAINING	exploration of tasking structures	exploration of spatial restrictions	exploration of equipment functions
EXECUTION	state dependent advices & guidance	integrated (synthetic) viewing	interfaces to device functions
MONITORING	task execution	collision avoidance	limit surveillance
RECOVERY	backtracking, support in diagnosis and alternative action subnet selection	support in correction or adaption of paths, backtracking	support in correction or adaption of parameter sets

Table 15. Basic Functions of Remote Handling Workstation. The selection shows examples of workstation functions suitable for the different work phases based on the different model types.

●   the development of remote handling procedures (e.g. represented by networks of system states and operator actions) in various degrees of detailing,

●   the development of motion pathes for the remote handling devices in the workspace, and

●   the generation of parameter sets for problem suited usage of the devices.

The result of the planning phase are three types of models representing the work in the three basic views onto the task. These three models are the basis for the complete simulation of the maintenance task showing what has to be done, where the task has to be done and how the working space is used, and how the devices can and should be used in the task execution.

There are basically two types of planning:

●   Planning of expected tasks in full detail providing a complete documentation of the procedures for various purposes

●   Ad-hoc and on-the-fly planning of unexpected subtasks. In this case a fast and easy performance of the planning-programming-simulation cycle is important for the operator to get fast an error free executable solution for the problem.

Planning includes programming because the result of the planning is in each case some sort of program in the sense of e.g. standard robot programs, or advices for a sequence of manual device usage, or a sequence of general advices for the operator on the tasking level. Programming support on the RHWS is provided on three levels:

a.   Programming with robot programming languages like PASRO, IRL, or IRDATA is provided by KISMET (or related subsystems) and requires qualifications and work as for C or Pascal programming.

b. Macro/script programming of the devices is provided by the DOIF-SS: bundling of generic device commands in an interpretative manner like shell programming in Unix.

c. Graphiccal graph based programming including advices to operators is provided by PEXOS on the tasking level.

2. **Training support**: Because the maintenance and repair tasks have to be executed not very often the operator has the possibility to get familiar with the task. This training is based on the simulation of the tasks and training suited remarks and guidance for the operator using also experiences gained during former task executions.The operator has the possibility

   *   to explore the task structure, describing which system states have to be passed and which elemental subtasks have to be executed,
   *   to explore the working space and the spatial abilities of the devices, and
   *   to explore the functional abilities of the devices.

3. **Execution support**: To minimize the rate of operator failures and to guarantee as far as possible that the procedures are executed as planned.

   *   the operator is guided step-by-step through the network of system states and actions by telling him what he has to do, how he should do the work, what he has to obbey. This guidance and advices are given comparable to a handbook, but enhanced according to the possibilities given by modern computer techniques: This means the workstation represents an interactive multimedial handbook and notebook (memo-book).

   *   The most important tool with respect to execution support is an integrated viewing feature, combining sensor controlled threedimensional computer graphics and standard TV-viewing to enhance scene presentation. Computer graphics is especially be used to generate overviews and special views not available by TV but computer graphics is also used to control the cameras.

   *   To control the various remote handling devices direct operating interfaces (DOIF) for all devices are available allowing for device control according to the control via the hardware control panel (if available) or the basic control interface for stand-alone device operation and/or testing.

4. **Monitoring support**: Monitoring is based on the three models available for planning and simulation: in the monitoring phase the process controls and/or synchronizes the simulation. Basic monitoring is done by the operator using the multi-level state presentation. Enhanced monitoring recognizes automatically deviations from the planned execution. Collision avoidance based on the spatial models and device parameter surveillance is the basic automatic monitoring module. Detected failures are presented to the operator in a readable form, the system is set in a secure state.

5. **Recovery support**: In case of errors the workstation helps the operator in recovering from the error state. Basis is a backtracking possibility (e.g. based on a logging facility). In addition the system supports the operator in replanning: generation/adaption of new motion pathes according to a new situation and generation/adaption of parameter sets for the devices.

There are two general functions/services available:

*   **Logging**: A special logging module keeps track of all actions of the operator and the system. This log-file is needed for review, analysis, and backtracking. Logging is

done on the different control levels separately: on the tasking level as well as on the functional and device/physical levels.

- **Annotation**: To learn as much as possible from experiences the operator has available on all operating levels a notebook function which allows him easily to make annotations related to the actual action.

Table 16 on page 49 shows in addition the special aim of the functions in relation to the general goals for maintenance work execution.

3.2.5 Architectural Features

The implementation is modular to make the system flexible for adaption to different working situations. The ERWS is configurable for different demands of the special task. All the basic modules are runnable alone to provide for example a minimally equipped ERWS. This modularity also facilitates a stepwise enhancement of the whole system with evolving state of the art. Another advantage of high modularity and loose coupling of modules is the easy testing, commissioning, and maintenance. Modularity is provided to facilitate different implementations with respect to the hardware: the possible allocation of modules to the hardware (computers) is flexible.

3.2.5.1 Functional Architecture

The functional architecture differentiates between three main modules of the ERWS (Figure 35 on page 53):

1. KISMET Kinematic Simulator, Monitor, and Programming Environment for Telemanipulation /20/) for spatial simulation and control
2. PEXOS (Procedure Editing, Simulation, Execution, and Monitoring System) for procedural simulation and control
3. DOIF-SS (Direct Operating Interface Subsystem) for functional/device oriented control and simulation

3.2.5.2 Kernel Subsystem Architecture

The kernel subsystems KISMET and PEXOS of the ERWS have an architecture as shown in Figure 36 on page 54. The DOIF-SS as a simplified implementation of a general functional simulation system is not based on an explicite model data base. Around a model data base six basic modules are grouped working with this model:

- The **model presenter** serves for visualization of the model for the operator.
- The **simulator** module provides the motion simulation based on the 3D model.
- The **monitor** compares actual states of the reality with planned states (simulated states) or checks actual parameters against their boundaries. An example of a monitoring subsystem in the case of spatial simulation is collision detection.
- To generate or to correct models an **editor** is available.
- Communication with the control system for realtime model updates (e.g. by joint angle sensor signals for geometric model or program execution messages for procedural model) and other RHWS subsystems are processed by two **message processors**.
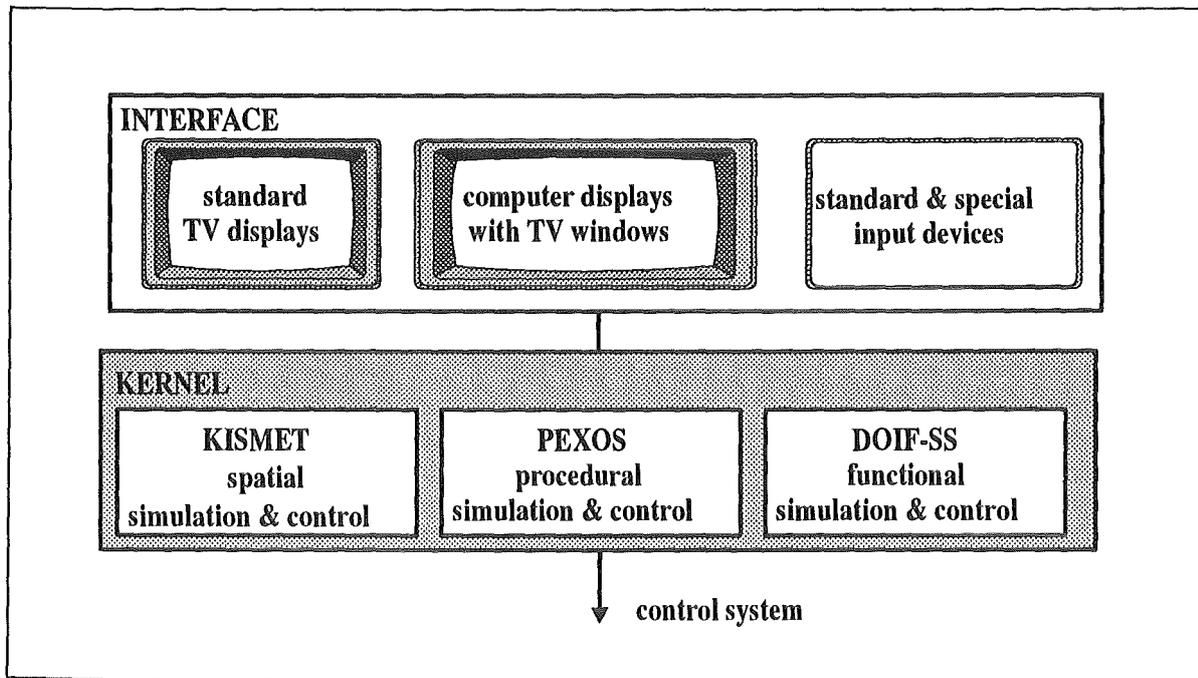
Figure 35. Functional Architecture of EDITH Remote Workstation. The ERWS consists of three basic sybsystems: KISMET, PEXOS, and DOIF-SS. The interaction between the operator and the workstation is done by standard input means and special motion control devices and three computer displays with live video windows and additional video monitors.

3.2.5.3 Hardware and Communications Architecture

According to the actual state-of-the- art in computer and communications the ERWS functions are implemented on of-the-shelf computer workstations. The basic difference in workstation technology is the graphics subsystem performance. Because the requirements for the implementation of the ERWS functions are different (high graphics performance for spatial simulation, knowledge engineering implementation tools for procedural simulation), the optimal solution was to use three different workstations for the three kernel subsystems. The hardware and communications architecture of the prototypical implementation of basic ERWS functions is shown in Figure 37 on page 55. The communication means are flexible enough to allocate subsystems on different computers (possibly with performance reduction). An example from the EDITH test environment for a minimal configuration of a ERWS: KISMET, DOIF-CAMERA, and DOIF-EDITH all together allocated on one workstation of type Silicon Graphics IRIS to run EDITH only with spatial support and a minimum of functional support. The communication is configurable by a special communications module (CCSS: Configurable Communication Subsystem) for each workstation. These communication modules provide the configuration dependent adaptions to the general purpose kernel subsystems.

3.2.6 Functions of Spatial Support Subsystem KISMET

Spatial simulation deals with the physical properties of the devices and the environment. Spatial simulation is based on a geometric/kinematic model of the equipment and the environment. These models are derived from the design data of the devices available on CAD-systems.

The prototype of such a spatial simulator is the KfK-KISMET system which is also used for the JET-TARM control and other non-fusion applications (e.g. space telerobotics). The predecessor of KISMET is GBsim developed and tested for the JET-BOOM control.
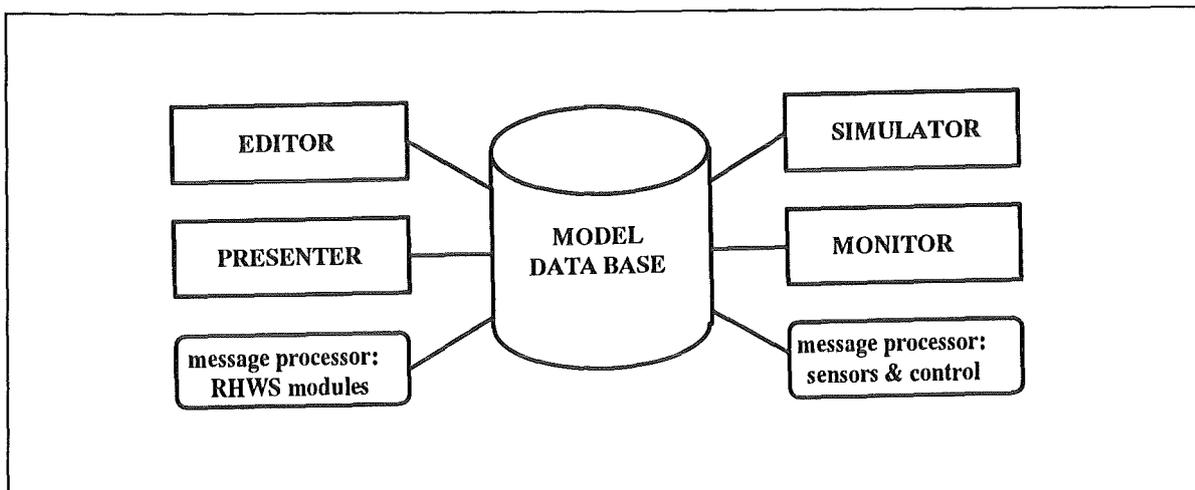
Figure 36. Kernel Subsystem Basic Functional Architecture

Basic features of the spatial simulation on a RHWS are shown in the following list. (see details in 3.3, "KISMET: Spatial Simulation for ERWS" on page 60 and /21/). All KISMET functions are accessible via a special textual command interface by keyboard and data communication link to other RHWS modules. This interface accepts also files of commands for execution (script files).

General geometric and kinematic features:

* **Model hierarchies** to scissor detailing to needs of task
* **Flexible kinematic description** of RH equipment
* **Parametric geometric elements**: Elements with variable geometry, e.g. crane ropes.
* **Task dependent rendering**: Rendering modes according to the state-of-the-art in graphics rendering are provided allowing a task oriented variable presentation of the scene.
* **Geometric auxiliaries**: Special geometric data like working coordinate systems, camera view pyramids are integrated into the scene if required for the actual task.
* **Editor with video option**: To manipulate models transferred from CAD or to generate new models or to correct models. To facilitate editing and quick model verification a video overlay option is available.
* **CAD interface**: Model transfer from CAD systems.

Motion oriented features:

* **On-line model update**: The variable model parameters are controlled by sensor signals if the simulator is running on-line.
* **Integrated Viewing**: A subsystem providing the integration of camera viewing, synthetic viewing, and camera control including camera tracking.
* **Path generation** for usage in special robot programming
* **Robot programming**: Basic features of editing robot programs executable on the device controllers to be able to work in the programming - simulation - execution loop for short not preplanned subtasks.
* Direct motion control of devices by using KISMET models of the devices as **synthetic master systems**
* **Program interpretation**: Control programs for the various devices are accepted for simulation (e.g. in the standard format IRDATA).
* **Inverse kinematics**: Real-time algorithms for the inverse kinematics including those for redundant kinematics to provide TCP (Tool Center Point) control.
* **Input device interface**: Especially for training purposes various input devices (e.g. master of master-slave manipulator) are attachable.
* **Collision detection**
* **Distance measurement**
* **Motion traces**
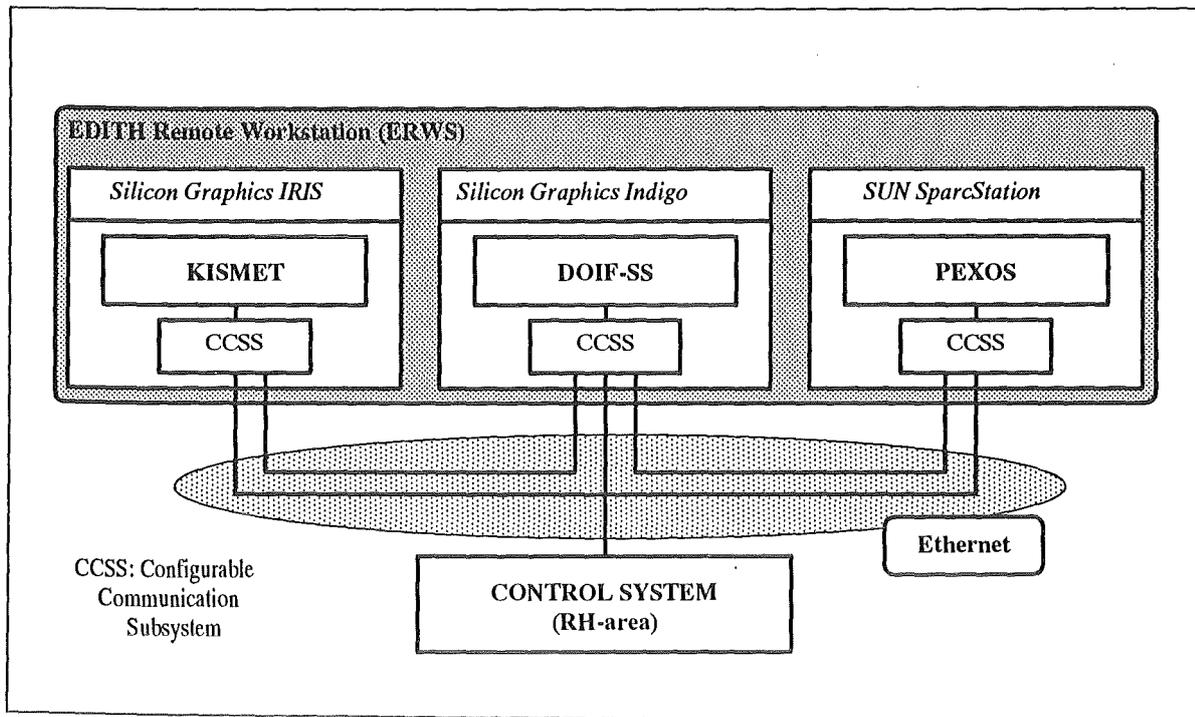
Figure 37. Hardware and Communication Architecture of EDITH Remote Workstation. The integration of the ERWS subsystems and the control system is realized by the CCSS (Configurable Communication Subsystem) via Ethernet.

Dynamic features to enhance modelling realism:
* **Real-time simulation of static deflections**
* **Real-time simulation of motion dynamics**
* **Real-time simulation of control system characteristics**

The spatial simulator is the basis for spatial monitoring, spatial guidance, motion planning and programming and for training of manual motion control. The functions of the spatial simulation system are accessible not only via its own interactive interface but also via a message interface providing access from other ERWS modules.

┌─── **Development status** ─────────────────────────────────────┐
│   │
│ KISMET is already a well known, approved, and self contained system in the status │
│ of a product available from KfK.   │
│   │
└───┘

3.2.7 Functions of DOIF-SS

All devices and functions of the remote handling system have to be accessible for manual control via the ERWS. This demands for interfaces on the RHWS for all these subsystems: The Direct Operating Interfaces. For example, a DOIF for the camera system provides a panel similar to the standard hardware control panel of a camera system, that means providing at least all the control functions of a hardware panel. The DOIFs represent (together with KISMET in case of motion devices) the devices on the workstation.

General features of the DOIF-SS are (for more details see 3.4, "Control Panel of the Direct Operating Interface for the EDITH Control System" on page 78):

* Enhanced support (e.g. graphical) to make device operation easier and more safe

- **Management of DOIFs:**
  - Initialization and positioning of DOIFs at system start according to a configuration file
  - Communication between the DOIF system and other RHWS modules
- **Helping:** Provision of operating guidelines, advices, help information.
- **Annotating:** Provision of operator notes management related to operating means (e.g. buttons).
- **General DOIF control functions** avilable for PEXOS or other external subsystems to unburden the operator from control work and to prevent operating errors:
  - Opening, closing (iconizing) DOIFs
  - Positioning of DOIFs
  - Activating, de-activating of control buttons or other input means
  - Activating, de-activating output means
  - Setting selection values or default selection values
  - Highlighting of control elements
- **Connection to KISMET** for device and function simulation
- **Textual command interface** for all DOIF functions and the ability of defining command macros to make manual control and coordinated usage of the devices more easy
- Operation relevant **parameter exchange** between DOIFs (devices)
- **Macro/script management** for bundling device functions of different devices

┌─ Development status ──┐
│   │
│ The DOIF-SS is running in a basic version, it will be ready in June 1993. │
│   │
└──┘

3.2.8 Functions of Procedural Support Subsystem PEXOS

PEXOS describes maintenance procedures as a sequence or a network of actions (e.g. gripping, welding, drilling, screwing) and facts (states). The start of an action normally is restricted by specific conditions (facts, states, e.g. welding machine available, bolt fixed) which must be fulfilled. Such a network is best modelled by a graph called Petri net /22/ /23/ bacause its graphics representation is easy to understand and because this type of graph visualizes best the dynamics of the procedures. This type of procedure representation is very natural because it explicitly represents all objects the operator deals with: actions, states, and their causality (connections). In addition the objects of the actions (the resources) are also represented explicitly (tokens, flowing through the network as a result of action execution). The generated procedures can be simulated. During the execution phase the procedure simulator is also used for procedure performance monitoring and as a basis for the advisory module. The simulator is then synchronized by the control system.

Comparing the procedure network with reality guarantees the correct performance of the planned maintenance (monitoring). Guidance for the operator is derived by generating advice for the operator from the actions in the net. The advancement is dependent on the message from the operator signalling completion of an actual subtask. In some cases this information may be derivable from signals in the control system.

The basic data structure allows for a hierarchical modelling of the maintenance process because an action itself may be modelled by a network of actions and states.

For the description of procedure elements multi-media techniques are used. In principle the PEXOS system represents an interactive multi-medial handbook/notebook usable in

all phases of work. Access to this multimedial information is possible via the elements of the procedure graph.

Basic features of the procedural simulation on a RHWS are (for more details see 3.5, "PEXOS: Procedural Simulation for ERWS" on page 80):

- **Graphic visualization** of procedures (action-state nets) for the operator including execution state visualization
- **Hierarchical data structure**: actions of a net can be described by subnets.
- **Simulation of action-state nets**
- **Coordination of man-machine cooperation**: the procedure description (especially the action description) includes advices and control commands for all other subsystems for coordination. In this sense the PEXOS system represents a high level programming and program interpreting subsystem.
- **System integration**: Integration of all remote handling subsystems to one operatable unit (devices & RHWS components). From the PEXOS view all other subsystems of the RHWS and the control system are servers providing devices and device functions (DOIF-SS) and simulation and viewing support (KISMET).
- **Procedure sychronisation by process**: In the execution and monitoring phase the procedure simulation is controlled and updated by messages from the device controllers or the RH-area controller; during off-line simulation synchronisation by KISMET and DOIF-SS is possible.
- **Execution monitoring** of procedures by comparing the planned execution with the actual performance.
- **Annotating**: A notebook function allows to attach operator notes (memos) to action and states.
- **Task (procedure) controlled information presentation**:
  Information about the state or next step are presented according to the actual state of the procedure and in relation to the actions, states, and tasking resources.
  Direct operating interfaces (DOIFs) are controllable by PEXOS as well as the KISMET simulations and views.
- **Complete procedure documentation**:
  Documentation and representation of procedures and procedure elements (actions, states, resources) using multi-medial information:
  - KISMET simulation sequences or static views of the working scene
  - DOIF-SS as functional device representation
  - Texts in the standard handbook fashion of information presentation
  - Form sheets for short infos
  - Drawings
  - Photos
  - Video sequences
  - Sound
- **Logging** for work documentation but also for back stepping is available.
- **Heuristic knowledge management**: Input and processing of heuristic knowledge about the task planning, execution, and problem solving is prepared.

Development status

PEXOS is running in a prototypic version, the integration with the DOIF-SS and KISMET is due to June 1993.

3.2.9 Implementation Related Features

- **Programmability**: Software solutions are used as far as possible for input and output functions (e.g. instead of function keys softkeys are used, instead of wireing software communication channels are used, instead of different displays or instruments software windows are used on larger computer displays). Allocation of selectable functions to a standard set of hardware input devices is possible.
- **Configurability**: The workstation is adaptable to control the work with different RH equipment in different environments. This reconfiguration is easy done by software and data (configuration files).
- **Modularity**: Modular architecture for easy maintenance and possibility of upgrading to include additional features. The architecture is especially designed for implementation in subsequent stages such that advanced new features of the RHWS can be added as needed.
- **Extendability**: Extendable to new or enhanced RH area control systems or enhanced operator support systems and man-machine interface devices. Extendability is also important because not all tasks will demand for a fully equipped workstation, but later extensions to full equipped state should be possible. Table 17 shows possible extension stages.
- **Implementation support**: Implementation is based on standard software and using standard software development and management tools (Table 18 on page 59).
- **Standardisation**: Standard internal interfaces (e.g.communication channels, progamming languages) to be adaptable to standard hardware and software products on the market (Table 18 on page 59).
- **Function distribution**: The CCSS allows to change the standard allocation of the RHWS subsystem to the available computers. In principle all subsystems may be allocated on one computer.

Configuration stages	Modules
CONFIG-1 **minimally equipped RHWS**	Spatial support (KISMET)
CONFIG-2	CONFIG-1 + Functional support (DOIF-SS)
CONFIG-3	CONFIG-2 + Procedural support (PEXOS)

Table 17. Possible Configuration Stages of EDITH Remote Workstation. The configuration stages allow for a stepwise extension of the ERWS suited to operational needs.

Topic	Solution
Hardware	IRIS, SUN
operating system	UNIX
user interface management system	no
programming language	C, Lisp
CASE-tool (Computer Aided Software Engineering)	no
program management (see: CASE)	no
database	no
communication network	ETHERNET
communication protocol	TCP/IP
CAD data format	STEP, native
motion data format	IRDATA /24/
control system communication	native, based on Unix IPC
window system	X-WINDOW
user environment	OSF/MOTIF, OpenLook
Graphics library	Silicon Graphics GL
TV	PAL

Table 18. Actually Used Implementation Utilities. This hardware, software and standards are used for implementation of the ERWS.

3.2.10 Interfaces

The ERWS is working in an environment which demands for different interfaces shown schematically in Figure 38 on page 60.

● **Man-machine interface**: This is the operator used interface realized with state-of-the-art graphical windowing techniques.
● **Internal Interfaces**: To be able to add for example special modules representing the operating interface for a special tool (e.g. welding machine) the ERWS has clearly defined internal interfaces to incorporate theses modules and to run them (Table 18).
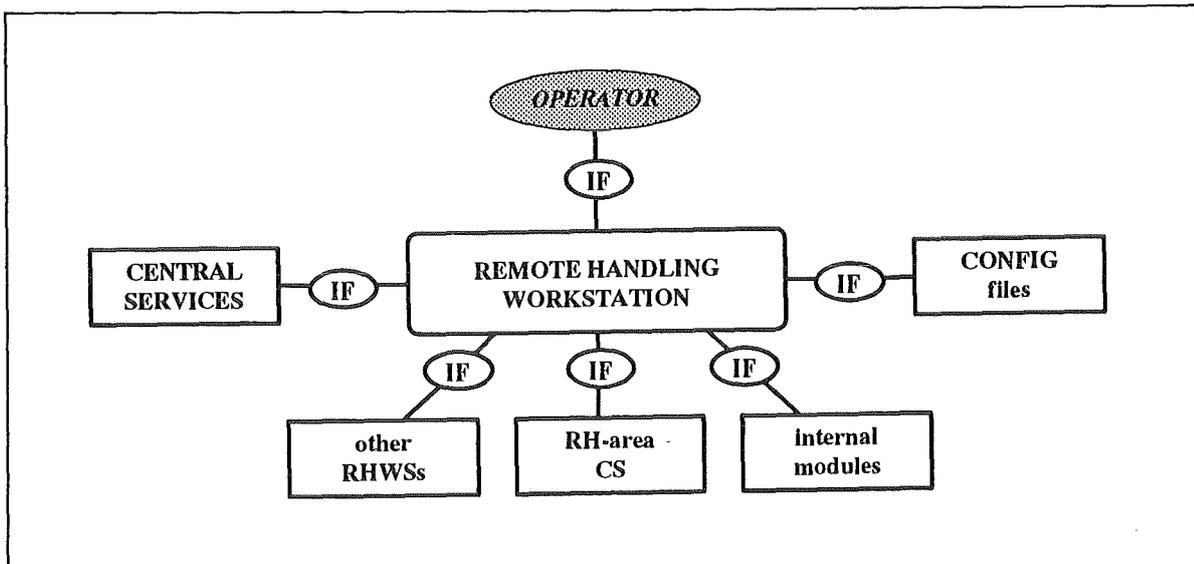
Figure 38. Interfaces of EDITH Remote Workstation

• **Configuration Interface:** The ERWS is configurable by reading data from a special configuration file which is easily editable.

• **Interface to Central Services:** The ERWS has an interface to central services which is actually the interface to the central CAD system for model transfer which uses STEP and KISMET native format.

• **Interface to RH-Area Control System:** The ERWS has an interface to the subordinated area control system actually the EDITH-area control system including a camera control system and later on the DHU control system.

• **Interface to Related RHWSs:** The ERWS is prepared to interface to other RHWSs controlling related work.

3.3 KISMET: Spatial Simulation for ERWS

KISMET (Kinematic Simulation, Monitoring and Off-Line Programming Environment for Telerobotics) is a software tool for effective planning, simulation, programming and monitoring of remote handling equipment, industrial robots and various types of mechanisms.

The program system KISMET was initially developed at the *Kernforschungszentrum Karlsruhe GmbH* (KfK) for use within the EC program for development of controlled thermonuclear fusion (JET, NET), and other nuclear oriented remote handling applications in hazardous environments.

Thus, the development work aimed at supporting operators and planners of handling equipment during preparation and execution of remotely controlled tasks in plant areas that are inaccessible for humans or difficult to observe. Execution of these teleoperation tasks is supported through **synthetic viewing**.

The operator orients himself within the working environment by means of camera observation (if available) as well as animated 3D computer graphics provided by KISMET. This combination of computer graphics and video technique is also called **integrated viewing**. For this type of applications, the graphical scene presentation has to be as realistic as possible and at the same time as fast as possible.

KISMET is implemented on a high-performance graphics workstation and allows via window techniques for several scene views at the same time with interactively selectable levels of detail.
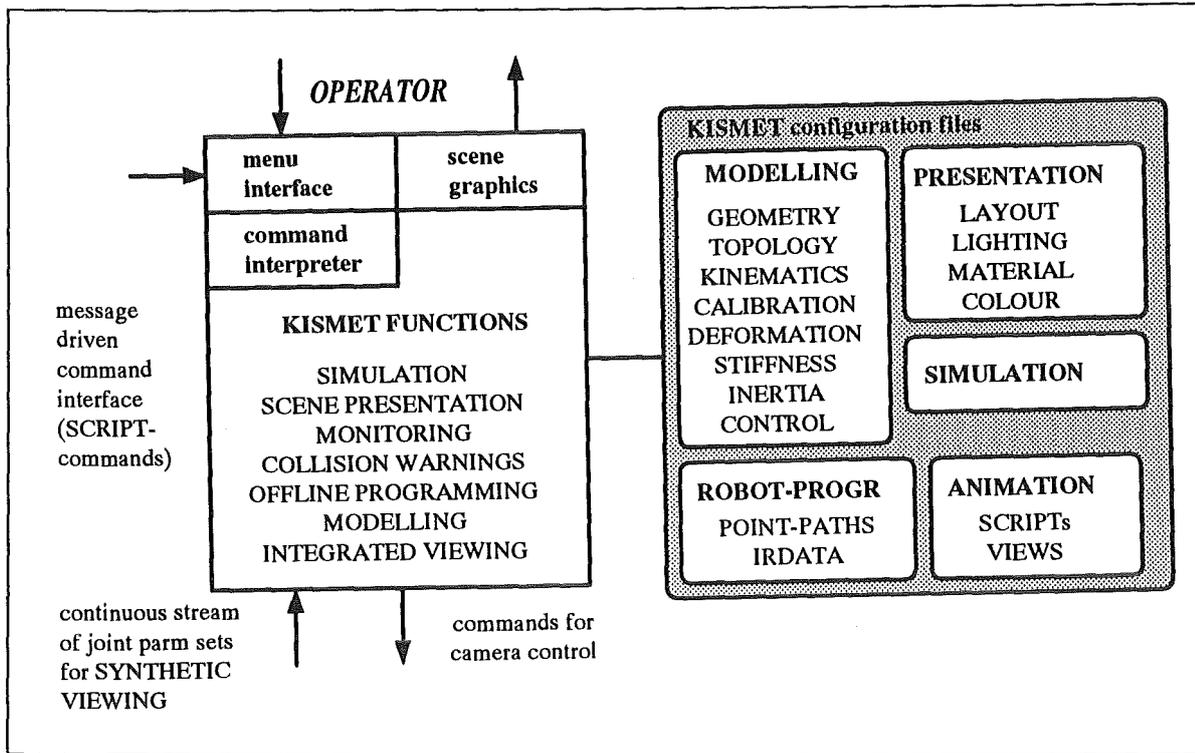
Figure 39. KISMET Interfaces. KISMET is dynamically controlled via the operator interface, the SRIPT command interface (used by external systems) and a continuous stream of sensor signals representing the information for synthetic viewing. Modal control of KISMET is possible via a set of special files containing the models and control informations for simulation, presentation, animation, and device motion.

Since KISMET is oriented towards neutral file formats for robot programming (IRDATA) and for geometry data transfer (STEP), it is easily integratable into a CAD environment.

KISMET allows for a real-time, synthetic generation of any view of handling and manufacturing cells in shaded display (Gouraud shading), as wireframe, or as transparent models. The rendering mode can be set interactively as a display attribute for any single geometry entity and/or for assembly parts. The integrated **raytracing** module can produce photorealistic images (non-realtime).

KISMET was written in the "C" language and is presently available for *Silicon Graphics IRIS 4D/xx* workstations running IRIX 3.x (UNIX 5.3). The KISMET software is implemented using the SGI graphics library.

In the newest version, device elastomechanics and dynamics can be simulated in real-time.

The concurrent simulation of any number and types of robots and other mechanisms, including tree-like mechanical structures and planar closed-loops, is possible.

3.3.1 Operating Modes and Applications

The "**Off-Line**" operating mode serves for

* creation of workcell layouts,
* editing of geometry, kinematics and workcell topology,
* operator training,
* remote handling operation planning,
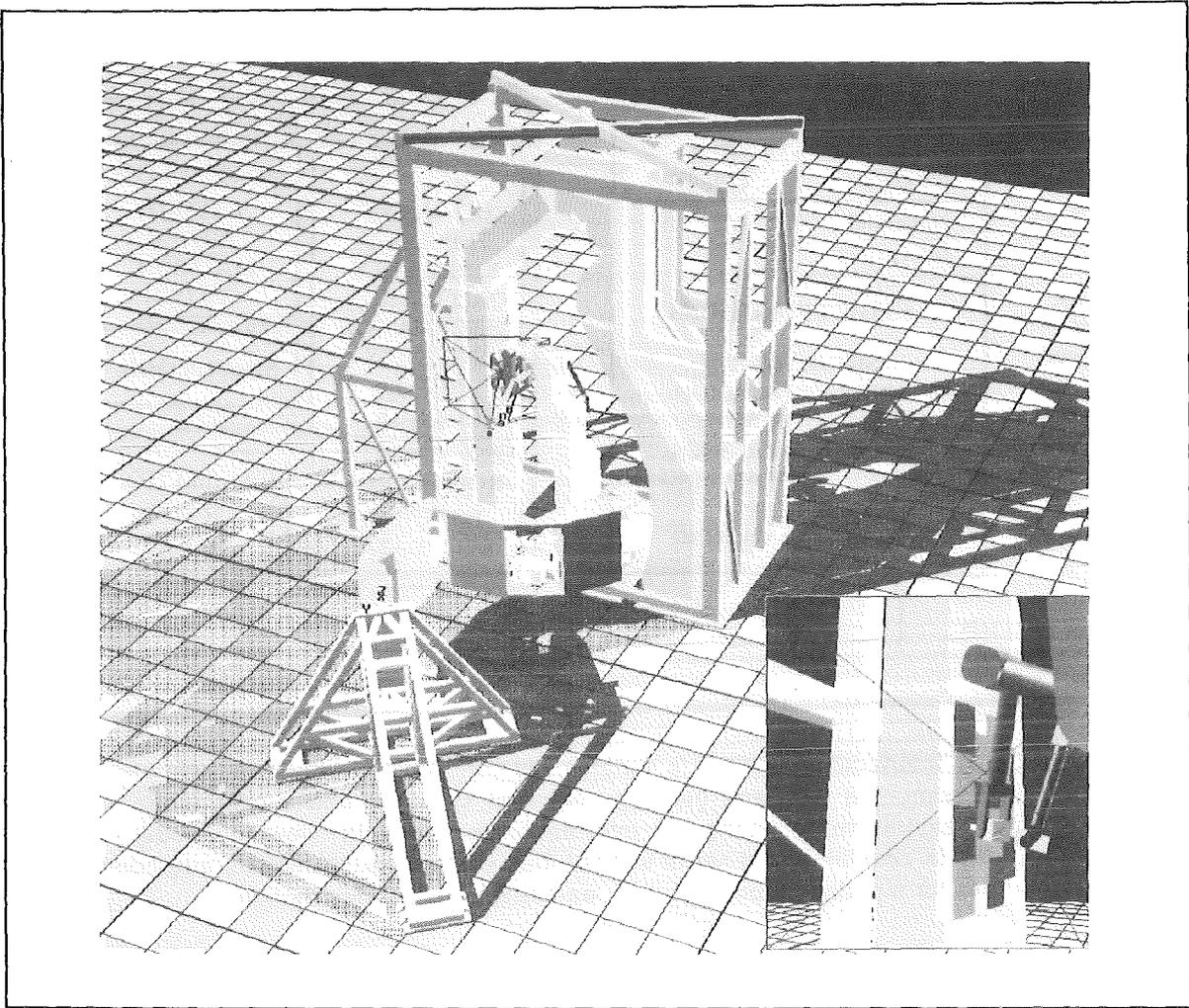
Figure 40. KISMET Model of EDITH Experimental Setup (fake shadows)

- accessibility tests,
- creation (textually, graphically), simulation, verification and execution time measurement of robot programs,
- numerical test for collisions,
- creation of animation sequences.

While in "On-Line" (supervisory mode, monitoring mode) the position sensors (resolver data) of the robot control units are used to drive the kinematic simulation model in KISMET and to present a realistic and problem suited display of the task area. Therefore, the graphical model configuration (robot joint position data, mounted tools, position or state of installed objects in the working environment) is controlled through the real running process. Hereby, the operator can specify any kind of view in the handling environment or manufacturing cell. While in this operation mode, the operator can supervise automatically running robot programs, or directly control the equipment by means of manual operation controls (man in the loop control).
KISMET supports the operator in this mode in terms of spatial scene presentation and, additionally, automatic collision warnings.
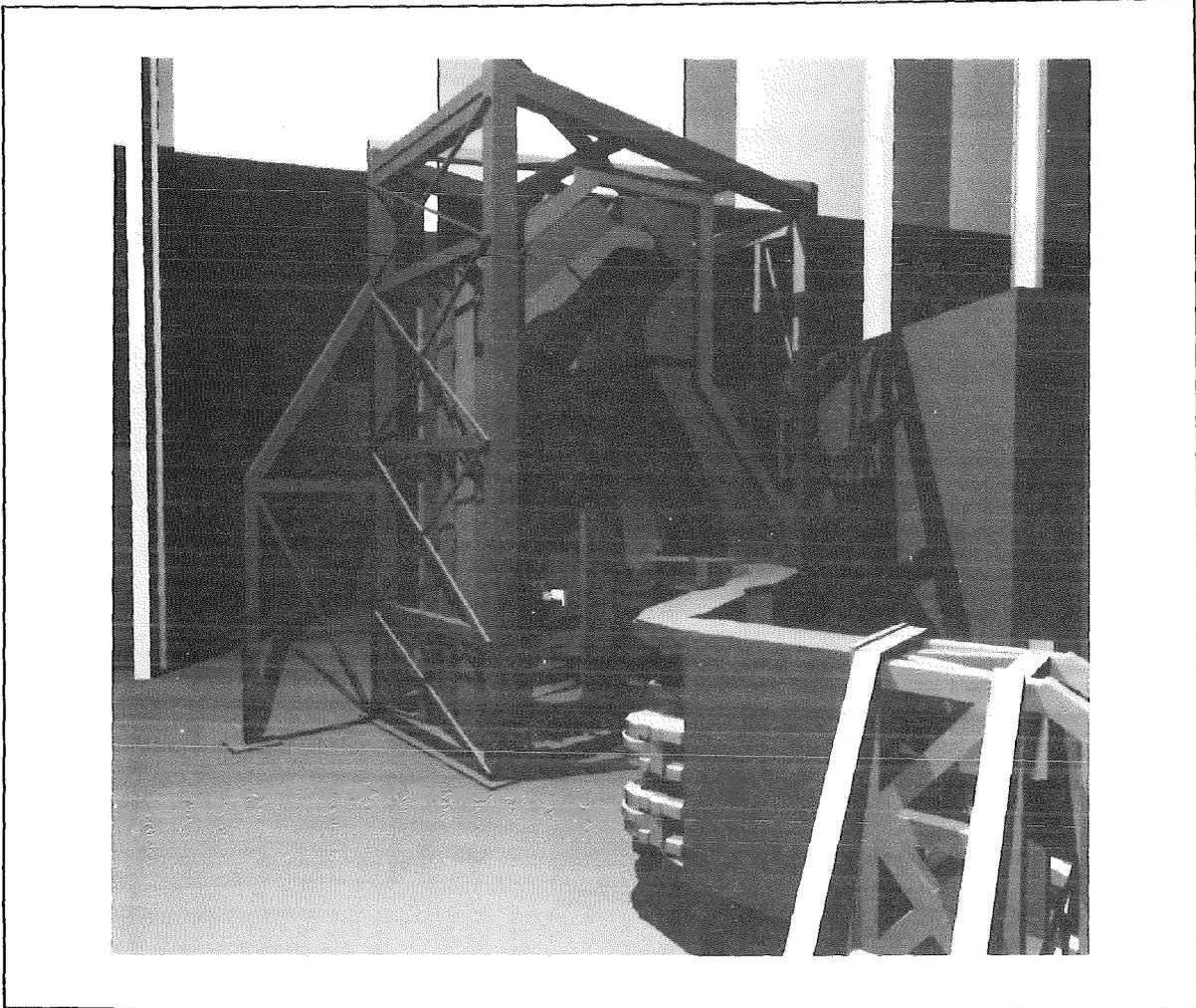
Figure 41. KISMET Model of EDITH Experimental Setup (photo-realistic)

3.3.2 RH-Simulation Applications

According to 3.2, "EDITH Remote Workstation: A Prototypic Guidance System" on page 46 and Table 15 on page 50 KISMET is used in all phases of remote handling work but also in kinematics design applications phase a pre-phase of planning.

However, it should be stressed at this point, that CAD-models which are basically produced (possibly in the ITER "standard" CAD-system CATIA) for **manufacturing** are not necessarily suitable for realtime RH-simulation. The models may be too detailed, or from the point of view of model topology not useful for further use. Topology means here

- which part belongs (kinematically) to which other part, and
- which subsystem belongs (kinematically, not logically) to which other subsystem;
- this information is essential for PICK, PLACE and TRANSPORT operations in the KISMET.

