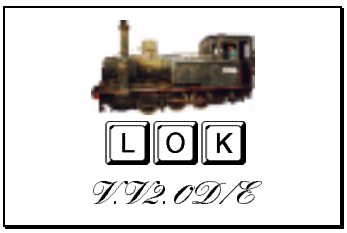
Dr. M. Michael König
Antoniter-Weg 11
65843 Sulzbach/Ts..

☎ 06196/75 01 20
06196/75 03 24
ra.dr.koenig@poboxes.com
http://www.germany.net/teilnehmer/100/76798 /digital.htm

Instructions and Description of the Software Controller 'LOK' for the Märklin Digital System (AC)



by Dr. M. Michael König © 1996-1998

Features:

LOK V.V2.0D/ E provides the following functions:

Generates the required control pulses for locos and solenoid devices in the Märklin Digital system with a PC without further hardware (except a booster).

Can be connected directly to the Märklin booster and via a simple interface to any other booster with balanced inputs such as the Elector complement.

Adjustable delay for short circuit shut-down.

Power-off for prolonged short circuits or emergency stops.

Controls up to 81 locos, i.e. addresses of 00 through 80.

Decoder dependent choice of data transfer protocol: old Motorola, new Motorola-Märklin and a 'mixed format'

Full support of the new Motorola-Märklin protocol: 4 extra functions, enhanced reversing signal, faster data transmission.

Permanently adjustable loco specifications: designation, acceleration and deceleration delay, maximum speeds for each direction and data-format.

Virtually increased number of speed levels from 15 to 25.

Speed and function status of all locos used are stored and shown when the loco address is re-entered.

Control of 20 two-pole or 40 single-pole solenoid devices (or a mixture e.g. for three way turnouts) within an address range of 324/648 (address 000 to 646).

Control of the turnouts via function-keys or via a simulated switchboard with ASCII-Grafik and mouse.

Control of solenoid devices with permanently selectable impulse duration: length of keystroke, one or two defined impulse(s).

Automatic reconstruction of the last turnout and signal positions at program start.

Loco control using a handheld controller or keyboard.

Calibration of handheld controller.

Automatic calibration of pulse generation at program start.

Requirements:

PC:
DOS PC with 80286 or later
min. 512 KB RAM, serial or parallel interface, optional game port
min. 5.25" DD floppy drive
DOS 2.1 or later

Additional hardware:

When used with a Märklin booster:
1 × resistor 47k

1 × resistor 47k
1 (2) × resistor 4k7
1 (2) × resistor 4k7
2 × Z-diode 5V1 (1N4148)

When used with an EDiTS booster:
7 × resistor 10k
2 × resistor 22k
2 × resistor 470
2 × Z-diode 12V
(1 × resistor 22k)
1 × 1N4148
1 × 1488

For each handheld controller:
1 x rotary potentiometer (with centre stop) 100 k
1 single pole switch

Introduction

Whoever had a model-railroad as a child knows the difficulties trying to run several locos on the same track. Excessive wiring was done for independent electric circuits and "dead tracks" without, however, achieving real multiple train control. Only an (expensive) overhead catenary made it possible to run two locos within the same electric circuit. Another solution - a high frequency carrier system - could not assert itself at least in the AC domain.

This has changed recently: Virtually every manufacturer offers a digital multi-train system, which not only gratifies the wish for controlling several trains together, but also allows flexible automatic control with a PC. At least in Germany, as well as probably in Europe the most widely spread system is the Märklin digital multi train control.

The disadvantage of the Digital control is - naturally - the price. Märklin digital equipment is very expensive, although the used electronics components do not justify the prices charged: A simple locomotive control based on the Motorola chip 145026 can be developed quite cost-effective.

However the self-construction of a comfortable control necessitates a large development expenditure and also exceeds the know-how of most fathers that have given their offspring a railroad. Thus buying Märklin seems inevitable and an original system with four handhelds and control of at least 32 points will cost about DM 1,500 to 2,000. This rate is extremely prohibitive, although Märklin on the other hand whets each customers appetite for digital-control with their starter packs, containing locos equipped with a "Delta" decoder. These packs nearly always offer an exceptionally good price/performance relationship: for instance the 1996 starter pack 29605 with a price as low as DM 330 is irresistible, because the contents list price is about DM 700; the loco by itself with the decoder being priced over DM 400. Only if the most interesting offers are seized a number of locos for the digital system accumulate in due course. In addition the Delta decoder already allows for 16 addresses by appropriate DIP switches settings so that medium sized layouts can easily be accommodated. Finally it is no big effort to modify the Delta decoder to include the auxiliary function of the Digital decoder. And even more, the four extra-functions of the new data-format, which are now available in some H0-decoders, can also be activated on the Delta: an information that Märklin conceals (deliberately?). The latter however is only possible with the newer Delta decoders equipped with the 701.17 chip.

