Development and Programming of Portable Robot Systems for Material Handling Tasks

Christian Brecher, Ben Schröter, Carlos Almeida,
Aachen University, Laboratory for Machine Tools and Production Engineering (WZL)

Abstract To improve the flexibility of industrial robots for material handling, a portable robot system has been developed. The robot system can be easily installed at machine tools. For the generation of robot programs a programming system is introduced that combines the traditional approaches of online- and offline-programming in order to allow fast and easy reprogramming of material handling tasks. The main feature of the programming system is an intuitive user-interface that can be operated without special qualifications.

Keywords: Robotic, Programming, Handling

1. INTRODUCTION

Supplying machines with workpieces is a task that even nowadays is largely carried out manually, especially in small and medium-sized enterprises (SMEs). This is mainly due to significant investment costs for dedicated automation solutions going along with insufficient utilization ratio in common production environments. In order to open up new potentials for automated material handling processes in SMEs the development of innovative and flexible systems for material handling purposes has been commenced. The development goes along with the demand for universal robot platforms that can be reused for many different product generations [1].

Within the German research project Porthos [2] the Automation Research Group of the Laboratory for Machine Tools and Production Engineering (WZL) at Aachen University develops a flexible, easy-to-use and portable robot system for material handling tasks. This system is easily movable from one machine or station to another, it is intuitively programmable by a graphical user interface and adapts semi-automatically to new tasks. Sensors monitoring intrusions into the working envelope will guarantee personnel safety in the vicinity of the system. A potential operation scenario is presented in fig. 1, where a single robot system is able to work on three machine tools with low utilization ratios.

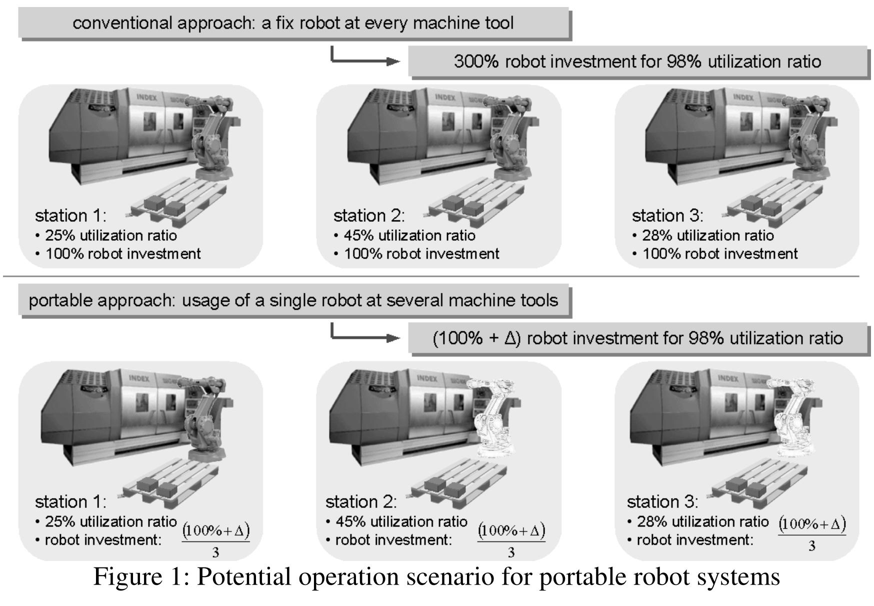

Figure 1: Potential operation scenario for portable robot systems

Such a flexible system lifts the constraint of installing specific automation solutions at each of the machining stations one plans to automate. This opens the prospect of using a single portable material handling system at any number of machine tools according to where the production bottleneck might occur [3]. After an initial ramp-up at each machine tool, which sometimes comprises some modifications at the machine, the robot is ready for operation. The result is a very high degree of utilization ratio of the portable material handling system. Additionally the key to acceptance by the operators will be the simplification of operating the system through the graphical user interface. In many cases these two points will decisively influence the economic viability of an automated material handling solution. The research and development project Porthos is partly funded by the German Federal Ministry for Education and Research (BMBF) within the national "Research In Production For Tomorrow" Program and managed by the Project Agency for Production and Manufacturing Technologies (PFT), Forschungszentrum Karlsruhe.

