# CAN in the rehabilitation field: M3S

**ir. M.W. Nelisse**
TNO Institute of Applied Physics, Delft, The Netherlands
nelisse@tpd.tno.nl

## Introduction

Not too long ago elderly and disabled people only had a very limited set of technical aids available to them. However in the last couple of years many technical aids became available for the purpose of increasing the independence of these people. An increased independence of this group would have a positive effect on nearly every aspect of their lives, both on personal and vocational activities.

Technical aids for these purposes, like wheelchairs, robotic arms and environmental control systems, are employed to an ever increasing extent. Until now these technical aids are developed in a rather uncoordinated way: every manufacturer has his own set of company specific devices. This results in many products which are not compatible to each other and prevents users from choosing an optimal set of devices fulfilling their needs: they can for example only choose from a limited set of input devices which are available from the same manufacturer as the wheelchair is.

Furthermore these systems offer very limited flexibility, while users tend to have changing needs, especially during revalidation phases and in case of progressing illnesses like muscular dystrophy. Therefore considerable additional costs related to customisation have to be taken into account in the price of the system. Because of these high costs, changes are very often disencouraged and the user has to live (quite often during the whole day) with a system not completely fulfilling their needs.

A next logical step is to present a modular system which integrates these technical aids and which offers disabled people better opportunities to function as independently as possible. The modular construction should allow users to compile a specific package of technical aids to a complete integral system, while still permitting them to extend or modify the system later on, because of changing wishes, needs and/or user environments.

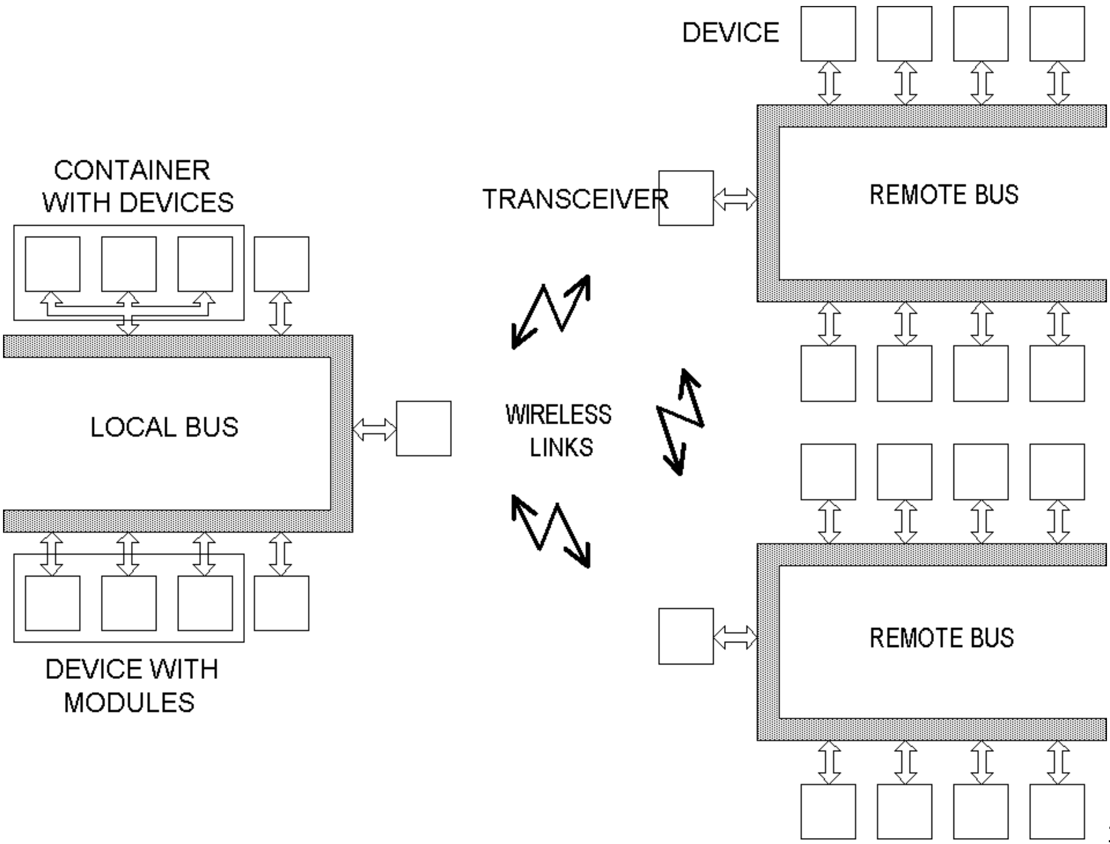
M3S, which stands for Multiple Master Multiple Slave, is the standard in the rehabilitation field which follows this concept. It improves the access to assistive technical devices by connecting different devices in a safe and easy to use way. It not only provides a standard interface between input devices and end-effectors, allowing devices from different manufacturers to be linked in the same system, it also enables the user to operate more end-effectors using a single input device. M3S was primarily intended for use on wheelchairs, however the architecture allowed a wider usage and it is now also used in integration with e.g. computers and homesystems.