However, it may be useful to share data between the standard CAD-system and the KISMET. Definitely a **CAD-interface** between the two systems is mandatory. **Model production** can be organised in various ways:

1. Geometry is done in CAD and transferred to KISMET when required Kinematics, dynamics etc. is added in KISMET after transfer.
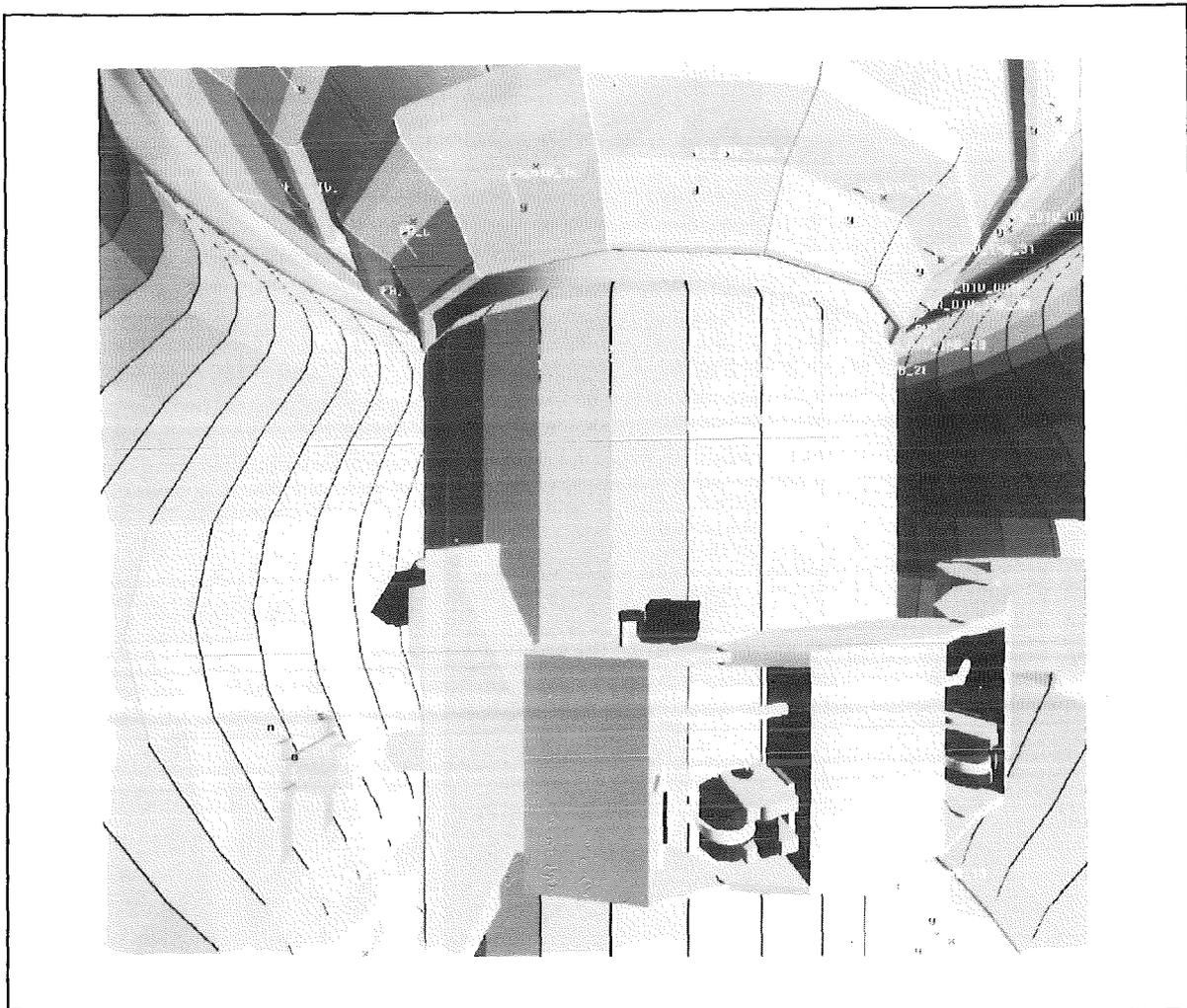
Figure 42. KISMET Model of Tile Handling with Master/Slave-Manipulator

2. Often re-modelling is required for the specific needs of the RH-team and/or of KIS-MET. This is together with point (1) the status-quo at JET.
3. Geometry and kinematics are created in CAD, then transferred to KISMET, specific informations for animation (high-quality rendering requires accurate material surface definition, lighting models, surface texture information, viewing setup and motion, object motion etc.) and real-time applications are added.
4. All is done in the KISMET. It is a stand-alone system and not related or dependent on the standard CAD-system.
5. Any intermediate form, as required

3.3.2.1 Kinematics design applications

The use of a KISMET for **kinematics design** may happen in all stages of the mechanical design of RH-equipment. Mostly this work is carried out by mechanical designers in the "drawing office" together with the responsible design engineer.

3.3.2.2 Planning and Pre-Programming of Device Applications

Usage of KISMET for **Remote Handling Planning** happens when the design of RH-devices and end-effectors is completed and during the phase after device (transporter) commissioning. This work is usually carried out by remote handling "experts", which are not necessarily involved in the design and sometimes not even in the physical operation of these devices. Planning means in this context gross operation planning, and in later stages robot "programming" for standard maintenance tasks and level 1 critical RH-operations. This work may involve the production of "RH-procedures", this is a textual description of each step of RH-operation. KISMET can be used in this stage for studies about device positioning, reachability and collision evaluation (trial and error method). This work may end up in animation files for KISMET used by PEXOS, the multimedia handbook, for presentation and documentation purposes.

For simulation of robot programs an IRDATA interpreter is integrated in KISMET (DIN 66313, part 1). Also available is a PASRO to IRDATA compiler (PASIR) for textual robot off-line programming. KISMET provides the following functionality for off-line programming and simulation:

- Parallel and concurrent simulation of multiple robot configurations is possible. For synchronisation of different system components there are statements available for 'wait', conditional program execution and program flow control (while, for, if-then-else etc.).
- The analysis of PTP (synchronous and asynchronous), linear and circular motions can be done quickly. For the calculation of time coordinated motion trajectories, acceleration and velocity profiles of single axes, robot arm configuration, motion restrictions are included in the simulation model.
- Robot program simulation can be executed

  - in "realtime", i.e. the simulation in KISMET takes the same time as the execution of the "real" robot program,
  - with an operator definable sampling interval.

In the latter mode, the calculation of robot program execution time is possible.

3.3.2.3 Operations Monitoring, Synthetic Viewing

In the stages of mockup tests and later in the operational phase of a fusion reactor, KISMET may be used as an integral part of a RH-workstation for Computer Aided Teleoperation (CAT). KISMET is used here to provide **synthetic viewing**. This is, the operator orients himself in a CAD-based **virtual world**. The synthetic views are generated in realtime from CAD-based model(s) of the plant, RH-devices (end-effector, tools) and sensor signals from the low-level control system.

Additionally, KISMET can be used for collision avoidance tasks and **integrated viewing**. This is, the system is used for positioning and/or tracking of real video cameras (KISMET acts here as a device controller). KISMET is best suited for this task, because of its "spatial knowledge" about plant and RH-device geometry. This kind of KISMET-usage was demonstrated with the KISMET at JET, in the TELEMAN project, and in the German space program (at DLR in Oberpfaffenhofen).

The use of KISMET for **3D synthetic viewing** and as integral part of a remote handling workstation (RHWS) has become an important application in the past years. An example is the use of the packages GBSIM (the KFK-predecessor of KISMET) and KISMET (KFK) at JET for In- and Ex-Vessel remote handling /25/.

Because of its "geometric knowledge" database, KISMET can be used for specific RH-subtasks as playing an active part in the control loop for some devices. An example is positioning and tracking of cameras and collision control.

The following KISMET features support teleoperation:

* The modelled RH-device can be controlled by sensor signals provided by the control system, such that KISMET allows for operations-monitoring and synthetic viewing.
* Update of robot motion on the KISMET display is not interrupted during user interaction, e.g. changing of viewing parameters.
* Sensor-simulation is available (e.g. distance sensor simulation).
* Camera simulation and camera view definition
* Graphics based camera control by pointing to the view target
* Target camera-tracking
* Camera-roll tracking

3.3.2.4 Training

KISMET is also designed for usage in **RH-operator training** for man-in-the-loop control. For this, the RH-devices should be modelled in the KISMET as accurate as possible for realtime response. This requires various kinds of modelling:

* Kinematics
* Kinematic control modes, e.g. single-joint and resolved motion (inverse kinematics)
* Robot dynamics, i.e. the dynamic behaviour of the mechanical system and the high- and low-level controllers
* Elastomechanical deformation (bending and torsion) due to device weight, load, position and possibly external forces. This feature is important as well for sensor signal based synthetic viewing, and off-line programming.

For all these subtasks, **realtime performance** of KISMET and in particular of the graphics subsystem is _essential_.

3.3.3 User Interface

The KISMET user interface is menu driven using:

* Popup/Pulldown menus, i.e. the menus are visible only when called
* Command-Panel like menus, i.e. a large set of menus, buttons etc. are visible during runtime on the KISMET-screen (and hides perhaps valuable space for scene viewing)

Most of the KISMET functions are also available via a textual command interface accessible through so-called SCRIPT files or a UNIX inter-process communication message interface (e.g. used by external processes like DOIF-SS or PEXOS).

The following input devices (command interfaces) are supported by KISMET:

* Keyboard, Mouse
* 6-dof Spaceball
* Dials- and Buttons-Box

3.3.4 Display Control

3.3.4.1 Graphical Features, Window Layout, Parameters Display

The graphical output (workcell display) is the most important simulation output of KIS-MET. Thus, special attention was laid on this topic.

3.3.4.2 Graphical Rendering, Drawmodes and Display-Options

Graphical rendering and drawmodes are most important features of KISMET, because they influence essentially the information content of the simulation. However, some of the features listed below are important only for fancyness.

The list of rendering features is:

1. Wireframe drawing, it is the only mode which does not require solid/surface geometric modelling. Antialiased wireframe drawing is available.
2. Flat-shaded rendering, possible with simple polyhedron models.
3. GOURAUD-shaded rendering, requires surface-normal information. This can be achieved only with SOLID or BREP geometric models.
4. Transparent-shaded rendering
5. Surface-texture mapping. This technique is used to map 2D-image data on 3D-surfaces to improve display realism.
6. Fake-Hidden-Line drawing
7. Fake-shadows. This is a projection technique used to simulate multilight shadows on a ground plane. The features (1)-(7) can be performed with realtime or near-realtime performance on high-performance workstations.
   All drawing modes (1)-(7) can be carried out Z-buffered, i.e. depth-sorted, hidden-surfaces removed for (2)-(7).
8. It is useful when the drawmodes (1)-(4) can be interactively defined in KISMET for each object or group of objects, i.e. one should be able to draw one set of objects in wireframe mode, another set opaque shaded and other parts transparent shaded (in the same scene-view). The last point is important for a **problem suited scene-display** during all stages of operation.
9. Raytracing. This technique can be used to achieve photorealistic image quality. Even with supercomputers it is not possible to calculate raytraced images for scenes with realistic complexity (number of parts, lightsources etc.).
10. Full RGB-colour schema (24Bit image memory per buffer)
11. User-defined lightsources (parallel light, pointsources, spotlights)
12. User-defined colour and surface-material (reflectance) schemas

3.3.4.3 Screen- and Scene-Window Layout

Special features of Screen- and Scene-Windows are:

* Multiple workcell views at the same time (multi-window layout)
* Position and size are user-definable (interactive)
* The window layout and initial viewing parameters are storeable for different models and can be automatically reloaded during KISMET model startup.
* It is possible to define for each view either orthogonal or perspective projection.
* A "full-screen" mode is available.
* Store and recall of a window-layout together with the viewing parameters for each viewport
* STEREO-viewing is supported.

- VIDEO-underlay of wireframe model drawing is supported. This mode is useful for model-verification and correction, using digitised video-camera data from the real-scenario.

3.3.4.4 Viewing Parameters and Operations

Views are defined by the following parameters:

1. Viewpoint (viewing position)
2. Lookpoint, also called point-of-interest (POI)
3. From (1) and (2) the viewing-direction is calculated.
4. By (1) and (2) also the focal-distance is defined.
5. View-twist (rotation around view-direction)
6. Field-of-view (zoom)
7. Perspective or orthogonal projection
8. Front- and back clipping plane distance
9. World-window volume to be displayed, this is an alternative to define parameters (2)-(6).

The following viewing operations are available:

1. All parameters as listed above are interactively controllable by the user, i.e. by input from mouse, spaceball, dials-and-buttons box etc..
2. Fast definition of POI (look at object function, object selected by picking)
3. Store, reload of all viewing parameters
4. Look-at scene centre function
5. Show area function
6. Realtime performance for viewing functions
7. A function to display (redraw) only the workcell parts currently visible in the scene viewport
8. STEREO-viewing with interactive modification of
   a. eye-distance,
   b. focal-distance.

3.3.4.5 Output of Numerical Parameters, Errors and Conditions

Additionally a list of numerical and textual parameters can be printed on the screen panels for user information. Such parameters can be:

1. Robot and/or motion parameters, i.e.
   - joint position, velocity, accelleration,
   - global (world) TCP-position,
   - TCP-offset relative to robot hand,
   - robot reference system relative to default robot base,
   - active ROBOT (the one attached to input devices),
   - active ROBOT program (under execution),
   - robot program execution time (cycle time),
   - robot program step (instruction) number.

2. Viewing parameters, e.g.
   - viewpoint location,
   - lookpoint location,
   - window (world) size.

3. Name of workcell
4. Name of active robot
5. Name of active, simulated camera

6. Dynamics parameters (forces, torques)
7. System time (i.e. a clock for the user)
8. Frame rate
9. Parameter display is optional
10. Output of error-conditions, warnings for
    - joint range (joint clash),
    - excess of joint-velocity / -accelleration during linear, circular motion modes,
    - collisions, which parts have collided,
    - user input faults,
    - robot program syntax / semantics.

3.3.4.6 Kinematics Control

The following **active motion** and **position control** and program simulation in joint-space, tool-space, object-space (objects are passive parts in the workcell) and world (workcell) space are available:

- Single-dof motion. This is trivial for simple kinematic chains, but not anymore for complex mechanisms with closed-loops and/or joint couplings.
- 6-dof TCP-motion in tool-coordinates, i.e. the translation is carried out along the current direction of the TCP-frame axes, the rotations around the TCP-frame axes.
- 6-dof TCP-motion, but the input is in world coordinates.
- 6-dof TCP-motion in any object coordinate system
- Move-to-Object positioning. I.e. KISMET allows for definition of object coordinate systems/frames, usually called "workframes" (WFRM). A WFRM corresponds in position and orientation to the robots TCP in the work-condition. WFRMs are also used to define intermediate target points for robot motion (flypoints, reference positions, tool-exchange positions etc.).
- KISMET allows for definition of a WFRM at the current TCP-position.
- WFRM are combined in a WFRM-PATH sequence. The PATH defines the sequence of motions for a RH subtask.
- WFRM-PATH sequences are combined in a WFRM-PATH-FILE. It defines all target and intermediate positions in a RH-task for one or multiple robots.
- An automatic mode is provided to transfer WFRM-coordinates to robot-coordinates and vice-versa after editing or modifying the cell model (to avoid reprogramming of target positions).
- Special KISMET features with respect to the **inverse kinematics problem**:
    - KISMET has a set of analytical solutions for standard robot kinematics (a robot library).
    - A general inverse solution for those robots which have no analytical solution allowing for under- or overdetermined robots (other than 6-dofs in space) handling optimum control methods (minimum energy, maximum speed criteria etc.).
    - KISMET provides special motion modes for redundant kinematics, like resolved-motion, joint-reconfiguration etc.

KISMET motion modes for **robot program simulation** are:

- Synchronous and asynchronous motions defined in joint space
- TCP point-to-point (PTP) motion
- TCP linear motion (LIN), i.e. the TCP moves with a predefined speed along a straight line or parametric curve in space.
- TCP circular motion (CIRC)
- Motion targets can be defined in absolute and relative coordinates (relative to the last position) and in a symolic way, i.e. the target is identified via ID-string.

- All motions are carried out as similar as possible as in the real world. Therefore KISMET provides the execution of joint-synchronized ramps with linear or sinusoidal accelleration period and the adequate motion planning.
- It is possible to execute pre-programmed motions in two modes:
  - In realtime
  - With a definable sample time, i.e. the execution time may be scaled compared to real life.
- It is possible to time (stop execution time) motion cycles and robot program execution cycles.
- Pre-programmed robot parameters (joint/TCP velocity, accelleration) must be included in the motion planning. An error is to be detected if joint min/max limits are exceeded during LIN/CIRC motions.

3.3.4.7 Textual Command Interface

Additionally to the direct, menu driven man-machine interface KISMET offers a textual **KISMET-SCRIPT** command language. **SCRIPT** commands can be entered into KISMET in different ways. If read in from an ASCII file, very complex animation sequences can be created.

However, SCRIPT commands can also be entered into KISMET be means of UNIX interprocess communication techniques. KISMET provides a 'message-queue' channel for other, concurrently running processes as a command interface (faster than 'pipes'). Using this method, each user can build his own MMI around KISMET.

SCRIPT-commands can also be entered as comments into IRDATA-programs. When these comments are sent back from the real robot controller during on-line program execution, KISMET can reconfigure its data structure upon process events or configuration changes (e.g. tool or end-effector exchange, grabbing of handled objects etc.).

3.3.4.8 Programming

Programming capabilities are provided by KISMET for a set of applications:

- Motion-Simulation (no robot programming intended)
- Robot-Programming
- Animation Programming (SCRIPTS)

KISMET allows for:

1. Graphical teaching
2. Textual programming: Textual programming can be used for generation of program bodies.
3. Combination of graphical and textual programming. With graphical editing methods a set of
   a. Workframes (WFRM). Each WFRM is kinematically connected to the workcell objects to be handled. When the object is moved (model correction, workcell calibration), the WFRM moves together with the object. The WFRMs are mainly used for definition of robot motion targets (TCP position).
   b. WFRM path sequences,
   c. WFRM-PATH files
   can be manipulated interactively. In another step, the textually generated robot program body and the WFRM-PATH files are merged to an executable robot program

3.3.4.9 Robot Languages

For off-line programming a set of robot/simulation languages is supported by KISMET. The languages and the simulation module support groups of statements such as:

1. Robot motion commands. Examples are:
   a. Point-to-point motions (synchronous and asynchronous), definable in joint and/or TCP-space
   b. Linear motion
   c. Circular motion
   d. Relative and absolute definition
   e. Specification of execution time (required velocity and accelleration are calculated)
   f. Move to home or reference position
   g. Support for additional axes

2. Definition of robot parameters, this is of:
   a. Joint speed and accelleration
   b. Joint limits (motion range)
   c. TCP speed and accelleration
   d. TCP-offset

3. Robot controller I/O commands (analogue, digital I/O) for sensor based robot programming
4. Signal exchange between robots
5. TAKE, RELEASE objects and tools (end-effectors)
6. Program flow control (while, for, if-then-else, subroutine calls, switch etc.)
7. Arithmetic operations for simple datatypes (integer, real, boolean, strings and aggregates of these) and of structures (vector, orientation, matrix), including trigonometric functions (sine, cosine, tangent, arctangent, exp, log , sqrt etc.)
8. Library functions can be included (elementary functions).

KISMET supports high-level robot programming languages (C- or PASCAL-like):

1. High-Level languages: IRL (Industrial Robot Language), available through the off-line programming subsystem TOPSY (in development at KfK); PASRO (Pascal for Robots), available through the programming subsystem GRIPS (developped at KfK).

2. Low-level languages (programming codes). Such codes are intended for direct execution in the robot controllers: IRDATA (the EDITH-CS has an IRDATA-interpreter).
3. Available are also compiler(s) and/or translator(s) from high-level programming language to programming code(s) and/or to custom programming codes.

A set of functions is available for program simulation:

1. Single step execution
2. Debugging
3. Execution timing (cycle time calculation)
4. Simulation of execution of multiple programs in parallel (quasi parallel) for workcells with more than one robot
5. Infringement of robot behaviour (excess of position, speed and accelleration limits)
6. Graphical display of TCP path
7. Collision test during program simulation (see 3.3.7, "Collision Test and Avoidance" on page 75)
8. Workcell calibration
9. Program down/upload functionality
10. Stop / reset / reverse program execution
11. Execution of robot programs from within an animation script

3.3.5 Modelling

In KISMET a hierarchical, recursive, tree like **data model** is used. The operator specifies at any time interactively and on-line the level of detail to be displayed. Components or model parts can be activated or deactivated for display at any time. This allows to display higher levels of detailing without loss of realtime performance.

Robots and other handling equipment or mechanisms (including tree-like mechanical structures and kinematically determined planar closed loops as typically found in hydraulic drives, gearings, crank mechanisms etc.) can be defined interactively, or can be imported from CAD systems.

3.3.5.1 Geometric Modelling

Geometric modelling is required to define the shape of the RH-devices and the working environment. Although much of the model geometry will be imported from the standard CAD-system, geometric modelling may be required for

* fast prototyping
* model correction during teleoperation

For creation of a RH-scenario only 3D modelling is important. 2D-drafting can be carried out in the standard CAD-system. For this a set of **geometric primitives** and a set of **modelling operations** is required.

Geometric modelling primitives:

1. 3D-line primitives: POLYLINE, parametric CURVE
2. BREP-primitives: POLYHEDRON and exact-BREP
3. 3D-solid primitives: BOX, SPHERE, CYLINDER, CONE, TRUNCATED_CONE etc.
4. 3D-extrusion (sweep) primitives (ROTATIONAL_SWEEP, TRANSLATIONAL_SWEEP)
5. High-level primitives, e.g. isometric PIPE
6. Parametric Surfaces, usually imported from surface modellers (SURF)
7. B-Spline Surfaces
8. Trimmed Surfaces (NURBS: Non Uniform Rational B-Spline)
9. Parametric dimensions are available (required for display of flexible bodies).

Geometric modelling operations:

1. Placement (rotate, translate) relative to kinematic reference frame
2. Scale part dimensions
3. Boolean operations (Union, Difference, Intersection)
4. INSTANCING
5. Definition of
   * surface MATERIAL,
   * weight, centre of gravity, inertia,
   * facetting during model startup,
   * definable accuracy error.

Features for workcell layout:

1. Automatic robot placement to get the optimum base position for a defined set of workframes
2. Interactive reach evaluation, i.e. this answers the question if the robot can reach all workframes from the current base position.
3. Collision testing

3.3.5.2 Kinematic Modelling

Kinematic modelling is provided and approved in KISMET for the definition of devices like:

*   Transport devices for in- and ex-vessel remote handling like articulated boom transporters (JET-AB, EDITH, TFTR-Boom), telescopic multiarticulated systems (TARM), autonomous vehicles carrying manipulator arms, rail based systems.

    All systems have redundant kinematics and require therefore special kinematics algorithm treatment.
*   End-effector devices like M/S-servomanipulators. Manipulators can have redundant kinematics too.
*   Special load transfer devices, e.g. divertor handling units (DHU)
*   Multi-dof articulated hands and grippers
*   Teleoperation support actuated mechanisms like cranes, camera-arms etc.

For the needs of simulation two kinds of kinematics features are distinguished:

*   **Mechanism modelling:** The task here is to define the input data for mechanism/robot simulation. It requires a set of modelling entities (joint and frame types) and editing features (placement functions).
*   **Kinematics Control Algorithms:** These are needed for mechanism motion definition and robot program simulation. A set of control modes for interactive positioning in KISMET and a set of algorithms or subroutines are provided for motion execution, including *inverse kinematics* transformation methods.

3.3.5.3 Mechanism Modelling Features

The list of KISMET features for **Mechanism Modelling** are:

*   Rotational (revolute) and translational (prismatic) joints
*   Higher level joints (spherical, cylindrical, slider/crank etc.). Although all higher level joints can be built as combinations of the basic revolute and prismatic joints.
*   Simple kinematic chains
*   Kinematic tree-like structures
*   Closed-loop kinematics
*   The possibility to define couplings (gears) between degrees of freedom. The motion of two or more joints is coupled through a functional relation, which is either implied by the mechanics or by the control system.
*   Parametric links which define during motion the shape (geometry) of the link. This is useful for simulation of extending crane-cables, spring and rubber parts, which change their length/shape during robot or mechanism motion.
*   Joint limits: the mechanical limits, and the programmable limits for program execution (to restrict the workspace)
*   Definition of actuator velocity and accelleration parameters (two sets, min. and max. values each)
*   Robot base reference system and tool-centre point (TCP, POR) system for robot control and programming reasons. Both systems allow for programmable 6-dof offset transformation.
*   the kinematics model can be interactively changed (edited) even for closed-loop kinematics.

3.3.5.4 Model Topology and Detailing

Model topology means description of

* which part belongs (kinematically) to which other part, and
* which subsystem belongs (kinematically, not logically) to which other subsystem.

This information is essential for PICK, PLACE and TRANSPORT operations in KISMET, that means if a part is interactively moved, all other parts kinematically connected to the part move also.

Important features regarding model topology which influence essentially system performance for complex models are:

* KISMET supports **Detail Levels** for part and geometry storage. This is, the workcell can be represented in various detail levels in the **same model database** to optimize in each stage of operation the KISMET performance.
* The user can quickly and interactively swap between detail levels.
* The detailing is not limited to a number of levels.

3.3.6 Dynamics and Elastomechanics Modelling

KISMET is a powerful tool for planning, programming and online control of remote handling tasks. Two features of KISMET were especially developed for remote handling tasks executed with the EDITH Boom: the calculation and visualization of the static deflections of the boom and the real time simulaion of the dynamic behaviour of the boom.

3.3.6.1 Real Time Calculation of Static Deflections

In the KISMET on-line mode, also called monitoring mode, the kinematical simulation model is driven by the real running process using position sensors. KISMET supports the operator in terms of spatial scene presentation and automatic collision warnings. "Stiff" models, neglecting the flexibility of the boom and the EEPU are not sufficient anymore. Due to its size, its weight and its flexibility there would be a difference between the shown position and the actual position. Therefore, it was necessary to enable KISMET to calculate and visualize deflections of the EDITH boom and the EEPU at certain points of interest (e.g. the joints and the Tool Center Point) due to gravity and load forces. The flexibility of the links can be modelled as well as the flexibility of the joints. Figure 43 on page 75 shows the "stiff" model of the EDITH-boom, drawn in the 'wireframe' mode, as well as the deflected model, drawn in the "Goraud-shaded-mode". It is very obvious that due to the accumulation of deflections of the boom and the EEPU these effects cannot be neglected to get a true information about the position of the end effector by KISMET.

3.3.6.2 Real Time Dynamic Simulation

Implemented algorithms enable KISMET to be used as a training simulator for operators of the EDITH boom by calculating the dynamic behaviour in real time. KISMET can be used as a training facility for many purposes:

* Direct Dynamics (the motion of the boom due to forces and torques) can be simulated as well as Indirect Dynamics (forces and torques due to a demanded movement of the boom).
* The operator gets a feeling for the dynamic behaviour of the boom by simulating a man in the loop control on joint level.
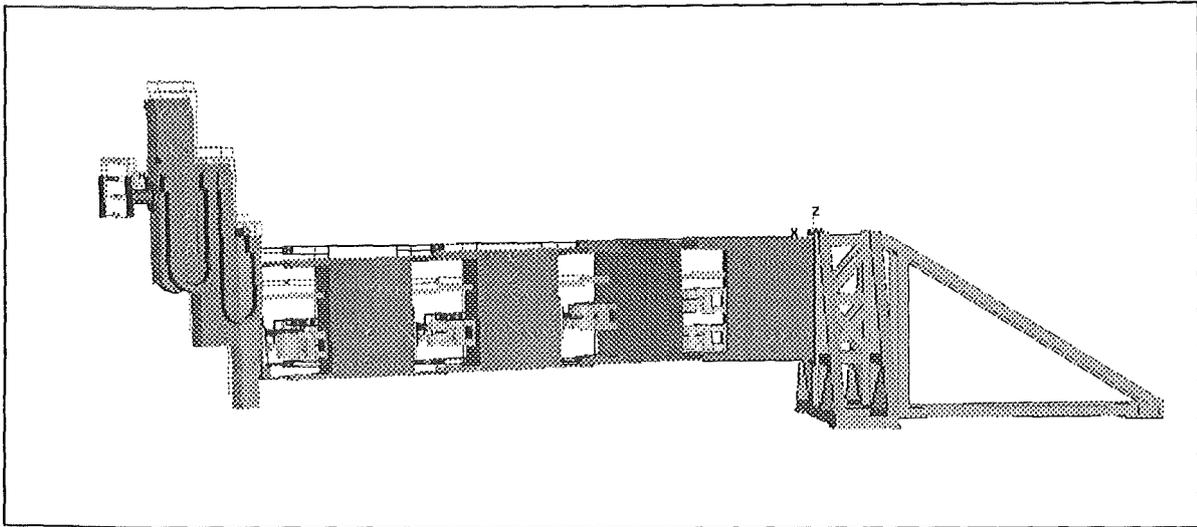
Figure 43. EDITH Boom and End-Effector Positioning Unit with Deflections

* Critical or dangerous situations can be trained.
* The maximimum speed of the boom for time critical remote handling tasks in a narrow environment can be evaluated in advance.
* The influence of changes in the the design of the boom or the control parameters on the dynamic behaviour can be simulated.

The dynamic model can easily be introduced into KISMET including:

* The flexibility of the driving system
* Damping and friction effects
* A joint based control system (position or velocity control)

Additional features support the operator with informations:

* Forces and torques can be visualized during the simulation. (Figure 44 on page 76)

* Certain parameters of interest, e.g. position, acceleration, torque, forces etc. can be stored in a file during the simulation in order to analyse them.

3.3.7 Collision Test and Avoidance

Collision test and avoidance is one of the major tasks for KISMET. The term means in this context **numerical** testing, not visually (by the operator), which can be done with any program capable of drawing 3D-parts.

Important features of the collision test subsystem are:

* The collision test does not interfere with some modes of operation or other system features.
* Collision avoidance is also available (path-planning).
* Minimum distance check is available and near-miss detection.
* The user can define the amount of testing, i.e. which robots, robot-links etc. versus which other robots and/or parts of the workcell environment.
* The test table is automatically initialised, the user has not to define the test part-table each time.
* The system provides warnings.
* It slows down robot-motion in danger of collision or minimum distance situations.
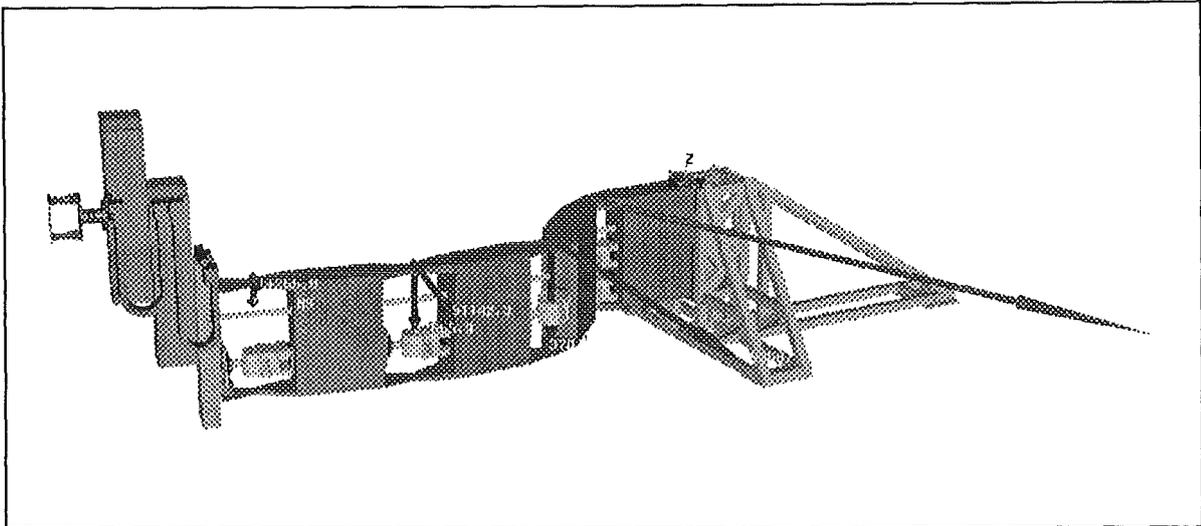
Figure 44. EDITH Boom with Visualized Forces and Torques

- The algorithm problems can cope with all geometrical situations.
- The user can specify if the robot(s) shall stop or not after a collision or near miss was detected during simulation.
- The colliding parts are highlighted.

3.3.8 File Interface

As shown in Figure 39 on page 61 KISMET has a file interface for the basic configuration and control of the system. These files are mostly ASCII files allowing easy editing with a text editor. The classes of information for controlling KISMET via those files are:

The SIMULATION file is the basic KISMET configuration file telling the system at start time which LAYOUT, LIGHTING, MATERIAL, COLOUR, CALIBRATION, and especially which model tree (TOPOLOGY) should be used for the simulation. This root of the model then references other TOPOLOGY files, GEOMETRY files, KINEMATICS files, and CALIBRATION files.

Presentation files

The viewing LAYOUT file defines the initial KISMET window layout: position and size of viewports on the screen and their initial viewing position, view direction, size of the world window, mode of projection.

The LIGHTING file allows the definition of up to 8 parallel- or point-lightsources.

The MATERIAL file specifies the surface properties required for diffuse or specular reflection which is calculated for shaded-redering.

The COLOUR file defines colour tables for KISMET.

Modelling files

The GEOMETRY files describe the geometry for the entities of the model.

The TOPOLOGY file is the mean for the model hierarchy definition. It is used to define the root and uppermost hierarchical level of a model tree and the subtrees. An TOPOLOGY file therefore can refer to other TOPOLOGY files, GEOMETRY files, DOF files, and CALIBRATION files. TOPOLOGY files are used for robots as well as for environments.

These DOF (degree of freedom) files are mandatory references in ROBOT-TOPOLOGY files only. They define input/mechanism functional relations.

The CALIBRATION files are mandatory references in ROBOT-TOPOLOGY files, if the robot is to be used in monitoring applications.

The DEFORMATION files contain material data, e.g. mass, modulus of elasticity, friction, etc.

The STIFFNESS files describe stiffness matrices of robot links.

The INERTIA files contain inertia matrices of robot links.

The CONTROL files contain the values of the control parameters for the joint based robot controllers.

Animation files

SCRIPT files describe animation sequences using a subset of the KISMET commands available through the menu interface.

VIEW files store the complete set of viewing parameters for all viewports. They can be recalled during an animation sequence to change the view(s).

Robot programming files

The WORKFRAME-PATH files describe the path of a robot motion by a sequence of poses.

An IRDATA file describes the motion of a robot in the IRDATA format.

3.3.9 CAD-Interfaces

CAD interfaces are available to import model data from

* the standard NET/ITER mechanical CAD-system (CATIA),
* 3D kinematic simulators (IGRIP, ROBCAD),
* dynamics simulation packages, for example ADAMS,
* FEM systems and FEM-preprocessors (PATRAN, FEMGEN etc.),
* animation system and high-quality rendering systems as ALIAS, WAFEFRONT, TDI.

The interfaces are based on neutral or standardized formats and/or information models. Such interfaces are:

* **IGES** (VDAIS): IGES is the oldest existing standard. Initially it was limited to 2D-drawing data and 3D-wireframe models. It was then further extended, but most CAD systems support only the initial functionality.
* **VDAFS**: This is a german standard, mostly used in the automotive industries. It is limited to the transfer of parametric free-form surfaces.
* **SET**: This is a french standard. It allows the transfer of solid models. Kinematics or other definition data (kinematics, dynamics, FEM etc) cannot be transferred.
* **STEP**: The STEP (STandard for the Exchange of Product data) standard (ISO 10303) is still in the process of standardization. It will replace all other standards in the near future. It is supported by most of the national standardization organisations and will allow to transfer not only drawing or geometry data, but all fields of product data.

The goal of the CAD-interface is, although the term *product model interface* is better suited to the problem, it should be able to import as much data as possible from other systems. This is not only GEOMETRY, but also model TOPOLOGY, KINEMATICS, DYNAMICS, MATERIAL definitions etc..

Such interfaces were already successfully used together with KISMET for generation of the JET ex-vessel RH-scenarios /26/and for generation of the NET in-vessel simulation at KfK /27/.

3.4 Control Panel of the Direct Operating Interface for the EDITH Control System

Manual operation of remote handling devices demands for an interfacing subsystem representing the device functions on the workstation and providing access to all those functions. The Direct Operating Interface Subsystem (DOIF-SS) provides the functional view onto the devices. In the simplest case an DOIF is a graphical control panel for a device. The DOIFs provide at least the functionality of the generic device interfaces which have to be available and usable for commissioning, back-up, and testing. But the DOIFF-SS provides more:

- Enhanced support of device operation for example by graphical representations
- Integration of the manual device operation into the workstation especially connection to KISMET and PEXOS. The KISMET link provides:
    - Simulation of the motion devices controlled via DOIF
    - Graphical data input for special DOIF (device) functions
  The PEXOS link provides task step dependent control of DOIF usage by the operator: e.g. activating/deactivating panels or panel elements to make manual operation more safe and more easy.
- Representation of devices in the simulation phase that means e.g. representation of device operation in the training phase
- Providing device classes to make the control more easy: The DOIF for cameras allows the control of different cameras via one generic interface.
- Integration of various devices into one operational tool, the DOIF-SS. This allows for parameter transfer between devices and combination of commands of different devices in one command macro
- Usage of modern human interfacing software tools also for devices with restricted computing power.

For the ERWS up to now two DOIFs are implemented in a prototypic version:

- DOIF-E: the DOIF for EDITH control allowing in principle the EDITH control as known from the handbox (the test and back-up interface for EDITH).

- DOIF-C: the camera control interface.

Both DOIFs are connected to KISMET for simulation and/or graphics parameter input. The DOIF-E/KISMET cooperation provides a realistic simulation and training tool for manual device operation; the DOIF-C/KISMET cooperation provides the "Integrated Viewing" component of the workstation combining standard camera viewing control with KISMET based synthetic viewing.
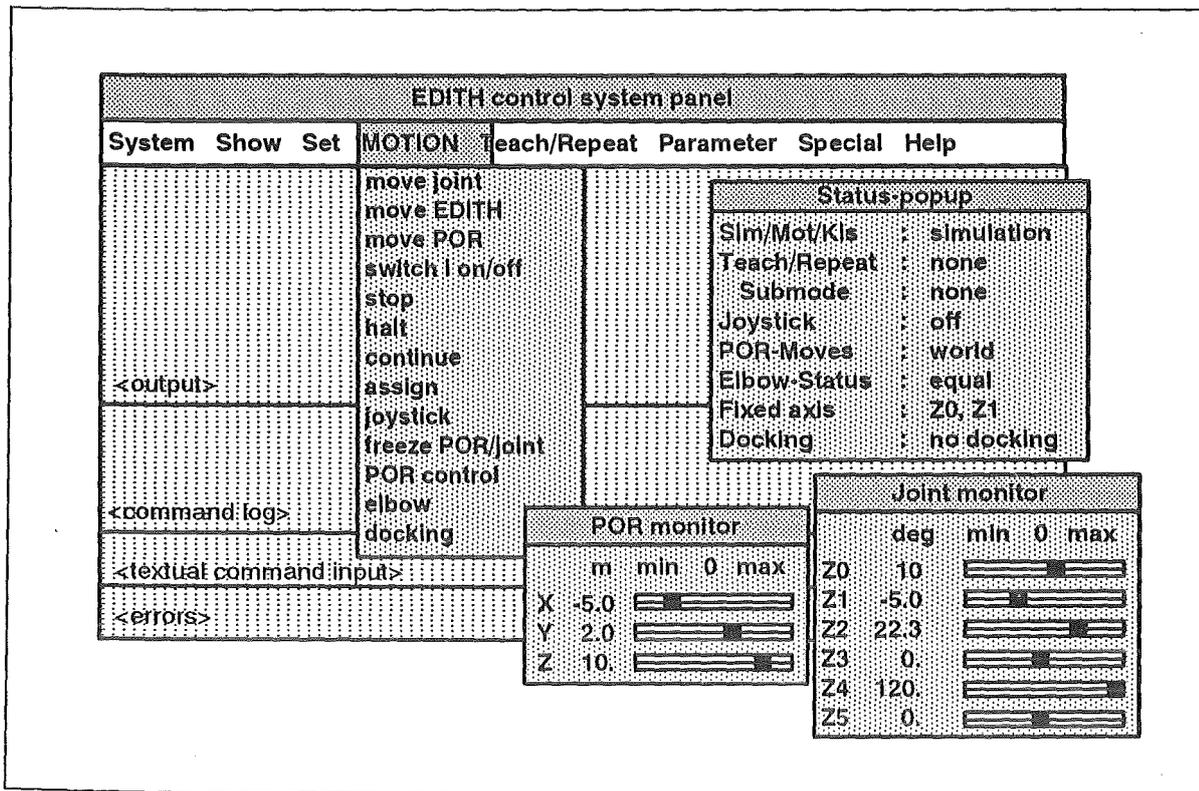
Figure 45. Control Panel of the Direct Operating Interface for the EDITH Control System. The main control panel provides pull-down menus for command input, the commands are arranged in classes shown in the top line of the panel. The basic window has additional fields for command logging, textual command input, and error messages. The status of the EDITH system can be displayed in a separate status window. Further monitoring windows are available e.g. for the joints and the POR (point of reference).

3.4.1 DOIF-E: Manual Control of EDITH

The DOIF-E provides access to the EDITH control system for manual control of the transporter. The command input interface and the parameter and status diplay is designed according to known interfacing standards. In the simulation phase the EDITH device is simulated by KISMET. The interface has a standard menu bar with basic command classes which might be pulled down. Especially for parameter display or input special pop-up windows are available for parameter groups e.g. joint angles. Motion control is possible via textual commands, joystick, mouse, or with KISMET as the synthetic master. The activation/deactivation of the whole panel or panel elements can be controlled in dependency on the task steps by PEXOS to enhance operational safety.