2. DEVELOPMENT OF PORTABLE ROBOT SYSTEMS

Nowadays industrial robots cannot be applied for handling small lot sizes in a profitable way because of the high investment costs and installation efforts. In order to open up this new market for industrial robots the robot's flexibility has to be increased, especially in terms of portability. Furthermore there has to be an easy-to-use software tool that guides the operator through the whole installation and programming process. Within the research project Porthos a robot system is developed that meets those demands of SME. The project includes the following objectives:

- increased portability of robots
- intuitive software tool for fast generation of robot programs
- flexible workpiece handling by developing multifunctional grippers and sensor integration
- sensor-based monitoring of the robot's workspace to guarantee personnel safety

2.1 Portability of the robot system

The portability of the robot system allows the operator to apply the robot at different machine tools, according to the demand (see fig. 2). The robot can be transported by means of a hand or fork lift. The physical installation to the floor is realized by mechanical elements at the portable platform that can be locked with corresponding elements in the ground. There is an option of realizing a self-centering connection between the robot and the floor. Otherwise the exact position of the robot has to be acquired with appropriate sensors for calibration.

The counterparts, that are fixed in the floor, can be closed by screws at ground level. So the coupling element cannot be affected by dirt and does not form an obstacle for other vehicles that are used in the shop floor.

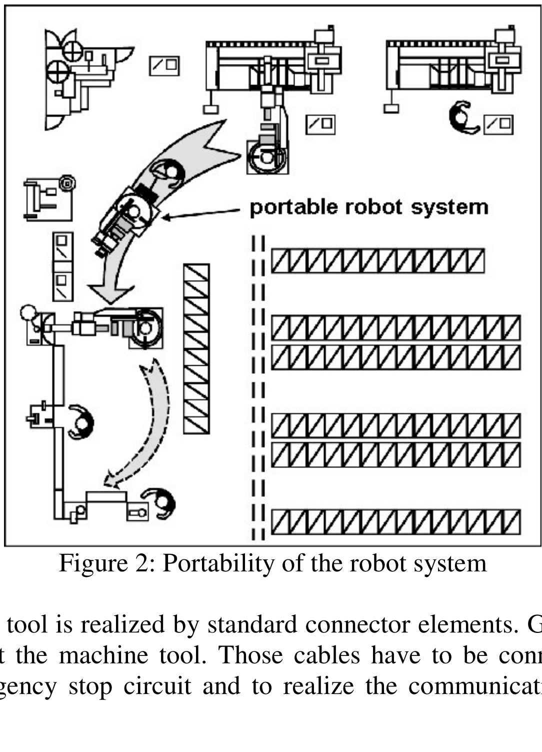

Figure 2: Portability of the robot system

The connection to the machine tool is realized by standard connector elements. Generally there will be only two cables that have to be installed at the machine tool. Those cables have to be connected to the robot in order to integrate the robot into the emergency stop circuit and to realize the communication between the robot and the machine tool, e.g. to operate doors and chucks or to start the manufacturing process. Before the initial installation at a machine tool, its PLC has to be prepared for the communication with the robot control.

2.2 Ramp-up and programming

The work and time required for the initial application as well as for the change of the location or the handling task have to be reduced significantly in order to keep the production downtimes to a minimum. The portable platform allows an easy reinstallation in front of another machine tool. After that the calibration and programming of the robot system takes place. In the case of initial applications at new machines a ramp-up procedure has to be carried out. As SME often lack of staff with deeper knowledge in the field of robotics, all working steps for the ramp-up need to be comprehensive and intuitive.

The reinstallation at machine tools where the robot system has already been applied before should be applicable by the machine operator. He can reuse old, adapted programs or create new definitions of handling tasks in a task-oriented way. Therefore it is sufficient to describe material handling processes in an abstract, generally understandable manner (e.g. “Place workpiece in machine tool” or “Pick workpiece from pallet”).