The Elector magazine stepped into the breach already some years ago and developed the control system EDiTS introducing it in December of 1987. Of course even a partial assembly of the EDiTS system for 16 controller with a powerful Booster is not cheap; despite favourable purchase I paid approximately DM 1,000 for the parts. The equivalent Märklin gear, however, would cost way over DM 5,000.

The disadvantage of EDiTS is on the one hand some bugs in the firmware that is not to be removed without a thorough knowledge of Z80 assembler. On the other hand and much more significant is the fact that the system does not support the new data format that Märklin - underhandedly - introduced in 1994. Consequently EDiTS does not provide the advantageous extended reversing signal and the four extra functions. Finally if EDiTS is limited to one controller the costs are not reduced considerably as for each controller only few components are needed amounting to about DM 5. Thus EDiTS seems only appropriate for larger layouts and has furthermore become obsolete in the meantime.

The basics of the Märklin digital control has been described in the respective Elector magazines as well as the Elector EDiTS book and the Märklin digital book. On the Internet you will find valuable information at:

http://rr-vs.informatik.uni-ulm.de:80/RR/Maedig/Maedig.html
http://www.germany.net/teilnehmer/100,76798/digital.htm
http://www.heise.de/ct/Redaktion/cm/maerklin.rtf
http://bolam5.lamel.bo.cnr.it/~scorzoni/motorola.html
http://www.marklin.com/subdir/digital/digital.html
http://www.sextant.it/modeltreno/dig_infi.html

Thus in the following only a brief summary of the Märklin system:

The Märklin digital train control is based on the principle that a Control-Unit generates control signals for every receiver (decoder in locos, wagons, turnouts) which are amplified by a Booster and 'impressed' on the operating voltage. This voltage effectual on the track is continually switched between +18..22V and -18..22V according to the control signal. The decoder built in locos, wagons, turnouts, etc. respond only to specific signal packets: every packet consists of an address and a control section. Only if the address matches the selected decoder address the control data are evaluated and executed.

Concretely the so-called Motorola protocol is used being implemented in the Motorola IC MC145026, MC145027 and MC145029 (the latter is not available anymore). It is worth to be noted that the address code does not correspond with the binary logic thus called bits but on a ternary logic: the trits or t-bits can adopt the values 0, 1 or open. For the transmitted pulse this is realised by two bits together representing a trit. Further information can be obtained at the above mentioned references. It is furthermore significant that a new or extended Motorola format - or rather Märklin format - was introduced in 1994 which is used if the DIP switches of the Märklin Control Unit (6021) are set accordingly. This new data format is excellently described by Dr. Andrea Scorzoni: briefly, the binary correlation between each bit of the four control data bit streams is dissolved thus every bit of the stream acts as a discrete bit. But EDiTS as well as simple self-made constructions for instance are based on the Motorola chip MC145026 which is not capable of this new format. If the features of the new format are to be used the digital model railroader is confined to the Märklin Control Unit, a - immanently flexible - software solution or a complete new hardware construction around a micro-controller.

Maximally 80 locos can be controlled independently in direction, speed (15 levels) and a direction dependent auxiliary function. The new protocol offers among others an improved reversing control as well as four supplementary functions, however, not dependent on direction. At the time being no loco decoder for H0 are available supporting the four extra functions. Up to 256 - with EDiTS 324 - turnouts and other devices can be switched. For further reading the interested is referred to the US Märklin WWW site and the Märklin book on digital train control.

At a closer look it turns out that the self-making of a controller for locos and turnouts can only be achieved with a high input of hardware and amounts to the construction of a special-purpose computer in the end. So why not use available remnants? Many of those outdated PC provide for a game card and thus a interface for an analogous joystick. Except for the controller and switches as well as a booster the complete hardware exist which otherwise is costly if EDiTS is reproduced or an own system is developed - the latter only to be constructed with considerable know-how. Thus the only hardware expenditures are needed for a handheld controller which amounts to approx. DM

1.50; if connectors, cables as well as the components (unfortunately) needed for the 'interface' to the EDiTS booster are taken into account you should get by with some DM 10. Supplied with the appropriate software the functionality - except for an automatic PC train control - of original Märklin equipment worth approx. DM 5,000 (!) can be reproduced. Data are transmitted via the parallel or serial interface which is provided by every PC. Thus only the software is missing - which is now available.