3.4.2 DOIF-C: Manual Camera Control

All different cameras of the EDITH system are controlled via the DOIF-C in cooperation with KISMET. The basic camera control panel allows for camera selection, pan-tilt control, and focus-zoom control. To select and vizualize camera parameters a special sub-window is available. The cameras and their viewing pyramid may be displaid and highlighted on KISMET. KISMET can also be used as synthetic master for the selected camera as well as KISMET may track a camera or to a moving or fixed scene target. In addition to camera control the DOIF-C provides the control of the video multiplexer to allocate video signals to displays including video windows on the computer screens.
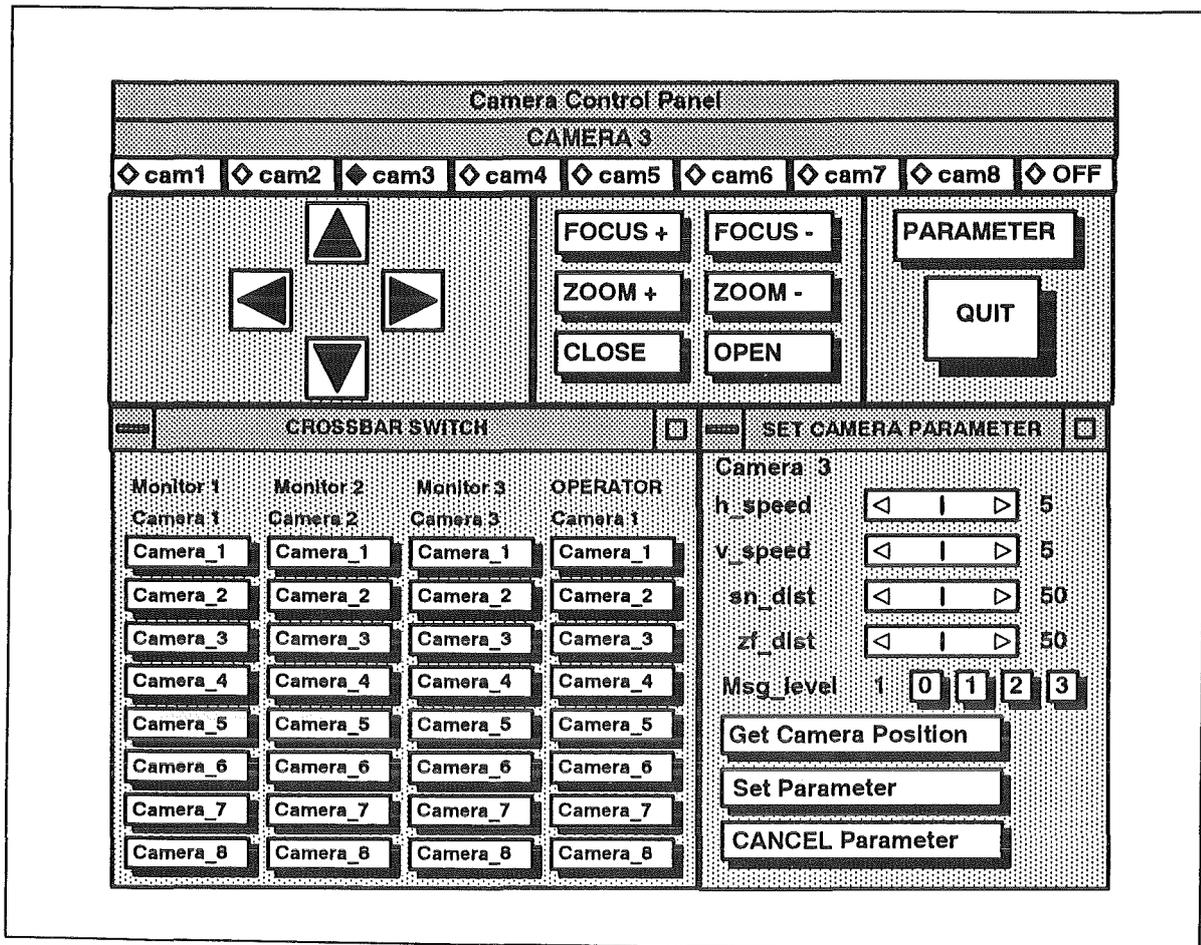
Figure 46. Control Panel of the Direct Operating Interface for Camera System. The selected camera (e.g. cam4, *) can be moved by button control (<: left, >: right, A: up, V: down), further buttons allow for focus and zoom control. In an additional window the camera parameters can be shown or set and the video signal multiplexer can be controlled.

3.5 PEXOS: Procedural Simulation for ERWS

PEXOS supports planning and execution of maintenance tasks on high abstraction level in dealing with actions, facts, their causality, and temporal relations. A well known technique for the representation of those action-fact networks for a system as well as graphically for the operator is the Petri net technique /22/ /23/. The Petri net graphic is simple enough to be easily accessible to non-experts. It is natural in some sense in representing the dynamics of a process and the causal dependences between activities and system states. The description and visualization of concurrence is included. The graphics representation of Petri nets is suitable for communication about processes like the maintenance process. The operator is confronted with ACTIONs and FACTs describing the procedures as a network or sequence of elementary actions to be done manually by the operator or automatically by the system. The execution of ACTIONs depends on FACTs describing the state of the task or the state of the maintenance equipment or the state of the plant. Therefore, the causality of ACTIONs and FACTs is described also in PEXOS by arcs connecting FACTs with ACTIONS (pre-conditions) or ACTIONs with FACTs (post-conditions, consequences of actions). To represent the dynamics of the procedure (network) a further object is used: the TOKEN. TOKENs are placed on FACTs and the ACTIONs move TOKENs from pre-FACTs to post-FACTs along the ARCs of the graph. The appearance of a TOKEN on a FACT means: This FACT is true (individual TOKEN on standard fact) or this PREDICATE (predicate FACT) predicates this TOKEN. In
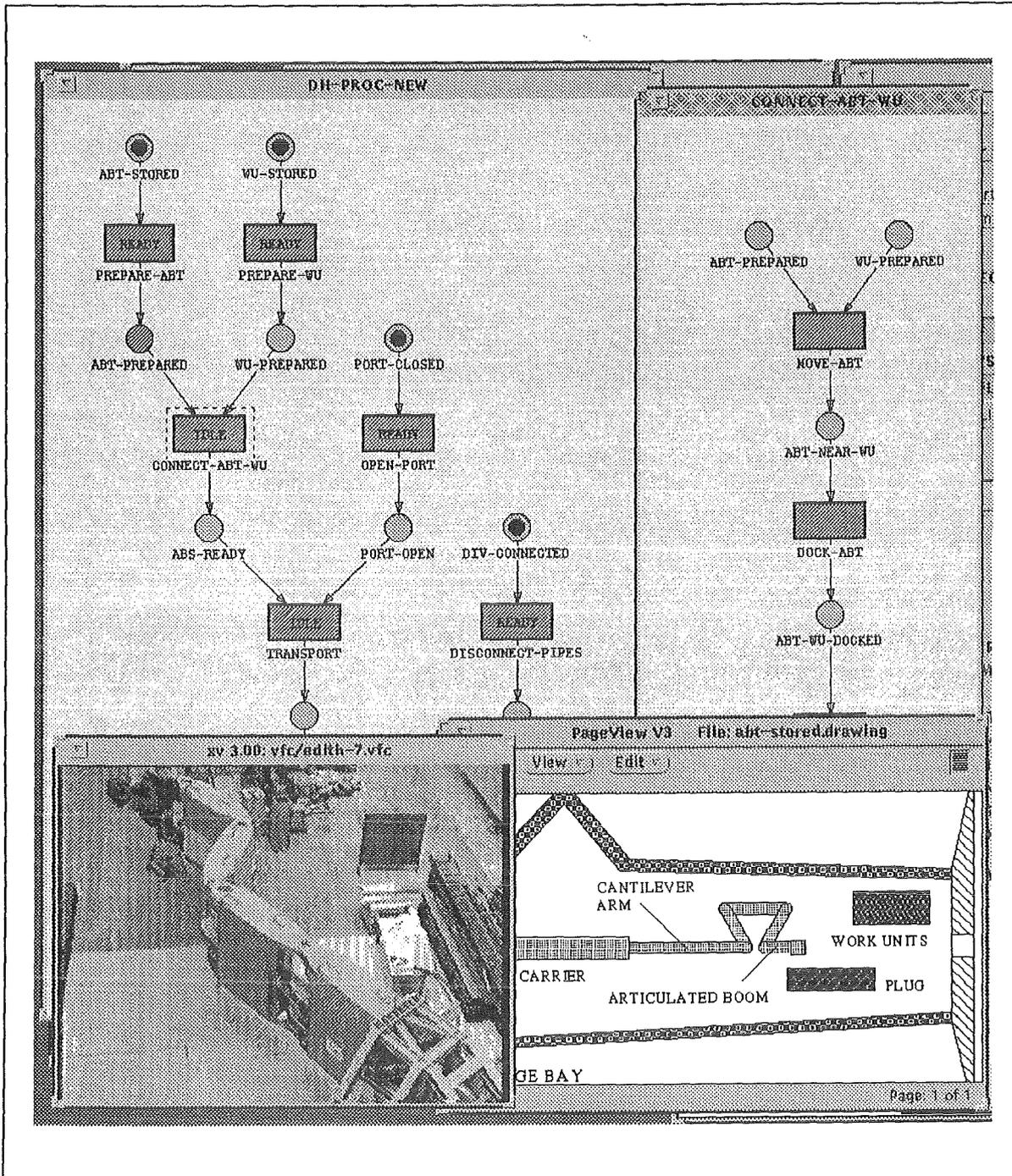
Figure 47. PEXOS Display. The display shows a basic procedure net (DH-PROC-NEW) with a subnet (CONNECT-ABT-WU) detailing an ACTION. Additionally a photo window and a drawing window are opened to describe the status. Access to the information is possible via object forms (Figure 50 on page 84 to Figure 54 on page 88) which are attached to the selected (highlighted) object in this graph. (Note: This is only a procedure example to show the principles of PEXOS).

Figure 48 on page 82 the FACT "ABS available" is true or in Figure 49 on page 83 the ABT-1 is available but not any other ABT. In the latter case of individual TOKENs the TOKENs represent the resources with which or on which the action works. The flow of TOKENs through the network represents the flow of state changes of the work and the system. PEXOS describes the tasks on a symbolic level but by integrating less abstract description means as KISMET device, environment, and motion path representations, functional device representations by DOIFs, and all standard description and documentation means like videos, photos, drawings, and texts.
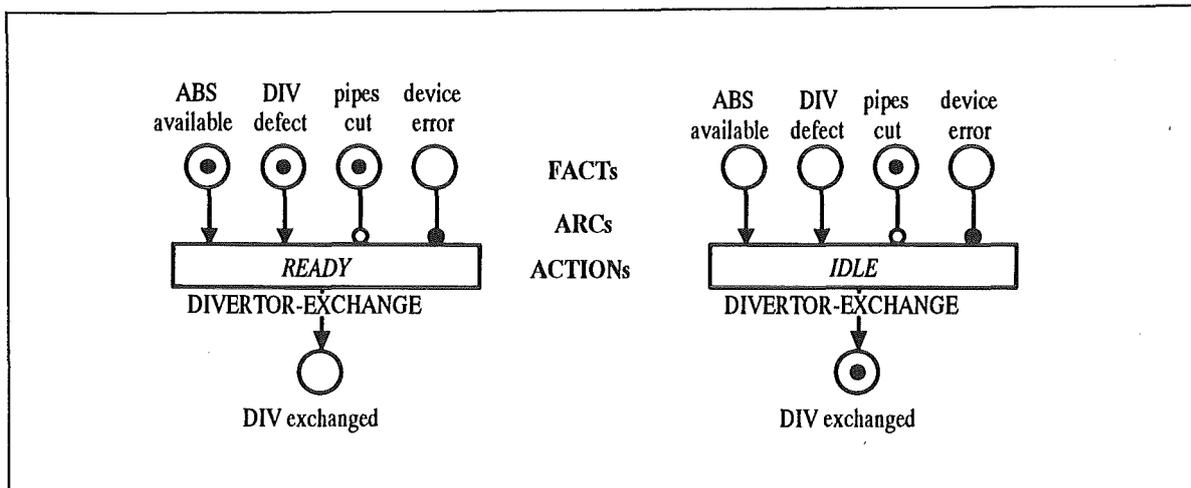
Figure 48. Simple PEXOS Procedure Example. The divertor-exchange ACTION is READY for execution because the articulated boom system (ABT) is available, the pipes are cut, no device error occurred, and a divertor is defect. All the FACTs connected to the ACTION by STABDARD-ARCs (with an arrow) or TEST-ARCs (open bullet) must be true, FACTs connected by INHIBIT-ARCs (closed bullet) must be false. The TOKEN in the FACT marks a true FACT. After execution of this action the divertor is exchanged (right hand side picture).

The operator deals with ACTIONs telling him what he has to do where and how, and with FACTs telling him about the state of the plant, the devices, and the execution state of the procedures. PEXOS provides the operator with various informations describing the FACT, e.g. with special drawings about the port door, or photos, and texts. To become familiar with the ACTION the operator may get related KISMET sequences, real video sequences from former excecutions of this task etc. In other words, the operator uses PEXOS in principle like a handbook, but this electronic handbook has much more capabilities. The basic capability is the simulation of procedures and the interaction of PEXOS with the control system for synchronisation of the handbook with the reality. PEXOS plays the role of the high level coordinator of all maintenance work, manual work as well as automated work.

In the following chapters PEXOS is mainly described by describing the properties and available functions of the PEXOS objects the operator has to deal with:

* PEXOS as a whole
* NETs, representing the work procedure as a whole
* ACTIONs, describing the work steps
* FACTs, describing the status of the work in general
* ARCs, describing the causality between ACTIONs and FACTs

In further chapters more details about the realization are presented.

3.5.1 PEXOS: Working with the System

To be able to work with procedures and their elements the operator is at first confronted with PEXOS as a whole represented on screen by the PEXOS main panel (Figure 50 on page 84). To start PEXOS he has to decide whether he will edit new procedures (programming) or execute procedures (simulate). For both modes an application domain has to be selected (e.g. EDITH, ROBERTINO). The applications domain selection provides the system with application specific information on ACTIONs and FACTs stored in system knowledge bases. Procedure selection is supported by short description, selected procedures can be opened for editing or execution, new proce-
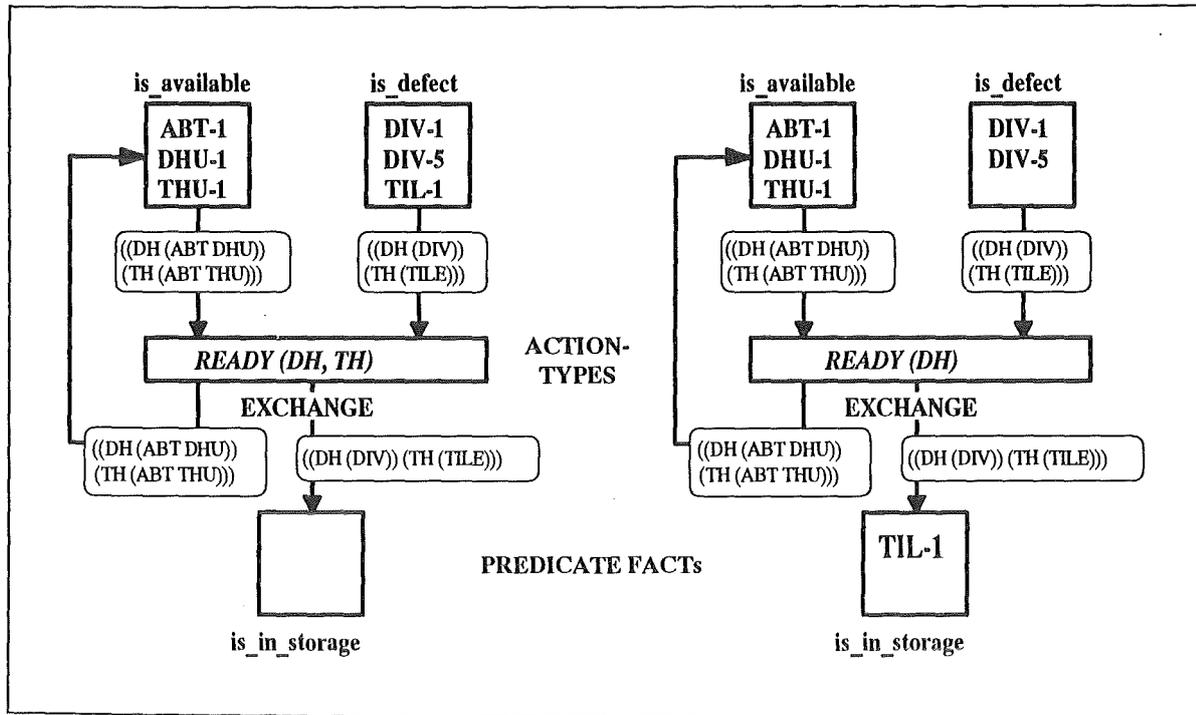
Figure 49. NET with Individual TOKENs and ACTION-TYPE. The articulated boom transporter 1 (ABT-1), the divertor handling unit 1 (DHU-1), and the tile handling unit 1 (THU-1) are available. Divertor 1 (DIV-1), divertor 5 (DIV-5), and tile-1 (TIL-1) are defect. The system is READY for divertor handling (DH) and tile handling (TH). The exchange task can start for TH: it needs an articulated boom transporter (ABT), a tile handling unit (THU) and a tile (TILE) for execution. During execution ABT-1, THU-1, and TIL-1 are allocated by the EXCHANGE ACTION. After tile exchange (execution of TH-type EX-CHANGE) the exchanged tile is in the storage and the ABT-1 and THU-1 are available but now the ACTION is only READY for DH.

dures can be created. PEXOS can be run together or without KISMET and the DOIF-SS. These two subsystems are represented in PEXOS by separate panels for supervising purposes (e.g. switching the connection ON/OFF).

3.5.2 NETs: Representation of Work Procedures.

PEXOS NETs or SUBNETs are graphs constructed with ACTIONs, FACTs, and ARCs (pre-arcs, post-arcs). A NET is interpreted as a work procedure describing work to be done by the operator manually and automated work which will be started by PEXOS if the related ACTION is started. A PEXOS NET can be simulated off-line, thus providing a guided training tour through the work package and on-line helping the operator by advices and information and unburdening the operator by doing directly some control work on the device control but also doing task and task execution dependent controlling of KISMET and the DOIF-SS to unburden the operator. PEXOS coordinates all the actions of the system and the operator. A NET can be seen as a high level program of the complete RH system including the operator.

The basic model structure of a NET is an ACTION with pre-FACTs (pre-conditions) and post-FACTs (consequences). An ACTION is ready for execution when the following conditions are fulfilled: The pre-FACTs must be true (FACT with TOKEN) if the connecting ARCS is a standard ARC (with an arrow) or a test ARC (with an open bullet), and false if the connecting ARC is an inhibit ARC (with closed bullet Figure 48 on page 82). The execution of an ACTION transports TOKENs through the NET, thus representing the NET dynamics by a TOKEN flow, by the state changes of the system and the ACTIONs. This
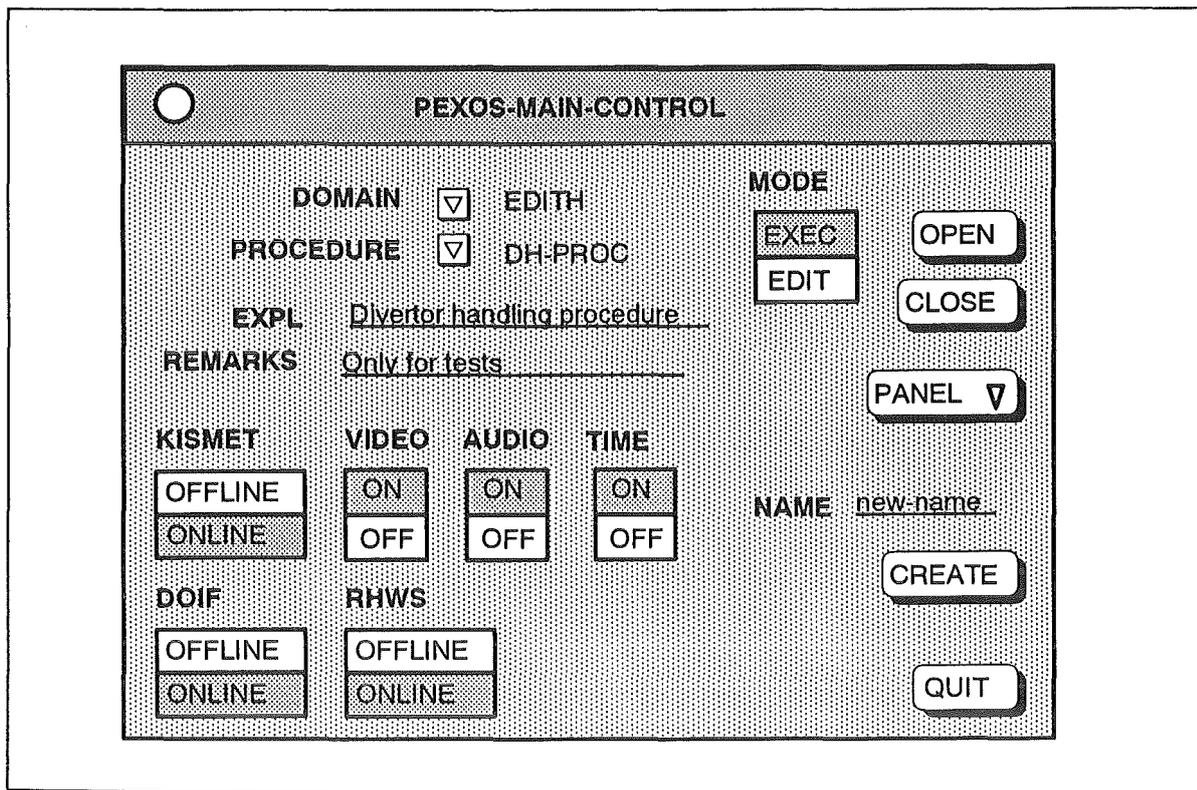
Figure 50. Main Control Panel. To work with PEXOS the operator has to select a procedure in an application domain, to select whether this precedure shall be edited or executed, and to connect PEXOS to KISMET and the DOIF-SS if needed. Short information on procedures is available to support procedure selection.

kind of TOKEN has no identity, they only are marking a FACT as true or false. A more powerful modelling technique uses individual TOKENs, representing resources of the work (e.g. a tool). The basic structure of this type of NETs is shown in Figure 49. These individual TOKENS are described by PREDICATE FACTs (if a TOKEN is placed on a FACT). In this case the ACTIONs have to be understood as ACTION-TYPES, they may act on different objects. In the example the EXCHANGE ACTION may be DIVERTOR-EX-CHANGE or TILE-EXCHANGE. The TOKEN transport conditions in this case are described by ARC annotations (rectangles with rounded corners) not displayed in the NET visualization but accessible through the ARC form. These annotations represent a transport filter or transport requirements for the ACTION. This type of modelling makes the structure of the procedure model more compact. Figure 47 on page 81 shows a procedure with standard TOKENs and concurrent ACTIONs: The example shows a very abstract divertor exchange procedure to give an impression about the graph graphics.

In addition to the structural representation of the NET, the NET as a whole is represented by a NET control panel (Figure 54 on page 88) allowing for NET manipulation and access to NET information. In addition to a textual description of the NET, videos, drawings, photos, KISMET simulations, may be available for information. The NET can be simulated step-by-step, back-stepping is possible to saved states and the execution time of the NET ACCTIONS may be monitored if demanded. More than one NET may be opened but only one NET is the actual NET to which the panel refers.
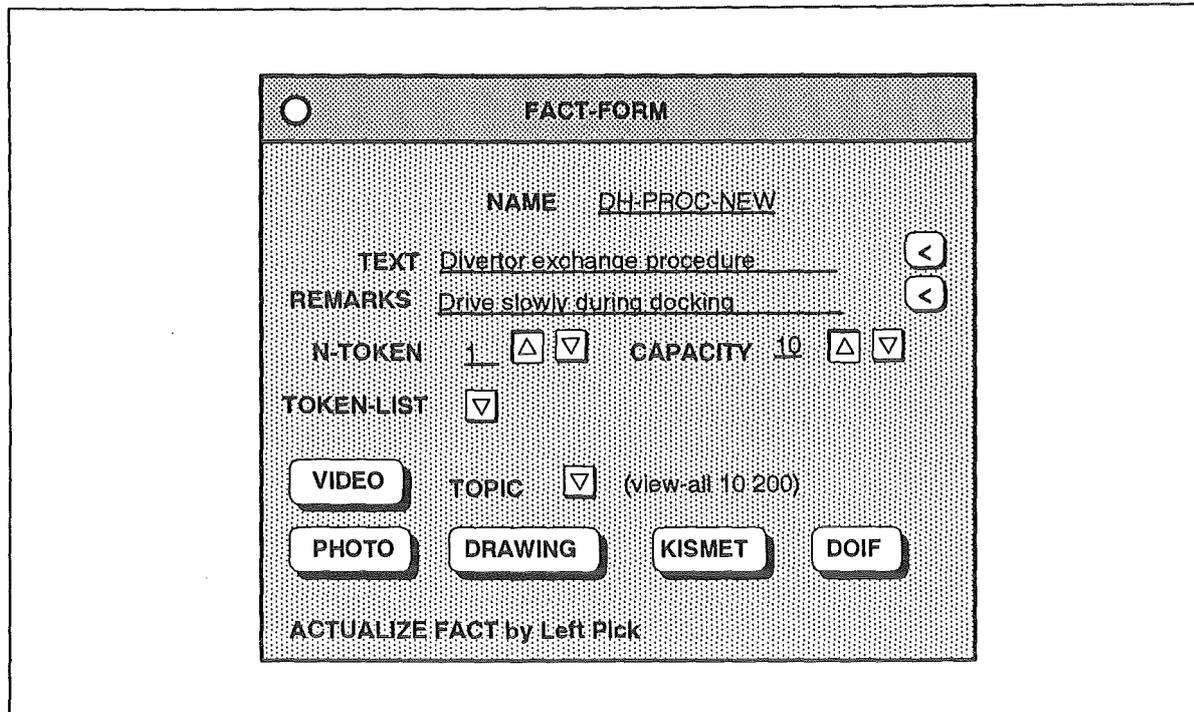
Figure 51. FACT Panel. The FACT form provides access to various information types by button control. The form also provides access to TOKENs placed on the actual (highlighted) FACT.

3.5.3 ACTIONs: Description of Work Steps.

The central object of a NET is the ACTION representing a part of work to be executed. The basic state of an ACTION is the IDLE state. If all preconditions of an ACTION are fulfilled the ACTION becomes READY. The pre-conditions are fulfilled when all FACTs are true (with token) or the required TOKENs (according to ARC inscriptions) are available on the PREDICATE-FACT. A READY ACTION may be started by the operator which sets the ACTION in the RUNNING state. A RUNNING action denotes in the manual mode that the operator is working, in the automatic mode, that one of the devices is working and the operator is supervising this action. An execution time may be attached to the action: this is the planned time which is used by the action monitor for detecting errors or problems. In the simulation case this time may be used for finishing an action automatically. The RUNNING state is ended by a done message coming from the external system or from the operator. An external system may send a message to stop the action. This brings it into the HALT state stopping also the action clock (monitor). Resuming an action has to be done by the operator. There may also arrive an error message from the outside which sets the action into the ERROR state. The error state will also be entered by the internal time monitor if the planned time is consumed. Also the error state has to be finished by the operator. The normal end of the RUNNING state is a DONE message which switches to the IDLE state. Starting an ACTION implicitly may send messages to external systems (e.g. to start an automatic action of a device or to RHWS subsystems) to unburden the operator in using KISMET or the DOIFs. A typical message to the DOIF-SS is to activate DOIFs for manual control of a device. A typical message to KISMET is to set the right view for monitoring a device in a certain situation. Via each ACTION the operator has access to various types of information about the ACTION.
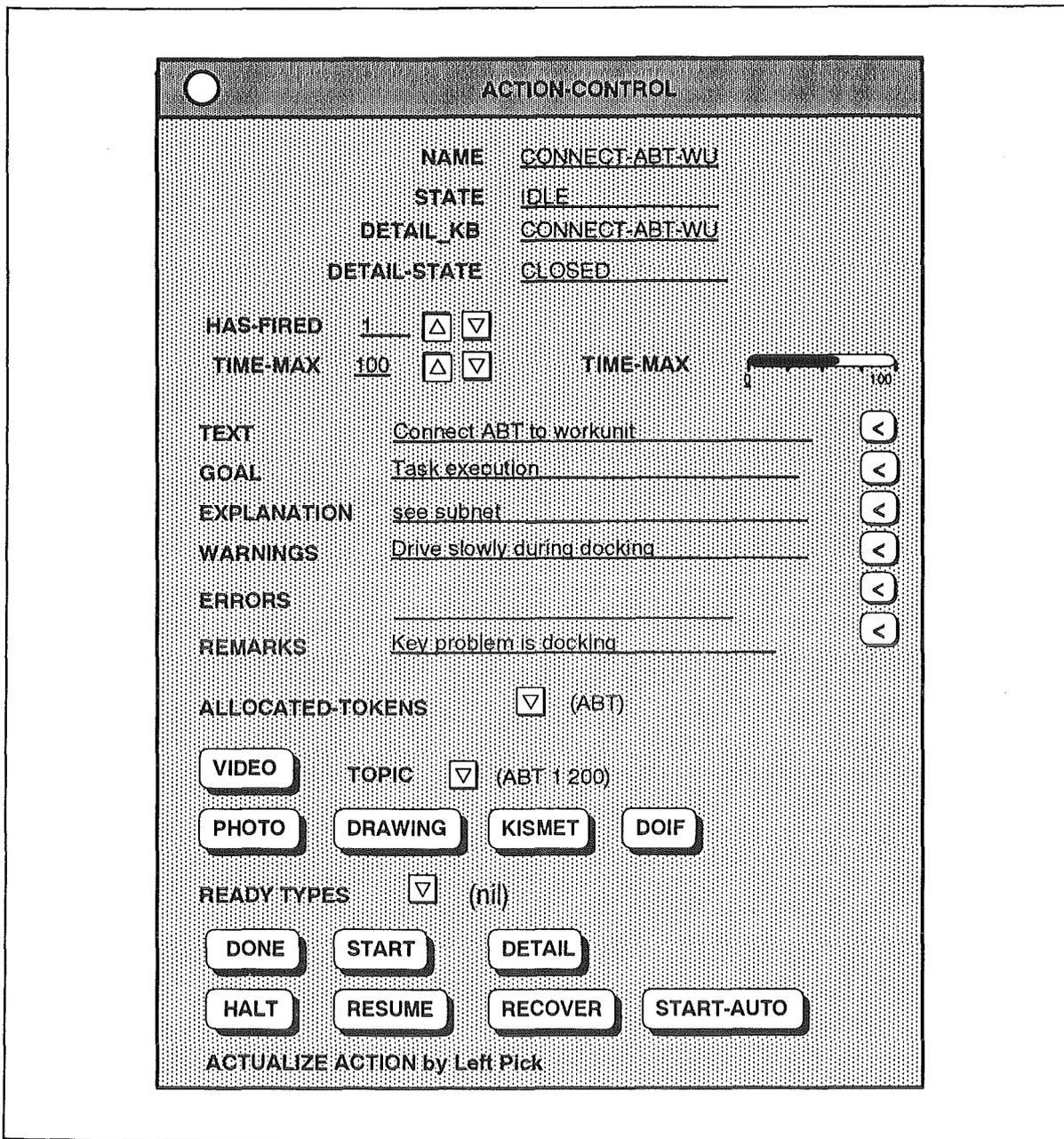
Figure 52. ACTION Panel. The ACTION panel is in the execution phase the most important panel because it provides all information and advices for the workstep to be performed (manually or automatically). Via this interface also the advancement of the work is manually controllable (START, HALT, RESUME, RECOVER).

Attributes

The ACTION has some attributes describing the general performance and behaviour of the action. An ACTION may be described by a TIME-MAX, a time span planned for the action execution. The monitoring subsystem of the action records the TIME-START and supervises the run time (TIME-RUNNING). If the run time exceeds the maximum time the action is set into the error state. If the running action is interrupted (HALTED) the TIME-HALTED is recorded. Furthermore, the system records whether the ACTION has been running or has been interrupted.

Refinement

An ACTION may be refined and described by a NET itself. The detailing NET has the preset and postset (the in-FACTs and the out-FACTs) as border elements to the environ-
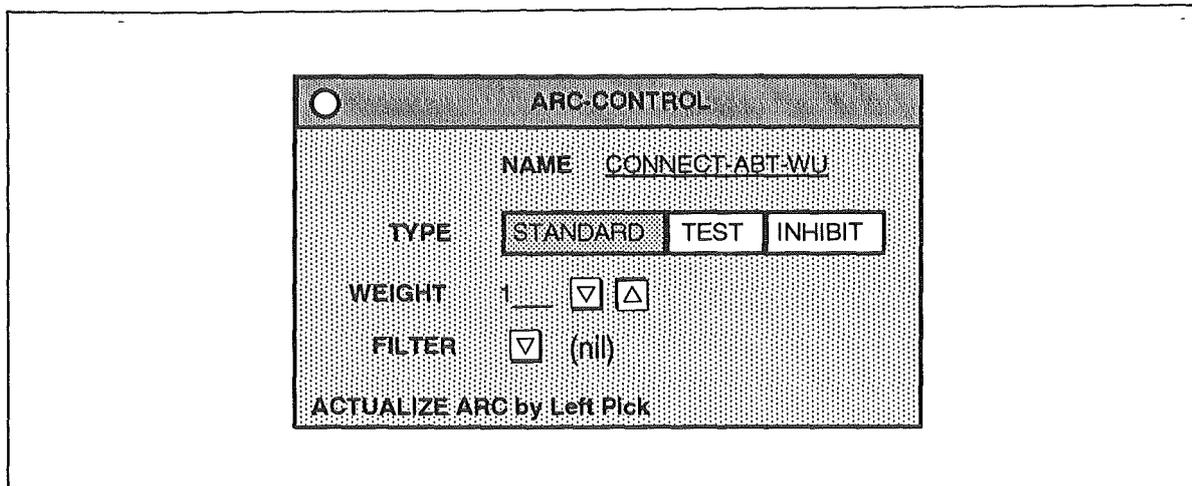
Figure 53. ARC/Connection Panel. This panel is of importance mainly in the planning phase to "program" the causalities between ACTIONs and FACTs.

ment. If a SUBNET is open the TOKENs of the preset are tranferred to the corresponding subnet FACTs if the ACTION is started. The activated father ACTION will be RUNNING during the execution of the subnet. If the subnet is finished the control can be given back to the father net and the father action will switch to the IDLE state.

Classes

ACTION objects may represent classes: an ACTION may represent "EXCHANGE-DIVERTOR" or may represent the class "EXCHANGE <type>". The ACTION description contains variables which will be bound to TOKENs (resources) in the case of execution. For example, the EXCHANGE action may describe the exchange of the divertors (using the divertor handling unit and the articulated boom transporter) and at the same time the tile exchange. That means the exchange action can be activated for divertor handling or tile handling. This technique of modelling procedures makes the procedure structure more compact.

User interface

The panel for the actual selected ACTION (Figure 52 on page 86) is the most important panel for the operator. With this panel he controls the advancement of the task performance and he also gets the most important information about the task. The operator has access through the panel to all information pieces available for the ACTION and he can make annotations the the ACTION for further usage.

3.5.4 FACTs: Description of the Work Status

The state of the system or of the task is described by the PEXOS FACT. Two classes of FACTs are available: STANDARD or simple FACTs (visualised by circles) and PREDICATE Facts (visualised by squares). The FACT "machine available" is true if the FACT hosts a standard TOKEN. The number of TOKENs can be interpreted as a capacity e.g. of available machines. With the PREDICATE FACTs a more flexible kind of state description is used: the predicate for individual TOKENs. Figure 49 on page 83 shows a net with such individual TOKENs and PREDICATE-FACTs instead of anonymous TOKENs on STANDARD FACTs. Individual TOKENs (objects, resources) are predicated according to the PREDICATE they are actually related to. With this construct a fact "machine M1 and machine M2 are available" can be expressed with one predicate "<machine> available" and the two TOKENs M1 and M2 on the predicate instead of needing two FACTs with the labels "M1 available" and "M2 available". The PREDICATE FACTs to-
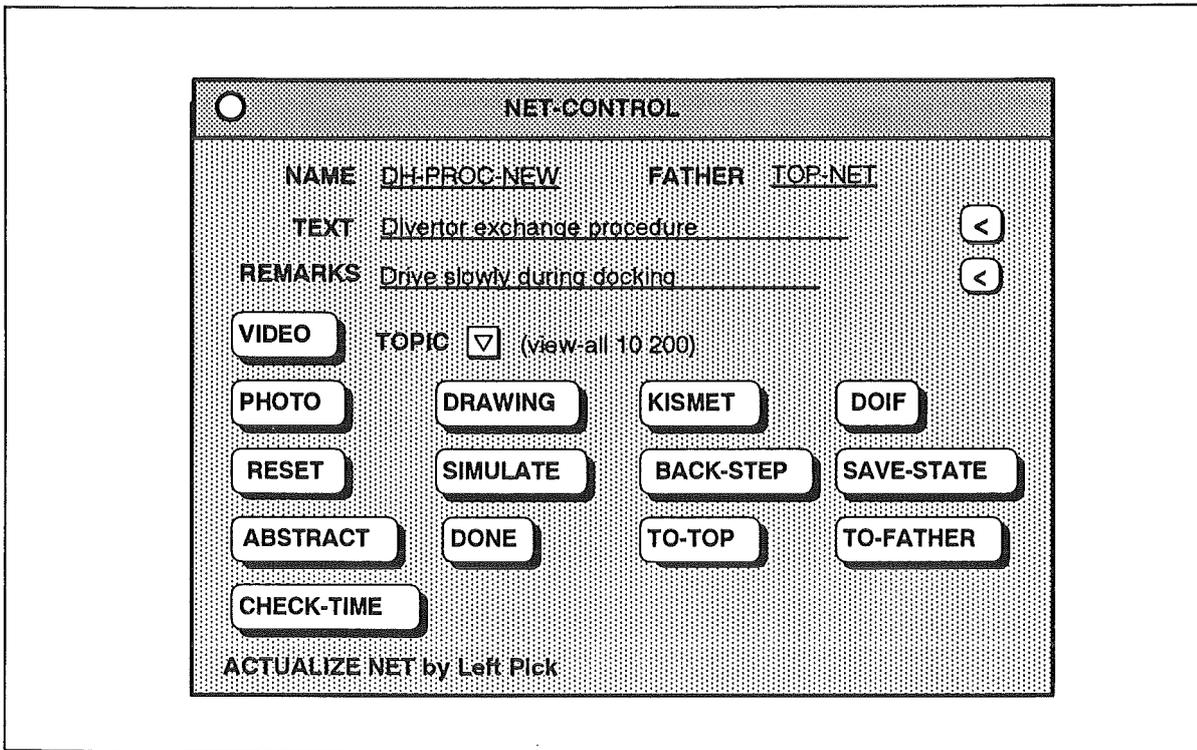
Figure 54. NET Control Panel. This panel allows for controlling the simulation of a whole procedure, but also provides information describing the procedure.

gether with individual TOKENs allow for a more compact description of task structures, they prevent for repetition of task structures.

User interface

In the actual implementation the STANDARD FACTs are drawn as circles and the PREDI-CATEs as squares. The highlighted square/circle is called the actual FACT and this is represented to the user by the so-called FACT-FORM (Figure 51 on page 85). The fact form describes the actual FACT directly by texts or allows for accessing further information by buttons.

3.5.5 ARCs: Causality between FACTs and ACTIONs

ARCs or connections are describing the causal dependencies between FACTs and AC-TIONs, they also represent the paths for TOKEN transport. An ACTION can become READY if all PRE-ARCs of the ACTION can fulfill the required TOKEN transport. In the case of STANDARD FACT as starting point of an ARC (PRE-ARC) the WEIGHT of the ARC denotes the number of anonymous TOKENs to be transported. In the case of individual TOKENs the ARC labels (these annotations in rounded boxes are not visible in the operator's NET) describe for each ACTION type the needed TOKENs. The ARCs are transporting TOKENs and the labels represent a filter for the TOKEN transport. The filter has the following syntax: <label> ::= (<action-type> (<token-type> ..... .). This means: If the ACTION of type <action-type> is started TOKENs of type <token-type> are transported from the PREDICATE to the ACTION. If the ACTION is finished TOKENs of the denoted types will be transported to the following FACTs.

User interface

The user interface only informs the user about the type of ARC. Its use is more important in the editing phase (Figure 53 on page 87).

3.5.6 Domain Specific Knowledge

The ACTION-FACT network represents the general working sequence independent of a certain application. But the system is prepared to deal also with the semantics of the work described. The ACTIONs and FACTs are interpreted for the operator by meaningful names, texts, and all the multimedial descriptions, but also for the system itself by domain specific subtypes of the ACTIONs and FACTs. The resources and their structure may be modelled to put into PEXOS information about the devices or the machine to be maintained. This type of knowledge will be modelled by PARTs and relations on parts relevant for maintenance. The goal is to model the knowledge on procedures as well as the knowledge on the tools available. This domain specific knowledge is used and will be used in future enhancements for the planned subsystem for heuristic knowledge to support the operator more intelligently. This domain specific knowledge describes the application devices, environment, and standard subtasks to enable the system to give support based on heuristic (rule based) knowledge. The system is therefore prepared to give domain specific reactions based on heuristic knowledge.

