When Kuka and STW teamed up to develop a new type of monitoring for a floor conveyor without a driver’s cabin, they made sure to include a safety-orientated freely programmable controller. The result was the Kuka OmniMove, an extremely agile type of drive for a floor conveyor. With its electrical drives, it provides unlimited manoeuvrability in all directions, as well as rotation on the spot. Through its omnidirectional drive wheels, the vehicle may be freely navigated and steered in all directions by remote control, even in the narrowest spaces, and boasts a positional accuracy of up to ±1mm. Compared with conventionally steered wheels, the logistics area required can be reduced by up to 50% and the production area correspondingly increased.

The OmniMove wheel has a special rim with vaulted rollers mounted on the periphery. Positioned at a 45° angle when the wheel is driven, two power components result that are parallel and perpendicular to the drive axis (Figure 1). Through diverse turning directions of the wheels, some of the power components compensate for the different wheels on the vehicle chassis while others control the movement of the trailer. Therefore, in addition to straight, sideways, and diagonal travel, it is possible to move in any curve or rotation around a freely defined middle point. Figure 2 depicts the concept for the transitional movement.

The Kuka OmniMove vehicles for internal logistics will effortlessly lift loads of up to 100 tonnes. There are 10 different vehicle models with numerous customer-specific options to meet any requirements. Anywhere from four to 20 drive wheels, as well as numerous additional load wheels, can be configured in order to facilitate the transport of extremely large and unwieldy construction components.

Flexible controller
To develop such a machine, Kuka needed to find a control system that would support many types of vehicles as well as customer-specific requirements. For years, Sensor-Technik Wiedemann (STW) has offered fast and easily adaptable solutions with its ESX family of controllers, which have proved themselves in many applications, particularly in agricultural and construction machinery, etc.

STW has continued this successful series with the ESX-3XL 32-bit controller. The ESX-3XL was conceived to provide an even higher level of flexibility: with six internal sockets for expansion boards, 84 of the 162 possible connector pins can be configured with distinct functions, from simple I/O but extending to encoders, special I/O data storage, and communication interfaces. The mainboard of the ESX-3XL also offers highly software-configurable functions. The 28 inputs are all 12-bit multifunctional (MFI) – in other words, their function can be completely defined by the application programmer in software.

Single callable functions allow the user to specify the input as analogue current or voltage input (measurement range 5V, 10V or 40V), or as digital/rpm (frequency). Furthermore, the 3XL software library enables an application-specific function to be called when the input signal changes (callback function). The rpm signal covers frequencies from 0.6-20kHz. Additionally, filtering may be applied to the MFI signal through software, with configurable debounce times and low pass filters. All the outputs can produce PWM signals, are protected against short circuits and are controllable via 12-bit current measurement and static feedback.

The core of the ESX-3XL is a 150MHz TriCore processor with 4MB RAM, 6MB (optional 34MB) flash memory and 32kB EEPROM parameter memory. The mainboard features four CAN interfaces and a serial (RS232) port. Additional communication interfaces (including...
The risk limit. An acceptable risk level is a situation where the risk is smaller than safety are fulfilled. Safety is defined here as course, that the criteria for functional relevant tasks – though this assumes, of hefty damage fees in applications for safety-of accidents with serious consequences or electronic controls originate?

Does the need for safety-orientated economically viable manner. From where safety requirements may be achieved in an of controller and memory components, grown. But with the increasing integration requirements for functional safety have customer applications has shown that the STW’s recent experience with numerous of expansion boards (Figure 3).

**Functional safety**

STW’s recent experience with numerous customer applications has shown that the requirements for functional safety have grown. But with the increasing integration of controller and memory components, safety requirements may be achieved in an economically viable manner. From where does the need for safety-orientated electronic controls originate?

These controllers act to reduce the risk of accidents with serious consequences or hefty damage fees in applications for safety-relevant tasks – though this assumes, of course, that the criteria for functional safety are fulfilled. Safety is defined here as a situation where the risk is smaller than the risk limit. An acceptable risk level is defined for the application; this risk is usually determined by a combination of the probability of occurrence and the likely severity of the damage.

Functional safety is part of the overall safety of a safety-monitored system that depends on the function of the safety-orientated electronic system, systems with other safety-relevant technologies, as well as external risk factors. The requirements of functional safety must be defined before certification: for example, which safety standards should be applied? These are defined mostly by the application.