M3S was a project in the pilot phase of the European Community research program TIDE (Technology Initiative for Disabled and Elderly people). In this project two demonstration platforms were assembled and evaluated at rehabilitation centres in the Netherlands and Switzerland. Further evaluation of six integrated systems in real life conditions took place in the IMMeDIAte project of the European Community program SPRINT (Strategic Programme for Innovation and Technology Transfer). As a follow-up several other European projects made contributions to the finalisation of the M3S specification. Now M3S is not only used in new European projects, but also the first companies have announced the availability of M3S devices. Other companies have shown interest in adapting their current devices to the M3S standard or already have demonstrated interfaces between M3S and their existing products. The M3S specification is now taken by an ISO committee defining 'a serial interface for wheelchair controllers', as the working draft for further standardization.

## The M3S system

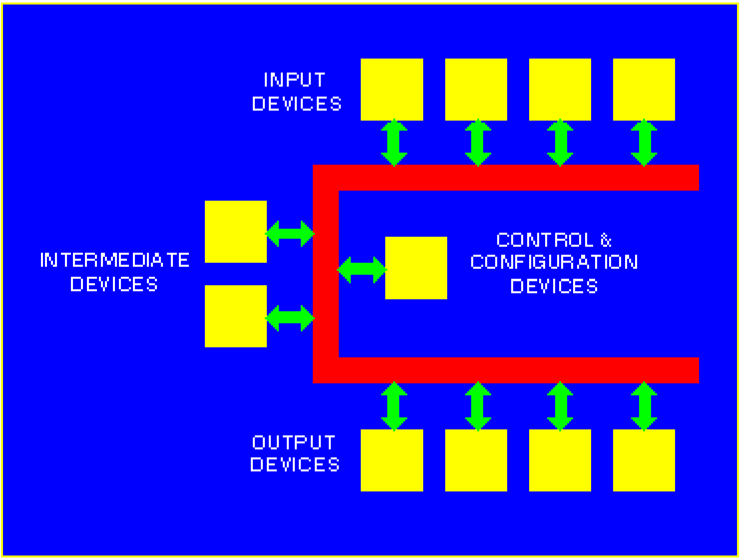
A M3S system is based on a *bus* and *devices*.

The bus contains three parts: 2 lines for digital communication, 2 lines for power distribution and 2 lines for safety features. At the beginning of the M3S project in 1992 a comparison study was made between different fieldbusses for the digital communication to be used in the system. As a result of that the CAN bus was chosen, because of favourable aspects in the fields of noise immunity, error detection & recovery capabilities, availability of silicon and real-time communication facilities. The CAN bus is operated at 250 kbit/s and uses a high speed interface. The power is distributed to all devices via the bus, since in most cases the system is a mobile platform and for weight reasons only a single set of batteries is available. In many industrial systems human operators are kept away from prime moving parts like robots and transport belts. However M3S systems typically include prime moving parts (like wheelchairs and manipulators) in close vicinity to the user, therefore special safety considerations should be applied. Based on the safety requirements set by the ISO committee defining 'a serial interface for wheelchair controllers', another study was started in the M3S project. This resulted in the addition of two safety lines to the bus, a *dead man switch* line and a *key switch* line. These two lines make it possible to stop any prime mover action or to switch off the system from any place.




1

The devices are connected to this bus. A single device may consist of more *modules* all connected to the same bus (eg a manipulator can have a main module which performs a high level control algorithm and several modules for the motors in the different joints which perform low level actions). It is also possible to integrate more devices in a single box (eg combine a display, keypad and speech-synthesizer device in a hand-held box), or even to run more devices on a single processor (eg different application programs running via some kind of multi-tasking mechanism on a single processor), these are referred to as *container* devices. Furthermore it is possible to link M3S systems together via a *wireless medium*. It has been shown in M3S that it is possible to run CAN via a wireless *infra-red* medium. Since it is not possible to extend the concept of safety lines over a wireless link without sacrificing some safety aspects, some restrictions apply to the use of prime moving parts on sections of a M3S bus which are accessed via a wireless link. The wireless link concept makes it possible for users with mobile M3S platforms, to access M3S device which are placed in a fixed environment, like for instance computers and home bus systems.