So the precondition is that old PCs should be re-activated since it is obviously not attractive to put a newly purchased high-end computer into the basement to control a model railroad. As on the other hand pulse intervals as short as 13 µs have to be realised and as the clock pulse of an old XT has already 220 ns, it is plain to see that essential parts of the software had to be written in assembler (a nowadays virtually extinct foreign language). Out of the question were also those modern gimmicks of (high-resolution) graphics or (moving) images as well as a Windows version. LOK - as this software is named - provides a high-resolution text mode (compared to a Z80 or Apple II) of 80×25 characters and a screen optics that will remind aged users (over 25) of the golden DOS era. Since old PC do not necessarily have a colour display LOK comes in nostalgic black and white. So the motto was "back to the roots" which does not mean to go back to the Stone Ages of Computer Technology. From the point of coding the development of a software running acceptably on old PCs makes higher demands than to tinker at a Windows database application in Visual-God-knows-what. Thus the making of LOK was not achieved during a coffee break.

LOK is intended to be used with an external handheld controller consisting of a potentiometer (100k, linear) connected to the joystick port. The most appropriate potentiometer are those with a centre click-stop position (unfortunately hard to find nowadays) easing the control notably. The mouse was purposely not designated as an input device for loco-control although technically possible and surely 'up-to-date'. This was regarded a bigger ordeal that the precautionary implemented keyboard control - although the latter is a stopgap still. Whoever does not believe this, please try to imagine to have your eyes both on the running loco and on the mouse pointer looking for the loco´s speed bar, function button, etc. A loco is directed by a handheld controller - and that's that. Anyhow, the - unfortunately code intensive - displays of the turnout/signal status are intended to serve those users who cannot but stare on their computer screen. Nevertheless, they have a definite benefit when locos disappear in tunnels, staging yards, etc.
This is not rue regarding the turnout-control via switchboard-simulation. Without additional "touchable" switches a mouse-control had to be implemented.

Finally a clarification: LOK is <u>not</u> intended for automatic layout control. Various software is available for this purpose but they all presuppose the presence of appropriate hardware, either Märklin equipment or substitutes (e.g. EDiTS). LOK should 'only' avoid the purchase of expensive hardware (except the booster). A Märklin or EDiTS compatible interface for data reception is not totally out of question but needs a considerable effort which is only worthwhile if popular interest exists. This, however, necessitates the use of a powerful processor and a multitasking environment - both colliding with the demand for the generation of exact pulse timing and the utilisation of remnants.

Concept

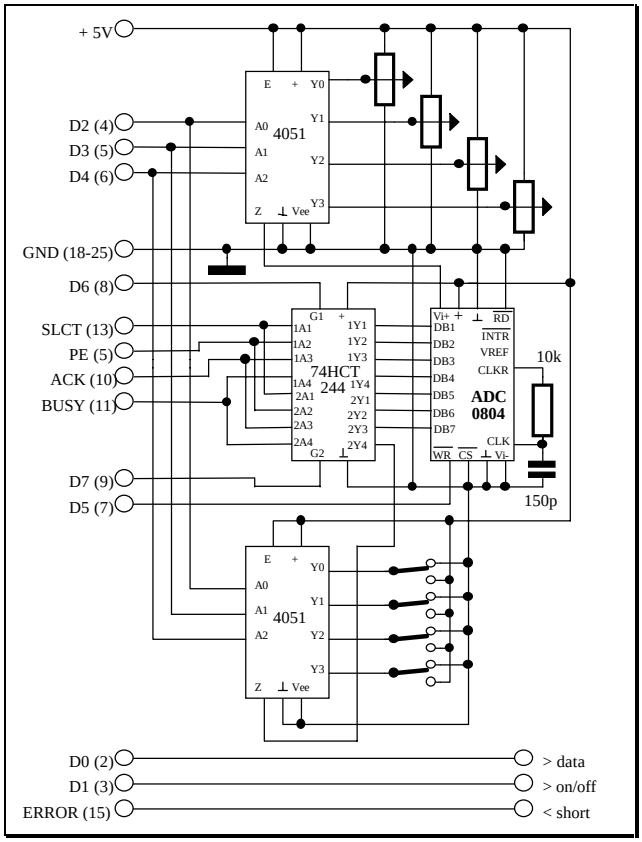
LOK works - much more is not to be said. But it is no great mystery that the exact pulse generation was only possible by machine-oriented coding. Thus a number of system resources are exclusively occupied which is why it is highly recommended to remove all TSR utilities. In plain language: LOK does not tolerate much else in memory. All not necessarily needed programs can only cause troubles.

As required by the protocol every 14 ms a packet of control pulses is generated and transmitted twice with a pause of 1.7ms. The meantime to the next sequence is the pause duration. After the second transmission of control data the handhelds (or keyboard, resp.) is checked and the values found are taken to calculate the new control data. Data for turnout control is also transmitted twice after pressing the key(s). If no new data are available the recent loco information is regularly repeated.