A more detailed description of the programming concept and its implementation and realization will be outlined in the following section.

2.3 Flexible workpiece handling

The flexibility of the robot system is not limited to its portability, i.e. the flexibility concerning its application sites. Another important factor is the ability of performing different handling tasks. The robot system has to be able to handle different workpieces in different surroundings (e.g. different machine tools, different types and positions of pallets etc.). These demands require both sensor based acquisition of the surroundings and adaptable solutions concerning the gripper.

Hence, a modular servo gripper has been designed and realized that automatically can adapt to a certain magnitude of workpieces (see fig. 3). The optimal positions for grasping the workpieces are calculated based on the given sensor data. Basically there are different types of sensors that can be applied to determine the actual grasping positions. A universal interface between the sensor and the robot control has to be implemented in order to modularly apply the optimal type of sensor for the given application or workpiece. Major criteria for the selection of an appropriate sensor are amongst others the robustness to environmental conditions, accuracy, size, measurement duration, as well as initial and operating costs.

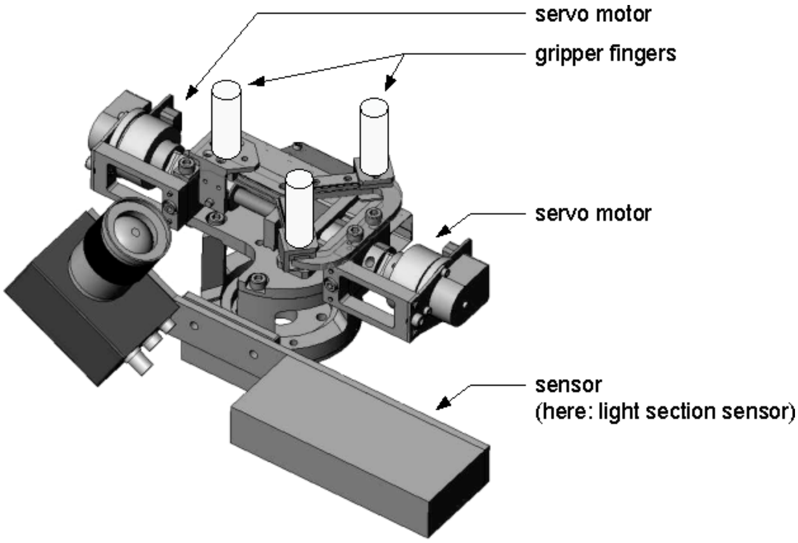

Figure 3: Servo gripper with integrated light section sensor (source: ABB)

The calibration of the robot or of important parts of its working environment is realized by the installed sensors, too. The calibration process has to be performed, when the robot system is installed without a self centering connection to the ground. Another reason can be the manual exchange of full or empty worpiece pallets. In this case it cannot be assumed that new pallets are positioned with appropriate accuracy. After a two dimensional determination of the new position the affected robot data can be corrected in order to guarantee an accurate operation of the robot. This task does not have high demands on the sensors. So even simple mechanical sensing devices can be applied for a precise calibration of components.

2.4 Personnel safety

Traditional safety equipment has to be permanently installed to the ground and separates the working area of the robot physically from the operators. This approach would require safety fences around every application site of the robot system. The result would be a loss of flexibility and high investment costs for the fences. To meet the demands of SME regarding safety systems, a sensor based solution has been installed on the portable platform. Within the Porthos approach, laser scanners monitor the borders of the working area and stop the robot as soon as someone tries to enter this area.

Fig. 4 shows a possible solution to implement such a portable safety system. Each laser scanner is freely programmable. The operability of the laser scanners is monitored automatically by acquiring the contour of the floor and other objects within the measuring range. The Porthos project aims for getting a certification for the developed safety system. After that it will be approved for the application in the shop floor.