2

Within this architecture four basic types of devices can be distinguished:

- control & configuration
- output
- intermediate
- input

*Control and configuration* devices deal with configuration functions, control functions and safeguarding functions (plug and play controller, transceiver). *Output devices* or *end-effectors* have primarily an actuator function (eg wheelchair, manipulator, environmental control system). *Input devices* have primarily an input or sensor function (eg joystick, keypad, voice recognition system). *Intermediate devices* have both input and output features (scanner, navigator, obstacle avoidance system).

Each device should have support for some basic M3S mechanisms, this is called the *generic* M3S functionality. It takes care of mechanisms like switching on and off the device, local safety checks, present the device to other devices on the bus, delivery of device configuration information, exchange of information between devices, etc. Devices can also support additional functionalities, like *CCF* (takes care of configuring the system and system wide safety checks), *arbiter* (takes care of allocating numbers to all devices on the bus), *transceiver* (takes care of establishing wireless connections to other M3S busses), *battery* (takes care of supplying power to the bus) and *feedback* functionality (takes care of presenting feedback to the user).

## System integration

In the rehabilitation field it is very important to recognize different kinds of users of the system. Not only on the kind of disability or handicap they have, but maybe much more important in the way they interact with the system. *Impaired users* will use the system as a tool, necessary for their daily activities. A *facilitator* can be someone like a therapist, a helper or a relative, who (together with the patient) will adjust the system to the personal needs and abilities of the impaired user. A *maintenance technician* will perform actions like repairing or replacing devices or just pieces of hardware and/or software. A *system configurer* will compile a system by integrating different products on the M3S bus in a technically correct way, but also satisfying the needs of the patient.

M3S is basically a distributed system which needs a central control and configuration method. The configuration method is based on a generalisation of devices. This ensures that new devices can be added to the system at a later stage without any further adaptions, therefore it must be assumed that an input device knows nothing about an output device and vice-versa. All devices on the bus have internally stored *device configuration* information, which describes the specific behaviour of a device in a uniform way (just like an electronic data sheet).

During start-up of the system, all devices take part in a mechanism called the *Serial Number Arbitration Protocol* (SNAP). The purpose of the SNAP mechanism is two fold. First it is used to identify all devices in a system, and secondly it is used to give each device a unique handle, called a *device number*. The SNAP mechanism completely eliminates the need for hardware jumpers or DIP switches and therefore making it easier to install devices in a system.

Every device has a unique 64 bit SNAP ID number that differentiates one device from another. This ID number contains a manufacturer ID, a serial number and an additional category number. On power up all devices are placed in an inactive state. They arbitrate for the CAN bus and the device with the smallest ID number (lowest message identifier) wins. Because all ID numbers are world-wide unique it is guaranteed that only a single device is *isolated* on the bus at a time. The arbitration is done in a byte oriented sequence, so a single arbitration cycle takes 8 steps. After a device has won arbitration, all other devices drop off. The first arbitration cycle is used to find a master to control the SNAP mechanism, this device is called the arbiter or *CCM*. The CCM will assign itself the first free device number (0). In following cycles the arbiter will repeatedly assign the next free device number to the devices which won arbitration cycles. Once a device has been assigned a device number, it no longer participates in the arbitration mechanism and arbitration of devices with higher ID numbers can proceed.

After the CCM allocated device numbers to the devices, it is able to retrieve the distributed configuration information from the devices and based on this data a *system configuration* can be build. This system configuration describes the setup of the complete system, all actions accessible by the impaired user and the settings which suit the special needs of the patient.

This configuration can be done manually be a therapist on the system itself, but since this is a tedious operation in which the therapist has to interact with the system via devices which are meant for an impaired user, a different approach is more often used. In that situation the device configurations are retrieved by the CCM and then transferred to a computer system (eg a PC running MS Windows). On this computer the therapist defines the mechanisms how the devices may interact within the system (use a joystick to drive the wheelchair, allocate particular switches to switch on or off the television via remote control) and also how the system gives feedback information to the user (eg errors via a speech synthesizer). This system configuration is then stored on the CCM and can from that moment on be used to start the previously defined operations.

