Protocol Validation Homework Assignment: Movable Patient Support for an MRI Scanner

Email your project document, mCRL2 specification and μ-calculus formulas to Bas van den Heuvel (b3.vanden.heuvel@student.vu.nl) and to me (a.ponse@uva.nl)

Deadline: appr. October 20, 2018 (will be confirmed)

The assignment consists of designing a controller for a small distributed and/or embedded system. You are supposed to carry out the assignment as follows:

1. Identify the global requirements on the whole system in natural language. A typical requirement is that 'the patient support system may never make a motorized horizontal movement while in emergency mode'.
2. List the interactions of the control system with the outside world. These are for instance 'turning a motor on', 'reading that the up button is released' and 'applying a brake'. Describe clearly but compactly the meaning of each interaction in words.
3. Translate the global requirements in terms of these interactions.
4. Depict an architecture for the control system.
5. Specify the behaviour of all components in the architecture in mCRL2.
6. Verify using the mCRL2 toolset that all requirements given in item 3 above are valid for your specification.
7. If not, modify your specification (or the requirements), and verify the requirements again.

The assignment must be documented in a technical report that covers all items above. This report should be a concise technical account of the system, written in such a way that from it the requirements, architecture and design, system behaviour, and action interface can be easily understood. It should also be clear how the requirements have been verified, in such a way that this can easily be redone without consulting any of the authors of the report.

The assignment is inspired by medical scanning machines developed at Philips Medical Systems. A novel system at Philips is a Movable Patient Support Platform, MPSP for short, which is a trolley bed on which a patient can lie inside a scanner, see

`http://www.google.com.ng/patents/US9078628`

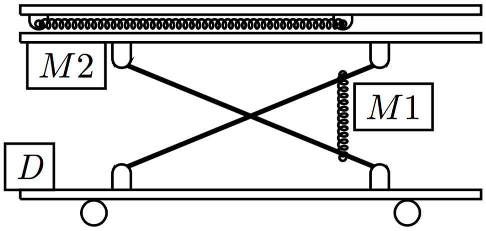

Figure 1: A movable patient support unit.

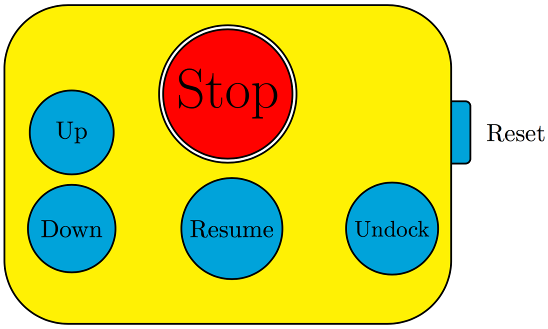

Figure 2: The user console on the MPSP.

The MPSP can be either disconnected from the scanner or docked, i.e. connected to the scanner. Since the MPSP can be disconnected from the scanner, a patient can be prepared in a separate room, while another patient is being scanned. Previously patients were prepared while in or near the scanner, and during this time the scanner must be idle.

There are two motors in the MPSP. Motor $M1$ controls the vertical and motor $M2$ controls the horizontal movement, see Figure 1. Both motors have brakes that can be turned on and off separately from the motors. Horizontal movement is only allowed when the MPSP is docked. When the MPSP is disconnected from the scanner, the bed must always be in the rightmost position (which is detected by a detector), as otherwise the MPSP might tumble over. When disconnected from the scanner, the horizontal brake must always be applied. The vertical brake must always be applied while the vertical motor is off. When a motor is on, the corresponding brake must not be used, as otherwise the motor could overheat.

The movements are controlled via a console on the MPSP, which is designed to be as simple as possible, see Figure 2.

The stop button puts the MPSP in emergency mode. In emergency mode, the horizontal brake must be released, to allow medical staff to manually drag the patient outside the scanner. This may be useful when an emergency occurs (a heart-attack while scanning), or when a system malfunction happens. The resume button puts the MPSP back to normal operating mode.

The undock button can be used to disconnect the MPSP from the scanning device. When the undock button is pressed a message is sent to the scanner which will undock the MPSP. For this a gentle spring mechanism is used that pushes the MPSP away from the scanner.

The undock message should never be sent to the scanner if the bed is not in the rightmost position, as otherwise the MPSP might tumble over.

The reset button is used for calibration. Every scanner can have a different height, generally dependent on how it is installed in the hospital. The MPSP can only be moved inside the scanner if scanner and bed are at the same height, which is called the standard height. Before use, the MPSP must be calibrated by setting the standard height. This is done as follows. The MPSP is docked. This is detected via a sensor in the docking unit D. Then using the up and down buttons the bed is moved to the correct height. By pressing the reset button once, this height is set to be the standard height. If the reset button is pressed while the MPSP is not docked, the standard height is forgotten and the MPSP goes into uncalibrated mode.

When the MPSP is docked and calibrated, the bed will halt when it has reached the standard height. If the up button is pressed at the standard height, the MPSP moves into the scanner. In order to avoid unexpected movements it is important that the up button is released before the inward movement is commenced. This means that releasing the up and down buttons are important interface actions also.

When the MPSP is docked and calibrated and the down button is pressed, the bed moves outward, until an outward horizontal detector indicates that the bed is completely outside the scanner. By releasing and pressing the down button again the bed will subsequently move downwards. While the MPSP is docked and calibrated, the bed cannot be moved above the standard height. When the MPSP is disconnected or uncalibrated, the up and down buttons can only be used to move the bed up and down. The bed is not allowed to move above some uppermost and below some lowermost position. There are two detectors, to detect when the bed is in the uppermost or lowermost position.

The assignment is to design a set of controllers that must operate the MPSP in such a way that no harm can ever be done to a patient or to the equipment. It is a strict requirement that at least three separate controllers are used: one for inputs from the console, one for inputs from the sensors, and one for outputs to the motors and the brakes. Of course these three controllers need to communicate with each other. You can assume that channels are secure (i.e., no messages are lost). The model must be such that qualified programmers can easily implement it. This assignment is underspecified. This means that in certain cases you have the freedom to make your own design decisions.

The actual platform is much more complex than the one described here. For instance, there are at least three emergency modes. And in reality the platform can also be controlled via a console on the scanner, or via an operator on a host computer. In all cases the platform can respond differently. For instance, it is not allowed to move the bed up or down via the host computer. And if the platform is in uncalibrated mode, the bed should only be moved up or down via the console on the platform.