3.5.7 Multimedia Data Organization and Processing

Under the handbook aspect the PEXOS NET plays the role of the user's guide, of an index, of a list of content, list of figures. But the PEXOS NET also ties together the various elements of information describing the working procedures as a whole, the ACTIONs, and the FACTs (system states). Through this net the operator gets task oriented access to the information elements. The various windows for information display are shown in Figure 55 on page 90. Figure 57 on page 92 shows by which processors this information is generated, stored, and displayed and Figure 56 on page 91 shows the structure of the multimedia database. PEXOS manages the following types of information:

* The graphical presentation of the working procedure in form of a bipartite directed **graph (2D graphics)** is the basis for further information access. To access more and detailed information of the net and the net elements the so-called actual (highlighted) elements are defined: actual net, actual action, actual fact, actual arc (connection). The additional information for each actual element is presented in a related, object type specific panel. The panels provide short information about the actual object, provide access to detailed multimedia information by opening additional windows by command buttons, and provide object control.
* The basic information about the procedure and its elements is **textual information** in the form of a handbook. This kind of information is generated using a standard word processor as for example ISLAND-WRITE or MS-WORDS. To be integrated into the PEXOS multimedia database (MMDB) the texts have to be formatted into a Postscript format, it is presented on the PEXOS screen with the PageView Postscript post-processor. In the database this file of information is qualified with ".manual".
* Actualization of a procedure element outputs **audio information** if available for this element. Audio information is normally used for special important information: warnings, state changes etc. The audio information is stored in the MMDB qualified with ".audio". Editing of audio information is done with the standard SUN Audio-Tool.
* Each net element or the net as a whole can often best be described by a set of **drawings** if geometric relationships are important. For editing or creating drawings the standard tool on the PEXOS-SUN is IslandDraw. Because the MMDB format for drawings is PostScript also drawings from other sources can be processed. A
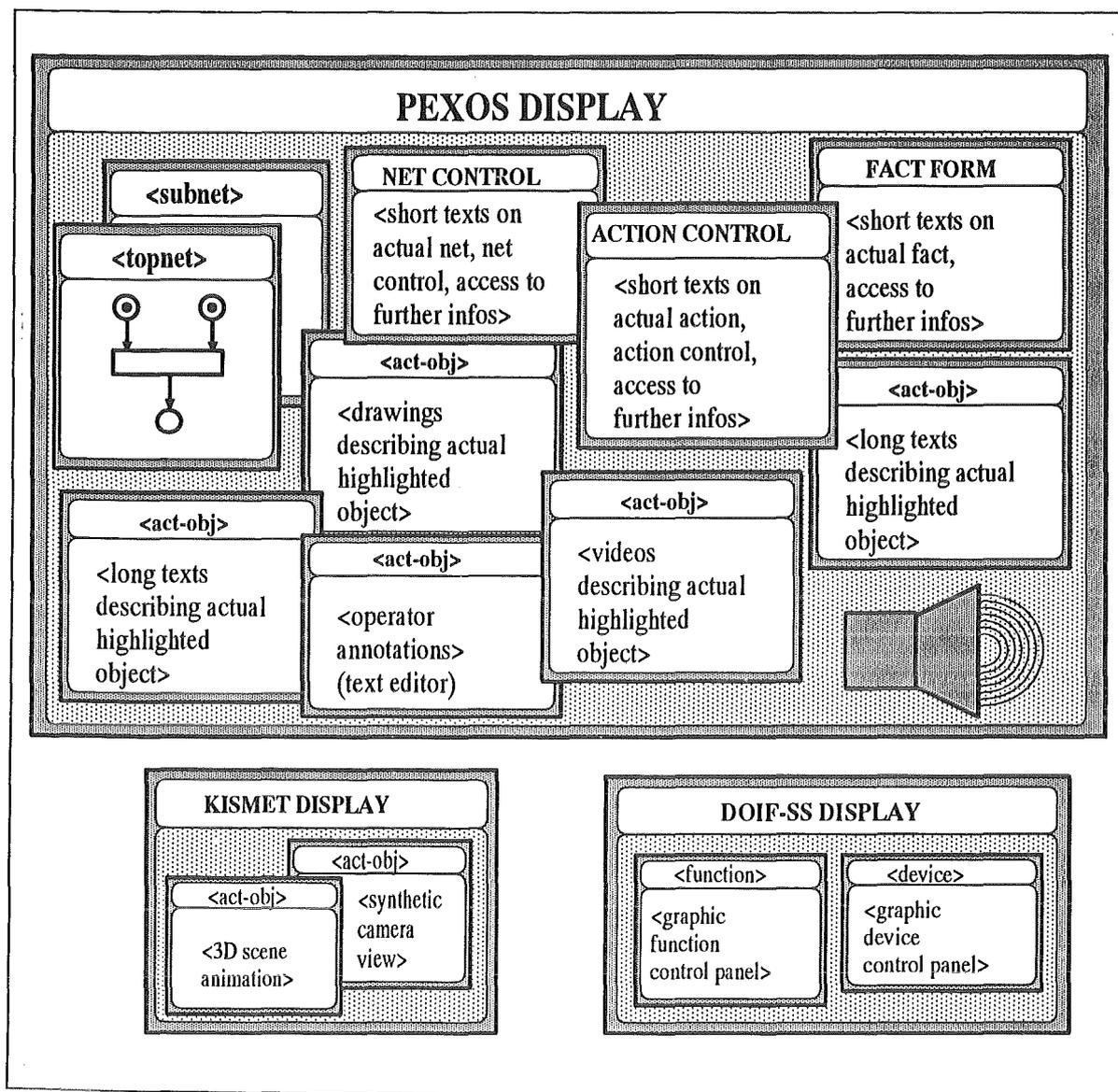
Figure 55. Multimedia information presentation on screen. The figure shows the maximal amount of information available on the three workstation displays (PEXOS, KISMET, DOIF). In normal cases not all of the specific windows are opened in parallel. In the actual implementation the operator controls which window (which piece of information) is viewable on screen. The actual object (<act-obj> , the object of interest) is highlighted in the PEXOS NET (!). On the DOIF screen the device and/or function panels related to the actual action is shown.

drawings file is qualified with ".drawing". For displaying drawings the PageView processor is used.

* To show situations or devices most realistic, **photos** may be attached to the object description. The format used for this type of information is the SUN-Raster format. The display of a photo in the special window is performed by the DISPLAY-PHOTO procedure.

* Special situations or actions can often be presented very impressive by **video sequences**. In PEXOS video spots may be stored on a video disc. This videos may presented on the PEXOS screen in a special video window which may also be used to display an actual image from a camera.

* Especially to instruct the operator about motion paths an **animated 3D graphics** from KISMET can be attached to an action. This information is activated via PEXOS

```
APPLICATIONS
|
|---<domaine>
|     |---<domaine>.u            object types of <domaine>
|     |---<domaine>.manual       domaine description
|     |---<domaine>.remarks      operator remarks, domaine related
|     |
|     |---<topnet-1>             top procedure 1
|     |     |---<topnet-1>.u       knowledge base
|     |     |---<topnet-1>.h       graphics
|     |     |---<topnet-1>.manual  procedure texts
|     |     |---<topnet-1>.remarks operator remarks, procedure related
|     |     |---<topnet-1>.drawings drawings for procedure
|     |     |---<topnet-1>.photos  photos for procedure
|     |     |---<topnet-1>.audio   audio data for proc
|     |     |---<fact-i>           subdirectory for fact data
|     |     |     |---<fact-i>.manual
|     |     |     |---<fact-i>.remarks
|     |     |     |---<fact-i>.drawings
|     |     |     |---<fact-i>.photos
|     |     |     |---<fact-i>.audio
|     |     |     |
|     |     |---<action-i>          subdirectory for action data
|     |     |     |---<action-i>.manual   action text
|     |     |     |---<action-i>.remarks  operator remarks, action related
|     |     |     |---<action-i>.drawings action drawings
|     |     |     |---<action-i>.photos   action photos
|     |     |     |---<action-i>.audio    action sound
|     |     |     |---<subnet-i>           subdir for detailed action
|     |     |     |     |---<subnet-i>.u       subnet knowledgebase
|     |     |     |     |---<subnet-i>.h       subnet graphics
|     |     |     |     |---<subnet-i>.manual  subnet texts
|     |     |     |     |---<subnet-i>.remarks operator remarks, subnet related
|     |     |     |     |---<subnet-i>.drawings subnet drawings
|     |     |     |     |---<subnet-i>.photos  subnet photos
|     |     |     |     |---<subnet-i>.audio   subnet sound
|     |     |     |     |---<fact-i>
|     |     |     |     |---<action-i>
|     |     |     |
|     |---<topnet-i>
|     |     |
|

<domaine> ::= EDITH | CATROB | INGRID | ROBERTINO | <others>
```

Figure 56. **PEXOS Multimedia Application Databases.** The PEXOS database is a hierarchical file system which is constructed automatically by editing a procedure graph. The multi-medial information of type manual, remarks, drawings, photos, and sound have to be introduced into database according to the above structure to be accessible via the procedure graph. The management of those infos is done with the standard tools.

but displayed on the KISMET screen. With 3D graphics and a moving viewpoint also complicated spatial situation can be made understandable.

*   From a functional point of view the devices are represented on the workstation by DOIFs. The DOIFs are visualized on the DOIF screen by a special **control panel graphics**. These DOIF panels are activated and deactivated via PEXOS. For example, if the operator has to transport an object with a device the device panel is activated showing the operator the control possibilities and making device helps available.

*   **Textual annotations** of the operator for each object can be presented and edited through a further window, in which a simple text editor is available. The annotation files are qualified with ".remarks" in the MMDB.

Figure 57. Multimedia Information Management. The PEXOS multimedia information elements are generated, updated, and displayed by standard processors. In addition to the display formats in some cases (manual, drawing) also basic format files are held usable for further editing.

3.6 Control System

3.6.1 Overview of the Control System
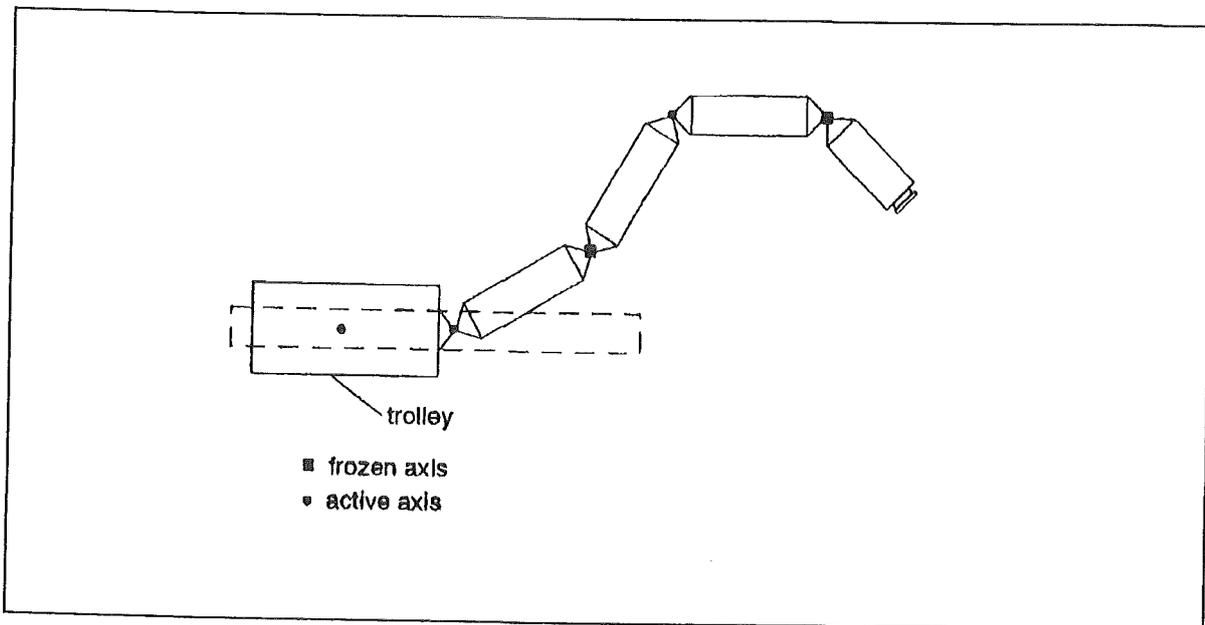
Figure 58. Redundant Kinematics. To get unique inverse kinematic transforms, the system is reduced to a non-redundant system by freezing two of five axis.

Controlling the EDITH ABS demands a specific Control System. So the EDITH Control System is designed to control a manipulator up to 13 axis. As far as possible it has been realized as a control system for a general manipulator, which is extended due to the specific requirements of the ABS.

The Control System provides two modes:

• Operator controlled or manual mode

• Automatic mode

In manual mode the manipulator is controlled by operator interaction. In automatic mode however operator action is reduced to starting the execution of a sequence file, the program, which has been taught before in an interactive mode, quite similar to the manual mode. Both modes are working in simulation mode (manipulator offline, closed loop control inactive) as well as in motion mode (manipulator online). Thus, with simulation mode the Control System offers a facility, which can be used for operator training and development of motion programs.

In manual mode there exist three control modes to move the ABS:

• Axis position control
  In axis position control mode, the operator will input the desired axis positions via a keyboard. Also he may add via points, i.e. axis configurations, which should be passed on the way to the target point without any stop.

• Axis velocity control
  The manipulator axis will be controlled by joysticks. The displacement of the joysticks will correspond to the axis velocity setpoint.

• Point velocity control (carthesian velocity control)
  In point velocity control mode the POR (POR = point of reference) will be velocity controlled by joystick. The POR is comparable to TCP in roboter technics. It is defined relatively to the work unit mounting flange of the transporter. The Control System contains no pathplanning algorithms for a redundant kinematics. To overcome difficulties with redundant kinematics, the operator is forced to freeze axis to get an unique inverse kinematic transformation (see Figure 58 on page 92).

Controlled in the automatic mode, the manipulator may be positioned in two modes:

• Axis position control with symbolic points
  This mode corresponds to the axis position control mode in manual mode. As a difference, the axis positions are assigned to symbolic points.

• Point positioning with symbolic points
  This mode provides the positioning of the POR by declaration of its position and orientation. As in point position control an inverse kinematics will be used for pathplanning.

Assignment of axis or POR positions to the symbolic points happens in an operator interactive mode during teaching: The operator has to position the manipulator (real or virtual in simulation mode) in manual mode and takes over the actual values as a new symbolic point. Point definition and command segment are separated in motion programs. So it is possible to transfer the point database of a teached program to a new one.

Actually there is no higher automation level implemented within the Control System. A later integration of sensor based or image processing based positioning systems will be possible.
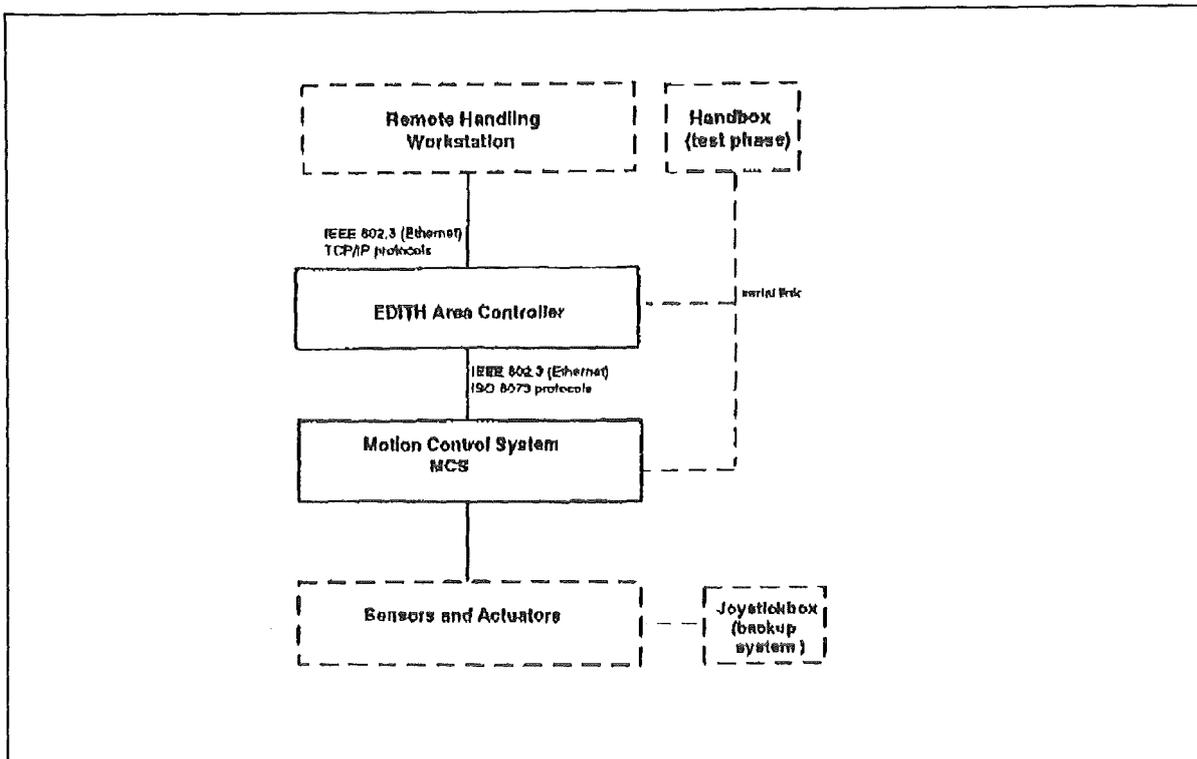
Figure 59. Architecture of the EDITH Control System

3.6.2 Implementation of Control System

Following the general requirements for the control system, a hierarchical architecture has been choosen. The control system has been split up into three subsystems (Figure 60 on page 96).

* Motion Control System (MCS), the realization of a device controller
* EDITH Area Controller (EAC)
* Operator Interface

The MCS contains the realtime section of the control system with process interfaces, i.e. digital and analog I/O-ports, path planning and closed loop control. It is connected to the EAC, which will manage motion programs and system parameters.

If the control system would break down, while the manipulator is inside the torus, it should be possible, to move the manipulator outside the torus whitout usage of Control System. In this case a joystick box will be connected directly with the servo drive units.

3.6.3 Motion Control System (MCS)

The MCS represents a simplified roboter control system. Its functionality is reduced to non-automatic operation, i.e. not programmable operation, quite similar to manual operation mode. It contains following functions:

* Path planning
* Closed loop position control
* Monitoring functions
* Tracing
* Input and output of control signals

3.6.3.1 Path Planning

There are path planning algorithms implemented in MCS, which support the four motion modes:

* Point position control mode
* Point velocity control mode
* Axis position control mode
* Axis velocity control mode

Path planning contains:

* Synchronization of axis movements in that way, that the duration of acceleration phase, constant velocity phase and deceleration phase are equal for all axis.
* Inverse kinematic transformations for cathesian movements
* Inverse kinematic transforms for reconfiguration

3.6.3.2 Closed Loop Position Control

The position loop for each axis is closed in software, using a programmable PID controller with velocity feed forward. As the EDITH testsite is an experimental device, it will be necessary to test other control structures. For this has been taken into consideration during concept phase, it will be easily posible to to implement advanced control algorithms like inverse model techniques or adaptive control schemes. The sampling time and control parameters may be set by the system manager and will be automatically restored for each work unit. The sampling time may be increased in steps of 10ms. At present, its lowest level is limited to 20ms.

As gear elasticity impairs the system's dynamic behaviour, there will be the approach of compensation by usage of motor resolver values in addition to joint resolvers.

3.6.3.3 Monitoring Functions

Various monitoring functions ensure the safe operation of the ABS. As error conditions there are:

* Exceeding of axis travel software limits
* Exceeding of axis velocity, acceleration, deceleration
* Unexpected motions and exceeding of allowed positioning error
* All digital inputs (range switches, state signals of servo drive units)
* Internal errors

The operator may define the system reaction upon any of these error conditions. There exist five options:

* EMERGENCY STOP would set a digital output which might switch off power supply of brakes and servo amplifiers. The actual motion would be stopped by switching the servo amplifiers inactive and engaging the brakes.

* STOP is quite similar to EMERGENCY STOP, as difference there would be set no digital output.

* HALT stops the movement with the operator specified deceleration.

* SIGNAL will set a digital output.

* INFO will cause a message telegram to the operator interface.

3.6.3.4 Hardware and Software platform

As a system which meets the requirements a Multibus II architecture with realtime operating system RMX III on it was selected. This combination guarantees both, the desired realtime behaviour and multitasking. Multibus II also offers scalability, i.e. the possibility to increase system performance by adding CPU-boards and distribution of tasks. This might be necessary, if there would be implemented more complex control algorithms to improve the system's dynamic behaviour.

Hardware
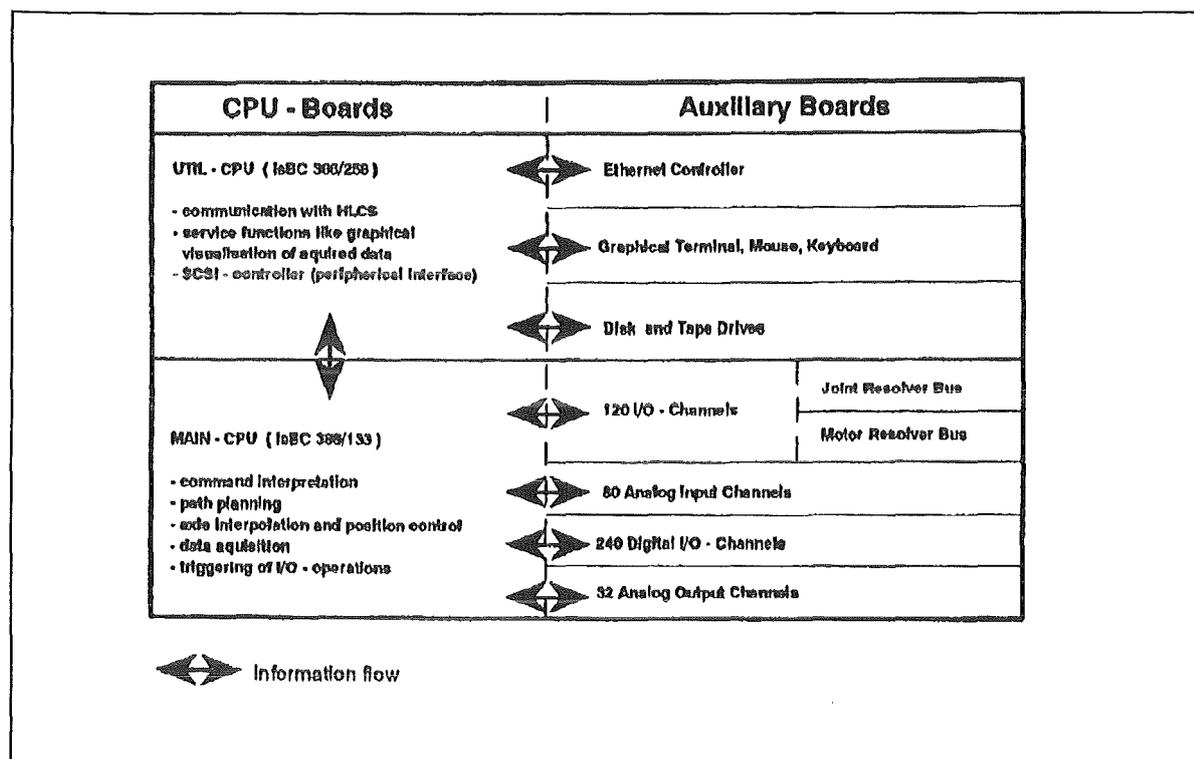
Figure 60. Hardware Architecture of the Motion Control System

As it is shown in Figure 60, the MCS consists of follwing hardware components:

* 1 386 SCSI-controller board (UTIL-CPU)
* 1 386 CPU-board   (MAIN-CPU)
* 3 digital I/O-boards (360 I/O-channels total)
* 1 Ethernet controller for connection to the EAC
* 1 analog input board (80 channels)
* 2 analog output boards (32 channels total)
* Mass storage devices (2 diskdrives, 1 tapedrive, 1 harddisk)

Sensors and actuators are interfaced by analog and digital I/O-channels.As sensors there are piezoelectric force sensors, resistive inclinometers and resolvers in use. In addition to the sensors, the actuators, i.e. the dive units, offer also some state and sensor signals. All digital channels are opto-decoupled and have an active level of 24V. The analog input and output signals have a range of ±10V. To reduce the number of digital I/O-channels, there have been built up two resolver buses, the servo resolverbus and the joint resolver bus Figure 61 on page 97. Each of these buses consists of an address bus and a data bus. In addition to these signal lines there are two handshake lines which ensure the correct timing of bus operation. Each address bus owns 5 address lines, to address up to 32 resolvers. Corresponding to the different resolution of motor and joint resolver, the motor resolver bus has 16, the joint resolverbus 24 data-

lines. To avoid disturbances, there has been choosen the RS422 signal standard with differential signal transmission between the multiplexer unit (containing the transducers) and the MCS.
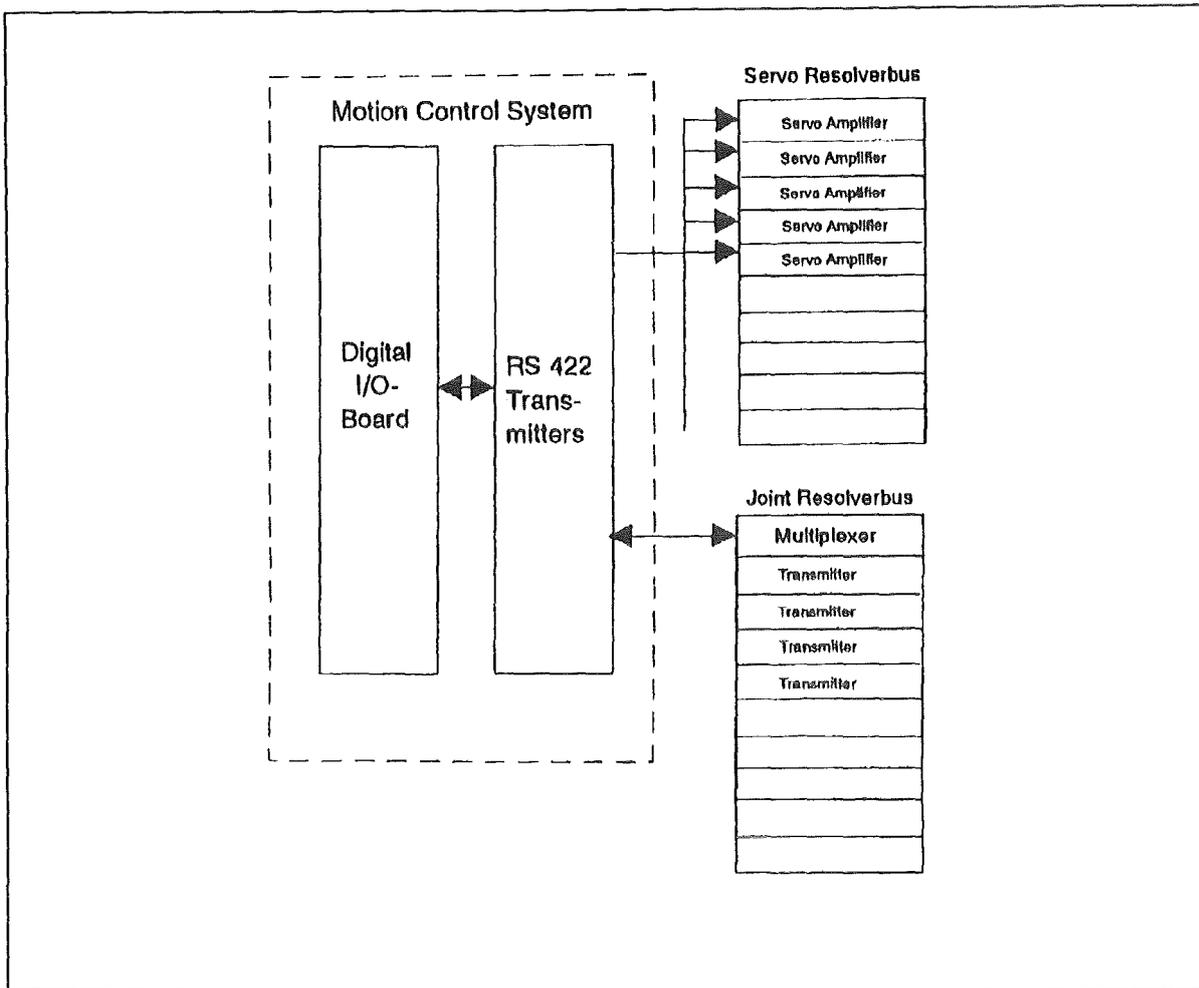
Figure 61. Resolverbuses. The resolver signals are combined to two resolver buses. The joint resolver bus has an central multiplexer which is decoding the addresses and aktivating the addressed transmitter. So there could be used standard transmitters. The servo amplifiers however have the address decoder on board. The addresses are setable via DIP-switches.

Software

Using the multitasking features of RMX, the functions has been distributed to different tasks. This gaurantees a well structured open system and will keep the system scalable. In addition, there have been added some auxiliary tasks, supporting inter board communication or communication with external devices like operator interface or EAC. There are also some additional tasks, which offer tools for graphic visualization of traced data. Inter task communication and synchronization happen via mailboxes and semaphors. There is a distributor task running on each CPU-board, receiving messages from other boards via the multibus and message passing coprocessor and distributing them to the tasks, running on the same board. System errors are bypassed and handled by an own error handler, which handles system errors and application errors commonly.

3.6.4 EDITH Area Controller (EAC)

The EAC will always be settled above the MCS. It contains higher control features like

* communication with operator interface,
* login,
* parameter handling,
* program management.

3.6.4.1 Login

Login of operators is controlled by the EAC. There are three operator sessions possible: system manager, operator and assistant operator. While the system manager is allowed to change system parameters, the operator is only allowed to change operator parameters but move the machine. The assistant operator has the lowest privilege. He is only allowed to display data but will not influence any motion.

3.6.4.2 Communication with Operator Interface

As the EAC controls the access of the MCS, all information coming from the operator interface must pass the EAC. This also makes sense, because all operator commands concerning automatic mode are performed on the EAC. The EAC controls also the system access.

3.6.4.3 System Parameters

System parameters can be manipulated only by an operator or system manager with system manager privilege. The following list shall give an overview about parametrisation of the control system.

* Parameters, which configure the communication between the subsystems (timeout, telegram frequency)

* Parameters for MCS:

  − Scaling and offset of I/O-channels
  − Definition of digital input channels (active high/low)
  − PID-parameters
  − Parameters for pathplanning (dimensions of manipulator links, limits for velocity, acceleration, deceleration)
  − Monitoring parameters (position, velocity, acceleration, deceleration limits, allowed positioning error)
  − Definition of error reactions

* Operator parameters are parameters, which will have an influence on pathplanning. In addition to velocity, acceleration an deceleration limits there are parameters, which define the position of POR relatively to the workunit mounting flange of the ABS or set an override factor to reduce the velocities due to the override factor. If there are parameters existing as system parameters and operator parameters, the operator parameters will be limited to the system parameter's value.

3.6.4.4 Program Management

As described earlier, anything concerning automatic mode is processed on the EAC. In this contenxt there exist three submodes: TEACH, REPEAT and EDIT submode. All submodes work in simulation mode as well as in motion mode. A motion program consists of three parts:

* General Description

* Point Database

* Program Sequence

The general description should contain a description of the motion program with start point and target point. The operator can input the descritional text in a text field during teaching or modificying a program. The point data base and program sequence however are created or modified automatically during teaching or modificying programs. To give a small overview about the program handling, the following sections describe the three submodes.

TEACH

The teach submode is entered if there will be recorded a new motion program. As there may already exist large point data bases within other programs, it is possible to import a database of another program. There can be imported only one database. Teaching happens in an interactive mode, which is comparable to manual mode. The operator inputs a motion command, which contains point names. If a point would not be defined, the EAC enters automatically the point definition submode. Now the operator commands the manipulator, using the motion commands, which are available in manual mode to the desired point. If he has reached the target point, he can accept the actual position and assign it to the symbolic point, which will be added to the point database automatically.

In addition to motion commands there may be teached also parameter handling commands, changing the operator parameters. There is also a PAUSE command available, which interrupts the program execution at a restpoint. The program execution then may be continued by input of a RESUME command. If the TEACH submode will be left, the program will be saved automatically to the sequence file specified while entering the TEACH submode.

REPEAT

As the MCS is designed to receive and execute only single motion commands, EAC interpretes the programs in execution mode. Program interpretation always follows the following scheme:

1. Reading and interpretation of a motion command line

2. Look up in point definition table

3. Test whether the actual position lies within the specified tolerance limits - if it is outside there are two possibilities:

   a. If the position difference is less than the tolarance limits for automatic alignment, the EAC will send a move command to Motion Control System, commanding the manipulator to the desired start position.
   b. If an automatic alignment would not be allowed, the EAC changes to manual mode. The operator is now requested to command the manipulator to the start position. If this would have happened, the operator may return to automatic mode. Program execution now would be continued with alignment check.

4. Send the motion command to the MCS

5. Wait until MCS has completed the movement

EDIT

The EDIT submode is an extension of the REPEAT submode. In addition, it offers the possibility to modify existing program files. Modification of programs means:

- Modification of program description
- Insertion of new command lines
- Deletion of command lines
- Redefinition of points

To allow a quick editing, it is possible to skip a number of commands.

3.6.4.5 Software and Hardware Platform

There has been choosen a VAX workstation as EAC with VMS operating system and programming language C on it. Multitasking capabilities allow a well structured segmentation of application software.

For communication with other system components there are different Ethernet adapters in use, which support different network protocols. Communication with MCS follows the ISO 8073 standard. However, for connection to operator interface, graphic workstation for kinematic simulation and remote workstation there are ethernet links with TCP/IP Protocol or a serial RS232 link available. To ensure that there is no overload on the ethernet connection between MCS and EAC, there is installed an own net segment, which is exclusively used by these subsystems.

3.6.5 Operator Interface

In the normal operation mode the operator interface is represented by the remote workstation as the generic interface. As operator interface during test phase there is a backup system available, the socalled handbox. It consists of an AT-compatible handbox with two 3D-joysticks which is connected to the EAC via the serial link.

3.6.6 Automation Possibilities

With exception of the repetition of motion sequences, at the present there are no higher automation functions implemented. As position measurement is only based on resolver values, torsion and deflection of the ABS will cause positioning errors. To compensate these positioning errors, it is planned to simulate the system elasticity and correct deviations caused by elastic deformation. The simulation will happen offline on the KISMET-workstation 3.3, "KISMET: Spatial Simulation for ERWS" on page 60 Based on the simulation, the positioning error may be corrected during offline programming.

The control system has been kept modular. Thus, it will be possible to automate subtasks step by step. As an example can be seen the KfK-Teleman project. Within this project there have been added automatic features based on an image processing system, using the standard interface of the motion control system.

3.6.4.4 Program Management

As described earlier, anything concerning automatic mode is processed on the EAC. In this contenxt there exist three submodes: TEACH, REPEAT and EDIT submode. All submodes work in simulation mode as well as in motion mode. A motion program consists of three parts:

* General Description

* Point Database

* Program Sequence

The general description should contain a description of the motion program with start point and target point. The operator can input the descritional text in a text field during teaching or modificying a program. The point data base and program sequence however are created or modified automatically during teaching or modificying programs. To give a small overview about the program handling, the following sections describe the three submodes.

TEACH

The teach submode is entered if there will be recorded a new motion program. As there may already exist large point data bases within other programs, it is possible to import a database of another program. There can be imported only one database. Teaching happens in an interactive mode, which is comparable to manual mode. The operator inputs a motion command, which contains point names. If a point would not be defined, the EAC enters automatically the point definition submode. Now the operator commands the manipulator, using the motion commands, which are available in manual mode to the desired point. If he has reached the target point, he can accept the actual position and assign it to the symbolic point, which will be added to the point database automatically.

In addition to motion commands there may be teached also parameter handling commands, changing the operator parameters. There is also a PAUSE command available, which interrupts the program execution at a restpoint. The program execution then may be continued by input of a RESUME command. If the TEACH submode will be left, the program will be saved automatically to the sequence file specified while entering the TEACH submode.

REPEAT

As the MCS is designed to receive and execute only single motion commands, EAC interpretes the programs in execution mode. Program interpretation always follows the following scheme:

1. Reading and interpretation of a motion command line

2. Look up in point definition table

3. Test whether the actual position lies within the specified tolerance limits - if it is outside there are two possibilities:

   a. If the position difference is less than the tolarance limits for automatic alignment, the EAC will send a move command to Motion Control System, commanding the manipulator to the desired start position.
   b. If an automatic alignment would not be allowed, the EAC changes to manual mode. The operator is now requested to command the manipulator to the start position. If this would have happened, the operator may return to automatic mode. Program execution now would be continued with alignment check.

4. Send the motion command to the MCS

5. Wait until MCS has completed the movement

EDIT
The EDIT submode is an extension of the REPEAT submode. In addition, it offers the possibility to modify existing program files. Modification of programs means:

- Modification of program description
- Insertion of new command lines
- Deletion of command lines
- Redefinition of points

To allow a quick editing, it is possible to skip a number of commands.

3.6.4.5 Software and Hardware Platform

There has been choosen a VAX workstation as EAC with VMS operating system and programming language C on it. Multitasking capabilities allow a well structured segmentation of application software.

For communication with other system components there are different Ethernet adapters in use, which support different network protocols. Communication with MCS follows the ISO 8073 standard. However, for connection to operator interface, graphic workstation for kinematic simulation and remote workstation there are ethernet links with TCP/IP Protocol or a serial RS232 link available. To ensure that there is no overload on the ethernet connection between MCS and EAC, there is installed an own net segment, which is exclusively used by these subsystems.

3.6.5 Operator Interface

In the normal operation mode the operator interface is represented by the remote workstation as the generic interface. As operator interface during test phase there is a backup system available, the socalled handbox. It consists of an AT-compatible handbox with two 3D-joysticks which is connected to the EAC via the serial link.

3.6.6 Automation Possibilities

With exception of the repetition of motion sequences, at the present there are no higher automation functions implemented. As position measurement is only based on resolver values, torsion and deflection of the ABS will cause positioning errors. To compensate these positioning errors, it is planned to simulate the system elasticity and correct deviations caused by elastic deformation. The simulation will happen offline on the KISMET-workstation 3.3, "KISMET: Spatial Simulation for ERWS" on page 60 Based on the simulation, the positioning error may be corrected during offline programming.

The control system has been kept modular. Thus, it will be possible to automate subtasks step by step. As an example can be seen the KfK-Teleman project. Within this project there have been added automatic features based on an image processing system, using the standard interface of the motion control system.

3.6.7 Operation Security

To provide a safe operation within the torus, there are different mechanisms integrated into the control system concept. There can be seen three levels, referring to the hardwareplatform:

* Automatically working mechanisms on MCS
* Automatically working mechanisms on EAC
* Operator interactive mechanisms
* Automatic collision control via remote workstation

3.6.7.1 Automatically Working Mechanisms on MCS

The MCS monitoring functions combined with the error handler provides a safe operation of motion control. They are implemented in own independent tasks, which have a high task priority. This system ensures, that monitoring will be processed even if other tasks with lower priority would hang. Monitoring functions check the values of axis travel software limits, axis acceleration, deceleration, velocity, unexpected motions during standstill and unallowed position differences. These parameters with exception of travel limits are handled as system parameters, which can not be handled by the operator. The travel limits however are operator setable, i.e. setable by an operator command. Thus it is possible to set these limits dynamically, with respect to the actual position of the manipulator to reduce the risc of a collision. In addition there are checked all digital inputs within the error handler. As described in a previous section, the error reaction can be configured as an emergency stop or gentle stop.

3.6.7.2 Automatically Working Mechanisms on EAC

In addition, there are some features integrated in EAC to provide safe operation of the control system in the automatic mode, which is processed on the EAC. As the most important feature can be seen the alignment check. Before giving a move command to the MCS the EAC checks, whether the actual position is identically with that one, when the program was teached. The alignment is then processed automatically or by operator interaction in the case of bigger deviations. This ensures, that even if there would be skipped commands, the manipulator will follow the desired path.

3.6.7.3 Operator Interactive Mechanisms

Beside the automatic working mechanisms the operator is able to stop a motion at each time. He can do this via command interface, giving in a HALT or a STOP command. These commands will be processed on the MCS and cause a gentle stop or an emmergency stop. If this would happen during program execution, the program will be aborted and can be continued, using the skip option with program starting. As a backup to the security function, based on the Control System, there exists an emergency stop system. Activating this system via an emergency stop button, the operator can switch of power supply of brakes and servo amplifiers.

3.6.7.4 Automatic Collision Control via the Remote Workstation

The remote workstation, especially the KISMET simulation, is the plate form which contains an exact model of the environment and can be used also for collision control. Detecting an anallowed approach to the reactor walls or installations, it will send a STOP or HALT command to the Control System.

3.6.8 Emergency Stop System

To provide a safe operation of the manipulator, i.e. to protect persons and the testsite from damage, an emergency stop system is installed which is independent of the control system. Within this system, there is realized a hierarchical power supply, forcing a fixed power on sequence to put the machine to operation. Also there is an emergency stop group integrated, which will switch off power of the servo controllers and motorbrakes. The emergency stop group will be activated by:

* Emergency stop buttons which will be placed in the control rooms and around the test area
* Dead-man's buttons
* Two limit switches at each joint
* Light barriers

Particularly, the limit switches form a backup system of the monitoring functions, which are integrated in MCS. They will protect the manipulator in that case, when the reaction on axis travel software limits would fail. Light barriers however have an additional function. They will prevent the manipulator from moving outside the testarea. This may be caused by a system fault of Control System or if the operator erroneously would command the manipulator outside. Also it will ensure, that the manipulator will stop, if anyone would enter the test area.

In addition to these automatically working security systems, there are emergency stop buttons placed around the testsite. So anyone will be able to stop the machine, if he would recognize a dangerous situation. Dead man's switches are foreseen for persons, who are authorized to stay in the testarea, while the manipulator is moving. They are forced to keep the button pressed, otherwise the machine would stop. When the dead man's button is token out of its mount, manipulator velocity will be reduced by a specified factor. This should be an exception but not the normal operation mode.

Also there are some additional features to increase security of operating by reduction of velocities.

* The velocity of the articulated boom will be decreased, if the EEPU has left its transport position.
* The velocity of any axis will be reduced, if it is controlled by the joystick box.

These measures will help to avoid collisions with the torus. In the first case, when EEPU came out, it would be very narrow in the upper parts of the torus. Therefore, there are only small motions possible, which might be also performed with reduced velocity without wasting time. In the second case however the operator exceptionally would control the manipulator by joysticks. Since he would not be used to do this often and because the manipulator's kinematics is rather complex, reduced velocity will offer him more time for reaction.

3.6.9 Qualification Tests

First qualification tests have been carried out. through. The first tests have were related to accertain the repetition accuracy. While measuring of absolute position accuracy shows systematic faults, the repetition accuracy gives information about stochastic error mechanisms. Systematic errors may be caused by:

* Torsion and deflection of the manipulator

- Nonlinear characteristic line of position measurement, caused by an eccentric attachment of joint resolvers
- Inaccurate adjustage of position measurement system, i.e. scaling factors and offset values
- Incorrect setting of Denavit-Hartenberg parameters for pathplanning

Stochastic errors may be caused by:

- Noise (for example quantization noise)
- Friction

As result,the specified repetition accuracy of ± 5 mm has been confirmed. The investigation of reasons for this deviation showed, that it has been nearly exclusively caused by axis positioning error. If the measured axis positions were identical, the target points on the graph paper could not be distinguished. So to improve repetition accuracy of positioning, there must be improved the positioning accuracy of each axis. The axis positioning error of the 4 main joints of ABT has been accertained to ± 4 resolver digits. Further optimization of control parameters might bring a slightly better repetition accuracy. Integration of motor resolvers, which have a resolution of about 26 times of the joint resolvers (transformed to the gear output side) would promise a repetition accuracy of about ± 1 joint resolver digits.

3.7 Drive Unit Control

The drive control system represents the lowest level of the EDITH Control System. Because the motors of each drive unit are redundant and are working in a master/slave mode the control system consists of master servo controller, slave servo controller and an electronic gear system. A system overview is given in Figure 62 on page 105, which shows the main devices of the drive system of one yaw joint with one master and one slave motor.

The servo controller includes the power amplifiers, the current controller, the electronic commutation, the speed controller as well as safety functions.

Standard servo controllers for driving the AC-servo motors were modified with some additional options specified by KfK. Generally, the interface between MCS and servo controller consists of an analog input for setpoint velocity or setpoint current. By switching an digital input of the servo controller, the control mode can be changed from speed control to current control.

The safety function of the servo controller consists of monitoring directly the motor winding temperature and the controller heatsink temperature and limiting the peak torque available so as to prevent thermal overload. The direct temperature monitoring and the method of processing used, ensure optimal torque availability from the system for all ambient temperatures and dynamic load conditions.

Beginning at a winding temperature of $+120°C$, the available motor current will be decreased proportional to zero, which corresponds to a temperature of $+150°C$ (at high temperature motors the corresponding temperature values are $+200°C$ and $+250°C$, respectively).