As LOK - as mentioned - is intended for the (re-)use of discarded old PC all gimmicks and especially a graphical user interface had to be eliminated - which does not really matter. Note that the shortest signal lasts 13 µs only. As the system timer 0 switches every 860 ns and the processor clock of an old 8-MHz-XT is 130 ns (thus a multiplication lasts longer than the shortest pulse) the absence of a GUI is understandable.

Slow computers in particular are considerably effected by every command; it is thus not to be ruled out that under certain circumstances a fine tuning of individual timing routines may be required. For the 'natural' tolerance of the decoders (up to 20% deviation) this was not necessary so far.

Finally I have implemented a routine that should allow for a control by handhelds on PCs without a game port (e.g. old notebooks) via an additional circuit. This routine is not verified for the lack of a circuit; the ensuing circuit diagram has been worked out in theory only. Both for the lack of time and of relevance neither a verification nor practical testing has been performed. The program routine, however is only activated if the printer port address is given as the game port address during parameterisation.



This routine as well as the circuit is based on the following concept:
Corresponding to the entered controller number the appropriate handheld is activated on the ADC. After the AD conversion the first four bits - D0 to D3 - of the output data of the ADC are read for the first nibble. Then the following three bits - D4 to D6 - of the output data of the ADC as well as the switch status of the corresponding controller number - D7 - is read as the second nibble. The transmission as nibbles are necessary for not all PCs can receive data on D0 to D7 of the parallel port and thus only SLCT, PE, ACK and BUSY are definitely available for input.

The software access works as follows:

1. addresses (D2 to D4) set "0", ADC-control (D5 to D7) set "1"
2. controller number as address on D2 to D4
3. ADC reset by D5 set "0"
4. D/A conversion started by D5 set "1"
5. loop about 180 µs, because D/A typ. requires about 100 µs
6. open latch for first nibble by D6 set "0"
7. read first nibble from ERROR, SLCT, PE, ACK (bit 0 to 3)
8. shut latch for first nibble by D6 set "1"
9. open latch for second nibble by D7 set "0"
10. read second nibble from ERROR, SLCT, PE, ACK (bit 4 to 7)
11. addresses (D2 to D4) set "0", ADC-control (D5 to D7) set "1"

Whoever wants to try this out, please let me know if it succeeded or failed; if a software error is verified a update is granted.

Loco control is possible throughout even from the keyboard except during initialisation and calibration. Turnout can only be controlled in the normal operation mode thus control is unavailable when adjustments of any kind are made.

Even if not obvious on first sight the coding of the turnout control via function-keys was very complex, as possible input errors should be excluded. For the mentioned preconditions and performance limitations no graphical interface was implemented and even a colour scheme was ruled out. Thus only the limited possibilities of the extended ASCII character set were available to depict all required information with relatively few key operations. You surely have to get used to my

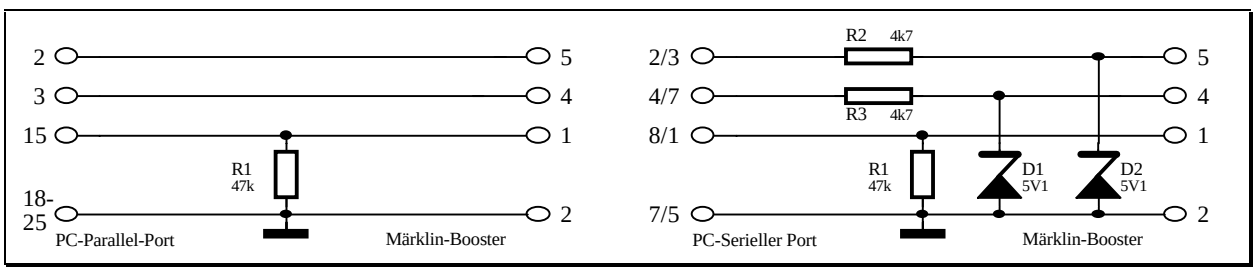
solution. But after a short time you will know how to handle the program as you only need the keys <F1> to <SHIFT>+<F10> and <TAB>+<F1> to <TAB>+<SHIFT>+<F10> after entering all input data.
This is also true regarding the switchboard-simulation. Because of the limitations of the ASCII-character-set the "graphics" is looking like self-made. But after some usage you can work with it without porblems.
With regard to the subroutines for turnout- and loco-data nothing much is left to say; only so much that they represent only in so far exceptional data base applications as numerous and complex feedback routines should exclude input errors. Hereby the program becomes somewhat sluggish especially notable on slow computers. This is, however, of minor importance as the significant process of data transmission is always guaranteed ensuring the current on the tracks.