The application described here involves the Kuka OmniMove floor conveyor. As this machine can transport loads of up to 100 tonnes, it must therefore meet a set of special safety requirements with respect to braking distance and maximum allowable vehicle speed.

The risk analysis of this application came out of the performance level ‘PL-c’, from the guidelines for machine safety DIN EN ISO 13849-1 that are applicable for the floor conveyor. STW produced the safety requirements and a safety plan for the OmniMove vehicle. The requirements were approved by the Institute for Work Safety IFA in Sankt Augustin, Germany.

In the current project, these conditions are implemented with a safety-orientated software module on the ESX-3XL electronic controller from STW. For that module, the software safety requirements for safety-related application software according to DIN EN ISO 13849-1 apply. The safety architecture is conceptualised based on Category 2 of the standard. The large unit consists of a freely programmable main controller and a test unit – the system supervisor (SSV).

The name of the system developed is OmniMove Speed Surveillance System (OM3S). It has the following core tasks (OM3S):

- **Communication of the current maximum allowed speed**;
- **Calculation of the current first-speed from the available signals of the three wheels**;
- **If the first-speed is too high or any other fault is sensed, an emergency stop must be initiated**.

The system supervisor is also a programmable controller that monitors the main controller by assuming watchdog functionality, monitoring system voltage, testing logical events and thereby allowing the utilisation of the controller in safety-critical applications. The main controller has diagnostic routines that continuously test the entire system in hardware and software and place the system into a safe condition in the event of a failure. The safe condition in this application is the emergency stop.

The Kuka OmniMove vehicles could reach speeds up to 6km/h depending on their construction type. However, the risk analysis has produced considerably lower allowed maximum speeds. In the project described here, there are two conditions for maximum speed limits. The first case is for the load condition. For an empty vehicle, the maximum allowed speed is 3km/h, which is the normal walking speed of a pedestrian (for example, the operator with the remote control unit). But for a loaded vehicle, the maximum allowed speed drops to 1km/h.

The second condition pertains to vehicles that are outfitted with laser scanners which create a protection zone around the vehicle. Laser scanners can bring a vehicle to a stop as soon as they sense an obstacle (e.g. personnel) in its path. There are situations where the laser scanner must be switched off (override conditions), for example driving through a narrow warehouse space. To guarantee safety under these conditions, the allowed maximum speed is reduced to 0.1km/h.

The maximum allowed speeds have been measured so that when the vehicle emergency brakes, it will come to a halt within the maximum allowed stopping distance. Because of the large number of variations of these vehicles, the maximum speeds can be further limited through parameterisation.

**The safety concept**

The safety-orientated software module for speed monitoring performs three tasks:

- Communication of the current maximum allowed speed;
- Calculation of the current first-speed from the available signals of the three wheels;
- If the first-speed is too high or any other fault is sensed, an emergency stop must be initiated.

The name of the system developed is OmniMove Speed Surveillance System (OM3S). It has the following core tasks (functional requirements):
• Reliable recognition of the load condition of the vehicle by redundant load sensors;
• Reliable recognition of the status of the laser scanner protective field (active or overridden) by redundant signal paths;
• Determination of the current allowed maximum speed based on load condition and override information;
• Reliable reading of the encoder signal from the individual drive wheels of the vehicle through redundant sensors with analogue or digital outputs;
• Calculation of the current first-speed of the vehicle from the drive wheel encoders. This can be accomplished by a maths model with the x and y co-ordinates and the instantaneous encoder counts of the three drive wheels;
• Fault recognition through probability analysis;
• Every recognised fault or a speed in excess of the current allowed maximum speed must result in an emergency stop for the vehicle, which is the safe state;
• The monitoring software must be suitable for all possible configurations (geometry, size, number of driven and auxiliary wheels, etc.) of the vehicle.

In order for the OM3S to meet the requirements of safety software with performance level PL-c according to DIN EN ISO 13849-1, the following quality attributes must be fulfilled:
• The parameters loaded in non-volatile memory (EEPROM) must be verified with CRC checksums;
• The working memory (RAM) must be verified against corrupting through suitable means;
• Program flow control must occur via a system supervisor offering watchdog functionality to guarantee the correct program processing.

Among the overall requirements for OM3S is the prerequisite for it to be implemented on the PL-d certified STW safety ESX-3XL controller with integrated hardware diagnostics. The STW ESX-3XL hardware diagnostics have two parts. The first is the start-up diagnostics that occur when the controller is switched on and test all of the basic functions, for example the safety relay. These tests cannot be repeated at a later time because they affect the operation of the vehicle.