Two digital inputs of the servo controller will be connnected with the corresponding limit switches to prevent the drive system from destruction by cutting off the direction of rotation until the limit position will be leaved.

The protection logic of the servo controller monitors the system's critical functions and safely shuts down the system in case of error. All fault conditions are signalled on the front panel.

Both master and slave motors are speed controlled. To prevent a mutual influence of both control loops due to the mechanical coupling, the motion of the motors of each link joint have to be synchronised. In case of EDITH the synchronous running will be realized by means of a digitally controlled electronic gear.

For this, the rotor position of the master motor (the encoder simulation module, which is part of the servo controller processes the digital absolute rotor position information to provide an incremental encoder type signal) will be used to control the position of the slave motor in a closed loop. The current slave position will be generated by the corresponding servo controller as well. The position control of the slave motor will be the main task of the electronic gear system. The function of the drive control system is shown in Figure 63 on page 105.

A second task of this device will be the electric preloading of the mechanical part of drive system in order to compensate mechanical backlash. This will be realized by varying the rotor position of the slave motor up to a determined twist angle between master and slave motor. The corresponding twist angle will be given by the MCS as a preload angle setpoint. The mechanical effects of electric preloading are shown in Figure 64 on page 106. Near the zero position of the drive unit the backlash will be totally compensated but the stiffness of the drive system will be decreased to the stiffness of a single gear.

4.0 AMBOSS Dynamic Simulation

In addition to the engineering design and structural analysis a dynamics simulation of the boom is realized. Main topics of the simulation are the evaluation of IVHU performance, selection and optimization of the actuator design and of the control algorithms.

In contrast to existing remote handling systems at JET and TFTR or common robotic tasks, the NET/ITER In-Vessel Handling Unit is characterized by high payload, large dimensions and narrow environment. These conditions force a design of the transporter system which has extremely high inertia and rather low gear stiffness. The influence of the gearbox and its non-linear effects on the dynamic behaviour increases as described in 4.1, "Influence of Mechanical Effects on Dynamics and Control" on page 107 the gear oscillation on the control circuit. The structural deformation of the system causes significant coupling effects between vertical loads at the tip of the boom and the actuators. Therefore, a reliable dynamics model of the transporter system must comprise the full three-dimensional non-linear multybody dynamics, the structural deflection of the boom and a precise model of the actuators and the control system including non-linearities such as backlash and friction. 4.1.1, "Physical Effects and their Representation in the Dynamics Model" on page 108 gives a detailed overview about the AMBOSS model and implementation. The following chapters are devoted to verification, control optimization and performance studies.

- Nonlinear characteristic line of position measurement, caused by an eccentric attachment of joint resolvers
- Inaccurate adjustage of position measurement system, i.e. scaling factors and offset values
- Incorrect setting of Denavit-Hartenberg parameters for pathplanning

Stochastic errors may be caused by:

- Noise (for example quantization noise)
- Friction

As result,the specified repetition accuracy of ± 5 mm has been confirmed. The investigation of reasons for this deviation showed, that it has been nearly exclusively caused by axis positioning error. If the measured axis positions were identical, the target points on the graph paper could not be distinguished. So to improve repetition accuracy of positioning, there must be improved the positioning accuracy of each axis. The axis positioning error of the 4 main joints of ABT has been accertained to ± 4 resolver digits. Further optimization of control parameters might bring a slightly better repetition accuracy. Integration of motor resolvers, which have a resolution of about 26 times of the joint resolvers (transformed to the gear output side) would promise a repetition accuracy of about ± 1 joint resolver digits.

3.7 Drive Unit Control

The drive control system represents the lowest level of the EDITH Control System. Because the motors of each drive unit are redundant and are working in a master/slave mode the control system consists of master servo controller, slave servo controller and an electronic gear system. A system overview is given in Figure 62 on page 105, which shows the main devices of the drive system of one yaw joint with one master and one slave motor.

The servo controller includes the power amplifiers, the current controller, the electronic commutation, the speed controller as well as safety functions.

Standard servo controllers for driving the AC-servo motors were modified with some additional options specified by KfK. Generally, the interface between MCS and servo controller consists of an analog input for setpoint velocity or setpoint current. By switching an digital input of the servo controller, the control mode can be changed from speed control to current control.

The safety function of the servo controller consists of monitoring directly the motor winding temperature and the controller heatsink temperature and limiting the peak torque available so as to prevent thermal overload. The direct temperature monitoring and the method of processing used, ensure optimal torque availability from the system for all ambient temperatures and dynamic load conditions.

Beginning at a winding temperature of $+120°C$, the available motor current will be decreased proportional to zero, which corresponds to a temperature of $+150°C$ (at high temperature motors the corresponding temperature values are $+200°C$ and $+250°C$, respectively).

Two digital inputs of the servo controller will be connnected with the corresponding limit switches to prevent the drive system from destruction by cutting off the direction of rotation until the limit position will be leaved.

The protection logic of the servo controller monitors the system's critical functions and safely shuts down the system in case of error. All fault conditions are signalled on the front panel.

Both master and slave motors are speed controlled. To prevent a mutual influence of both control loops due to the mechanical coupling, the motion of the motors of each link joint have to be synchronised. In case of EDITH the synchronous running will be realized by means of a digitally controlled electronic gear.

For this, the rotor position of the master motor (the encoder simulation module, which is part of the servo controller processes the digital absolute rotor position information to provide an incremental encoder type signal) will be used to control the position of the slave motor in a closed loop. The current slave position will be generated by the corresponding servo controller as well. The position control of the slave motor will be the main task of the electronic gear system. The function of the drive control system is shown in Figure 63 on page 105.

A second task of this device will be the electric preloading of the mechanical part of drive system in order to compensate mechanical backlash. This will be realized by varying the rotor position of the slave motor up to a determined twist angle between master and slave motor. The corresponding twist angle will be given by the MCS as a preload angle setpoint. The mechanical effects of electric preloading are shown in Figure 64 on page 106. Near the zero position of the drive unit the backlash will be totally compensated but the stiffness of the drive system will be decreased to the stiffness of a single gear.

4.0 AMBOSS Dynamic Simulation

In addition to the engineering design and structural analysis a dynamics simulation of the boom is realized. Main topics of the simulation are the evaluation of IVHU performance, selection and optimization of the actuator design and of the control algorithms.

In contrast to existing remote handling systems at JET and TFTR or common robotic tasks, the NET/ITER In-Vessel Handling Unit is characterized by high payload, large dimensions and narrow environment. These conditions force a design of the transporter system which has extremely high inertia and rather low gear stiffness. The influence of the gearbox and its non-linear effects on the dynamic behaviour increases as described in 4.1, "Influence of Mechanical Effects on Dynamics and Control" on page 107 the gear oscillation on the control circuit. The structural deformation of the system causes significant coupling effects between vertical loads at the tip of the boom and the actuators. Therefore, a reliable dynamics model of the transporter system must comprise the full three-dimensional non-linear multybody dynamics, the structural deflection of the boom and a precise model of the actuators and the control system including non-linearities such as backlash and friction. 4.1.1, "Physical Effects and their Representation in the Dynamics Model" on page 108 gives a detailed overview about the AMBOSS model and implementation. The following chapters are devoted to verification, control optimization and performance studies.

Figure 62. EDITH Drive Control Devices of One Link Joint
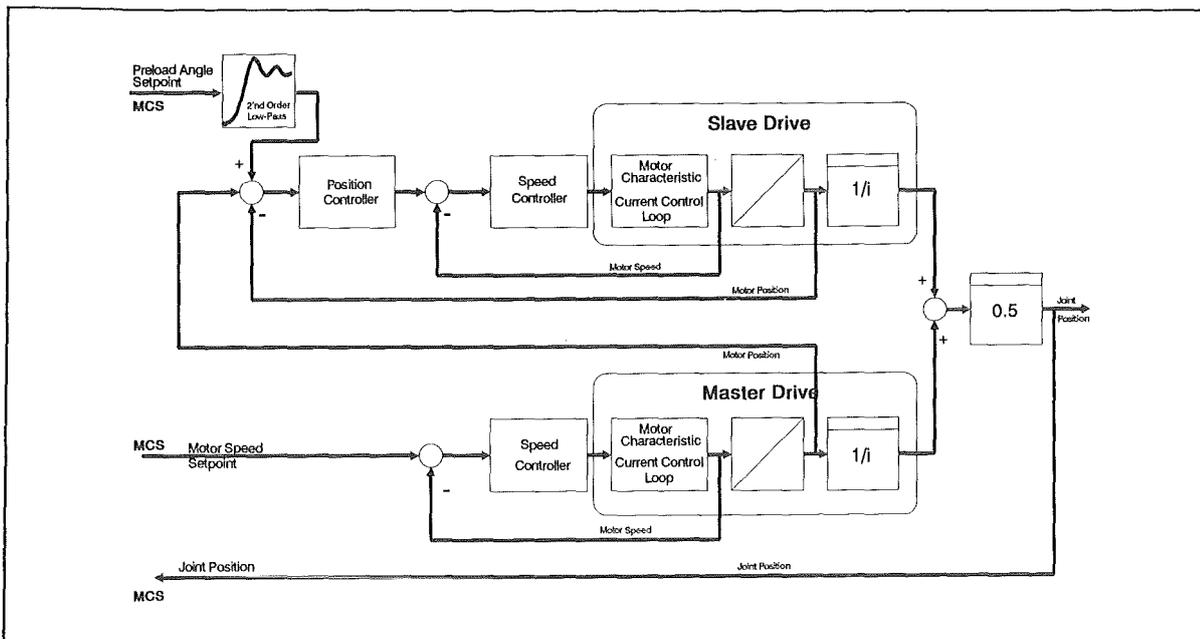
Figure 63. EDITH Drive Control of One Link Joint
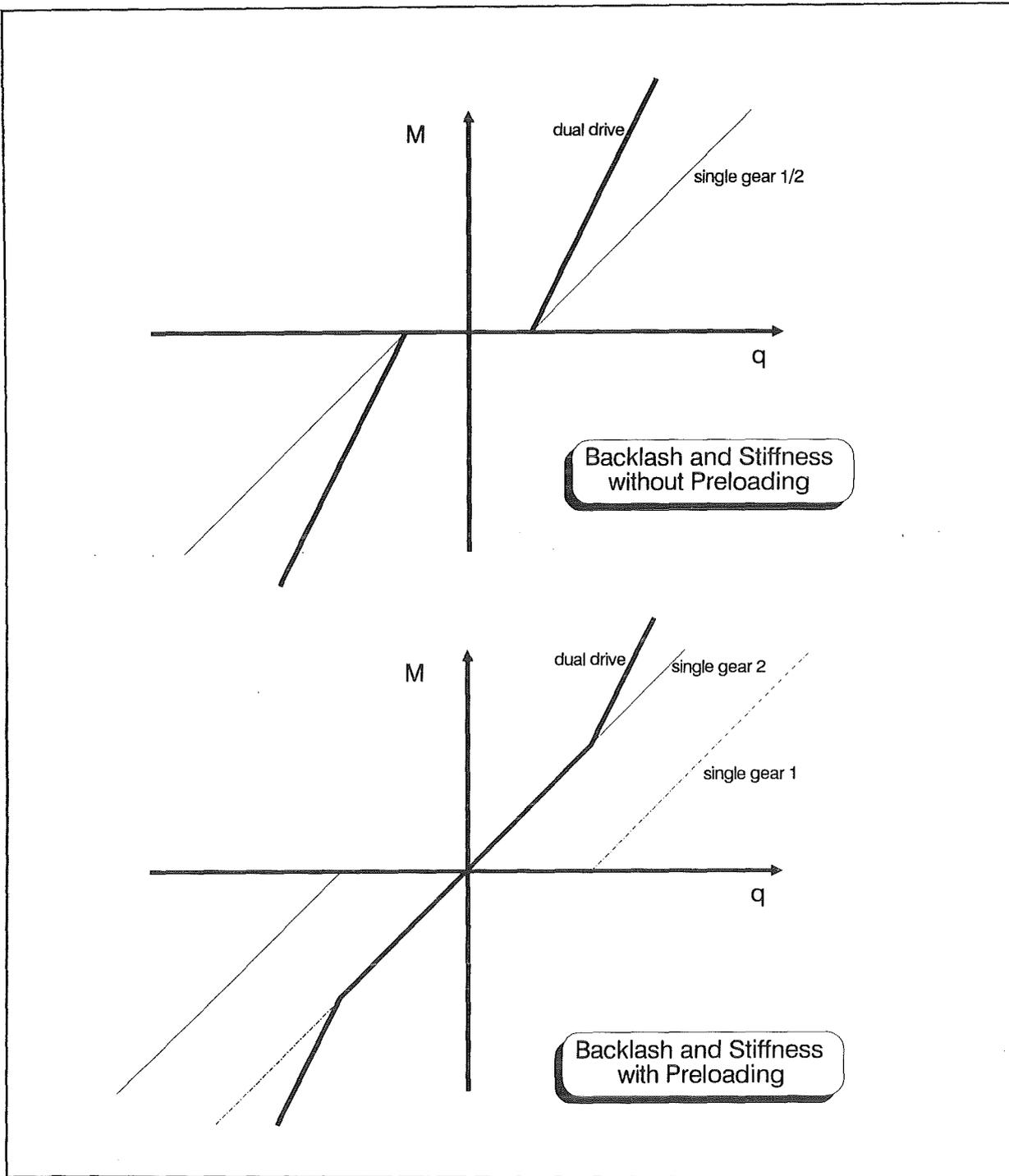
Figure 64. Effects of Preloading on Drive Systems. M symbolizes the joint torque and q the twisting angle of the motor.
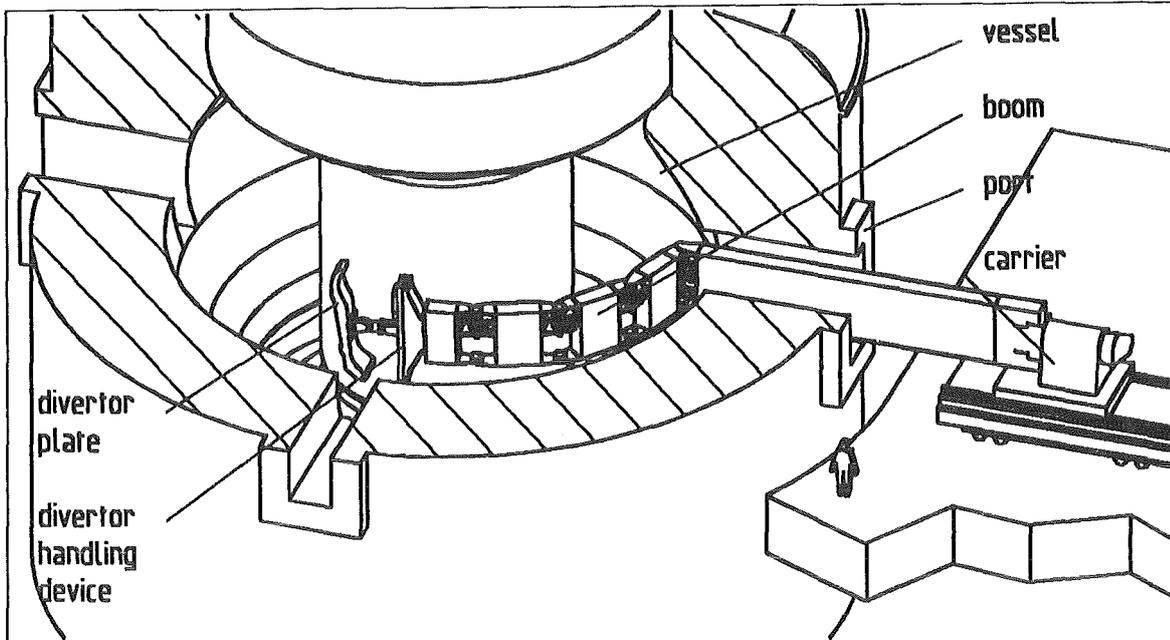
Figure 65. In-Vessel Handling Unit in NET/ITER Torus Environment

4.1 Influence of Mechanical Effects on Dynamics and Control

The motion of the IVHU is generally described by the equations of motion of the links and the actuator elements. Joint constraints are coupling the equations. Therefore, the system becomes a highly non-linear differential equation system especially when super-imposed rotation occurrs. The following parameters have essential effects on the solution and control layout:

1. Mass and inertia of the links
2. Length of the arms on which mass and inertia forces act on the joint
3. The proportion of gear stiffnes and link inertia which determines the natural frequencies of the system
4. Desired velocity and acceleration

A comparison of the EDITH device with a common industrial robot points out the following main differences:

1. The in-vessel transporter has heavy and long links. Both enlarges the acceleration coupling between the joints.
2. High velocitiy of an industrial robot induces relatively high centrifugal and coriolis forces. On the EDITH joints they are small.
3. Common actuator layout keeps the natural frequencies of the gear oscillation in a narrow range. This is achieved choosing a motor inertia and a transmission ratio that keeps similar kinetic energy at the motor side and in the corresponding link motion. In case of the in-vessel transporter a conventional layout will cause extremly heavy actuators in need of a high motor inertia or transmission ratio. Therefore, compared with an industrial robot a relatively low gear stiffness and a wide range of natural frequencies must be accepted for the in-vessel transporter. These effects increase the influence of the gear oscillation on the control circuit.

The mentioned phenomenons require a detailed control analysis and a dynamics model including the gear box oscillations and the acceleration coupling of the joints.

4_1.1 Physical Effects and their Representation in the Dynamics Model

4_1.1.1 Elastic Links

The behaviour of elastic bodies is generally described by Hook's equation

$$\underline{\sigma} = \underline{E}\,\varepsilon \tag{18}$$

The tension matrix $\underline{\sigma}$ which is notated as vector contains the normal tensions in its first three elements and the shear tensions in the later elements. In the same way ε contains the relative displacements.

The elastic links of the IVHU are modelled using two parts connected by a beam or six dimensional spring element. More precise numeric solutions are not necessary because higher modes of the structural oscillation are not excited by the actuators or loads at the tip of the boom (see 4.1.3, "Verification of the Structural Model" on page 125). The next sections discuss two possible approaches for the structural model.

Beam with Constant Cross-Section

Within a static beam model the load-deformation dependency between two ends of a massless beam is described. Mass properties are specified for the bodies at both ends of the beam element. Translational (d) and angular (α) displacement of the cantilevers free end are solution of the hook's equation. Integrating this equation (18) over the constant cross section area A and the length of the beam l the result can be written as stiffness matrix $\underline{K}_B$. It contains the loads for a unit displacement and describes the linear elastic relationship between the load vector $\underline{f}$ and the deformation vector $\underline{d}$.

$$\underline{f} = \underline{K}_B\,\underline{d}$$

$$
\begin{bmatrix} F_x \\ F_y \\ F_z \\ M_X \\ M_Y \\ M_Z \end{bmatrix}
=
\begin{bmatrix}
\dfrac{EA}{\ell} & 0 & 0 & 0 & 0 & 0 \\[2mm]
0 & \dfrac{12EI_{zz}}{\ell^3} & 0 & 0 & 0 & -\dfrac{6EI_{zz}}{\ell^2} \\[2mm]
0 & 0 & \dfrac{12EI_{yy}}{\ell^3} & 0 & \dfrac{6EI_{yy}}{\ell^2} & 0 \\[2mm]
0 & 0 & 0 & \dfrac{GI_{xx}}{\ell} & 0 & 0 \\[2mm]
0 & 0 & \dfrac{6EI_{yy}}{\ell^2} & 0 & \dfrac{4EI_{yy}}{\ell} & 0 \\[2mm]
0 & -\dfrac{6EI_{zz}}{\ell^2} & 0 & 0 & 0 & \dfrac{4EI_{zz}}{\ell}
\end{bmatrix}
\begin{bmatrix} d_x \\ d_y \\ d_z \\ \alpha_x \\ \alpha_y \\ \alpha_z \end{bmatrix}
\tag{19}
$$

The planar moments of inertia I and the cross section area A depend on the geometric shape. The modulus of elasticity E is a material constant. The described beam approach has an exact solution for the static case. In a dynamics simulation it is an approximate solution because one beam element only covers the basic mode for each degree of freedom.

For a good approximation the beam approach is only valid for long beams with realtively small cross section. This requirement does not fit with the IVHU geometry that has link boxes with a height of 1.8m and lenght of 1.9m. To improve the capabilities of the model it is detailed using a general linear elastic approach.
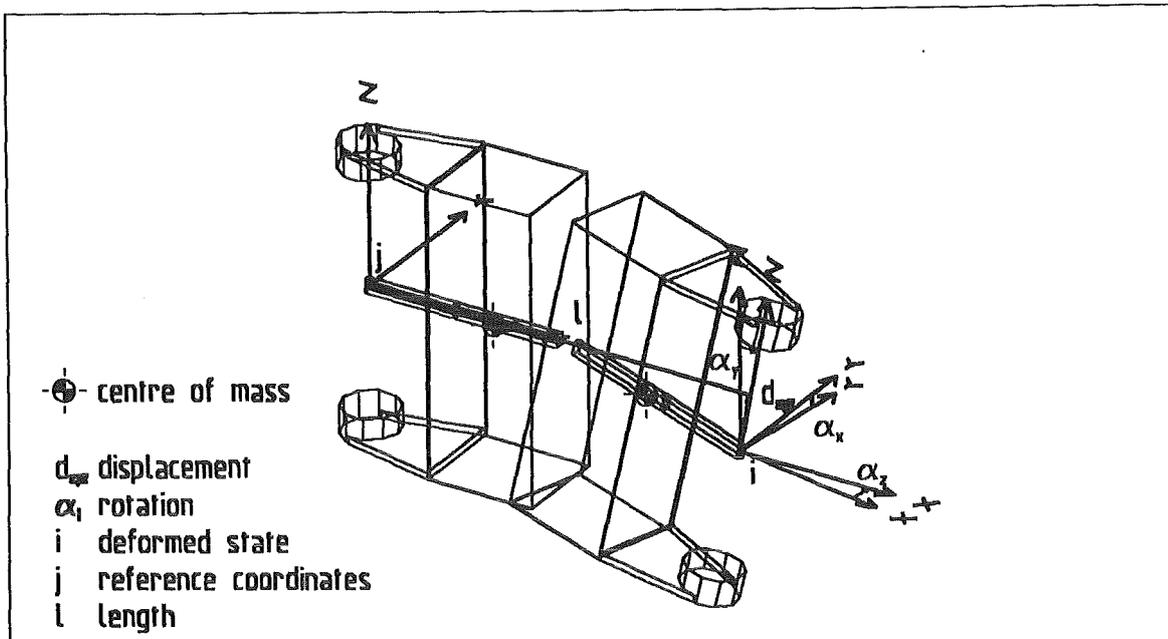
Figure 66. Parts and Lokal Coordinates of the Elastic Link Model

General Linear-Elastic Model

For the solution of the load-deformation dependency in one link the following linear equation system is set up.

$$\underline{f} = \underline{K}\,\underline{d} \qquad\qquad (20)$$

while $\underline{K}$ is the matrix of the stiffness parameters. Stating that the system should be conservative, the stiffness matrix $\underline{K}$ must be positive semi definite and symmetric. Compared with the beam approach the linear elastic solution has two significant advantages:

1. The formal analogon with the Finite-Element-Method (FEM) allows to transfer results from static linear-elastic FEM calculations to the dynamics model.

2. The general equation of the load-deformation dependency includes the capability to describe phanomenons not covered by the beam approach. Concerning the IVHU-links the model can cover the coupling between torsional load and bending displacement. This is caused by the extremly reduced weight of the lower joint plates which have to carry only low bending stress. Applying a torsional load on the boom, upper and lower joint plates are affected by equal paired forces but the lower plates have larger displacements due to their small cross section. The second effect not covered by the beam model is the shell shape of the link boxes.

Sitffness matrices of the links are calculated in a static Finite-Element analysis. The model which has been set up during the structural layout of the boom applies the load in the main axis of the cross section. Load transfer to the joint bearings is simulated by a rigid imaginary axis. Each degree of freedom is solved for unit loads on the link. The results are stored in a deformation matrix /10/. Calculating the inverse the desired stiffness matrix $\underline{K}$ is derived.

The linear elastic approach exactly models the static behaviour of the more precise Finite-Element model. Respectively dynamic deformation effects only the lower modes of structural oscillation are resolved. Higher mode oscillation can only be approximately solved or is neglected by this model, because of the limited degrees of freedom of the

model. Verification experiments described in 4.1.3, "Verification of the Structural Model" on page 125 showed afterwords that this simplification is valid in the complete dynamics model of the IVHU.

4.1.1.2 Motor Model and General Joint Controller

The main properties of an electric actuator are covered by the differential equation for the motor shaft's motion. All friction effects of the motor itself are summarized in the damping v_M. With the torque constant K_T, the motor shaft's inertia J_M and the output torque $\frac{1}{i} M_G$ the differential equation is

$$J_M \, \dot{\omega}_M = K_T \, i_M - v_M \, \omega_M - \frac{1}{i} M_G$$
$$\dot{\alpha}_M = \omega_M$$

(21)

This equation of motion is formulated in the input language of the multi-body dynamics program. While motion and inertia of the motor shaft are performed by the analysis software motor torque, damping and applied load are introduced via applied forces. The first two effects act between motor shaft and motor housing. They are combined in the motor torque M_M

$$M_M = K_T \, i_M - v_M \, \omega_M$$

(22)

The motor current i_M is derived from the differential equation of the coil

$$\frac{di_M}{dt} = -\frac{R}{L} \, i_M + \frac{1}{L} \, u_M$$

(23)

with the voltage u_M, the inductivity L and the resistance R.

This equation is the plant model of the current control loop. Coil model and current control are introduced as proportional integrating elements between joint controller output and motor torque. The delay time T_M and the gain K_M are evaluated from the current controller model.

Die voltage u_M at the coil is calculated with /28/ (page 1194).

$$u_M = R \, i_M + L \, \frac{di_M}{dt} \, .$$

(24)

Stating an exact controlled voltage the transfer function of the current control loop is

$$G_S(s) = \frac{I_M(s)}{U_M(s)} = \frac{\frac{1}{R}}{1 + \frac{L}{R} s}$$

(25)

The term

$$T_E = \frac{L}{R}$$

(26)

is the delay time of the plant model. With a proportional feed back the closed current control loop as shown in Figure 68 on page 112 has first order delay characteristics.

motor shaft
= gear input

motor torque
- motor friction

load transfer
in gearbox M_G

previous link

friction torque M_g

motor

motor angle α_M

gear output

next link
(tip direction)

joint angle q

Figure 67. Parts and Applied Torques in the Actuator

$$G_W(s) = \frac{I_W(s)}{I_M(s)} = \frac{K_M}{1 + T_M\, s}$$

$$\text{with } K_M = \frac{\dfrac{K_R}{R}}{1 + \dfrac{K_R}{R}}$$

$$\text{and } T_M = \frac{T_E}{1 + \dfrac{K_R}{R}}$$

(27)

The gain K_R has influence on the constant control error. Because the circuit has unlimited stability only the allowed voltage limits the response time of the current control loop. For simulation purposes the gain is estimated for the case that maximum motor current is reached at maximum voltage of the power supply.

$$K_R < \frac{u_{max}}{i_{max}}$$

(28)

For the selected brushless DC servo motors the delay time T_M results from the following estimation. Stating a proportional torque characteristics the motor needs at maximum torque of 60 Nm a current of 40 Ampere. The voltage is limited to 220 V by the power supply. The estimated controller gain is $K_R = 5$. Equation (27) gives the characteristic parameters for the electric part of motor model.

$$K_M = \frac{\dfrac{K_R}{R}}{1 + \dfrac{K_R}{R}} = 0.9$$

$$\text{and } T_M = \frac{T_E}{1 + \dfrac{K_R}{R}} = 10^{-3}\ \text{s}$$

(29)

Figure 68. Motor Model with Electric Delay

Compared with the influence of the motor's inertia the delay time T_M is quite small. Therefore, in most cases simulation can run with the simplified model ommitting the current control circuit of the servo motor. Its block diagram is shown in Figure 69 on page 113.

To allow an optimization of the control layout without continuous changes of the model topology a general linear feedback of the main sensor values has been implemented. The EDITH instrumentation allows to sense the position of the motor shaft α_M and the joint axis q and their velocities ω_M and $\dot{q}$.

The general feedback shown in Figure 70 on page 114 provides proportional gains for the four mentioned sensor values. An additional PI-block allows an integrating feedback. The set point and the actual value are compared at the input of the PI-block. An additional input channel provides the capabillity to superimpose external calculated feed-forward values. The general controller toolkit contains the integrator's differential equation with the connection of the input values

$$\dot{y} = K_V \dot{w} + K_W w - K_{WM} \omega_M - K_{AM} \alpha_M - K_{WJ} \dot{q} - K_{AJ} q \tag{30}$$

and the output equation to evaluate the motor current and the motor torque.

$$i_W = K_{RI} y + K_{RP} \dot{y} + K_{FFC} M_{FFC} \tag{31}$$

The further processing takes care of the above mentioned two motor mo dels

1. with current control circuit

$$\dot{i}_M = K_M i_W - \frac{K_M}{T_M} i_M \tag{32}$$

2. with out current control circuit

$$i_M = i_W \tag{33}$$

Figure 67. Parts and Applied Torques in the Actuator

$$G_W(s) = \frac{I_W(s)}{I_M(s)} = \frac{K_M}{1 + T_M s}$$

$$\text{with } K_M = \frac{\dfrac{K_R}{R}}{1 + \dfrac{K_R}{R}} \tag{27}$$

$$\text{and } T_M = \frac{T_E}{1 + \dfrac{K_R}{R}}$$

The gain K_R has influence on the constant control error. Because the circuit has unlimited stability only the allowed voltage limits the response time of the current control loop. For simulation purposes the gain is estimated for the case that maximum motor current is reached at maximum voltage of the power supply.

$$K_R < \frac{u_{max}}{i_{max}} \tag{28}$$

For the selected brushless DC servo motors the delay time T_M results from the following estimation. Stating a proportional torque characteristics the motor needs at maximum torque of 60 Nm a current of 40 Ampere. The voltage is limited to 220 V by the power supply. The estimated controller gain is $K_R = 5$. Equation (27) gives the characteristic parameters for the electric part of motor model.

$$K_M = \frac{\dfrac{K_R}{R}}{1 + \dfrac{K_R}{R}} = 0.9$$

$$\text{and } T_M = \frac{T_E}{1 + \dfrac{K_R}{R}} = 10^{-3} \text{ s} \tag{29}$$

Figure 68. Motor Model with Electric Delay

Compared with the influence of the motor's inertia the delay time T_M is quite small. Therefore, in most cases simulation can run with the simplified model ommitting the current control circuit of the servo motor. Its block diagram is shown in Figure 69 on page 113.

To allow an optimization of the control layout without continuous changes of the model topology a general linear feedback of the main sensor values has been implemented. The EDITH instrumentation allows to sense the position of the motor shaft α_M and the joint axis q and their velocities ω_M and $\dot{q}$.

The general feedback shown in Figure 70 on page 114 provides proportional gains for the four mentioned sensor values. An additional PI-block allows an integrating feed-back. The set point and the actual value are compared at the input of the PI-block. An additional input channel provides the capabillity to superimpose external calculated feed-forward values. The general controller toolkit contains the integrator's differential equation with the connection of the input values

$$\dot{y} = K_V \dot{w} + K_W w - K_{WM} \omega_M - K_{AM} \alpha_M - K_{WJ} \dot{q} - K_{AJ} q \tag{30}$$

and the output equation to evaluate the motor current and the motor torque.

$$i_W = K_{RI} y + K_{RP} \dot{y} + K_{FFC} M_{FFC} \tag{31}$$

The further processing takes care of the above mentioned two motor mo   dels

1. with current control circuit

$$\dot{i}_M = K_M i_W - \frac{K_M}{T_M} i_M \tag{32}$$

2. without current control circuit

$$i_M = i_W \tag{33}$$

Figure 69. Motor Model with Proportional Torque Characteristics

Finally Table 19 on page 113 lists the parameters of the controller toolkit.

K_T	torque constant		T_M	coil circuit delay time
K_M	coil circuit gain		K_R	current control gain
K_{RI}	integrator gain		K_{RP}	proportional gain
K_V	velocity feed-forward gain		K_W	set point input gain
K_{WM}	motor velocity gain		K_{AM}	motor position gain
K_{WJ}	joint velocity gain		K_{AJ}	joint position gain
K_{FFC}	feed-forward gain			

Table 19. Parameters of Motor Model and Controller Toolkit

Besides the system calls on actual and set point values the simulation code needs an integrator to model the coil differential equation and the integrating controller. Controllers tested and optimized in 4.1.4, "Layout and Optimization of the Joint Controller" on page 126 are coded using the presented method.

4.1.1.3 Special Treatment of Non-Linear Actuator Elements

Caused by the utilized solution scheme based on a predictor-corrector iteration and because of the high number of equations the non-linear actuator elements need special treatment to fit for the simulation task. The following example points out the sudden change of state caused by non-linear actuator elements:

1. The system is moving straight on. Nearly constant centrifugal and acceleration forces act on the actuators. Backlash and static friction remain in the bounds of their linear characteristics. The solution strategy performs relative large integration time steps.
2. A sudden load at the tip of the boom or a change of the set point causes a change of load or direction in an actuator. Along the non-linear area which is crossed when the load changes, a small integration step size is necessary to solve the high frequency exitation of the non-linear actuator elements.

To trap both cases in an efficient simulation the integration step size control and the non-linear actuator elements must work together.
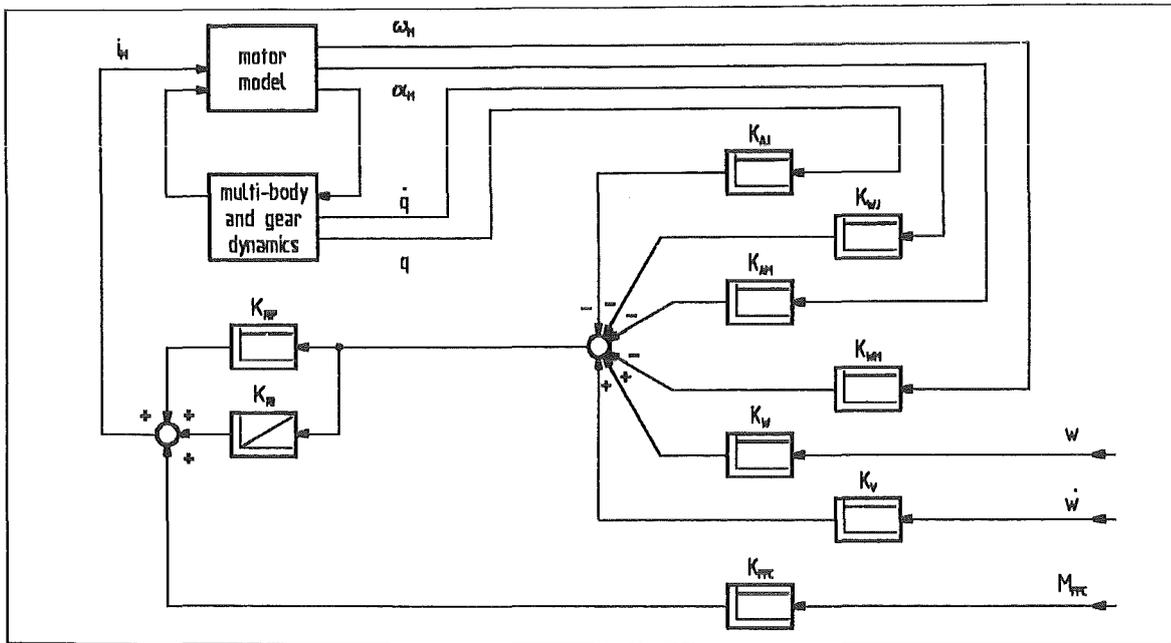
Figure 70. Controller Toolkit with Proportional and Integrating Feedback

When the utilized simulation program ADAMS steps forward to the next time step the start solution for the corrector iteration is estimated in the predictor step. This prediction is performed evaluating a Taylor polynom. The rest error controls the time increment. Evaluating a linar interpolated approach the error estimation contains second derivatives. This is the reason why the model has to provide continuously second derivatives for all force and torque expressions.

Against this requirement non-linear actuator elements have discontinuous first derivatives. Because simulations with fixed step size that do not need the error estimation consumed extremly high computation effort an adapted formulation of the non-linear actuator effects has to be developed.

4.1.1.4 Backlash of the Gears

The behaviour of an elastic gear box is described with a linear spring equation with the stiffness C and the damping D. The gear deformation is

$$\alpha_G = q - \frac{1}{i}\,\alpha_M \tag{34}$$

and its derivation, the deformation velocity

$$\omega_G = \dot{q} - \frac{1}{i}\,\omega_M \tag{35}$$

With these equations the resulting torque at the gear output shaft is

$$M_G = \begin{cases} -\,C\,(\alpha_G + \alpha_B) - D\,\omega_G & \text{für } \alpha_G < -\alpha_B \\ 0 & \text{für } -\alpha_B \le \alpha_G \le \alpha_B \\ -\,C\,(\alpha_G - \alpha_B) - D\omega_G & \text{für } \alpha_B < \alpha_G \end{cases} \tag{36}$$

Figure 71. Transition of Gear Spring Characteristics at the Backlash Stop

with the half backlash α_B. Evaluating the motor angle α_M the full cycles of the shaft have to be counted. The discontinuous spring characteristics and the temporarly switched off damping of equation (36) disturbe the predictor step.

To smooth the edge of the spring characteristics a forth order polynom is selected. Compared with the half backlash α_B the range of the polynom is very small. Taking a polynom with symmetric second derivative the first derivative of the spring characteristics is just the desired approximation for the switched damping in (36). Figure 71 shows the spring characteristcs and defines the above mentioned angles α_B and α_D, the stiffness C and the damping D. The parameters of the polynom approximation are evaluated with the boundary conditions requiring a smooth second derivative. The implemented approximation for the torque at the gear output shaft is

$$
M_G = \begin{cases}
- C\,(\alpha_G + \alpha_B) - D\,\omega_G & \text{for } \alpha_G \leq -\alpha_B - \alpha_D \\
- M_C - M_D & \text{for } -\alpha_B - \alpha_D < \alpha_G < -\alpha_B + \alpha_D \\
0 & \text{for } -\alpha_B + \alpha_D \leq \alpha_G \leq \alpha_B - \alpha_D \\
- M_C - M_D & \text{for } \alpha_B - \alpha_D < \alpha_G < \alpha_B + \alpha_D \\
- C(\alpha_G - \alpha_B) - D\omega_G & \text{for } \alpha_B + \alpha_D \leq \alpha_G
\end{cases}
$$

with

$$
M_C = \left[- \frac{1}{12}\left(\frac{|\,2\alpha_G - \alpha_B + \alpha_D\,|}{\alpha_D} \right)^4 + \text{SIGN}(\alpha_G)\,\frac{1}{3}\left(\frac{|\,2\alpha_G - \alpha_B + \alpha_D\,|}{\alpha_D} \right)^3 \right]
$$

$$
M_D = \frac{3}{4} D \left[- \frac{1}{3}\left(\frac{|\,2\alpha_G - \alpha_B + \alpha_D\,|}{\alpha_D} \right)^3 + \left(\frac{|\,2\alpha_G - \alpha_B + \alpha_D\,|}{\alpha_D} \right)^2 \right] \omega_G
$$

$$(37)$$

Selecting a small interpolation interval α_D the model error becomes neglegible small.

4.1.1.5 Friction Model

General friction effects are divided into viscous damping and Coulomb friction. The first one describes surfaces sliding on a liquid film for example an oiled bearing. Because the friction torque is proportional to the velocity with the damping v_M its major influence appears at high velocities. In the IVHU case this is at the motor shaft. The friction law is

$$M_{RM} = v_M \, \omega_M \tag{38}$$

The Coulomb or static frction appears at dry contact surfaces and has a load dependent friction law /28/ (page 129)

$$F_R = \mu \, F_N \tag{39}$$

The friction force F_R is proportional to the normal force F_N. Sliding friction and friction of rest have different parameters μ. The following paragraphs derive the equations for the gear and bearing friction of an IVHU actuator.

Gear Friction

Static friction in a gear box is generally described by its efficiency η_G

$$\eta_G = \frac{P_G - P_R}{P_G}$$
$$\Rightarrow P_R = (1 - \eta_G) P_G \tag{40}$$

Applying equation (39) on a rotational motion the friction law becomes

$$P_G = M_G \, \dot{q}$$
$$\text{and} \quad P_R = M_{RG} \, \dot{q}$$
$$\Rightarrow M_{RG} = (1 - \eta_G) M_G$$
$$= \mu_G \, M_G \tag{41}$$

while M_G is the transmitted torque (see (36)), $\dot{q}$ the joint velocity and μ_G the gear friction constant.

Bearing Friction

The rolling friction of the joint bearing mainly depends on the bearing load that is changing with the boom's position. The friction torque M_{RL} in one bearing is approximated referring to /28/ (page 424)

$$M_{RL} = F_N \mu_L \, \frac{d_L}{2} \tag{42}$$

with F_N radial bearing load, μ_L the friction parameter and d_L the bearing's nominal diameter.

The IVHU joints have paired bearings at the upper and the lower joint plates. The arm introducing the bending torque M_{JY} and the torsional torque M_{JX} is half the distance between the bearings. Taking two bearings the radial load F_N on each bearing is

$$F_N = |F_J| + \frac{1}{h_L} \sqrt{M_{Jx}^2 + M_{Jy}^2} \, , \tag{43}$$

with the resulting joint force $|F_J|$ and the axial distance h_L. Combining equation (43) and equation (42) the bearing friction torque is

$$M_{RL} = \mu_L \frac{d_L}{2} \left(|F_J| + \frac{1}{h_L} \sqrt{M_{Jx}^2 + M_{Jy}^2} \right) . \qquad (44)$$

Limitation of Static Friction Torque

The friction laws described above do not regard that friction must not apply energy to the system. Therefore, a limitation mechanism has to be added to the friction laws. On the other hand the simulation system requires friction characteristics with a continuous second derivative. The following implemented solution satisfies these two requirements.

$$M_R = -\text{SIGN}(\dot{q}) \, M_{Rmax} \left(1 - e^{-v_s \, |\dot{q}| \, \frac{M_0}{M_{Rmax}}} \right) \qquad (45)$$

with the joint velocity $\dot{q}$ and the maximum friction torque M_{Rmax} which is the result of the superimposed friction laws. The discontinuous characteristics of the signum function and of the absolut value in the exponent keep the balance. Therefore, the function still has the required continuous derivatives.

The selected solution provides the following advantages:

* At low $\dot{q}$ arguments the function is nearly independent of the maximum friction torque M_{Rmax}
* The existing exact derivatives are used to stabilize the solution

The disadvantage that the damping applied at low velocity can not stop motion becomes neglegible small when a high value v_s is selected. Figure 72 on page 118 shows a series of function plots evaluated for different maximum torques M_{Rmax}.