Connections:

The following pins of the parallel/serial interface (25pol/9pol) are used:

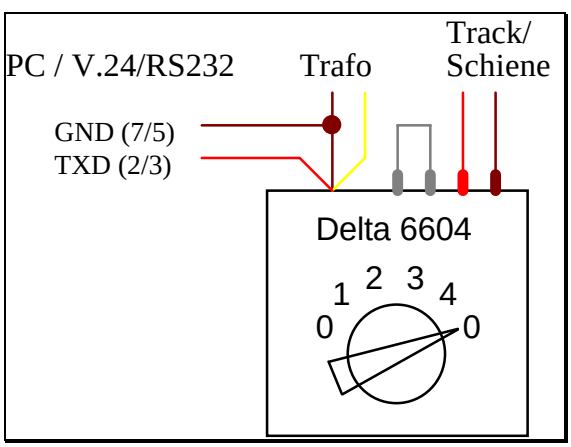
parallel	pin	serial	pin (25/9)			
D0	2	TXD	2/3	data	off	active = high = 5 V
D1	3	RTS	4/7	power on/off	off	active = high = 5V
BUSY	11	DCD	8/1	shorts/emerg. stop	on	active = high = 5 V
ground	18 to 25	ground	7/5	ground		

The connection to the Märklin Booster is quite simple:

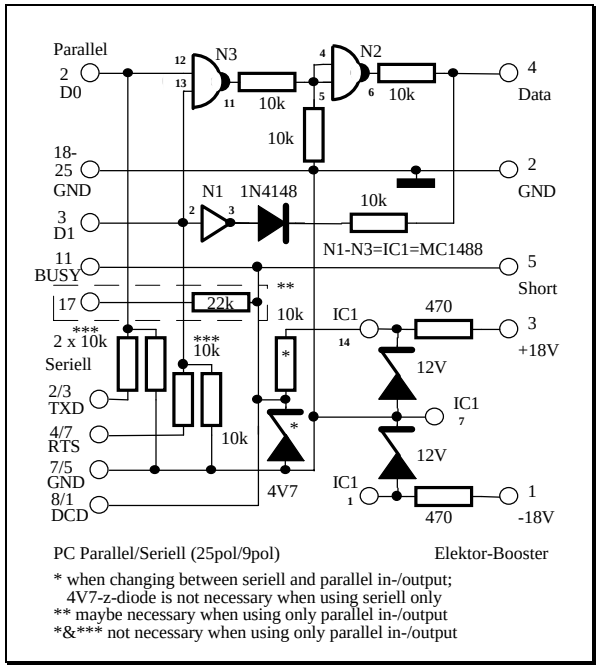


The pin numbers of the serial interface denote the 9-pole/25-pole D-SUB connector. If necessary the zener diodes can/must be replaced by plain 1N4148 with parallel 4k7 resistors.

The Delta-Controller 6604 can be used as a booster, too. But easily this can be done only using the serial port. The ensuing diagram shows the necessary connections. Please take care the knob is turned to the right STOP-position already when switching-on the trafo.
Of course you can use as "better" interface (also for using the parallel port) the following circuit.

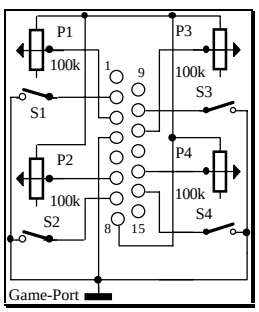


The hook-up to an EDiTS or other booster with a symmetrical inlet via the parallel port unfortunately a little bit more complex for the requisite transposition of 0/5V to +15/-15V as well as the shut-down at short circuits/emergency stops.
The ensuing diagram depicts a combined interface for the parallel and serial port. It can of course be used for one port only and the nonessential parts can be omitted. The pin numbers of the serial interface denote the 9-pole/25-pole D-SUB connector. Note that the reduction of the operating voltage of the MC1488 is obligatory.



The handhelds are connected as follows:

It is advisable to use rotary potentiometers with a centre click-stop; unfortunately they are not easy to get.



Installation:

As the readiness of the (German) public is not particularly marked to observe the copyright law for software I could not but provide LOK with a copy protection. This comprises the usage of a dongle. In case of a damage the dongle will be replaced on return and reimbursement of the low cost-price. No copy protection is perfect. You should, however, have in mind that enhanced updates are only available if the production and distribution of LOK is economically feasible. In other words: if pirate copies are distributed no updates will be forthcoming.

Usage:

At first start-up LOK generates empty data base files and then initiates the parameterisation routines. You will have to enter the required data as described below and if necessary the handhelds are calibrated. After that the calibration of the computer is offered which is strongly recommended. With <-> or <+> you (de)activate the respective handheld and next the standard mask is displayed which is directly accessed on every subsequent program start. Below a brief command description you will find the function display of the handheld and to the right you see the status of the function keys <F1> to <SHIFT>+<F10> in standard and <TAB> mode.