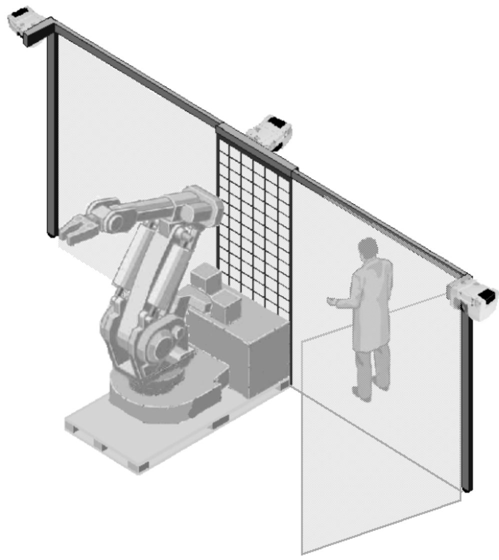

Figure 4: Implementation of the sensor based safety system (source: SICK)

3. SPECIFICATION OF THE PROGRAMMING SYSTEM

3.1 State of the art in robot programming

Robot programming systems can be distinguished into online or offline programming systems. The *Online programming* is performed with support of the robot. First, the program body has to be written without any position data. This working step requires deeper knowledge in the given robot programming language. The second step is the acquisition of the position data by moving the robot to the appropriate positions and copying the point information into the empty program body. However, the generation and testing of those robot programs can turn out to be very time and work consuming.

Offline programming methods hold other disadvantages. Before the user can start generating a robot program, a complete geometric model of all objects within the robot's working range has to be built up. Especially if the robot has to be integrated into the existing shop floor infrastructure, it cannot be assumed that usable CAD data is available for all machine tools and other relevant objects. Above all the application of offline programming systems leads to a poorer quality of robot programs because of differences between the geometric model and the real environment of the robot.

3.2 Specifications for an efficient programming systems

Based on the disadvantages of common robot programming systems, the specifications of a programming system can be derived that meets the demands of SME with varying material handling tasks.

- *Operability without any special qualifications*
The daily operation of the robot system, i.e. the reinstallation at already prepared machine tools and the generation of robot programs, has to be carried out by an operator with limited experience in the field of robotics. The working stages that require more technical knowledge, e.g. the ramp-up at new stations, have to be temporally separated from the daily operation.
- *Adaptability of the programming system to specific conditions on the shop floor*
To guarantee a fast ramp-up at new application sites, the programming system has to be expandable in terms of specific components that are used in the given environment. Examples for those components are grippers that demand certain signals for operation, or special pallets that are used in the given company. An easy integration of such user-defined components is essential for the acceptance of the programming system.
- *Time optimized acquisition of position data*
As the robot system shall be easy to integrate into long existing shop floors, it cannot be assumed that CAD data is available for all relevant machines and objects. Therefore, position data has to be acquired via teach-in. The accuracy and the completeness of the position data has to be assured, as they are essential for an error-free automatic generation of robot programs. Nevertheless, the duration of teach-in procedures can be seen as down-time and thus has to be minimized.
- *Integration of communication cycles*
Almost every material handling task contains the exchange of control and synchronization commands between the robot control and peripheral components, e.g. for operating machine doors or starting the manufacturing process. To be independent from the type of machine tool, the communication cycles should be limited to digital inputs and outputs instead of using any proprietary protocols.
- *Sensor integration*
In the case that either workpiece depots or the workpieces are not placed at designated positions, fast algorithms have to be realized to calibrate cell components or to identify and localize workpieces. The sensors, which carry out these tasks, should be set up and programmed by the programming system. Therefore, the goal is to develop methods, which simplify the integration of external sensors significantly, offer an intuitive programming of the robot-sensor-system, and ensure a reliable execution of sensor-guided, adaptive material handling tasks.
- *Task-oriented generation of robot programs*
A grouping of movement instructions in task-oriented operations, e.g. “Pick workpiece from chuck”, can be a forward-looking possibility for simplifying the programming system. The whole handling task can be easily composed of such operations in a generally understandable way. The visualization can be realized by demonstrative flow charts.
- *User support*
The user input should be checked for plausibility whenever possible. Examples for plausibility tests are the composition of handling tasks (each pick operation is followed by a place operation) or during the ramp-up (no finishing of the ramp-up procedure if position data is missing).