Friction of Rest

The change from friction of rest to sliding static friction is described with a similar exponential approximation. The supplement to the static friction μ_G when motion stops is the constant friction of rest torque M_{H0} and a load dependent term covering bearing influences.

$$M_H = M_{H0} + \mu_H \frac{d_L}{2} \left(|F_J| + \frac{1}{h_L} \sqrt{M_{Jx}^2 + M_{Jy}^2} \right) \qquad (46)$$

When motion starts this friction torque has to be removed. The approximation solution which has to take away the sign of the velocity input is set up with the basic function $f(x) = e^{-x^2}$. Introducing equation (46) the friction of rest component of M_{Rmax} is

$$M_{RH} = M_H \, e^{-\left(\dot{q} \, v_s \, \frac{1}{M_{HF}} \right)^2} . \qquad (47)$$

Figure 73 on page 119 shows example plots of this characteristic function.

4.1.1.6 Assembly of the Model for an Actuated Joint

The complete model of an IVHU main joint shown Figure 74 on page 120 is combined from the equations of the physical effects and the designed approximation terms. Two
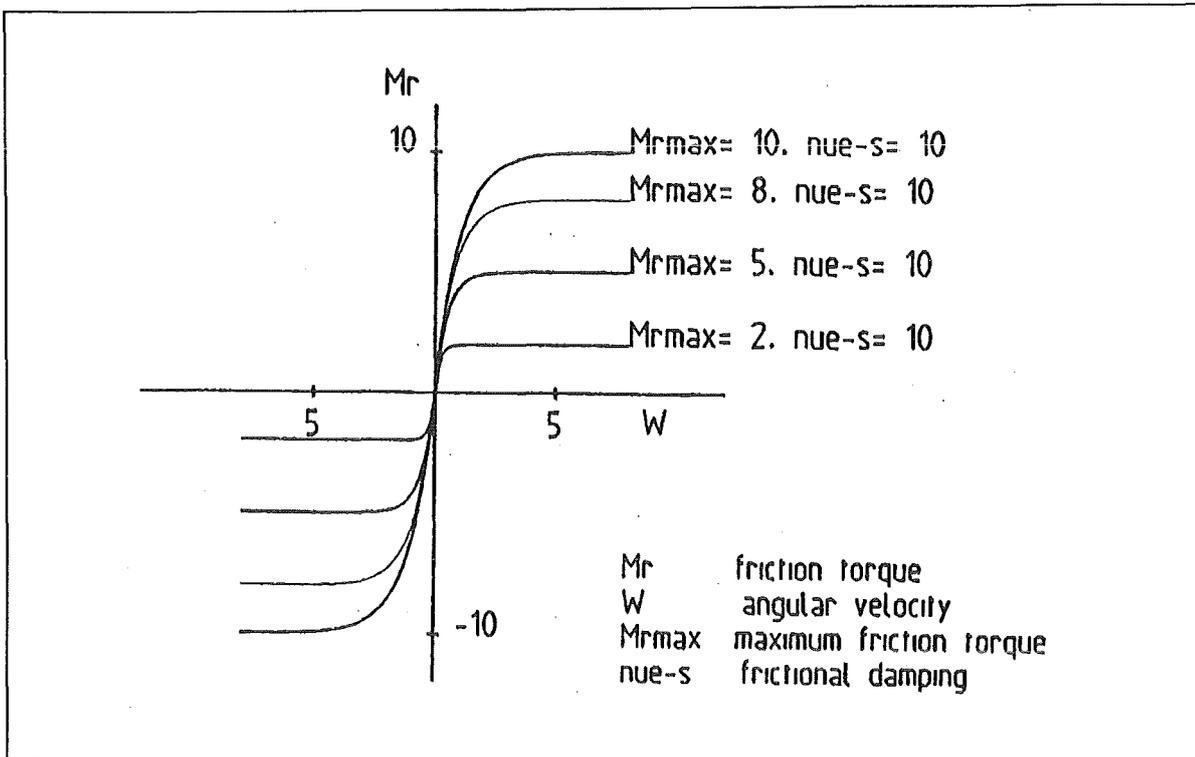
Figure 72. Function Limiting Static Friction Torque

main torque equations supply the coupling between the actuator model and multi-body dynamics.

1. The resulting axial torque on the motor shaft M_{Mres} acts between motor shaft and reacts on the previous link.

$$M_{Mres} = M_M - \frac{1}{i} M_G \tag{48}$$

In this equation the reaction torque of the gear input reacts on the actuator housing instead of the gear input. In the next equation this is compensated putting the whole gear output torque as reaction torque on the gear housing and omiting the gear input torque on the motor shaft.

2. The second torque M_A comprises all the applied torques on the joint which are static friction and the gear output torque.

$$M_A = M_G - M_R \tag{49}$$

While the gear output torque is well known from equation (37) the friction torque is superimposed of the equations (44), (46) and (45).

$$M_R = -\text{SIGN}(\dot{q}) \, M_{Rmax} \left(1 - e^{-v_s \, |\dot{q}| \, \frac{M_0}{M_{Rmax}}} \right) \tag{50}$$

with the maximum friction torque M_{Rmax}
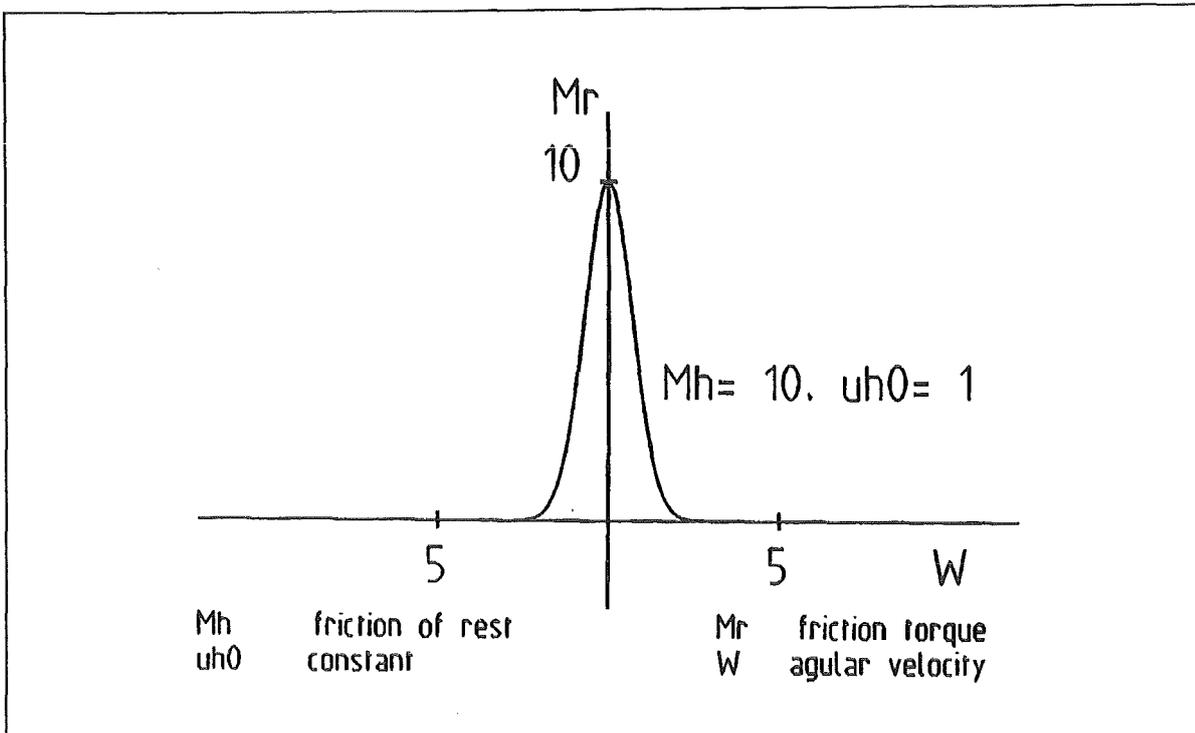
Figure 73. Function Modelling Friction of Rest

$$M_{Rmax} = \left(\mu_L + \mu_H\, e^{-(\dot{q}\, v_s\, M_{HF})^2} \right) \frac{d_L}{2} \left(|F_J| + \frac{1}{h_L} \sqrt{M_{Jx}^2 + M_{Jy}^2} \right)$$

$$+ \mu_G\, |\overline{M}_G|$$

$$+ M_{H0}\, e^{-(\dot{q}\, v_s\, M_{HF})^2} \tag{51}$$

$$+ M_{R0} ,$$

the damping v_s that stops motion via static friction and the torque unit M_0. The other symols were introduced in the equations (42), (43) and (47).

The block diagram Figure 75 on page 121 visualizes the force equation (50), which is covered in the block "static friction" of Figure 74 on page 120. Figure 76 on page 122 summarizes the static friction characteristics for varied bearing and gear output load. The friction torque (ordinate) is plotted versus the joint velocity.

4.1.2 Software Implementation

For the implementation of the above described model the commercial dynamics analysis package ADAMS is utilized. The following advantages are provided combined with the shortened development time:

- The mechanical system is formulated in a comfortable input language.
- Differential equations are generated automatically by the system.
- Some basic interfaces for graphic input and output are already existing.

For graphic input and output the CAD-system Bravo3 by Applicon-Schlumberger has been used. Own developed pre- and post- processor software has been implemented to manage the source model assembly and the result data processing. Figure 77 on page 123 gives an overview about the software components and the data flow between them.
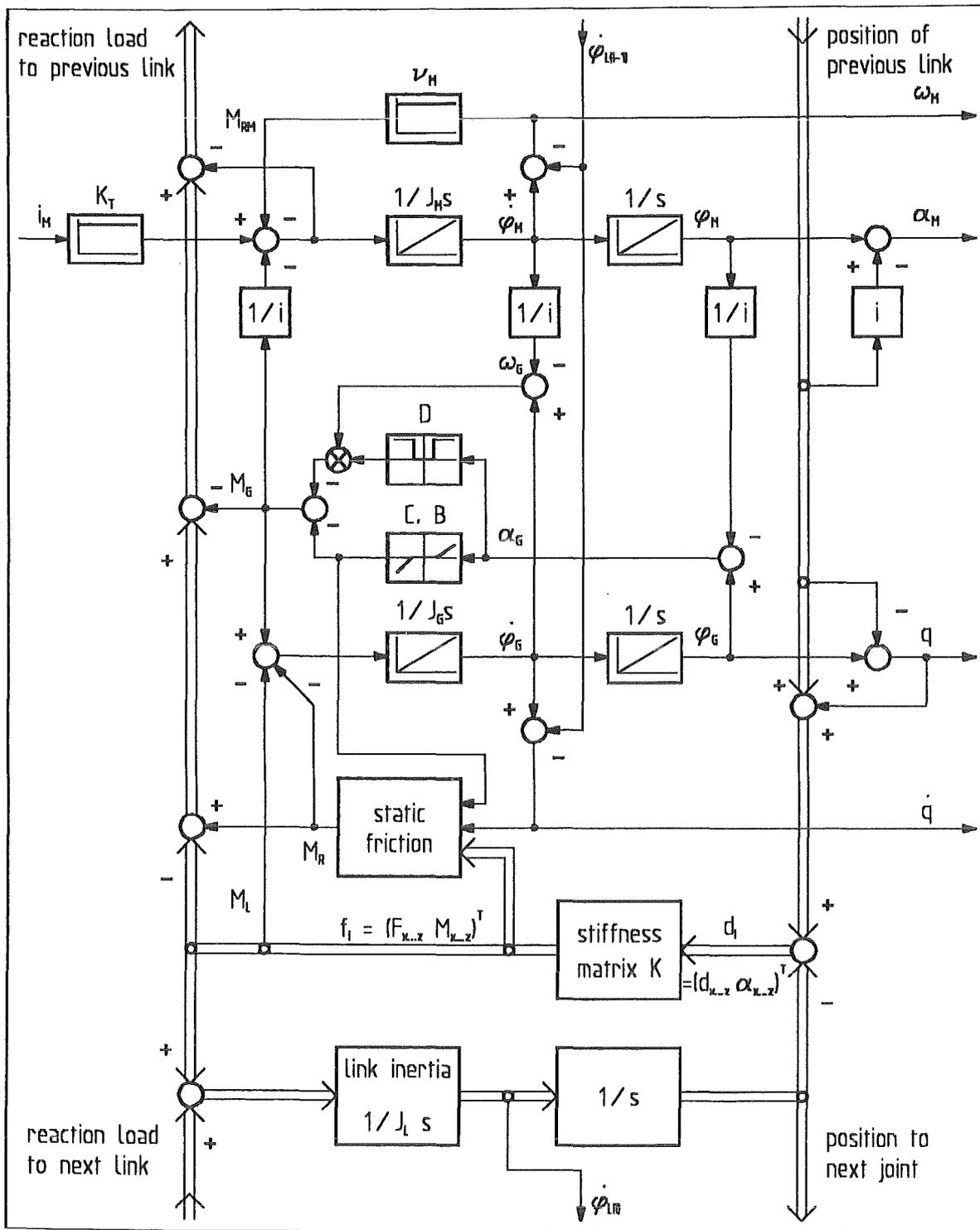
Figure 74. Block Diagram for a Joint with Complete Actuator Model.

Symbol definitions: i_M motor current, K_T torque constant, v_M motor damping, J_M inertia of motor shaft, M torque, i transmission ratio, D damping of gear deformation, E gear stiffness, B backlash, J_G inertia of gear output and half link, K stiffness matrix, d_i deformation vector, f_i load vector (in link), J_L inertia of half link and fixed parts of the next drive unit, φ absolute position, α relative position

To provide a save and consistent maintenance of the huge amount of input parameters the model is structured modular. The pre-processor assembles the model components and sets the parameters. The modules are divided into four sections indicating their different origin:

Figure 75. Block Diagram of the Friction Model

1. Significant positions of the boom are generated with ROBOT, a KfK software product embedded into the Bravo3 CAD-system. ROBOT has been developed to add kinematic properties to a CAD-model. It provides a kinematics simulation that allows a graphic generation of robot positions. The generated positions are written into the AMBOSS post-processor format. The post-processor processes the positions and generates a path with reasonable velocity and acceleration curves. The result is stored for simulation input.

2. Geometry, basic kinematics, and mass properties are edited in the Bravo3 CAD-system. The Bravo3 MECHANISM interface drops these data which is stored as pre-processor module.

3. The modules containing the actuator model, an optional simplified kinematics model of a joint, control algorithms or stiffness matrices are coded by manual editing. They are included into the simulation via conditional switches.

4. At least simulation control and model assembly is driven by a main module containing basic simulation commands and pre-processor macros.

A general command procedure provides the correct configuration of the model and the compilation of the user subroutines linked with the ADAMS program. After simulation the post-processor prepares the results for the presentation in MECHANISMS, ROBOT, KISMET /29//20/ or plotted curves. Optionally torque data for feed-forward control are generated.

The complete environment is called ADAMS Modelled BOom Simulation System (AMBOSS).

Figure 76. Static Friction Characteristics

program	task	author
VAX/VMS	operating system, command procedures	Digital Equipment Corporation
Bravo3	CAD-system	Applicon-Schlumberger
MECHANISMS	Interface to ADAMS inside Bravo3	Applicon-Schlumberger
MATRIXx	control analysis	Integrated Systems Incorporated
ROBOT	generate kinematic configurations	Dr. A. Ludwig KfK
ADAMS	dynamic solution	Mechanical Dynamics Inc.
PREP	pre-processor	J. Reim KfK
AMBOSS post-processor	process ADAMS results prepare path data	J. Reim KfK
PLI FORTRAN LINK	software developement	Digital Equipment Corporation

Table 20. AMBOSS Software Components

Table 20 summarizes the software components. The following description of the model synthesis introduces their use more detailed.
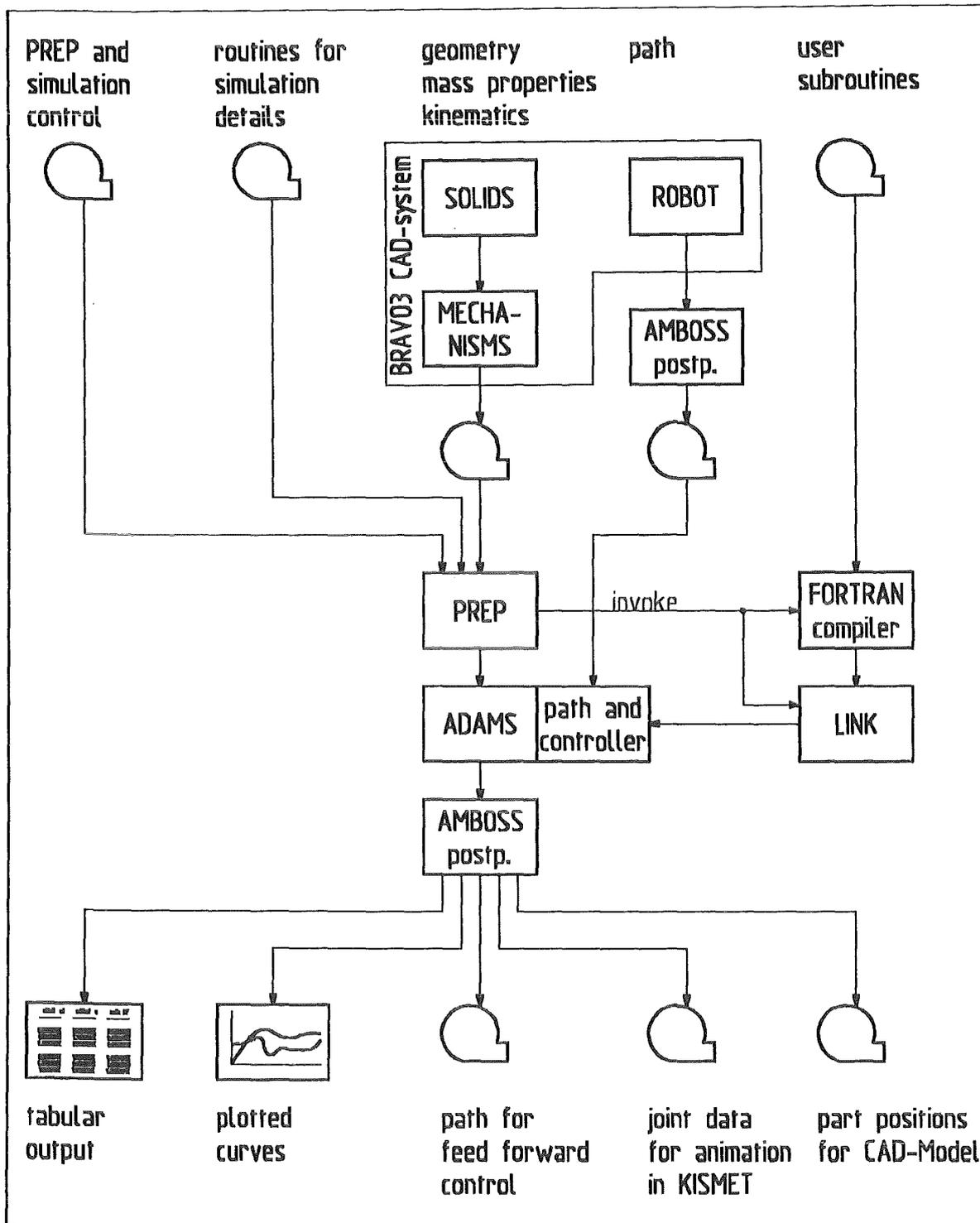
Figure 77. Data Flow and Software Architecture of the AMBOSS Simulation

4.1.2.1 Geometric Modelling of the IVHU in the CAD-System

The geometric model of the IVHU is based on the design data of the EDITH device. On the CAD-system a shape model is generated that contains besides the graphic representation information about material and density. Mass properties are evaluated automatically by the system. The solid model of each main link of the IVHU is divided into two equal sized parts. The elastic element modelling link compliance is positioned between them.

Legend

center of mass
(in simulation divided into two parts)
n.) joint identifier in simulation
Zn joint name in design
Bn link name

2.45 m   2.45 m   2.45 m

1.09 m   1.24 m   1.22 m   1.88 m

q_1

q_2 q_3 q_4 q_5

1.) translation Z0

2.) joint Z1   link B1 m= 2103 kg
3.) joint Z2   link B2 m= 1466 kg
4.) joint Z3   link B3 m= 1292 kg
5.) joint Z4   link B4+DHD+Last m= 3900 kg
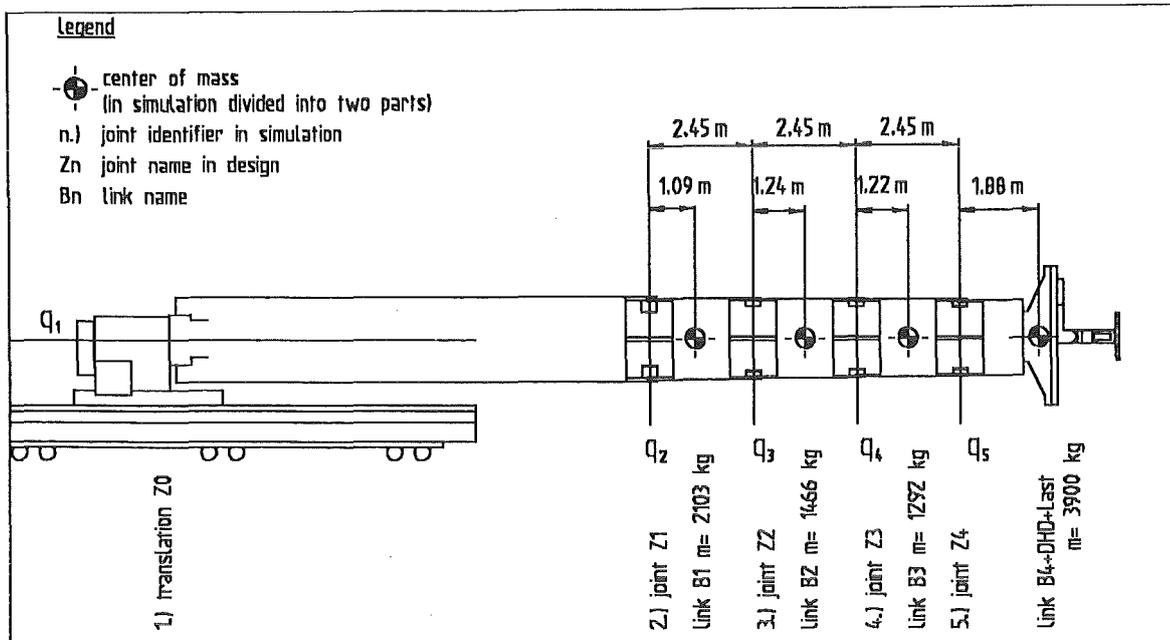
Figure 78. Dimensions and Mass Properties of the In-Vessel Handling Unit

To add kinematic properties the MECHANISMS application is utilized. The parts are assembled to one mechanism. Geometric properties like joint connection points, force reaction frames, or output locations are attached to the parts. The partly finished model dropped as AMBOSS module.

Figure 78 summarizes joint names and basic properties of the model.

4.1.2.2 Pre- and Post-Processing for the AMBOSS Simulation

Because model changes and parameter variations were not consistent controllable by hand the pre-processor has been integrated into the AMBOSS simulation. Having the modular structure mentioned in Figure 77 on page 123 whole modules or conditional parts are replaced keeping model consistency. The text oriented pre-processor selects the modules and replaces the symbolic parameters by their actual numeric value. The implemented heredity of parameters to the called modules provides a save method to keep changes consistent.

The pre-processor is invoked by a command procedure which selects the fitting user-subroutines and links them with the ADAMS-program.

The post-processor contains many specialized routines which were necessary during the working period. They perform the following subtasks:

- Store and manipulate ADAMS results in a compressed binary form.
- Prepare path data and robot positions.
- Evaluate simulation results with scaled dynamic displacements.
- Plot curves on HPGL-plotter or as Bravo3 CAD-geometry.
- Print tables or calculate minimum and maximum values.

A Command interpreter allows to drive the post-processor with macro files.

4.1.3 Verification of the Structural Model

Figure 79 on page 125 compares statics deformations from a statics simulation with those measured with the GMS (Geometry Measurement System of KfK) laser theodolite system at the EDITH device. For the dynamic case Table 21 on page 126 shows the natural frequencies. In the experiment the EDITH device is excited by a sudden lost of 200 kg load. Acceleration sensors on each link and at the end-flange observe motion. Two sensors checked that lateral oscillation is small enough to be neglected. Table 21 on page 126 shows the natural frequencies of the first and second vertical mode for several positions.

The simulation results are based on the stiffness data extracted from preliminary FEM (finite element method) calculations. Rising differences between experiments and simulation in bowed boom positions point out that the FEM calculations assumed a rather low torsional stiffness of the links.
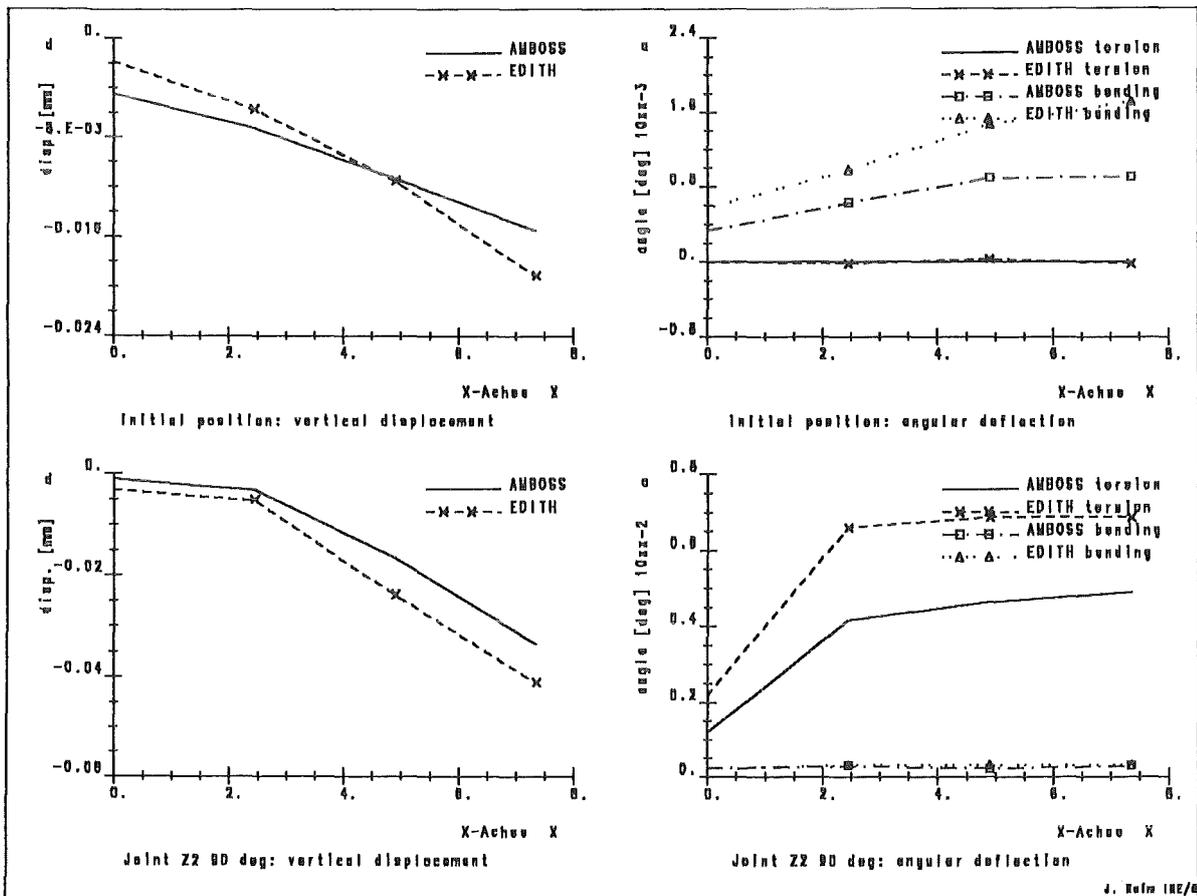
Figure 79. Statics Deflections in Simulation and Experiment

position	Z1-Z4: 0 deg	Z4 90 deg	Z3 90 deg	Z2 90 deg
EDITH experiment				
1. Mode	4.2	4.4	4.0	3.5
2. Mode	14.3	12.8	10.2	12.7
AMBOSS model				
1. Mode	4.24	3.9	3.0	2.6
2. Mode	14.9	14.9	11.7	10.5

Table 21. Natural Frequencies in Simulation and Experiment

4.1.4 Layout and Optimization of the Joint Controller

This chapter reports about the conventional and advanced control studies performed within the AMBOSS simulation. In several development steps the controller and the utilized reference plant models are improved and detailed. The design strategy regards the special dynamic effects appearing at the IVHU that are

1. the influence of the gear oscillation on stability and control results,
2. the disturbation of the joint controllers by coupling effects,
3. the influence of non-linear effects in the actuators such as backlash and static friction.

Further the common requirements of a control system must be fulfilled:
4. Low overschooting
5. High stability
6. Low lag error
7. Low controller output

To solve this complex task the joint controllers are layed out with linear equivalent-models.

Different linear control schemes are tested utilizing the linear equivalent-circuit of the complex plant model. This intermediate step allows conventional layout and parameter trimming of the control system, because classical methods such as bode diagram and root locus diagrams can be applied. Another benefit is that the estimated layout can be checked in short test simulations with the equivalent-circuit. For these first layout checks the applied plant model has to cover all the phanomenons that the selected controller takes into account. Therefore, the detail level of equivalent-circuit is determined by the basic control layout. The linear joint controllers compared in this chapter can be tested with analytic computable plant models for one joint.

Nevertheless, the test simulations with the single joint equivalent-circuit do not allow a statement about stability and quality of the complex system behaviour. Therfore a final check and comparison of the designed controllers utilizing the full AMBOSS dynamics model is necessary. The following development steps are resulting:

1. Basic control layout with differential equation and block diagram
2. Set up equivalent-circuit fitting to the selected controller type
3. Trim controller parameters.
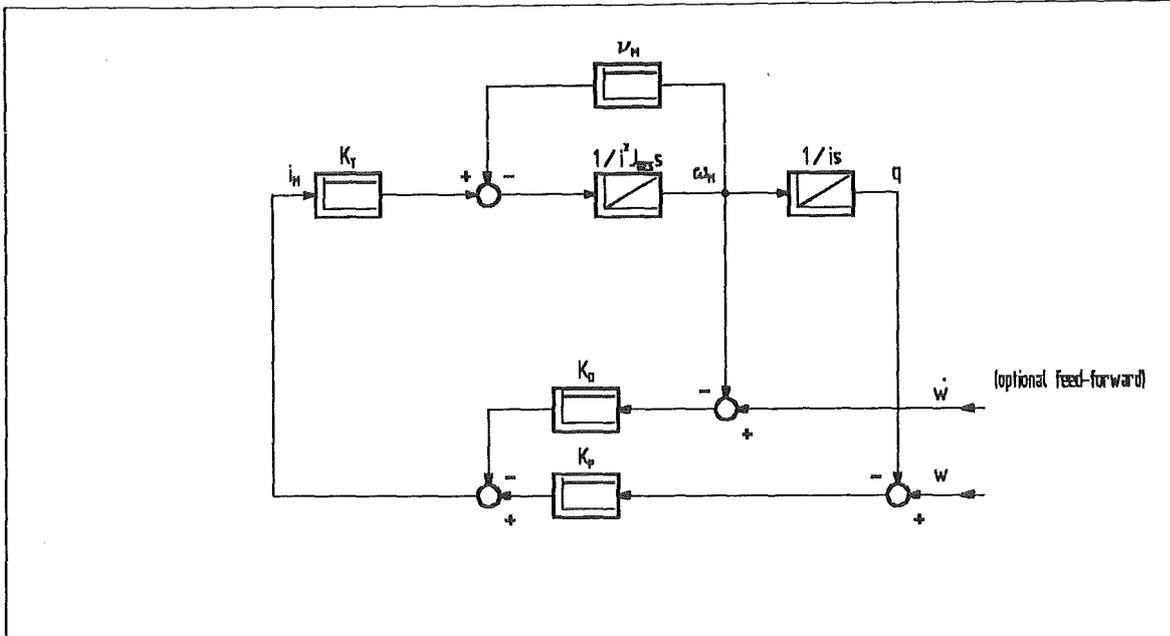
Figure 80. PD-Joint Controller with Equivalent-Circuit of one Joint

4. Verify controller layout in a comlex simulation.
5. Evaluate layout quality.

The next paragraph discusses a joint position controller with proportional-derivative feedback.

4.1.4.1 Single Part Equivalent Model with PD-Joint Control

The layout of this PD-Joint controller is documented in /30/. It is based on an equivalent model that transforms motor current proportional via an ideal stiff gear box without retardation on the joint. All occurring friction effects are summarized in the motor damping v_M. The differential equation of this system is

$$\frac{1}{i^2} J_{ges} \dot{\omega}_M = - v_M \omega_M + K_T i_M$$

$$\dot{q} = \frac{1}{i} \omega_M$$

(52)

Figure 80 shows the block diagram of a control with equivalent plant model and PD-controller.

The control layout in /31/ calculates the control parameters for given poles of the closed circuit. It starts with the frequency response of the plant
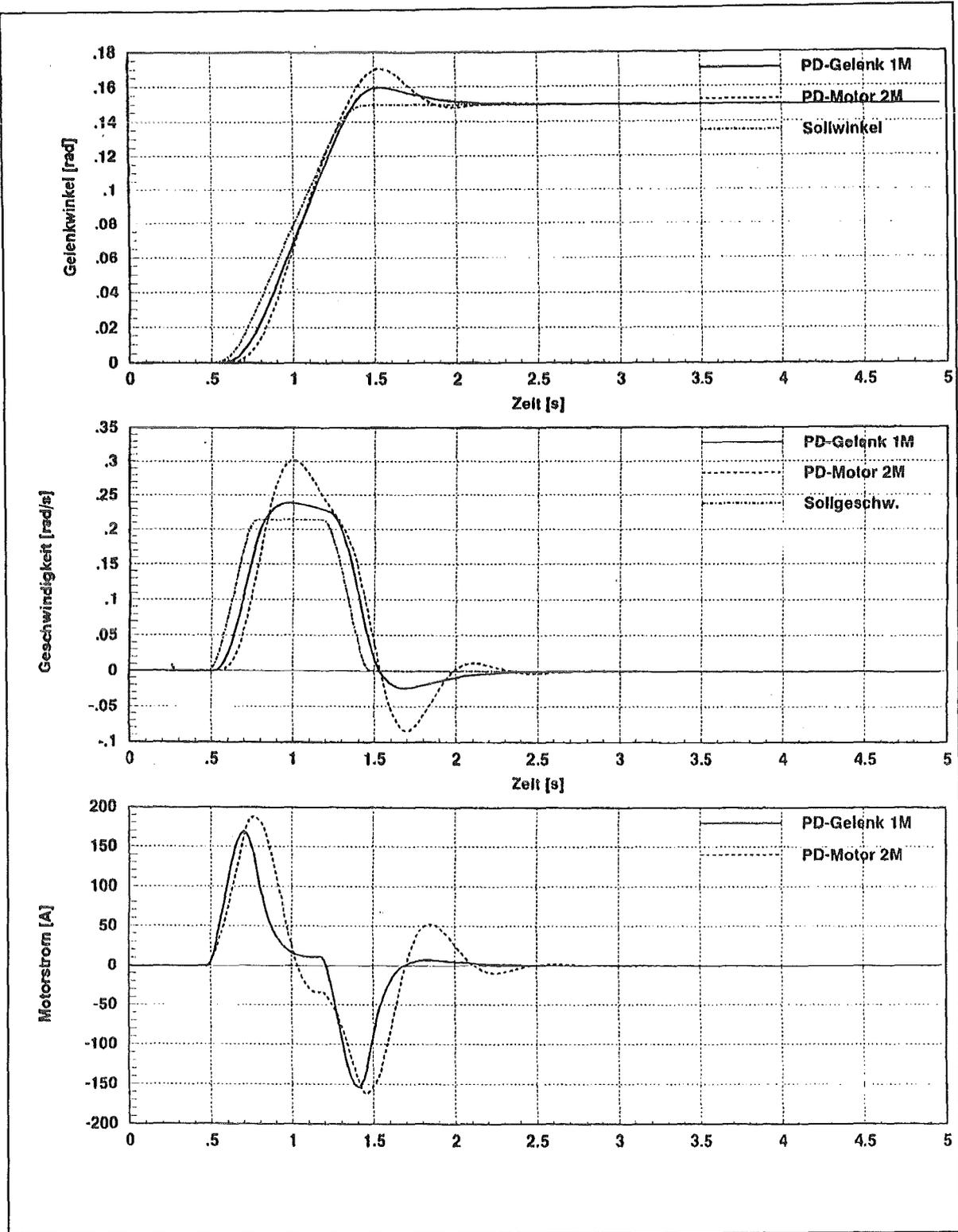
Figure 81. Simulation with PT1-I-Equivalent-Circuit and PD-Joint Control.
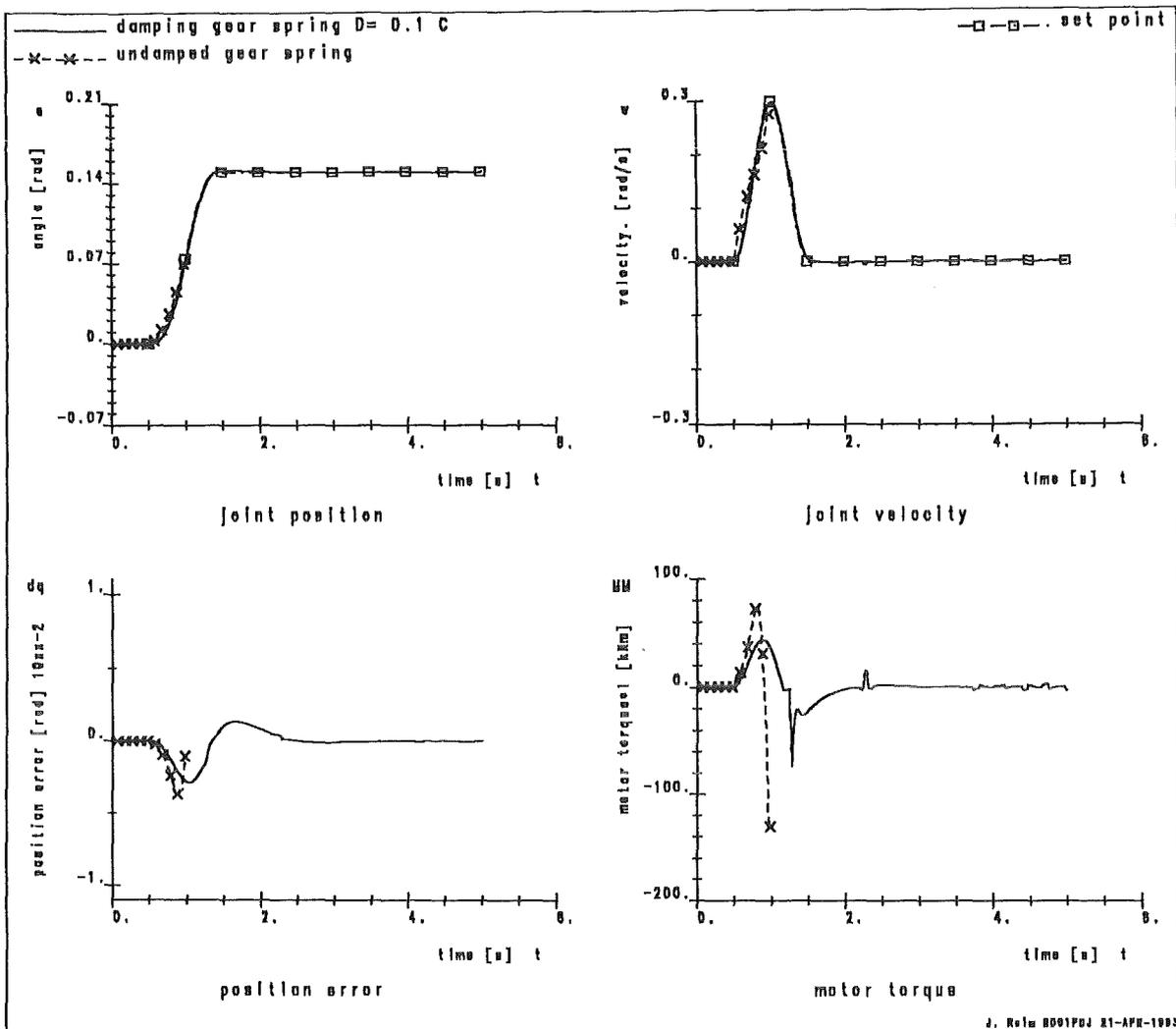
Figure 82. PD-Joint Control with Varied Damping of the Gear Spring

$$G_S(s) = \frac{Q(s)}{I(s)} = \frac{1}{s} \frac{\dfrac{iK_T}{i^2 v_M}}{1 + \dfrac{J_{ges}}{i^2 v_M} s}$$

$$= \frac{K_S}{s(1 + T_S s)} \tag{53}$$

with the delay time $T_S = \dfrac{J_{ges}}{i^2 v_M}$

and the gain $K_S = \dfrac{iK_T}{i^2 v_M}$

The second order retardation of the plant is compensated by a PD-feedback. The controller parameters are gained from the analysis of the closed control circuit neglecting the influence of the rated velocity. With the feedback transmission function

$$G_R(s) = K_P + K_D s \tag{54}$$

the closed loop transmission function becomes

$$G_W(s) = \frac{K_P\, G_S(s)}{1 + (K_P + K_D s)\, G_S(s)}$$

$$= \frac{1}{s^2 + \dfrac{1 + K_D K_S}{T_s}\, s + \dfrac{K_P K_S}{T_S}}$$ (55)

Evaluating the characteristic polynom the complex poles of this transmission function are

$$s^2 + \frac{1 + K_D K_S}{T_s}\, s + \frac{K_P K_S}{T_S} = 0$$ (56)

Optimal master control is gained placing both natural values together at the aperiodic case. The complex part responsible for oscillation disappears. The controller gains K_P and K_D result comparing the coefficients of the characteristic polynom. Equation (57) evaluates the terms for a desired time constant $T_w = 0.15$ s of the closed loop.

$$K_P = \frac{T_S}{K_S\, T_W^2} = 6000\,\frac{A}{RAD}$$

$$K_D = 2\, T_W\, K_P - \frac{1}{K_S} = 1775\,\frac{As}{RAD}$$ (57)

Figure 81 on page 128 shows the results of test simulations with the single mass equivalent plant model. In the upper diagramms joint velocity and joint angle are printed above the time axis. The curves refer to a simulation of EDITH joint Z1 in stretched boom position.