Input of Loco Data

It is recommended to enter the data of the available locos first. For this you select "Loco Data" and enter the required data for the loco with the specified address. Every 'normal' character - including <SPACE> - is accepted. Addresses can only assigned once. Since the Delta decoder can use 16 addresses in any case and with some tinkering the whole range of 81 (0 to 80) this should be no problem ensuring an individual definition for all locos. The symbols "▲" and "▼" stand for the acceleration and deceleration delay, resp.; each speed step is given a delay of $x \cdot 0.1$ seconds with x being an integer between 1 and 9. The direction dependent maximum speed is symbolised by "→┤" and "├←" allowing integers from 6 to 15. "Fmt" stands for data format: "0" defines the old Motorola format and "1" the new Märklin-Motorola protocol. <ESC> leaves this module.

To put it clear: This choice of data format is not identical to the selection of the operating mode during the initial parameter setting. With the latter it is only defined if for the chosen loco the format registered in the Loco Data or the new data format is used to which the reversing signal of the old data format is added (so-called 'mixed format').

Input of Turnout Data

Now comes the data entry for the available solenoid devices with "Turnout Data". The familiar Märklin-organisation of decoder and keyboards is replaced with the more consistent decoder-organisation of 8 sub-addresses per main-address. Attached you find a overview including the relation to the Märklin-C-turnouts. Using this the sub-adresses are contolled with the switches 9 + 10: "9 10" is sub-adresses 0+1, "- 10" is sub-adresses 2+3, "9 -" is sub-adresses 4+5 and "- -" is sub-adresses 6+7. In each column the possible 81 addresses are displayed (0 to 80, Märklin originally uses only 64) while each rows contains 8 sub-addresses (switching outlets) to which the type of solenoid device and its pulse control can be entered. Under the row heading "1", "2", "3" ... "7". the device type is entered, the heading "1∞", "2∞", "3∞" ... "7∞" the pulse control can be set.

The following abbreviations for solenoid devices are available

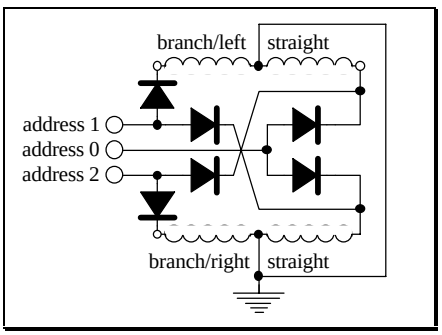
D =	three-way turnout
L =	left turnout
R =	right turnout
K =	double slip turnout
S =	signal w 2 pos.
F =	signal w 3 pos.
M =	motor
E =	uncoupling track

The possible entries are listed in the screen page footer. Please note that the address code of a solenoid device must not exceed one decoder. For instance a three-way turnout must not start at sub-address 6 or even 7 as this device needs 3 addresses and in order to avoid input errors only the assignment of consecutive addresses are allowed.

The pulse control is coded as follows: "0" means that the solenoid is activated as long as the key is pressed. Accordingly "1" indicates a single pulse of 0.3sec duration and "2" two of these pulses with a pause of 0.3sec.

For solenoid devices with multiple poles only the first address has to be entered which is "straight" for turnouts or "green/orange" for signals. The other required addresses for the corresponding type are marked with "=" for the ease of input. For two-pole turnouts the address for "straight" has to be followed by the address for "branch". For two-pole signals the address for "green/orange" is followed by "red", the address sequence for three-pole signals is "green", "red" and "orange". The three-way turnout is defined by "straight", "branch/left" and "branch/right".

This program method requires some re-wiring in three-way turnouts: the connections of the solenoids to the decoder outlets has to be modified according to the opposite figure.
It is only a schematic circuit diagram that illustrates how the individual solenoids - if necessary with transistors - can be controlled by three switching outlets.



Note that for three aspect signals a direct change from "orange" (Hp2) to "green" (Hp1) is not allowed and thus the program will not permit this. The signal has always to be set to "red" (Hp0) first. Although the procedure vice versa from "green" (Hp1) to "orange" (Hp2) is technically possible it is also not admissible prototypically so first a "red" (Hp0) is required. To avoid that the program is completely idle if the invalid change of "Hp2" to "Hp1" and vice versa is entered, a warning beep is heard and the signal is switched to "Hp0" instead.

Invalid inputs are not possible - they are 'approved' by a warning beep. Entries are deleted by <SPACE>, <ESC> exits this subroutine.

Note that all pulse controlled solenoid devices have to be equipped with a terminative deactivation or a protection similar to that shown in http://www.germany.net/teilnehmer/100,76798/weichdek.htm. Otherwise the shut-off command can be missing and the solenoids can burn out if the PC hangs/crashes in the wrong moment.