4. CONCEPT OF THE TASK-ORINTED PROGRAMMING SYSTEM

4.1 Modeling of the robot cell

The presented programming system is based on a two-stage conception (fig. 5). In the first stage a data model of the robot cell is created, that contains all information necessary for generating robot programs. It is composed of pre-defined library elements that can be adapted to the real cell by varying parameters. The library is easily extendable by user-defined elements, e.g. special grippers, workpieces, or workpiece depots. During the modeling stage the operator is guided by the programming system in order to guarantee a complete modeling of the cell. After finishing this stage the data model encompasses task-independent information about the cell layout, target positions, movement parameters, and interaction with other machines. The whole procedure of modeling the robot cell can be split up into the following working steps:

- Selection of cell components from library file
- Adjustment of the cell component’s parameters
- Definition of inputs and outputs
- Modeling of communication cycles
- Acquisition of position data
- Optimization of movement parameters (e.g. speed, acceleration)

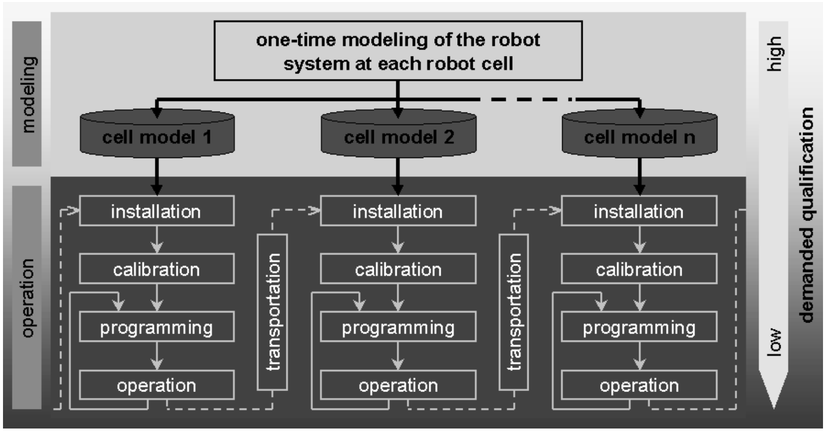

Figure 5: Two-stage programming concept

In contrast to real offline programming systems that require extensive geometric data, for the components of the presented programming system the user only has to enter the absolutely mandatory data. The amount of this data corresponds with the type of component (see fig. 6). Altogether there are four types of components: A robot, a gripper, one or more workpieces and one or more workpiece depots. In this context the type "workpiece depot" also comprises machine tools. E.g. a turning machine is reduced to its chuck, the relevant part of the machine when performing a material handling task. The communication cycles are also assigned to the corresponding workpiece depot.

Figure 6: Components and their assigned data

The most time consuming part of the modeling stage is the acquisition of the position data. As the programming system is limited to the programming of material handling tasks, the operations the robot can perform are limited to pick and place operations. So all positions for every pick and place operation at each workpiece depot have to be stored in the data model of the cell. In many cases the number of points to be taught can be reduced by reusing the points of the pick operation for the place operation or by applying the same points for the movements to and from the grasping position.

When the user starts teaching the cell he can freely decide about these reductions of the point acquisition. After that a teach instruction is generated. The teach instruction is a robot program that guides the operator step by step through the teach-in process. During teaching the points, the operator can define the amount of points needed for the specific operations. So both, complex operations at e.g. turning machines and simple components like flat pallets can be taught with the same tool. Finally, the position data is automatically transferred from the robot control into the data model.

4.2 Task-oriented programming

As all relevant data for generating robot programs has been stored in the cell model during the modeling stage, the programming stage can be reduced to a generally understandable description of the material handling task. Each workpiece depot has a number of operations that can be put together to the handling procedure in a graphical-interactive way. The structuring of the program flow can be realized by the usage of elements, such as loops, wait functions, or if-then-else statements. The conditions of those elements can consist of digital inputs or outputs of the robot control as well as of the number of workpieces in the depots. The filling, emptying or exchange of workpiece depots can be integrated into the program flow, too. In this case, after the robot stops, the operator is informed on which depot has to be loaded or unloaded. After finishing the loading work, he has to approve or correct the new number of workpieces.