The PD joint controller is tested in simulation with the complex AMBOSS model of the in-vessel transporter. These simulations point out that this controller layout is sensitive to parameter variations in the actuator model and has only limted stability.

In Figure 82 on page 129 the simulations with varied damping of the gear spring are compared. While the first simulation with a damping of 10 % is stable, simulations with undamped gear box oscillation are obviously unstable.

From the high influence of the gear oscillation on the stability of the control one can conclude that the single mass equivalent plant model is neglecting relevant effects and not valid for control layout.

4.1.4.2 PD-Position Control of the Motor Shaft

The next development step achieves a stable control feeding back the motor side position and velocity data. This removes the disturbing oscillation of the gear box from the control circuit. The advantage of nearly unlimited stability has on the other hand the disadvantage of uncontrolled gear oscillation because its oscillating mechanics is positioned outside the control loop. This PD servo controller has been the first access to stable control of the boom and is the basis for further optimization.

Figure 83 on page 131 compares simulation results with 10 % damped and undamped gear spring. The motor torque and gear torque curves have significant oscillations and visualize the gear deformation. Load changes at the beginning and at the end of the ramp curve excite the gear box oscillation. In the damped case the influence of the gear oscillation causes stick slip motion having reached the set point value. The absolute
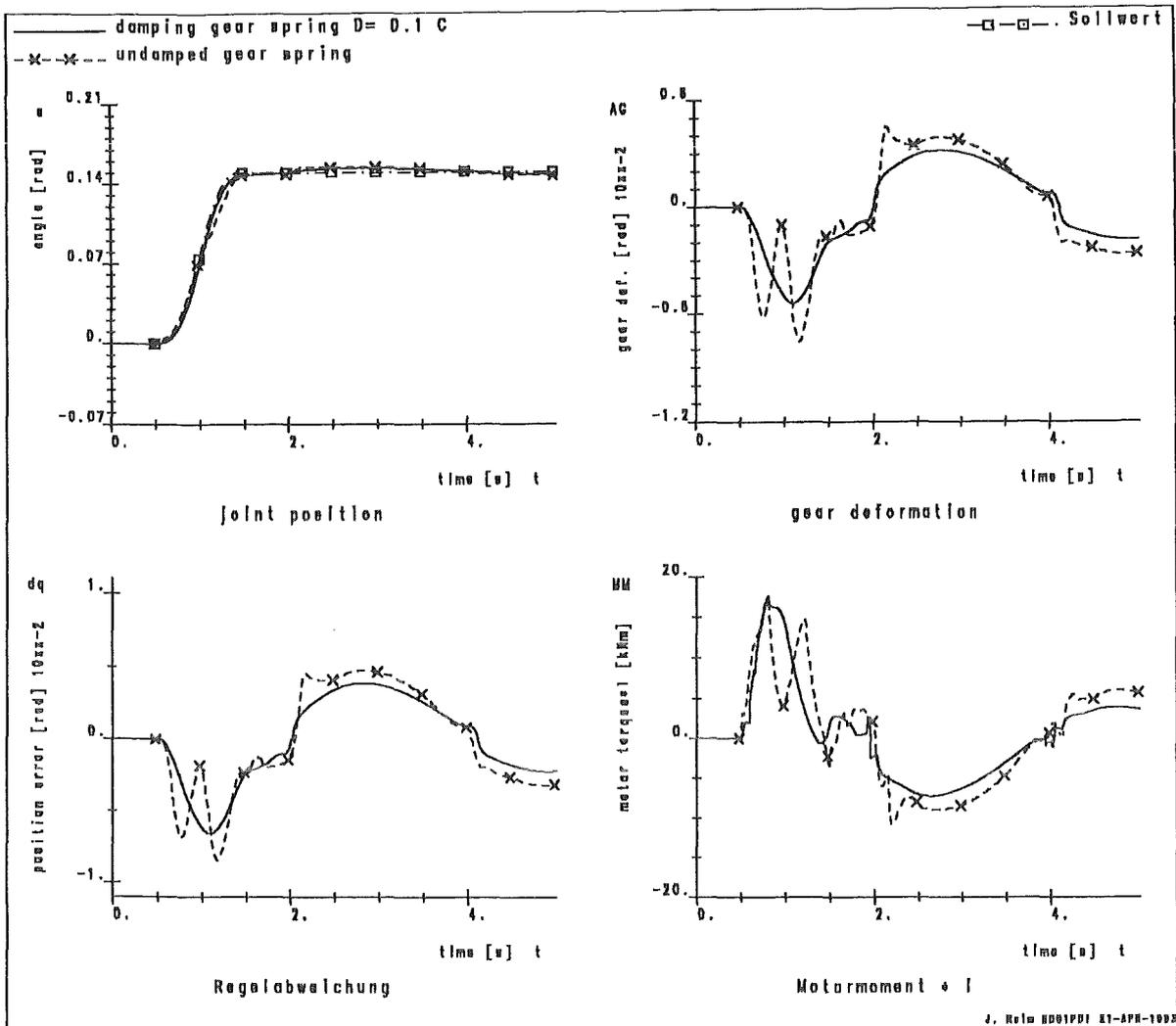
Figure 83. PD-Motor Position Control with Undamped Gear Oscillation

control error is about 10^{-2} RAD. This corresponds to a position error at the tip of the boom of about 10 cm.

Limited stability of the PD joint controller and high deviation of the simple servo controller state that both test layouts do not fit for the task controlling a rather compliant gear box. Putting the disturbing elements into the feed back control loop an improvement of the control results should be possible applying classical control design methods. The next development step examines cascaded control loops with an inne r servo and an outer position control loop. The layout requires a more detailed equivalent circuit which comprises the gear oscillation to be damped.

4.1.4.3 Equivalent Circuit with Compliant Gear Box and Servo Controller

Mass and inertia of the single mass equivalent model are distributed on two parts, the motor shaft and the link itself. Equation (58) shows the coupled equations of motion of this system.

$$
\begin{aligned}
J_M \, \dot{\omega}_M &= -D\,\omega_M + D\,i\,\dot{q} - C\,\alpha_M + C\,i\,q + K_T\,i_M \\
\dot{\alpha}_M &= \omega_M \\
J_L \ddot{q} &= D\,\frac{\omega_M}{i} - D\,\dot{q} + C\,\frac{\alpha_M}{i} - C\,q \\
\dot{q} &= \dot{q}
\end{aligned}
\tag{58}
$$

Figure 84. Cascaded Control Scheme with Plant Model Including Gear Oscillation.
  PPI-Cascaded Scheme: $K_D = 0$
  PDP-Cascaded Scheme: $K_I = 0$

For test simulations and control analysis the differential equations are transformed into the state space description with the system matrix A and the input matrix B.

$$\dot{x} = A x + B u \qquad\qquad (59)$$

The resulting equations of the two part plant models are

$$\frac{d}{dt}\begin{pmatrix}\omega_M \\ \alpha_M \\ \dot{q} \\ q\end{pmatrix} = \begin{pmatrix} -\dfrac{D}{J_M} & -\dfrac{C}{J_M} & \dfrac{iD}{J_M} & \dfrac{iC}{J_M} \\ 1 & 0 & 0 & 0 \\ \dfrac{D}{iJ_L} & \dfrac{C}{iJ_L} & -\dfrac{D}{J_L} & -\dfrac{C}{J_L} \\ 0 & 0 & 1 & 0 \end{pmatrix}\begin{pmatrix}\omega_M \\ \alpha_M \\ \dot{q} \\ q\end{pmatrix} + \begin{pmatrix}\dfrac{K_T}{J_M} \\ 0 \\ 0 \\ 0\end{pmatrix} i_M \qquad (60)$$

The selected cascaded control scheme is added to the equations of motion as shown in the block diagram Figure 84 on page 132. The examined variants are covered omiting selected controller elements .

4.1.4.4 Layout of two Cascaded Scheme Controllers

Laying out a cascaded control scheme for an actuator with elastic gears an inner servo controller is put to the outer position loop. While the outer loop controls the desired joint position the inner servo loop stabilizes the motor speed. Stating a general linear feedback both controllers can have PID characteristics. But having twice derivation or integration in the total feedback the elements will disturbe or elliminate each other. Therefore, one controller of the cascaded scheme should have proportional feedback.

In the literature there is no unique hint which controller should have proportional characteristics. Therefore, a general selection between two suggestions is performed in a simulation with the complete model.

Figure 84 on page 132 summarizes signal processing in the cascaded control scheme. The outer position loop has PD charcteristics. Its differential part is ommitted in the PPI cascaded case ($K_D = 0$). Servo control is layed out as PI controller. The integrating part avoids a steady deviation. In the PDP cascaded case the integrating part of the servo controller is set to zero ($K_I = 0$). The resulting P servo controller (block K_P) is fed by the output of the PD position controller. An additional velocity feed forward (K_V) takes care of a fast controller reaction on set point changes.

The above discussed PD controller parameters are taken as initial layout of both cascaded schemes. The gain of the newly added proportional feedback element and an adapted total gain are derived as discussed in the next section:

1. **Adding a P-Position Controller to a PI-Servo Controller**

   The PPI cascaded scheme utilizes the already implemented PD-position controller as PI servo controller. Only one parameter is missing to fix the outer position controller. Root locus curves are calculated for the closed position loop. Figure 85 on page 134 shows in its lower left area the dependency of the gain K_L in the complex plane. The position control is generally stable having a negative real component. The desirable aperiodic case is achieved at equal sized real and imaginary components (on 45°-line). The evaluation of the curves points out that the position loop has only limited stability at low gain values. An optimal trim of this position loop seems to be reached at $K_L = 0$ when the position loop is completely omitted. For simulations a quite low gain of $K_L = 0.25$ is selected.

$$\begin{aligned} K_L &= 0.25 \\ K_D &= 0 \\ K_P &= 6000 \\ K_I &= 1775 \end{aligned} \qquad (61)$$
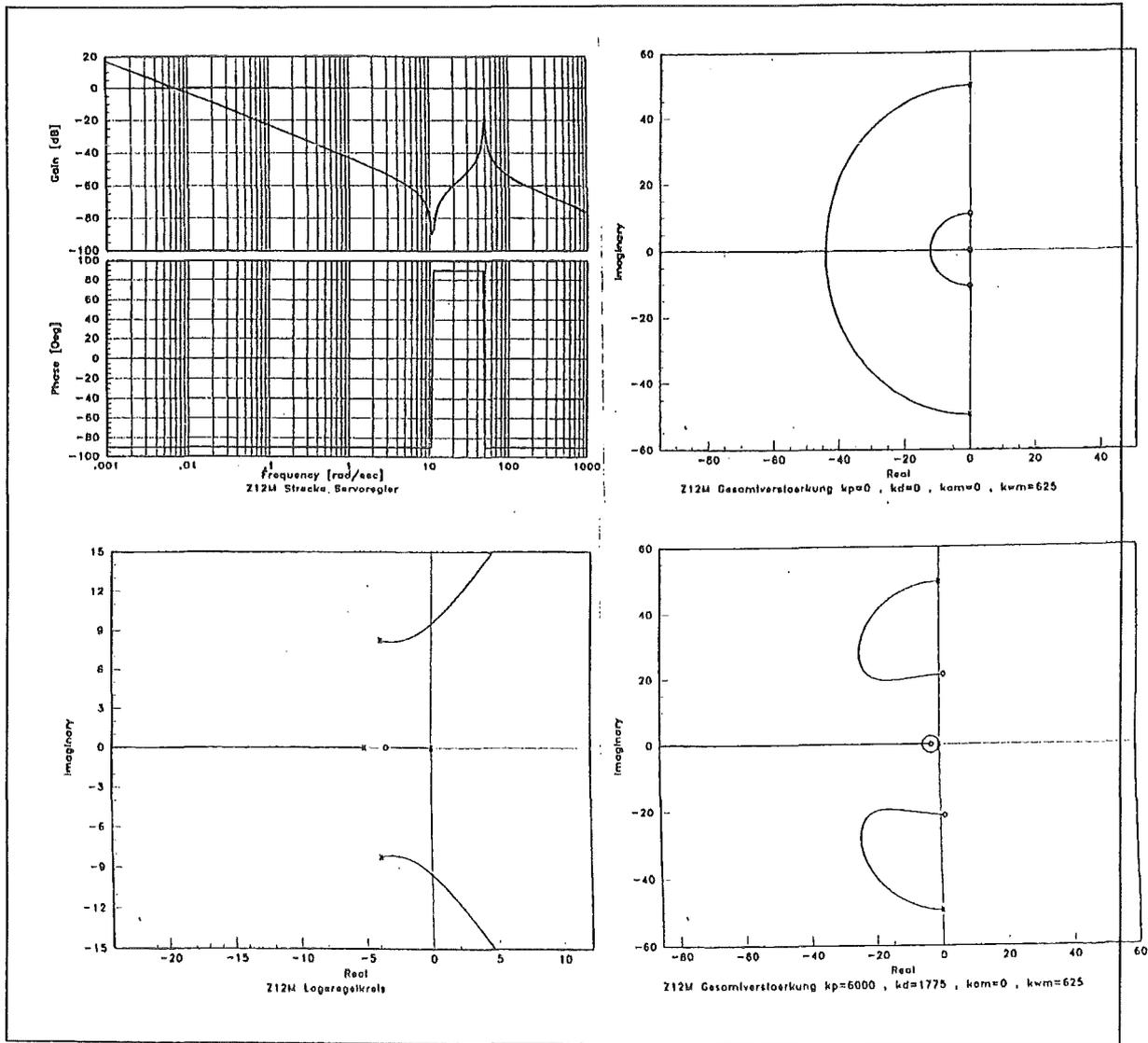
Figure 85. Layout of a Cascaded Control Scheme in Frequency Domain.
upper left: Frequency response of the servo circuit as bode curve
lower left : Root locus curves of the P-Position loop with PI-Servo control
upper right: Root locus curves of the P-Servo loop
lower right: Root locus curves of the PDP-Cascaded scheme

Calculations with the equivalent circuit and the complex model fit with the results of the root locus analysis. The PPI-cascaded scheme is not able to improve the control result compared with the PD-motor position controller.

2. **Adding a P servo controller to a PD position controller**

In a similar procedure the existing PD controller is taken as basic layout for the PD position feedback in the cascaded scheme. The newly added proportional servo circuit has to be trimmed to complete the layout. Again the root locus of the closed control loop is calculated (see Figure 85 in the upper right). Since the poles have a negative real component for the whole parameter range of the gain K_P the proportional servo loop is unlimited stable. The optimal gain is selected at $K_P = 624$ where the closed loop has a high damping and low amplification of disturbances. The parameters of the PD position control are divided by the new derived servo gain K_P to keep the effect of the position loop. The resulting parameters referring to Figure 84 on page 132 are
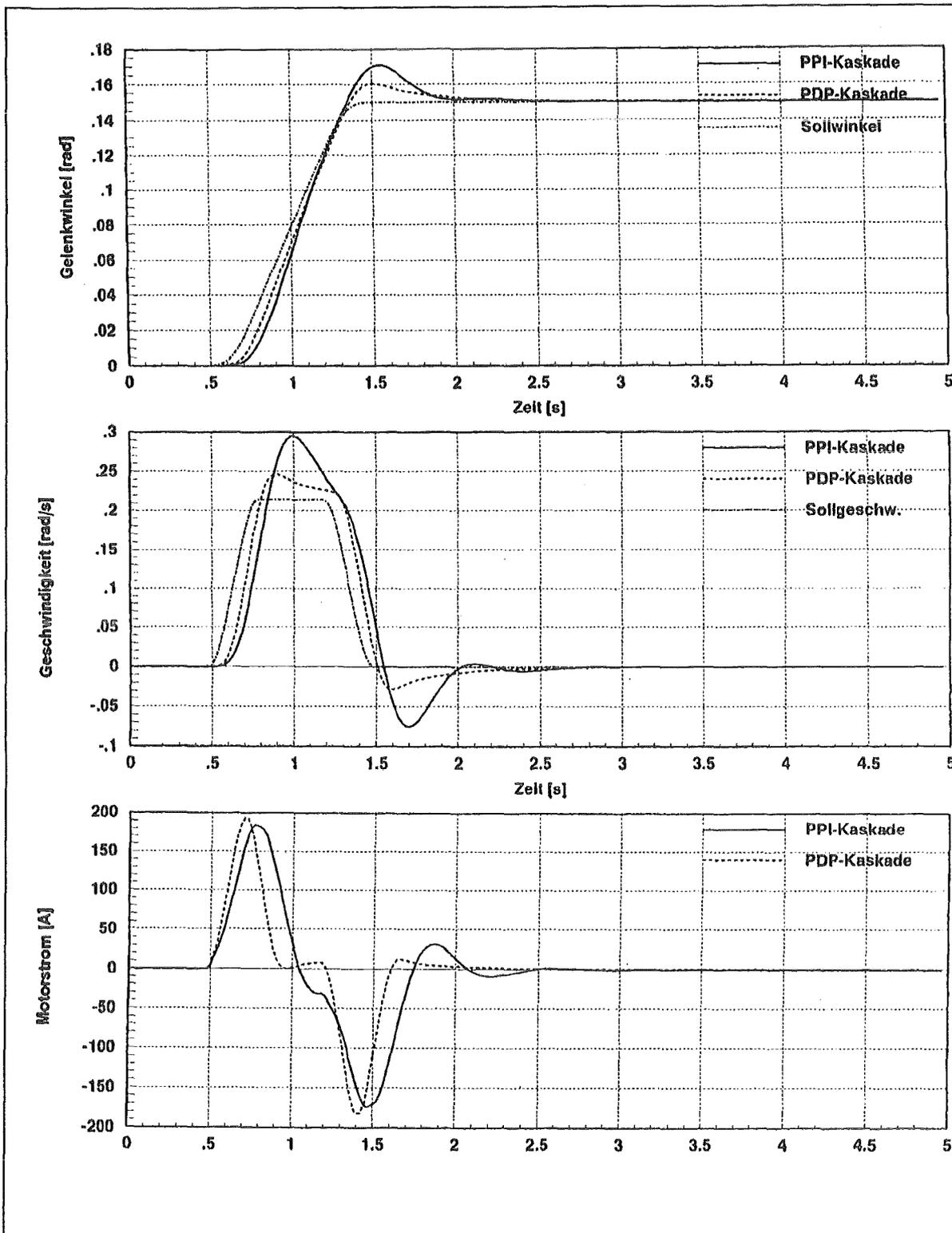
Figure 86. Simulation of the Cascaded Control scheme with forth Order Plant Model

$$K_L = 6000/625 = 9.60$$
$$K_D = 1775/625 = 2.85 \tag{62}$$

Finally the root locus are evaluated for the complete cascaded scheme to correct the servo gain K_P for the superimposed case (see Figure 85 on page 134 lower right). Scaling the primary evaluated servo gain to the optimum total gain the control parameters for the PDP cascaded control are
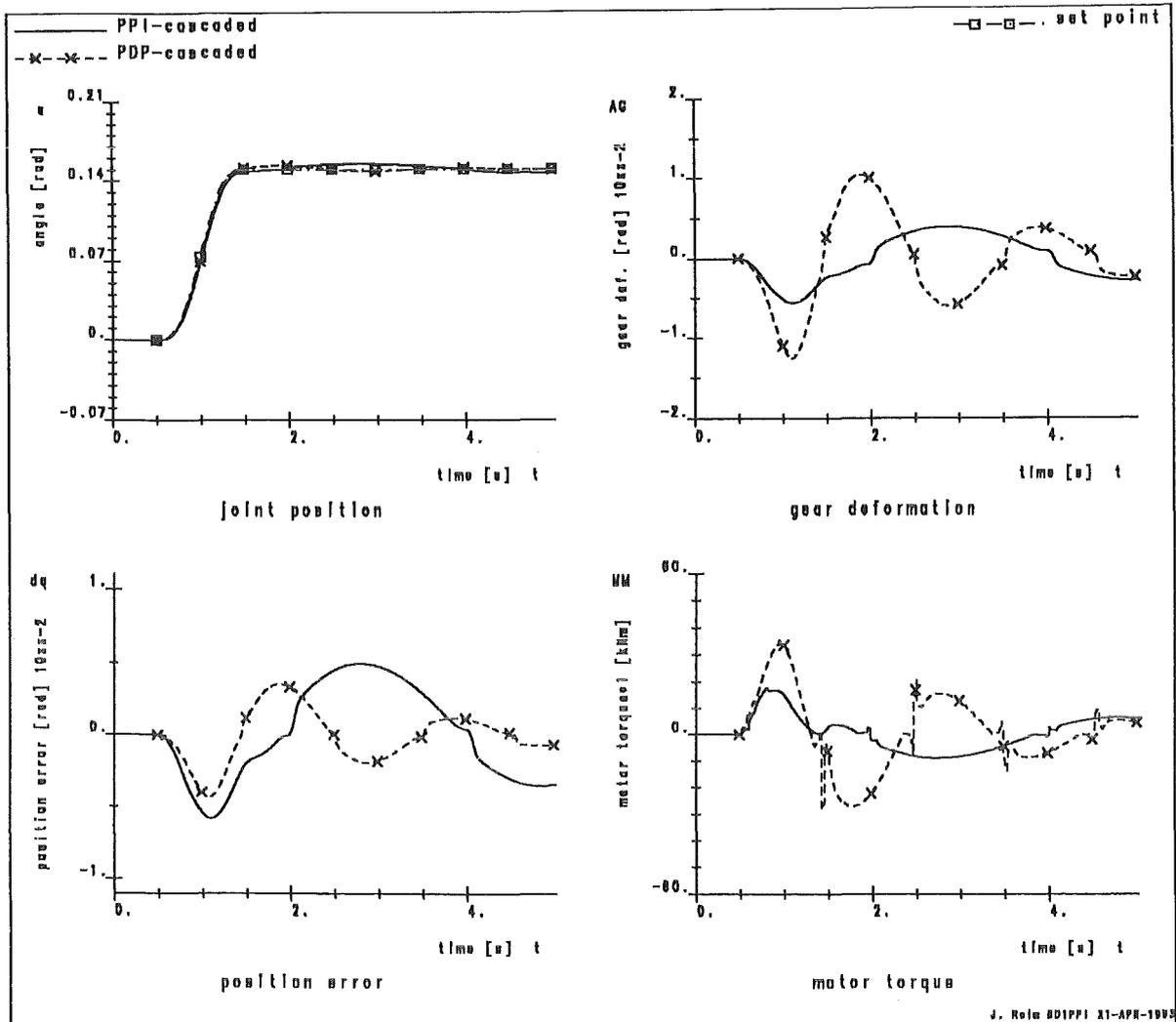
Figure 87. Comparison of the two Cascaded Schemes in the full AMBOSS Model

$$K_L = 9.60$$
$$K_D = 2.85$$
$$K_P = 530 \tag{63}$$
$$K_I = 0$$

Both cascaded control schemes are tested in a simulation with the equivalent circuit (Figure 86 on page 135) and with the complex overall IVHU model (Figure 87). In all important topics the PDP-cascaded scheme beats the PPI-scheme as

1. lower set point deviation repectively position error, time lag and overshooting,
2. higher damping of the gear box oscillation,
3. shortened time constant of the closed loop system.

4.1.4.5 Feed-Forward Control to Reduce Coupling Effects

The next optimization step tries to reduce coupling effects between the joint control loops. Therefore, the manipulated variable is superimposed at the controller output with motor current necessary to follow the desired path. This feed forward value covers most of the necessary motor current. Therefore, the controller only has to work against disturbances and errors. Figure 88 on page 137 shows the joint torque curves that were calculated for the joints Z1 to Z4 of the IVHU in a simulation without actuator effects.
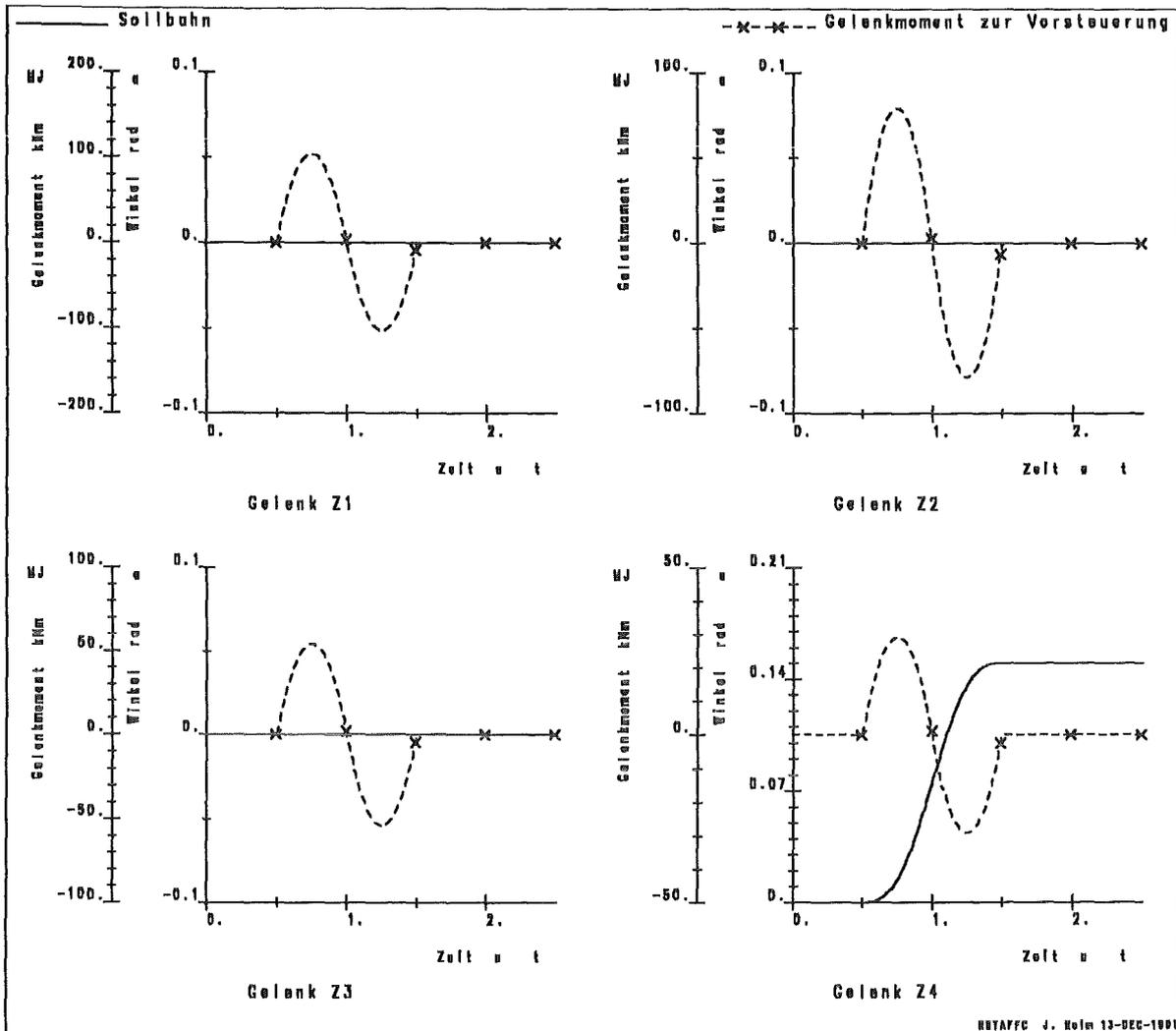
Figure 88. Joint Torque Curves Calculated for Feed Forwaed Input

Having elastic gears the feed forward of the joint torque necessary for motion has only few improvement of the control result. This is caused by the retarding behaviour of the gear box. Therefore, in a further optimization step besides the joint torque the estimated gear deformation is calculated from the necessary joint torque and added as feed forward motor position. Figure 89 on page 138 shows the results of the two implemented feed-forward schemes for the above used reference ramp set point. A significant improvement is achieved superimposing the two methods (FFC + WWM). In the lower diagrams the deviation of the disturbed joints Z1 to & rbl.Z4 is shown.

To check the influence of erroneous feed forward signals in all simulations the feed forward angle and torque were applied with 25 % error. The torque feed forward seems to be quite insensitive for feed forward errors. At the motor angle feed forward technique erroneous input sometimes causes high torque peaks.

4.1.4.6 Influence of Non-Linear Actuator Elements

The incluence of non-linear transmission elements of the actuators is an other important criterion for control layout. The main non-linear effects disturbing control are the slacknes of the gears and static friction. Both effects are taken into account by the AMBOSS model. To estimate their influence on the designed control simulations with omitted, enlarged, and average non-linear effects are performed. Figure 90 on page 139 shows that backlash and friction have only few influence on the PDP-cascaded
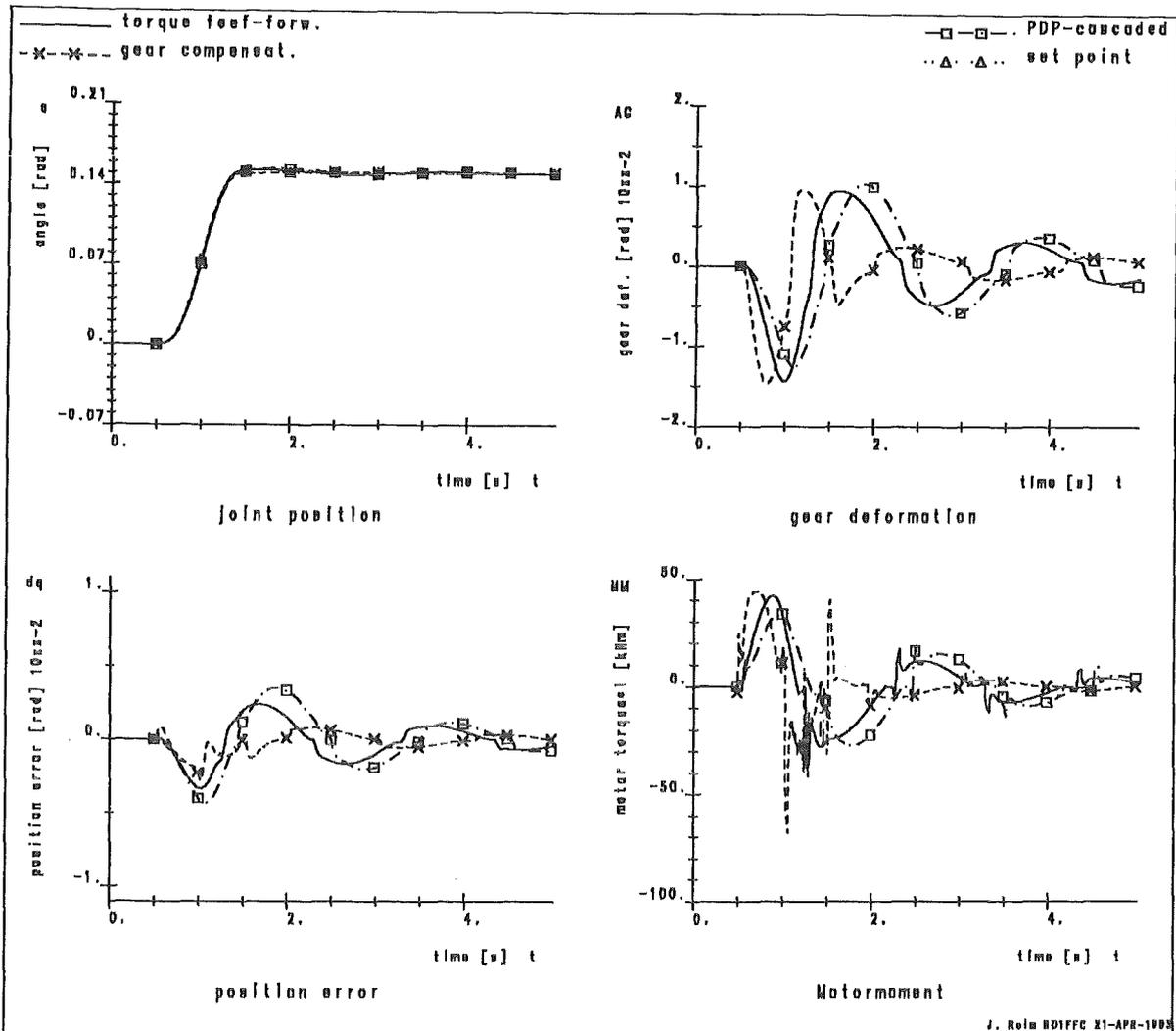
Figure 89. Comparison of Joint Torque Feed Forward and Set Point Correction

control. Changes are only as large as the parameter variation affects the model. The position error rises with the amount of the backlash enlargement and the motor torques need additional strength to compensate the friction torque. In opposite to the quite robust PDP cascaded control the PD joint controller, which is the originally planned control layout for EDITH, reacts quite sensitive to nonlinear actuator effects. A slight increase of backlash or static friction causes an instable control circuit as shown in the figure.

4.1.4.7 Final Performance Test with Triangle Path

To check the overall control behaviour with realistic superimposed joint motion a triangle path near a realistic working position is tracked. The selected path starts in a half unfolded position as shown in Figure 91 on page 140. In nearly ten seconds a triangle with about 10 centimeters edge length is tracked. In every edge motion stops and restarts. Looking on Figure 92 on page 140 the different controller variants can be compared. The PD motor position controller is characterized by high overshooting. Controller layout with additional feed forward control (FFC, WWM) have reduced position error in high speed areas.
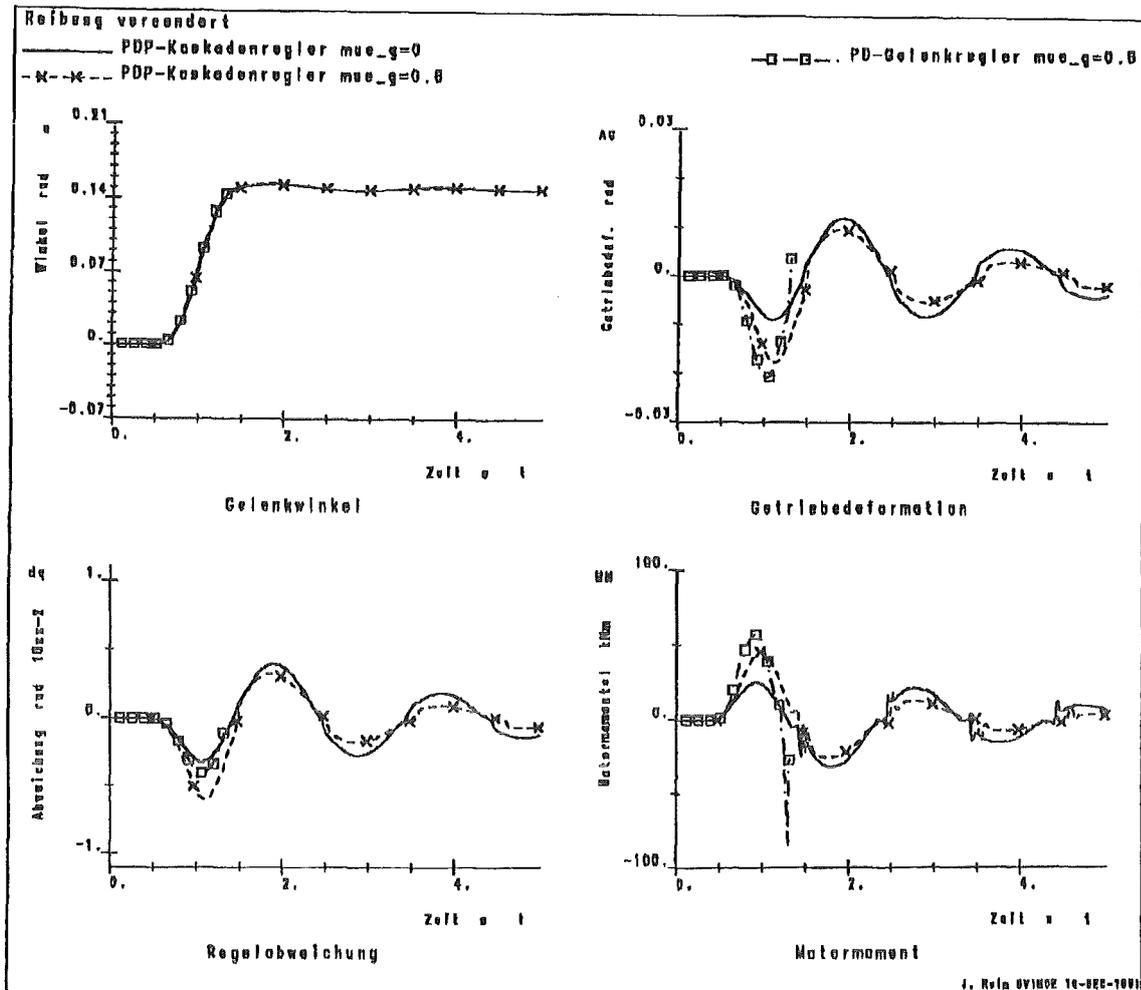
Figure 90. Simulations with enlarged Nonlinear Transmission Effects.
 upper: Enlarged backlash
 lower: Enlarged static friction

4.2 Performance Studies for Critical Operations Carried out with the In-Vessel Handling Unit

4.2.1 Passing the Port of the Vessel

The in-vessel transporter must get into the vessel through a narrow port. A point-to-point path has been generated for the motion - unfolding outside the vessel - inserting the boom - moving to the point 90° away from the port. Therefore, several positions were fixed. Between them the path was scaled to move with maximum velocity and acceleration of $0.1 \dfrac{rad}{s}$ or $0.1 \dfrac{rad}{s^2}$ respectively. The whole path is run about five times faster than assumed in the layout.

For the PPI controller the performance is not satisfactory because the gain at the first joint Z1 had to be reduced to achieve stability when the boom is stretched.
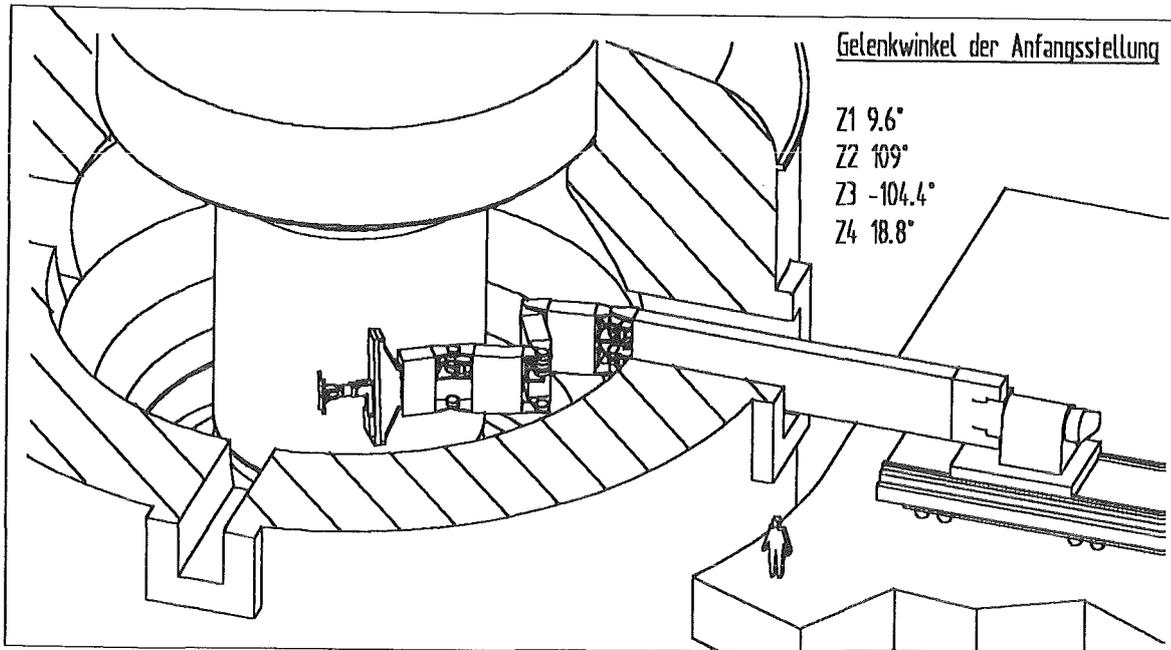
Figure 91. Initial Position of the In-Vessel Handling Unit for the Triangle Path
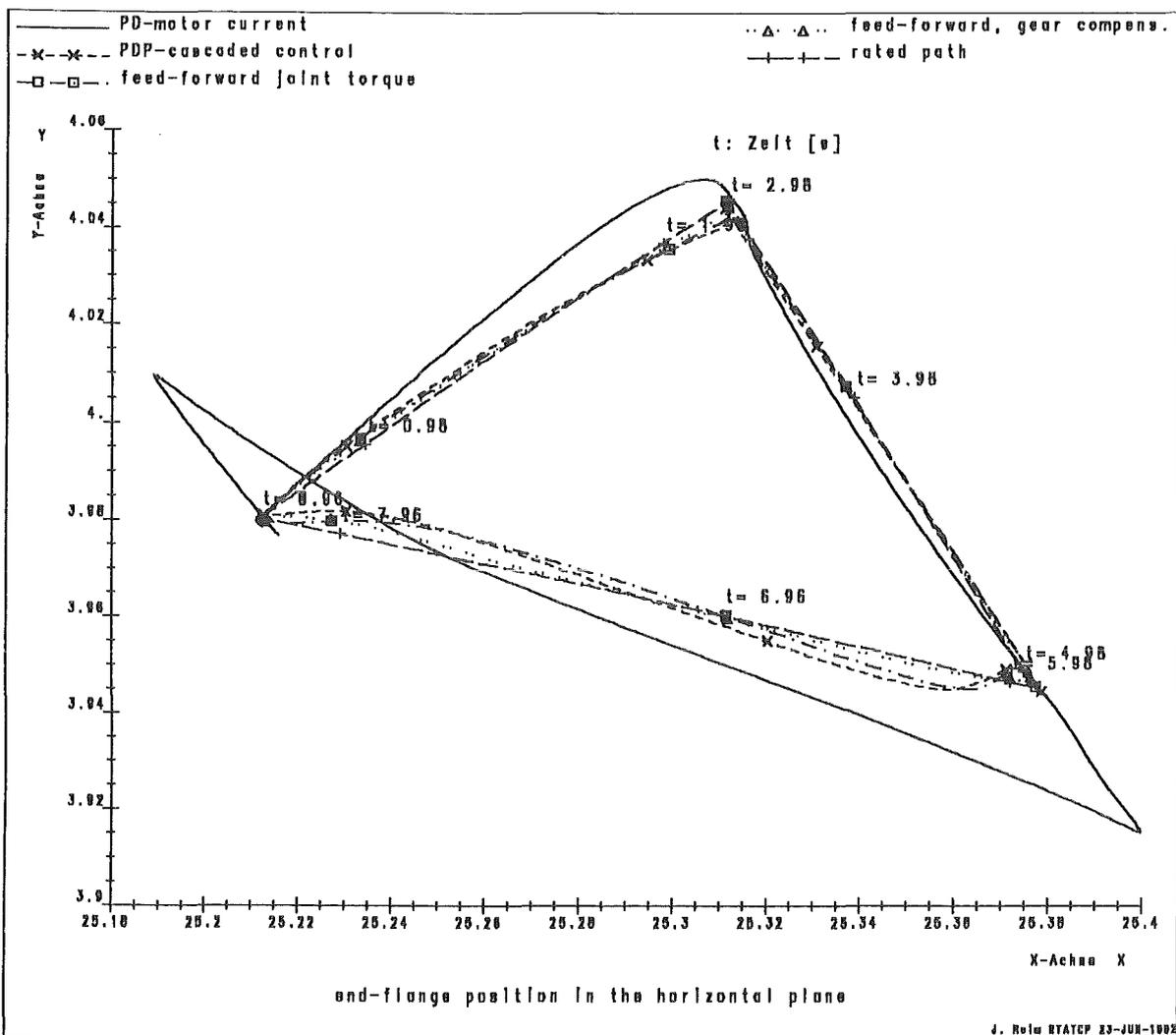
Figure 92. End-flange Tracking at Triangle Path
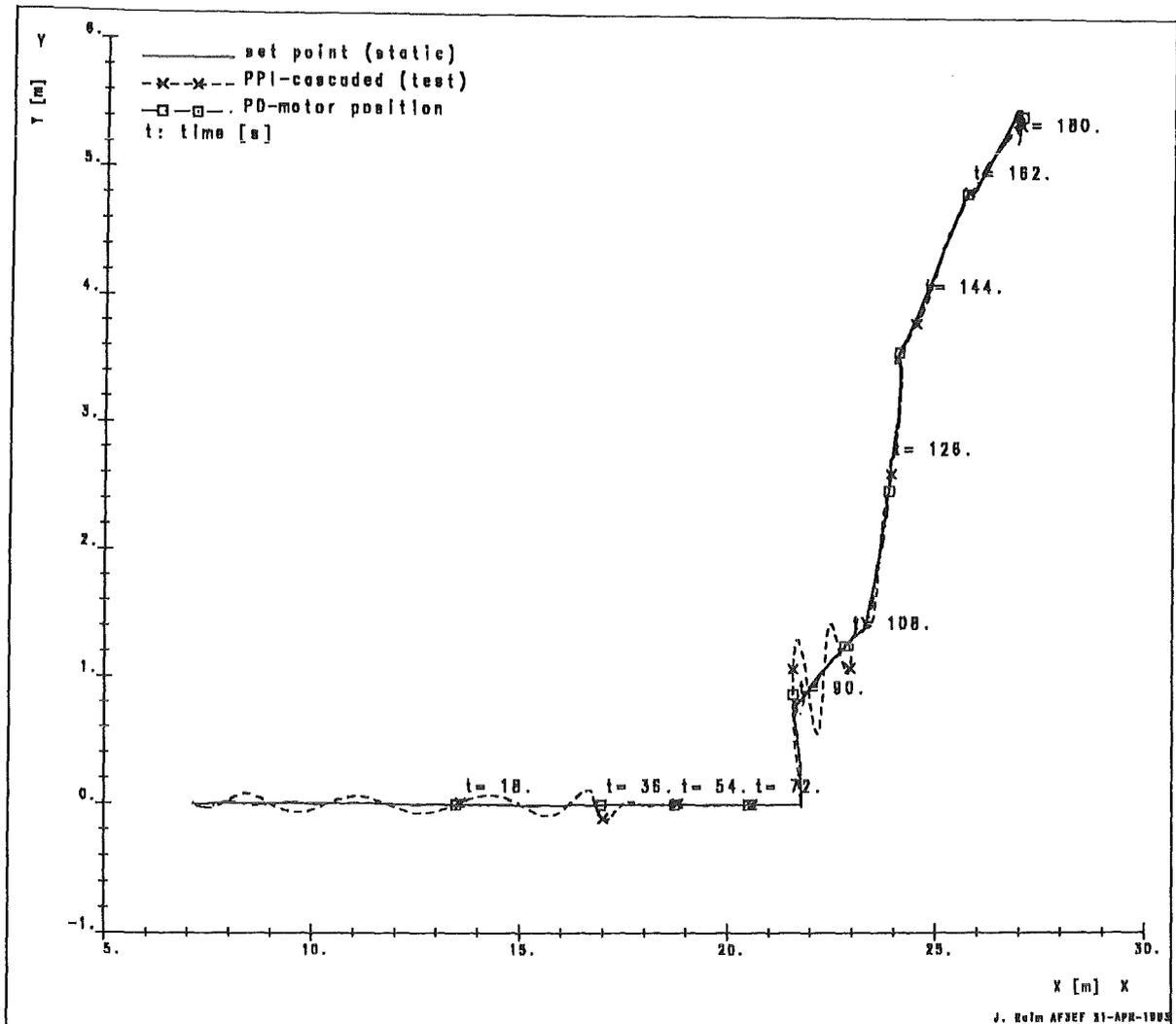
Figure 93. Path of the End-Flange in the horizontal Plane

Figure 93 shows the nominal path of the end-flange and the results for the PPI-cascaded control scheme and the PD-position controller.