Since most system are rather simple systems, containing only a limited amount of devices (eg a joystick, some switches, a wheelchair and maybe an environmental control system), it is possible to use a true *plug-and-play* (PnP) configuration method. In this mechanism the devices are simply connected to the bus and no special manual configuration has to be done. During the SNAP procedure, one of the devices becomes the CCM and all other devices make themselves known to this CCM. The CCM can then retrieve the device configuration from these devices and try to find some matches in the supported operations of the devices. When one or more matches have been found, the CCM starts the devices and the user can work with the system right-away. During user evaluations in some of the European projects mentioned above, it has been shown that using this PnP facility, it is possible to setup a M3S system consisting of a wheelchair, joystick and display device, and have it running within less than two seconds after the system was first switched on (in that case a wheelchair driving).

In case of manual configuration it is possible to speed up the SNAP mechanism, since the CCF arbiter can directly assign device numbers to devices based on information stored in an existing system configuration (only for extra devices not part of the system configuration some additional SNAP cycles will have to be performed). In case of PnP configuration, no system configuration is available and the full SNAP mechanism is performed. Finding a SNAP arbiter takes 180 ms and additional SNAP cycles for configured devices take 450 ì s, while SNAP cycles for unconfigured device take 160 ms.

## Using the system

In case of simple systems, using the system is rather easy (eg moving the joystick results in driving in that direction). However when systems consist of more devices, more combinations between input and output devices become possible and it can be harder for the user the understand which actions will have which results. Feedback can provide the extra information necessary in these kinds of situations.

Therefore M3S supports mechanisms in which the patient, uses his normal input device to select some kind of action from a set of possible actions. After having performed this action, he can select another one without changing the input device needed to do the selection or performing the actions (eg it is possible to use the same joystick to operate a wheelchair, a manipulator and the selection mechanism to choose between these options). As the number of devices in a system grows, the number of options to choose from will also increase. To help the user in ordering these options, M3S supports a hierarchical organisation of the different actions the system can provide.

An additional safety mechanism associated with the key switch line in the M3S bus, makes it possible to determine from which device the system was switched-on. This then allows M3S to automatically move the view on the hierarchical tree to the device where the system was switched-on from (eg it is possible to switch on the system and immediately start driving, without first explicitly selecting this option). This key switch mechanism also gives a simple but powerful possibility for attendants to give assistance to a user needing help.

> An example: A user collapsed and his body lies over a keypad he normally uses to drive the chair. This input device is therefore inaccessible for an attendant. When another input device is present (eg a special attendant's joystick at the back of the wheelchair), it is possible for the attendant to press the on switch at this joystick, automatically move the view of the system to this device and then drive the chair (via this joystick) to a place where help can be given to the patient.

# Safety

Within reasonable levels of cost, complexity and ease of use, systems cannot be 100% safe. The M3S bus concept aims at a high level of safety within a usable system. Impaired users with high levels of disability will often benefit most from using complex systems. However, they will also tend to have the greatest limitations in controlling the system.

### Safety monitoring

The purpose of the safety monitor is to regularly check the safety of the system at all levels and when appropriate to shut down or restrict functions for specific parts of the system. The safety monitor mechanism is based upon a global safety monitor in the CCM: the *system safety monitor*, a local safety monitor in each device: the *device safety monitor* and additional safety monitors in the transceivers of a link when a system contains local and remote busses: the *local link safety monitor* and the *remote link safety monitor*.

In order for the system to remain active, the system safety monitor should monitor the overall system safety and should receive confirmation from all device safety monitors that the devices are in a safe operating condition. This is achieved via periodic status requests and responses. No device should be activated without a prior safety check. This requires a response from the device safety monitor indicating a safe state.

Each device safety monitor ensures the safe operation of its own device. The device safety monitor verifies the periodical safety checks from the system safety monitor. If the requests are interrupted, a time out will occur and the output device safety monitor should then execute an immediate stop. Device safety monitors on prime movers will shut down the motor power circuitry directly and so give a rapid response to serious errors.

The local and remote link safety monitors ensure the correct operation of the communication mechanism on the link. Due to the time delays between the local and remote bus (caused by the link), an emulation of the periodical status check procedure is necessary over the link. The remote link safety monitor must therefore also be capable of carrying out all immediate safety actions. If the link is broken, there is no CCM on the remote bus anymore and the remote link safety monitor itself must carry out a shut down procedure on the remote bus.