5. REALIZATION OF THE PROGRAMMING SYSTEM

The efficiency of the presented concepts has been validated by implementing a first prototype of the programming system. Meanwhile this prototype offers almost all functions that have been designed during the conception phase.

5.1 Modeling user interface

All components of the cell are displayed via the modeling user interface. To each component a contextual dialog can be shown that offers the possibility to visualize and edit all data concerning this component (see fig. 7, left part). The communication cycles, the so called handshakes, can be defined by using this dialog, too. Next to those component depended functions the user interface of the modeling stage also offers buttons, e.g. to generate the teach instruction or to start the semi-automated calibration of the robot or certain components of the robot cell.

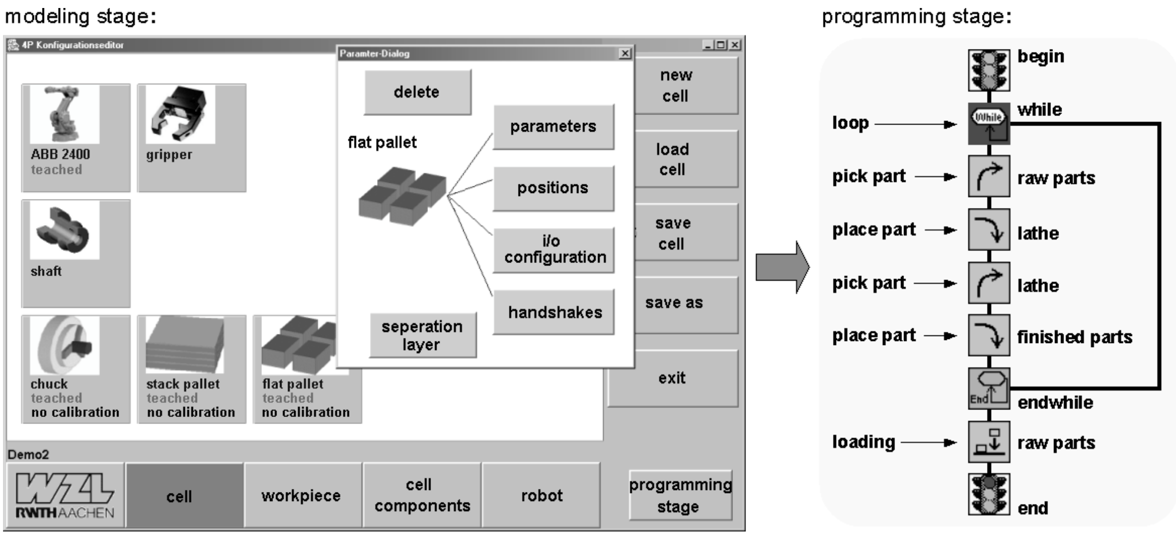

Figure 7: Prototype of the programming system

5.2 Programming user interface

For the programming stage a similar user interface to the modeling stage has been designed and implemented. The function blocks can be added to the program flow (see fig. 7, right part) by simply clicking on them. Some function blocks need further adjustments, e.g. the condition of loop-elements. Those blocks are marked red until all necessary parameters are defined.

Before the program flow can be translated into the robot language, the operator has to describe the initial status of the robot cell. The programming system has to know, how many workpieces are placed in the modeled workpiece depots. This order related information normally changes quite often. Therefore it has to be checked and corrected if necessary.

5.3 Application example

The prototype of the programming system has been tested at a real world robot cell (see fig. 8). It consists of two fenced boxes for the workpieces (pots) and a rack between the boxes to temporarily putting down workpieces. During the modeling phase the operator has to pay special attention to the boxes. Firstly, there are sheets of cardboard as separation layers between the different pot levels. Secondly, the pots have to be grasped with different orientations in order to avoid collision between the gripper and the fence of the box. Both peculiarities of this kind of workpiece depot can be modeled easily by few mouse-clicks. However, it is mandatory for a successful modeling that there is a gripper attached to the robot that can handle both, the main workpiece and the separation layer.