Then there are also the periodic tests of memory and CPU for the integrity of content and function, which run in parallel with the actual customer application in the background.

In this project, Kuka already uses the STW safety ESX-3XL controller, on which the vehicle’s controller software runs. In addition, the OM3S will run on this controller. Interfaces to the following external entities exist for the OM3S:
• Control software from Kuka;
• The hardware abstraction layer (HAL) of the ESX-3XL ECM (electronic control unit).

STW development process
The development process of safety software according to DIN EN ISO 13849-1 follows the V-model (see Figure 5).

The specifications of this project are processed exclusively in database-orientated requirements management software. This promotes the atomic capture of individual requirements (individual entities) that are referenced by SD and can be individually tested. This leads to testable requirements and the individual requirements can be tracked through the lifecycle of the application and back to their source.

The individual specifications go through formal reviews by various working groups based on the tough criteria of unambiguity, completeness and testability. The results are reviewed so that they can be processed in the next version.

The system architecture must be designed before the design of the individual software components. An outside software architect must review the architecture and then it must be refined.

Information on architecture reviews
Frequent and early reviews prevent the development effort heading in the wrong direction or into a dead end. Through the process of explaining one’s own design to another person, shortcomings in design will become evident. It will soon be clear whether the underlying considerations are logical and complete.

Reviews also uncover misunderstandings, differences in interpretation or potential
incompleteness of the requirements. Reviews during the design process should be carried out to explain the solution: why it was chosen, what alternatives there were, and why they were not chosen.

In this way, the work of the architect can be reproduced and either confirmed or corrected. The review is an important test that can prevent errors from being propagated in future projects.

There must also be a unit test written for each function that is implemented in a software component. Each software change can therefore be tested completely and instantaneously at the push of a button. This ensures that all previous functions still work error-free after the change.

After the requirements have been put in the specification, they must be evaluated for testability. System tests will then be produced for testable requirements in a system test specification.

A configuration manager will also be assigned to administer the configuration of the system and provide an overview of exactly which versions of the individual development components (specifications, system test specifications, architectural designs, implementation and test protocol) go together.

The importance of the software tests will be emphasised by the attendance of a test manager, who will be assigned by project management. He produces a test management plan, organises the tests and co-ordinates the activities of the software testers. The software testers also have the task of managing defects. The defects must be entered into a ticket system. Once entered, the processing of the defects will be tracked, including their status (such as developer analysing, tester confirmed resolution, etc.).

Once a first version of the specification has been completed and released, work can then begin on the software architecture and system test specification. In practice, work can begin earlier when a stable set of requirements have been identified.

After this, desired changes can only be introduced through a change-management system. Each change request is entered into a ticket system and must proceed through a series of different status conditions before it can be finally approved and implemented.

Once all the conditions of functional safety have been fulfilled, it should be safe to expect the floor conveyor to be able to function without failure.

Availability is an important benchmark for the customer. In the case of the OM3S safety application, STW must increase the availability, because the construction of the OmniMove wheels leads to inherent system vibrations. A good result is reached by building in 'mean filters' on the encoders of the drive wheels. To achieve this result, the project required multiple iterations of test drives of the OmniMove vehicle, along with discussions of the possible technical improvements with safety professionals and certain modifications to the OM3S requirements specifications. An acceptable solution was found from all points of view, including cost.

Full functional safety

The safety-orientated software application fulfilled all aspects of functional safety, maintaining the availability of the system and the flexibility in terms of the large number of variants of the floor conveyor vehicle. Parameterisation was built into the safety considerations to allow future customer-specific developments to be implemented in a simple manner.

A system context diagram (SCD) depicts all external entities that can interact with the system:
- The SCD depicts the system to be built as a black box;
- It graphically depicts its interactions with external entities;
- It determines the flow of information and control between the system and the external entities.

A context diagram should depict in a simple manner exactly what relationship the system has to the surrounding participants. However, it should not attempt to represent the architecture of the system. In this overall view, the system is handled as a black box. The emphasis is placed on how the system relates to its environment, not on its architecture or decomposition.

Christoph Filser is project manager; Rolf Jung is vice manager R&D/functional safety manager; and Tobias Heilig is a computer scientist at STW

CONTACT

www.sensor-technik.de
info@sensor-technik.de