4.2.2 Load Transfer at the Upper Divertor

The other critical in-vessel operation examined with the AMBOSS model is an accident situation at the upper divertor plate. A sudden load transfer of 1000 kg is assumed in the 90° position shown in Figure 94 on page 142.

The boom, initially in a static balance suddenly changes to the loaded state and builds up to the new transient position with an additional deflection of about 20 mm. The path of the end-flange oscillation after load transfer is drawn in the three-dimensional plot in Figure 95 on page 143
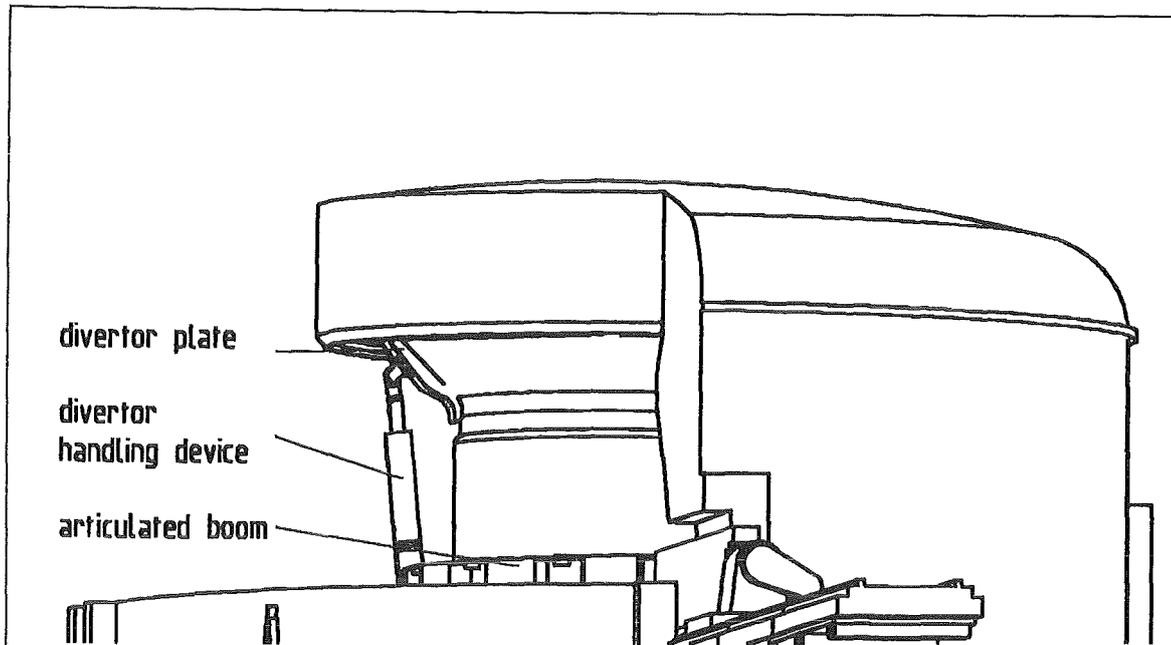
Figure 94. Load Transfer at the Upper Divertor

Peak torques arising in joints and links amount between 550 and 1100 kNm. The maximum torque in the actuators is about 50 kNm which is between the nominal and the maximum peak torque of the actuator layout /35/.

Symbols and abbreviations

B	half backlash	C	gear stiffness
D	damping of gear spring	G	transmission function
i	gear ratio (input over output)	i_M	motor current
J_{ges}	total inertia on a joint	J_M	motor inertia
J_L	arm inertia	K	constant transmission
K_T	torque constant	q	joint position (angle)
T	time constant ??	w	rated joint position (angle)
v_M	motor damping	μ_G	sataic friction coefficient
α_M	motor angle	ω_M	motor velocity

5.0 Full Scale Mock-up

The full scale mock-up as shown in Drwg. IT-OUT-14-2 and Figure 96 on page 144 simulates the upper part of a NET/ITER torus sector. It can be positioned at different arrangements with respect to the position of EDITH (see Figure 97 on page 145). The mock-up is made up of the following main subassemblies:

* Basic structure
* Inner structure
* Erecting scaffold

The base structure is a combined welding- and screw-structure. At the base structure the inner structure is mounted. Dependent on its arrangement the upper or the lower
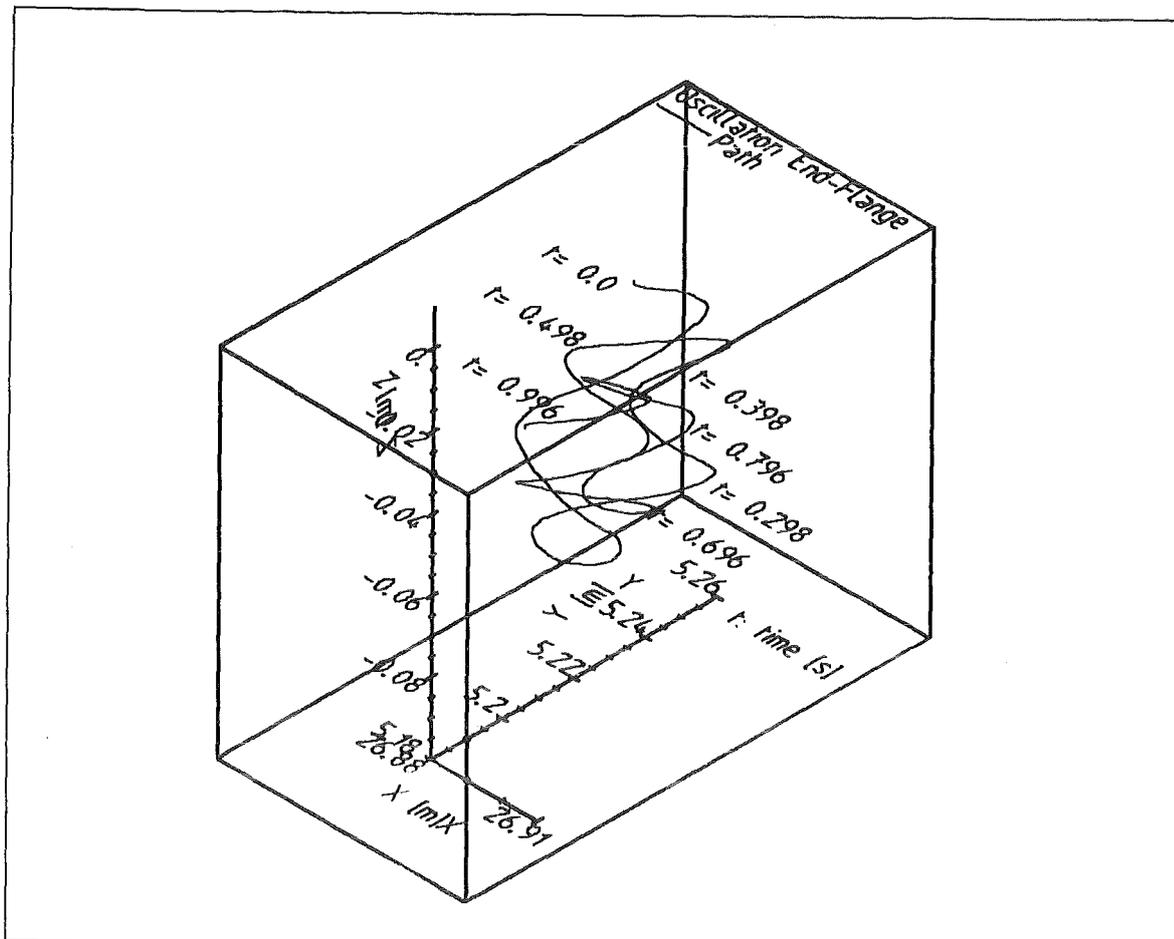
Figure 95. Path of the End-Flange after Load Transfer

torus part is simulated. Inner structure as well as errecting scaffold are made up as a screw-structure.

The change from the upper to the lower torus part is possible by mounting the inner structure upside down (see Figure 98 on page 146).

At the inner structure dummies of the in-vessel components can be mounted for demonstration of handling operations. In a first step, a part of the first wall with protective armour tiles is simulated which serves as a basis object for the demonstration of remote detachment and and re-positioning of the tiles.

6.0 Outlook

EDITH and the full scale mock-up were commissioned as described in the previous chapters. The present performance of the handling system and the mock-up give the basis to carry out tests which are required to demonstrate that the plasma facing components and the equipment are designed such that in-vessel maintenance and especially the replacement of the components can be performed with the anticipated reliability and time. In particular, the following items are the main tasks for the qualification and demonstration tests:
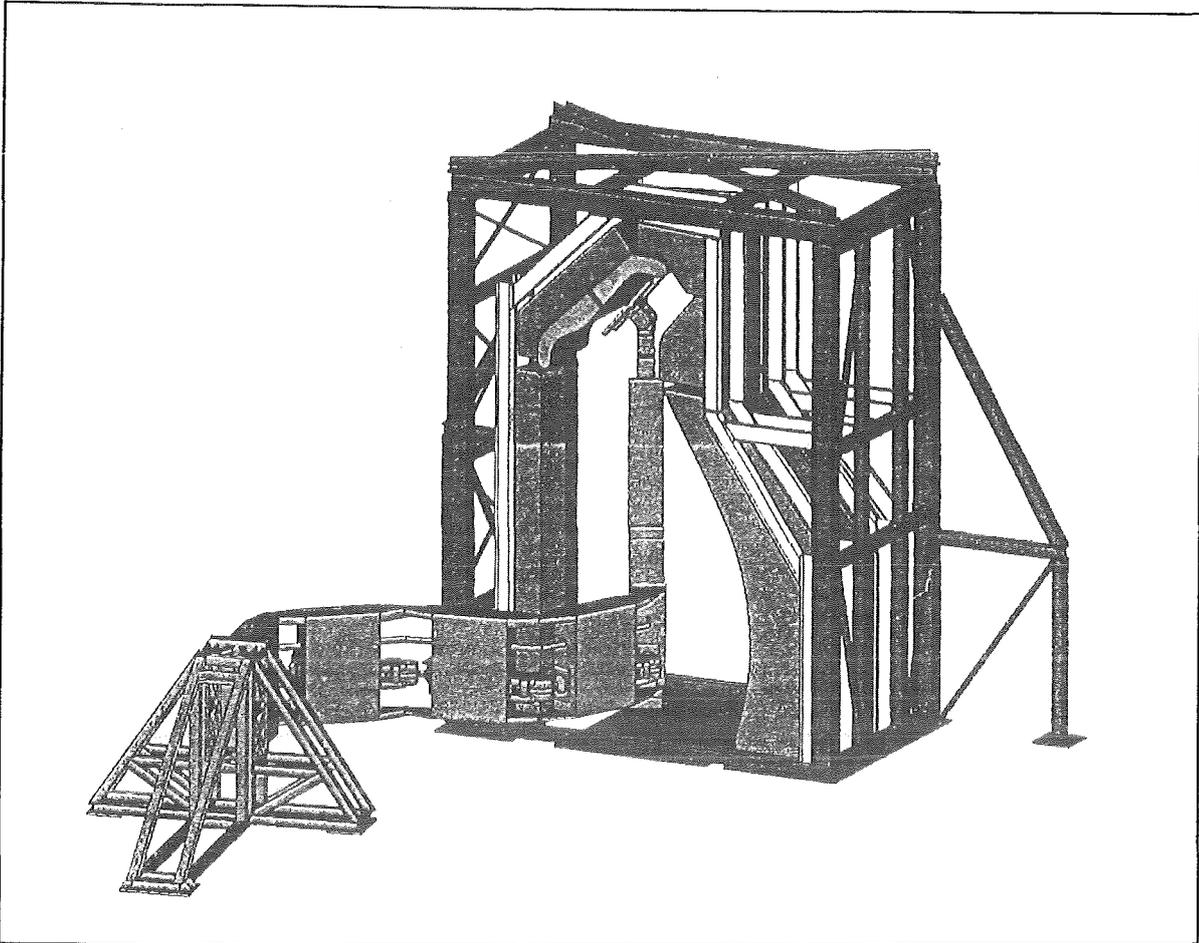
Figure 96. Full Scale Mock-up with EDITH. Divertor handling in the upper part of a torus sector

*   Integral tests and checks of combined effects of handling components and subassemblies. E.g., this includes also tests to varify the results of the stress analysis related to deflection.

*   Investigation of dynamic behaviour with respect to controllability and manoeuvrability.

*   Verification and demonstration of typical maintenance tasks. In a first step the replacement of protective armour tiles by means of tools which will be handled by the MHU will be investigated. Tools to be used are the tile detachment device and the tile positioning device.

*   Identification of possible improvements of the handling system and eventually of the machine and components to be handled.

*   Determination of maintenance time for specific tasks

*   Assessment of margins concerning reliability, availability and positioning accuracy.

7.0 List of Drawings

IT-OUT-08-423B   Teststandgerüst (Support Structure)
IT-OUT-08-428   Gelenk 2L (Link Joint Z2)
IT-OUT-08-446   Articulated Boom Transporter
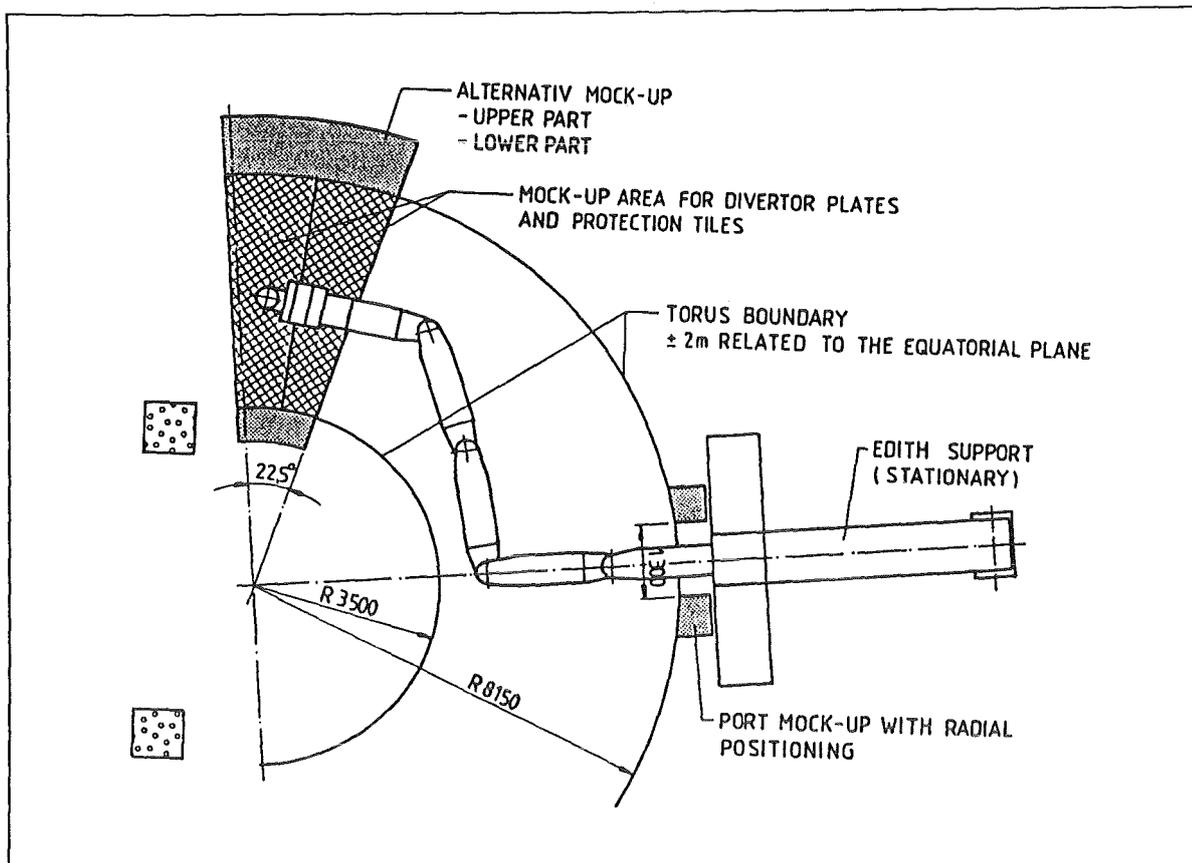
Figure 97. Scheme of Full Scale Mock-up - Plan View

IT-OUT-12-214	Coaxial Drive Unit
IT-OUT-14-39	Mock-up
IT-OUT-21-008A	EEPU - External Loads and Forces
IT-OUT-21-40	End-Effector Positioning Unit
IT-OUT-24-52	Manipulator Handling Unit
IT-OUT-24-53	Manipulator Positioning Unit

8.0 References

/1/ Suppan, A. et al.: The NET Articulated Boom: Preliminary Investigations and Justification for a Full Scale Prototype, KfK 4809, December 1990

/2/ The NET Team, NET Predesign Report, Fusion Engineering and Design, Volume 21, Feb. 1993, pp.92-106

/3/ Honda, T. et al.: ITER Assembly and Maintenance. ITER Documentation Series, No. 34, IAEA, Vienna, 1991

/4/ Maisonnier, D.: NET/ITER In-Vessel Transporters - Technical Specification for their Conceptual and Preliminary Design, Doc.Nr. N2/P/3240/1/B, issue: A, 24.10.1989

/5/ Reeve, T.: Divertor Maintenance Using IVHU - In-Vessel Operation, Doc.Nr. N2/P/0230/2/A, 11.05.1989

/6/ Reeve, T.: Definition of Requirements for the Design of Prototype Divertor Handling Equipment, Doc.Nr.N2/P/023132/2/A, issue: Draft Jan. 1990
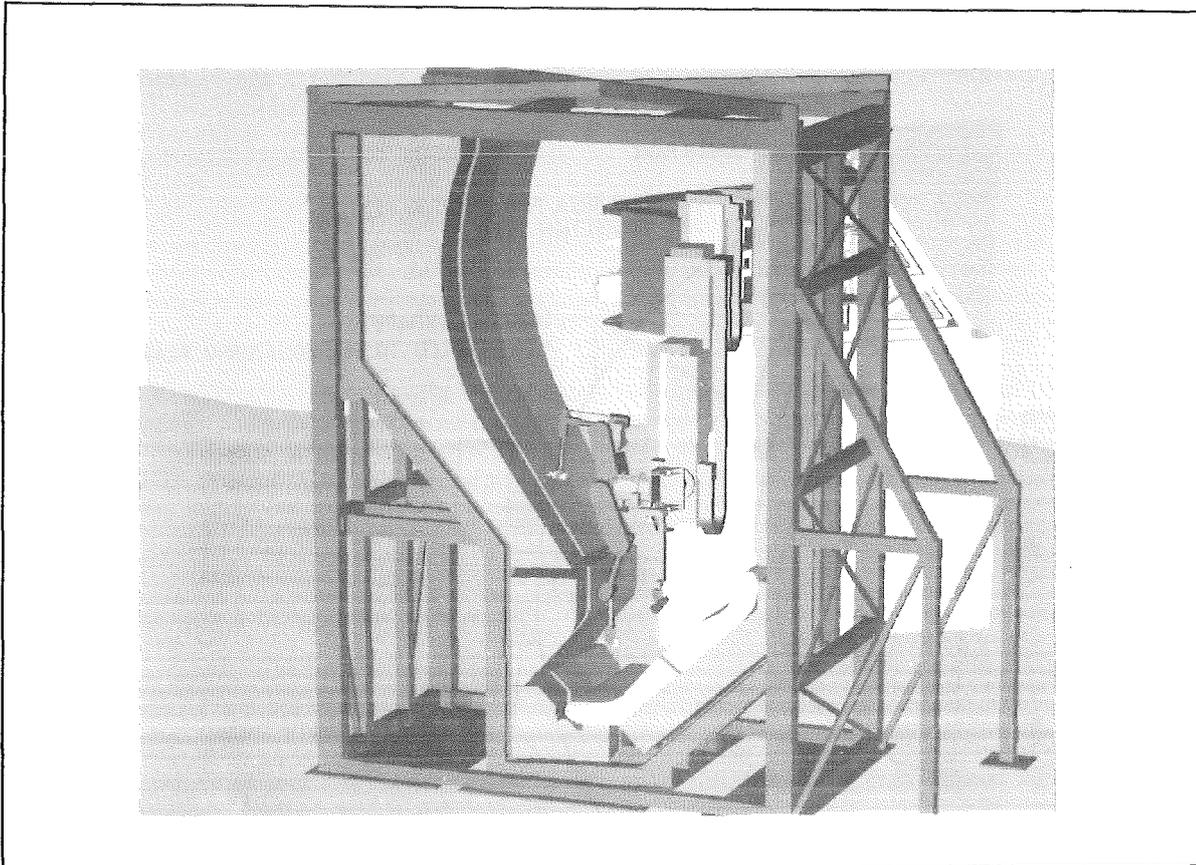
Figure 98. Full Scale Mock-up with EDITH. Tile handling in the lower part of a torus sector

/7/ Suppan A., Hübener, J. Reim, J., Woll, J., Maisonnier, D., Reeve, T.J.: EDITH - A Prototypical Articulated Boom System for NET/ITER, 17th Symposium on Fusion Technology, Roma Sept. 14-18, 1992, Fusion Technology 1992, Vol.2, pp.1599-1603

/8/ Ognibene,I., Sanguinetti, G.P.: Coaxial Drive Units for EDITH-Test Report, Job Nr. T32000, 08.07.92

/9/ Rieke, M.: Statische Berechnung der EEPU, 15.01.93

/10/ Krieg, R., Wehner, E., Hübener, J., Suppan, A.: Design Optimization of an Articulated Boom for NET In-Vessel Handling Unit, Trans. 10th Int. Conf. on Structural Mechanics in Reactor Technology, Vol. N, Anaheim, Cal. 1989, pp.91-96

/11/ Szabo, I.: Höhere Technische Mechanik, Springer Verlag, Berlin, Göttingen, Heidelberg 1960, pp. 23 and 278

/12/ Browne, M.L., Bombi, F.: Functional Specification of the NET Control System. NET N/R/6000/4/A, 26.09.1989

/13/ Browne, M.L., Bombi, F.: Operational Requirements of the NET Control System. NET N/R/6000/5/A, 26.09.1989

/14/ Leinemann, K.: Tele-operator support for fusion plant maintenance: The EDITH remote workstation. Fusion Engineering and Design 18(1991) pp.517-523

/15/ Leinemann, K., Kühnapfel, U., Katz, F., Knüppel, H., Krumm, H.-G., Olbrich, W.: KISMET based multi-media workstation for operational support in fusion plant maintenance. 17th SOFT, Roma, September 14-18, 1992, Fusion Technology 1992, Vol. 2, pp.1579-83

/16/ Benner, J., Leinemann, K.: Supervision and automatic control of robotic systems in nuclear environments. Proc. Third Intern. Conf. on High Level Radioactive Waste Management, Las Vegas, April 12-16, 1992, Vol. 1, pp.966-973

/17/ Suppan, A.: Unpublished report, KfK, March 1988

/18/ Leinemann, K.: Unpublished report, KfK, November 1988

/19/ Leinemann, K.: NET Remote Workstation. KfK-4785, October 1990

/20/ Kühnapfel, U.: Graphische Realzeitunterstützung für Fernhandhabungsvorgänge in komplexen Arbeitsumgebungen. KfK 5052, May 1992

/21/ NET-RHB1-1 task RDD document

/22/ Reisig, W.: Petri Nets, An Introduction. Springer, Berlin, 1982

/23/ Baumgarten, B.: Petri-Netze. BI-Wissenschaftsverlag, Mannheim, 1990

/24/ IRDATA - Schnittstelle zwischen Programmierung und Robotersteuerung. DIN 66313, Teil 1, 1989

/25/ Kühnapfel, U., Leinemann, K., Schlechtendahl, E.G.: Graphical Support for JET Boom Control. Proc. ANS, Pasco Wa., 1987

/26/ Mittelstaedt, M.: Unpublished report, KfK, June 1989

/27/ Kühnapfel, U., Ludwig, A.: Graphics and CAD Support for NET/ITER Boom Control. Proc. 16th SOFT, Sept. 3-7, 1990, London, pp. 1347-1352

/28/ Beitz, W.; Küttner, K.-H. (Hrg.): Taschenbuch für den Maschinenbau (Dubbel), 15. Auflage, Heidelberg 1983

/29/ Kühnapfel, U.; Ludwig, A.: Graphics and CAD Support for NET/ITER Boom Control, Proceedings 16th SOFT London 3-7.9.1990

/30/ Süss, U: Unpublished report, KfK, April 1991

/31/ Süss, U.: Unpublished report, KfK, März 1989

/32/ Holloway, C., Salpietro, E.: Maintenance Concepts for NET. IAEA-TECDOC-495 (1989) pp.75-87

/33/ ISO/DIS 9506 Manufacturing Message Specification, 1988

/34/ Breitwieser, H.; Frank, A.; Süss, U.; Leinemann, K.: Specification of EDITH Motion Control System, KfK 4787, September 1990

/35/ Reim, J.: Dynamic Simulation of the NET In-Vessel Handling Unit, Fusion Engineering and Design 18 (1991), pp. 471-476

Schnitt A-B

Schnitt C-D
(ohne Pos.3 gezeichnet)

Ansicht Z

33 bei Montage gebohrt

Mitte Gliederarm

nicht bemaßte Schweißnähte 5 △ umlaufend
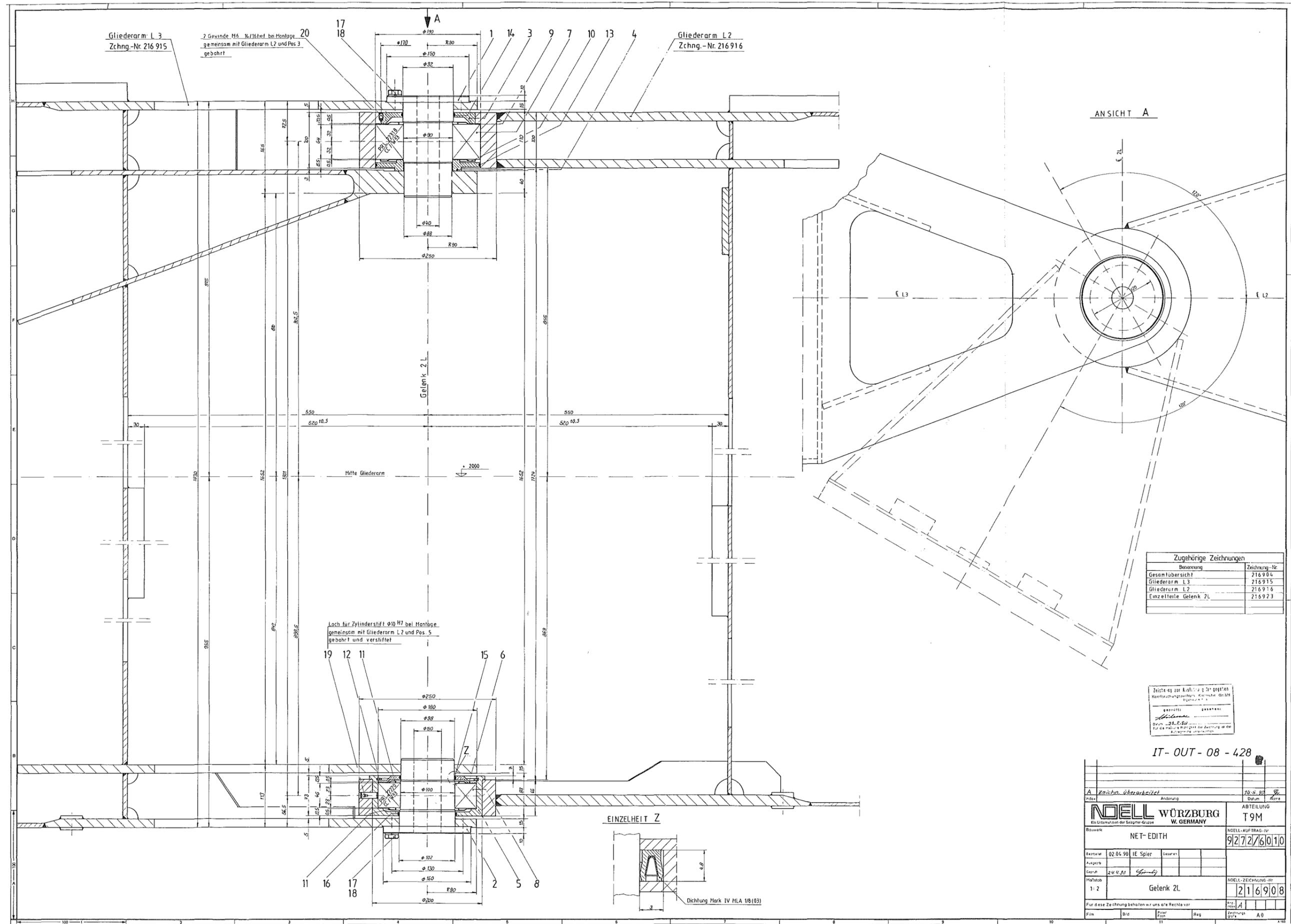
Gliederarm L3
Zchng.-Nr. 216 915

2 Gewinde M6 14/16tief bei Montage
gemeinsam mit Gliederarm L2 und Pos 3
gebohrt

Gliederarm L2
Zchng.-Nr. 216 916

ANSICHT A

Gelenk 2 L

Mitte Gliederarm

Loch für Zylinderstift Ø10 H7 bei Montage
gemeinsam mit Gliederarm L2 und Pos. 5
gebohrt und verstiftet

EINZELHEIT Z

Dichtung Mark IV MLA 1/8 (03)

IT- OUT- 08 - 428

Zugehörige Zeichnungen

Benennung	Zeichnung-Nr.
Gesamtübersicht	216904
Gliederarm L3	216915
Gliederarm L2	216916
Einzelteile Gelenk 2L	216923

NOELL WÜRZBURG W. GERMANY
Ein Unternehmen der Saarpiler-Gruppe

ABTEILUNG
T9M

Bauwerk
NET- EDITH

NOELL-AUFTRAG-Nr
9272/6010

Index	Änderung	Datum	Name
A	Zeichn. überarbeitet	10.5.90	

Bearbeitet 02.04.90 IE Spier
Gepr. 24.4.90

Maßstab 1:2

Gelenk 2L

NOELL-ZEICHNUNG-Nr
216908

SUPPORT
STRUCTURE

CANTILEVER ARM

LINK

END-EFFECTOR
POSITIONING UNIT

Z1   Z2   Z3   Z4

C1   B1   B2   B3   B4

COAXIAL
DRIVE UNIT

MANIPULATOR
HANDLING UNIT

(a) MAX. STROKE ± 5075 mm

12 480

2000   2450   2450   2450   590   1500   ~1040

± 115°   ± 115°   ± 115°   ± 105°   ± 90°

MANIPULATOR
POSITIONING UNIT

0   2000 mm

ARTICULATED BOOM TRANSPORTER (ABT)

25.02.93

IT-OUT-08-446

HOUSING

TORQUE ARM

Kernforschungszentrum
Karlsruhe GmbH

Benennung

Coaxial Drive Unit
Elevation View

Zeichnungs-Nr.
IT-OUT-12-214

EDITH

Zulässige Freimaßtoleranz in mm					
-	6-	30-	120-	400-	
6	30	120	400	1000	
± 0,1	± 0,2	± 0,3	± 0,5	± 0,8	

Maßstab | Gewicht

	1993	Tag	Name
Bearb.		2.6.	
Gepr.			Böhm
Ges.			
P.I.			
Vorhaben			

Zust. | Änderung | Datum | Name | Urspr.

Ersatz für: | Ersetzt durch:

Blatt 1
3 Bl.

MOTOR

Kernforschungszentrum
Karlsruhe GmbH

Benennung

Coaxial Drive Unit
Top - View

Zeichnungs-Nr.
IT-OUT-12-214

Ersatz für:

Ersetzt durch:

Blatt
2
3 Bl.

EDITH

Zust. | Änderung | Datum | Name | Ursp.

Vorhaben

P.I.

Ges.

Gepr.

Bearb.

Werkstoff

1993 | Tag | Name

Böhm

Maßstab

Gewicht

Zulässige Freimaßtoleranz in mm

Zulässige maximale Rauhtiefe Rt in µm

IDLE GEAR

CYCLO GEAR

SPUR WHEEL GEAR

Zust.	Änderung	Datum	Name	Urspr.	Ersatz für:	Ersetzt durch:

Vorhaben
EDITH

Benennung
Coaxial Drive Unit

Zeichnungs-Nr.
IT-OUT-12-214

Kernforschungszentrum
Karlsruhe GmbH

KfK

Blatt 3
3 Bl.

Maßstab

Gewicht

Werkstoff

	Tag	Name
1993		
Bearb.	07.06.	Schneider
Gepr.		
Ges.		
P.I.		

Zulässige Freimaßtoleranz in mm

- 6	6- 30	30- 120	120- 400	400- 1000
± 0,1	± 0,2	± 0,3	± 0,5	± 0,8

Zulässige maximale Rauhtiefe R_t in μm

| — | ∼ | 63 | 8 | 16 | 4 | 1 |

INSTALLATION MEANS
FOR DIVERTOR MODULE

DIVERTOR MODULDE

ATTACHMENT
STRUCTURE

INSTALLATION MEANS
FOR TILES

OUTBOARD SIDE

INBOARD SIDE

2000

SUPPORT STRUCTURE

SECTION A-A

10000

1520

B

A

A

B

4310

SECTION B-B

R 3140

R 8300

OPEN. ACCESS FOR THE ABT

2510

7495

Datenbank ITM1:[LINK]NETDN22A;20 / D-MOCK-UP4-A3-A3
Diese Zeichnung wurde mit CAD erstellt. Änderungen dürfen nur mit CAD durchgeführt werden.

Zulässige Freinaßtoleranz in mm					Zulässige maximale Rauhtiefe R_t in µm							Kernforschungszentrum Karlsruhe GmbH Hauptabteilung Ingenieurtechnik
- 6	6- 30	30- 120	120- 400	400- 1000							KfK	
± 0.1	± 0.2	± 0.3	± 0.5	± 0.8	—		63	16	4	1		

Maßstab 1:75	Gewicht	Werkstoff				
			Tag	Name	Benennung	MOCKUP mit Montagegeruest
		Bearb.	04.06.93	LINK		
		Gepr.				
		Ges.				
		P. L				
		Vorhaben			Zeichnungs-Nr.	Blatt
		IVHU			IT-OUT-14-39	BL.
Zust.	Änderung	Datum	Name	Urspr.	Ersatz für	Ersetzt durch

10 kN
(20 kN)*

10 kN
(20 kN)*

2°

4 kN

4 kN

4,2 kN

4,8 kN

4,2 kN

4,8 kN

900   350

6275

5325

3750

3990

*) Peak loads

Divertor plate

Gripper arrangement

Joint arrangement

WUIF
Trolley
Slide I
Slide II
Support frame

10 kN

1550

4 kN

4,2 kN   4,8 kN

1335   490   100

2450   675

250

290

200

10 kN

Divertor plate

900

4,8 kN

8,2 kN   350

DRIVE UNIT   LIMIT SWITCHES

RESOLVER

WUIF

TROLLEY

CABLE CHANNEL

GLIEDERARM

SLIDES 1 AND 2

SUPPORT FRAME

974
940
195   195
100
600
550
250
500
250
2600
7335
810

210
60
255
1275
129.5   140
127
286
60
1341.5
2017
1302.5
1302.5
105   100
60
60
277.5
293.5
2758
2952
C4
40
1248
R 250
R 250
500
15   15   40
1499

Datenbank ITMO [RIFFEL]IT-OUT-21-40;8 / D-OUT-21-40-A3
Diese Zeichnung wurde mit CAD erstellt. Änderungen dürfen nur mit CAD durchgeführt werden.

Zulässige Freimaßtoleranz in mm					Zulässige maximale Rauhtiefe R_t in µm				
– 6	6– 30	30– 120	120– 400	400– 1000					
± 0.1	± 0.2	± 0.3	± 0.5	± 0.8	—	63	16	4	1

Maßstab	Gewicht	Werkstoff		
			Tag	Name
		Bearb.	25.05.93	Riffel
		Gepr.		
		Ges.		
		P. l.		
	Vorhaben			
	3Z1A			
Zust.	Änderung	Datum	Name	Urspr. IT-OUT-21-35

KfK
Kernforschungszentrum
Karlsruhe   GmbH
Hauptabteilung Ingenieurtechnik

Benennung
END-EFFECTOR POSITIONING UNIT

Zeichnungs-Nr.
IT-OUT-21-40

Blatt
BL.

Ersatz für   Ersetzt durch

EMSM3

MONTAGE FLANSCH
(ATTACHMENT FLANGE)

DREHEINHEIT
(ROTATION UNIT)

SCHULTERSTÜCK
(SHOULDER PIECE)

VERLÄNG. ARM
(EXTENSION ARM)

ADAPTEREINHEIT
(ADAPTATION UNIT)

EMSM2-B

Bewegungen Movements		Bereiche Range	
1	Oberarm neigen Shoulder pitch	+65°	−225°
2	Oberarm drehen Shoulder roll	+210°	−210°
3	Unterarm neigen Elbow pitch	+225°	−70°
4	Unterarm drehen Elbow roll	+210°	−210°
5	Hand neigen Wrist pitch	+60°	−210°
6	Hand drehen Wrist roll	+ ∞	− ∞
7	Greifen Gripping	150 mm	
8	Gehäuse neigen Casing pitch	+210°	−210°

Bewegungen Movements		Bereiche Range	
Z	Oberarm neigen Shoulder pitch	+45°	−45°
X	Oberarm drehen Shoulder roll	+45°	−45°
Y	Unterarm neigen Elbow pitch	+45°	−45°
	Unterarm drehen Elbow roll	+180°	−180°
	Hand neigen Wrist pitch	+50°	−135°
	Hand drehen Wrist roll	+180°	−180°
	Greifen Gripping	90 mm	

595

445

847,9

618,5

CYCLO DRIVE

MULTI TURN RESOLVER

AC-SERVO MOTOR

±90°

K1

MHU