The complete modeling of this cell can be done in less than 15 minutes as the design of the workpiece depots are rather simple, i.e. the pick and place operations can be performed with a minimum number of different positions.

Furthermore, there are sensors attached to the gripper, that can detect the actual position of the workpieces. Based on the sensor data the workpieces can be securely grasped even if they are not on their programmed position. So it is sufficient to teach the pick operation of one workpiece. The robot positions for the other workpieces and for the place operations can be calculated, as the approximate palletizing pattern is known and stored in the data model of the cell.

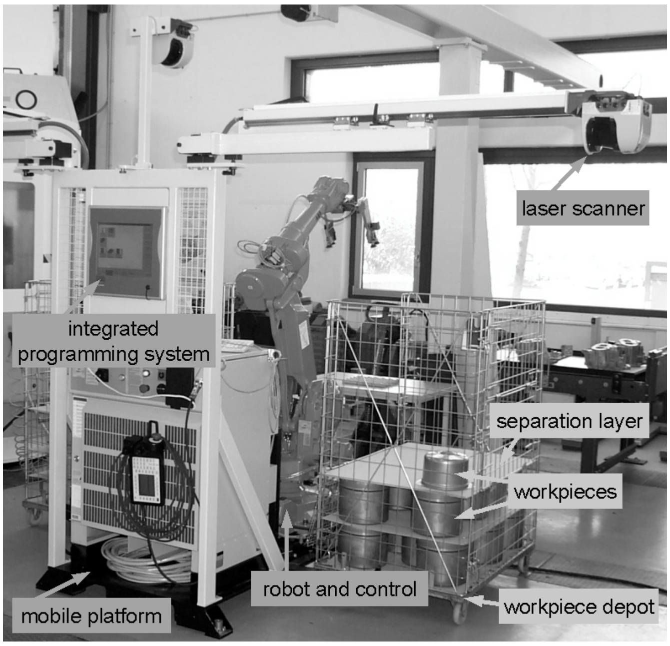

Figure 8: Setup of the Porthos robot system

6. CONCLUSION

The application of industrial robots in small and medium-sized enterprises requires the development of flexible and easy-to-use robot systems. The robots have to be portable and must be able to adapt to different application sites or material handling tasks. The time and work required for reinstalling the robot system have to be low and the operation of the system must be feasible for machine operators. By fulfilling these demands, a cost effective application of the robot in SMEs with varying lot sizes and partly low utilization ratios of machine tools can be achieved.

The presented project “Porthos” aims to meet the demands of SMEs by integrating appropriate flexible, mechatronic components into one single robot system. The mechanical setup is complemented by a graphical programming system that guides the user through the whole ramp-up procedure and reduces the efforts for generating robot programs significantly. The programming system is based on a two-stage approach. In the first stage a data model of the robot cell is created, that contains all information necessary for generating robot programs. After finishing this stage the data model encompasses task-independent information about the cell layout, target positions, movement parameters and communication cycles with peripheral machines. The second stage is the so-called operation stage. Its main task is the generation of the robot program. This can be done in a task oriented way by the graphical composition of operations like e.g. “Pick workpiece from chuck”. So the programming itself is reduced to an abstract and therefore intuitive description of the robot task.

Both the robot system and the programming tool have been built up as a prototype. The next step will be the evaluation of the prototype in a real shop floor environment. This application will lead to a better determination of the efficiency, both technological and economical, of the Porthos robot system.

REFERENCES

[1] N.N., 2004, Key to the market 2003, VDMA Robotik + Automation

[2] Porthos web page, URL: http://www.porthos-roboter.de

[3] Gottschald, J., 2001, Place&Play Roboter – Ein portables Handhabungssystem für die Werkstatt (Place&Play robot – A shop floor oriented portable handling system), Ph.D. thesis, RWTH Aachen University