Assignment of the function keys

Finally the function keys are adjusted to activate the solenoid devices. By selecting the command "F-Keys" the current assignment of the function keys, i.e. the addresses of the respective device are displayed. With the addresses combined with the status symbols as well as the sign "┌─" it is apparent if a two- or a three-pole device is connected. These status symbols are:

← = left turnout - branch
→ = right turnout - branch
↑ = turnout - straight
X = double-slip turnout - straight
⟩ = double-slip turnout - branch
⁄ = signal "green" (Hp1)
∅ = signal "orange" (Hp2)
– = signal "red" (Hp0)
Φ = motor
Π = uncoupling track

The symbol "∩" indicates that only the <TAB> - function of the respective function key is assigned to a single pole solenoid device.

After the function key to be assigned is entered - if necessary with <TAB> - the window with the turnout data opens up; select the desired solenoid device and confirm with <ENTER>. Invalid inputs are rejected; furthermore a existing assignment cannot be overwritten, it has explicitly to be deleted with <DEL> first to avoid input errors. At the same time it is possible to modify the data of the solenoid devices. <ESC> exits the sub-routine.

Switchboard

Selection of the handhelds

If the handheld is activated with <ALT>+<1> or <1> the following actions are possible:
(a) assignment of a new loco address
(b) activation of the four extra functions in the new protocol
(c) reversing in the old protocol
(d) online modification of actual loco data
(e) if operated by the keyboard - activation of the auxiliary function

<ESC> exits this function which; otherwise the exit command is given 10 sec after the last key is pressed. The exception to this rule is when one of the following functions is selected some of which require further inputs:

Reversing
Since the old Motorola data format does not provide for a absolute reversing command the actual loco direction can deviate from the controller position, e.g. if the loco did not receive the reversing command due to transmission difficulties. In this case the reversing command can be sent manually by selecting "Reverse"; this works even if the loco is moving.

Select loco/assignment of loco address

The loco address can be changed using the <-> on the numeric pad and selecting "Address". Originally it was intended that this operation is only possible after a manual deactivation of the handheld. Test users found that awkward; thus the handheld is deactivated automatically if "Address" is selected. Shown are only locos that have been defined in "Loco Data" as well as the currently active loco.
Loco data can be modified here too: a new address entry is confirmed by <ENTER>, <ESC> cancels the process. The handheld is re-activated with <+> on the numeric pad. LOK saves the speed and the

status of the four extra functions of the new protocol - in case of a keyboard control even the status of the auxiliary function - of all locos used.
To set an example

Loco 01 is active. During operation the handheld is deactivated. Loco 01 keeps on running based on both the internal data backup as well as the continuous transmission of the last commands entered. With the new data format the information on the four extra functions is sent also. A change to loco 02; the activation of the handheld denotes that all further data are transmitted to loco 02 only. Excessive shunting follows. Then - after selection and deactivation of the handheld - back to loco 01. If the handheld is now used the first speed bar shows the (saved) speed of loco 01 as well as the status of the extra functions. By the appropriate setting of <F1> to <F4> - if necessary also the auxiliary function when keyboard controlled - as well as adjusting the handheld which actual setting is displayed in the second speed bar a seamless transition is seen to after the re-activation of the handheld.

One further clarifying remark: when the handheld is deactivated the current data for the running locos are regularly repeated. On re-activation the actual data are send to the selected loco.

Acceleration delay
With "Speed up" the acceleration delay can be entered online. This variable accepts integers from 0 to 9 to be set with <-> and <+> on the numeric pad; the respective loco can be run simultaneously to watch the effects immediately. The delay is implemented by changing the speed level every $x \cdot 0.1$ seconds, where x is the integer entered. If the handheld is turned from "Full stop" to "Full speed" the loco does not receive the command for the highest speed level instantly. Rather every speed level is passed separately and is delayed according to the value entered. The artificial intermediates generated by permanently switching between two speed levels are also allowed for. <ENTER> accepts, <ESC> rejects all newly input data.

Deceleration delay
With "Slow down" the deceleration delay is determined analogous to the preceding description.

Maximum speeds
With "Speed" the maximum speeds for both directions can be set independently analogous to the preceding description; first the maximum speed for the forward than for the backward direction.

Extra functions 1 to 4
With <F1> to <F4> locos equipped with the 701.17 decoding chip and a supplementary circuit the four extra functions of the new protocol can be switched if the appropriate data format is assigned. A corresponding display informs about their status.

Auxiliary function
With "Lights" the auxiliary function of a Digital or a modified Delta decoder is switched if the loco is keyboard controlled. The status is displayed as well.

Initialisation

"Init" activates the sub-routine where the most important parameter are set, namely the operating mode (data format), the LPT- or COM-port address, the game port address as well as the short circuit shut-down delay. Moreover the handheld is calibrated, a procedure frequently to be repeated as the game card components drift considerably. This is only to be avoided if the parameter are saved once at operating temperature and future playing is always delayed until the PC reaches this state.