### Safety equations

Since M3S is an integrated but distributed system, it is possible that devices can put the system or parts of it in a potentially dangerous position. To be able to handle these kinds of situations, the concept of *safety equations* was introduced in M3S. Devices which can detect potentially dangerous situations, can signal *safety events* to the CCM. Together with safety equations defined in the system configuration, the CCM will respond on these events by signalling *safety restrictions* to other devices in the system. These devices will now be limited in their usage.

> For example a seatcontroller can signal a safety event when the seat is not down, a manipulator when it is not folded in or a wheelchair when it is driving at high-speed. Safety restrictions can limit a seatcontroller to move the seat up, to fold out a manipulator or to drive a wheelchair at high-speed.

Both the safety events and safety restrictions are boolean type of signals. The relationship between the safety events and safety restrictions is specified in safety equations. These safety equations are defined in what is called 'a sum of products' form. This form describes the formula, how to calculate the equations using the boolean operators AND, OR and NOT.

Some examples of safety equations, where the left side of the '=' sign defines the safety restriction and the right side defines the boolean operations on the safety events.

AllowSeatUp = NOT SpeedIsHigh
AllowHighSpeed = SeatIsLow AND NOT ManipulatorBeingUsed
AllowManipulatorUsage = NOT SpeedIsHigh

**Key switch function**

The key switch function can be operated by the user or a helper and is meant to turn on or off the complete system. This also allows the system to be stopped immediately in case of an emergency. The key switch functionality consists of one or more key on and off switches, a hardwired KEY line as part of the M3S bus and some circuitry on all devices to interrupt the power supply to the device.

The key on switch is a non-latching physical switch meant to turn on the complete system. The key on switch is associated with a key off switch which can be integral or independent. The key off switch is a physically operated switch, but the key off switch can also be operated by the CCM on request of the user or when major faults occur. The KEY line provides an enable signal to the local power supply circuitry on the devices. No device should be able to operate unless the KEY line is active. The KEY line is connected to the battery plus via the key on switches in the input systems. Pulling the KEY line to battery minus should turn off the local power supply circuitry and hence turn off the system.

Although the key on switch is a non-latching physical switch, the key switch circuitry should provide a latching function. A key switch status signal at the input systems is used to check the latch status. Where duplicate key on switches are provided, only one should be active at any one time, which means only one key switch is latched. The CCM can monitor the key switch status and ensure that only one is active. This guarantees that the system can be turned off with any key off switch in the system, independently of the actual key on switch used to turn on the system.

**Dead man switch function**

The dead man switch (DMS) is a function, which requires a continuous positive action from the user to allow the operation of devices which are capable of powered motion and which can affect the safety of the user. It allows the user to halt these devices safely and quickly by releasing the DMS.

The dead man switch functionality consists of one or more dead man switches, a hardwired dead man switch line as part of the M3S bus and some circuitry on the output devices to interrupt normal power supply to the prime moving parts. In case duplicate dead man switches are provided, only one should be active at any one time (to be ensured by the CCM). The dead man switch function on a link is unidirectional. It can only go from the local bus to the remote and not from the remote to the local bus.

The addition of the DMS line to the CAN bus, provides an additional safety backup to turn off the prime movers independent of the CAN bus and the control function of the CCM. This means that even if a non-trapped fault were to develop in the CCM, the user can still shut down the prime movers safely by releasing the dead man switch.

# Developing M3S devices

During the beginning of the M3S project in 1992, CAN was a rather new bus and development tools were not widely available. Also no CAN application layer protocols fulfilling the requirements of the ISO committee 'a serial interface for wheelchair controllers' were present. So a lot of work had to be started at that time and it was decided that the results of this work should be made available in the public domain.

This resulted in an open M3S standard: the M3S specification can be used for free, a free software development kit (SDK) is available containing the programs necessary to develop M3S devices, the complete protocol has been implemented in an application programming interface (API) which is available in C source code at no costs and several M3S interface and prototype cards are available at minimal costs in so called *M3S starter kits*. Furthermore to coordinate the activities related to the development and support of M3S, a M3S dissemination office was started.

**Correspondence:**


M3S Dissemination office
TNO Institute of Applied Physics
PO Box 155
2600 AD Delft
the Netherlands

phone: +31 15 2692004
telefax: +31 15 2692111
web: http://www.tno.nl/m3s/