Data format
If the data format for operation is "0" all information are transmitted to the respective loco according to its specifically set data. Thus an adjustment - in contrast to the Märklin controller - is not required.

Port addresses
They are entered in hexadecimal format corresponding to the used ports. Standard values for the OUT port are "0378" for LPT1 and "0201" for the game port. Valid inputs for the OUT port are 0278, 0378, 02BC and 03BC for LPT as well as 02FC, 03FC, 02EC, 03EC and 02E4 for COM. The COM-adresses are *not* the base-adresses but the actual used port-adresses. The entry "0000" for the game port sets the loco control to the keyboard.

Short circuit shut-down delay
Since short-lasting short circuits are frequent during operation an immediate shut-down it is not desirable. A delay can be adjusted between 0.5 and 3.0 seconds.

Lights

LOK provides the option to switch with <B> the 8. subadress of each turnout-mainadress constantly to "on". When switching the lights of the turnouts with this 8. subadress you can control the lights of all turnouts with only one key. The input of "1" enables this light-funktion and "0" switches this off.

Automatic processor calibration
If the program should not verify the saved values for the pulse generation at every start-up enter "0".

Automatic recovery of turnout/signal position
The input of "1" denotes if the software should reconstruct the turnout and signal positions according to the last saved session. "0" switches this off.

Calibration of handhelds
Following the instructions the values for the left, right and centre potentiometer position are determined. If the program registers identical or out-of-range values the routine assumes a faulty or missing potentiometer: the game port is set to "0000" thus the keyboard control is activated.

Short circuit/emergency stop

At a short circuit the data transmission is interrupted after the shut-down delay is expired. Removing the defect and pressing a key resumes the normal operation. The same applies to the use of <SPACE> which serves as a emergency key throughout except when entering loco or turnout data.

Program exit

<ESC> leaves the program and the current data are saved automatically. This does not include the information on processor and handheld calibration: they are only updated if the corresponding query is affirmed, otherwise the old values from the start-up are used.

Loco and turnout control

Loco control
If handhelds are attached they control the speed. Speed level 0 is in the centre position; turning the knob to the right increases and to the left decreases speed. If the knob is turned over the centre

position to the left, the loco is reversed. But the absolute assignment of turning direction to loco direction is only available with the new data format in combination with the decoding chip 701.17. With the old data format only a specific flip-flop on the decoder is responsible for the reversing command. In the case of transmission difficulties the decoder may miss the reversing command: the loco keeps on running unperturbed.
The contact on the handheld switches the auxiliary function to which usually the lights, a steam generator or the TELEX coupler is assigned as long as the loco is equipped with a Digital or a modified Delta decoder.

If no handhelds are attached or should not be used by entering "0000" for the game port the loco is controlled by the key combinations <SHIFT>+<1> in forward and by <CTRL>+<1> in reverse direction. Continually pressing these keys the speed level is counted up or down, respectively. With fast computers and short pause as well as retrieval times this can result in very fast responses. In this case the "delay" value in the .INI file can be set to "1" or else the value for the retrieval time can be increased adequately.

Turnout control
Turnouts and other solenoid devices are controlled by simply pressing <F1> to <F10>, <SHIFT>+<F1> to <SHIFT>+<F10> and <TAB>+<F1> to <TAB>+<SHIFT>+<F10>. A short activation suffices if a single or double impulse was selected; otherwise the device is switched corresponding to the duration of the keystroke. The key currently being pressed is highlighted on the screen display. Following the definitions in the turnout data as well as function key section the corresponding status of the device is displayed according to the symbols mentioned above. This is, however, only relevant if the turnouts or signals are not visible. For three-pole devices the switch for the third status is on the <TAB>-level of the following key; the device status is nevertheless shown over the 'actually' assigned function key. To avoid misinterpretations three-pole devices are further identified by "┌─" between the upper function keys to symbolise that the <TAB>-level of the subsequent key is associated with the previous key.

The keys for single pole solenoid devices is not highlighted, as these components are mono-stable as a rule such as uncoupling tracks, turntable motors etc.

LOK provides as consequence of the advantage of one decoder in each turnout the option to switch the 8. subadress of each turnout-mainadress constantly to "on". When switching the lights of the turnouts with this 8. subadress you can control the lights of all turnouts with only one key. Turnout-decoder which take care of this feature (e.g. the improvement of my M-track-turnout-decoder or the C-track-turnout-decoder of Mario Binder and Bo Braendstrup) make shure that the light dies not flicker when switching the turnout. You can switch the lights with <B> when the light-function is enabled in the initialisation. Of course you loose one subadress from each turnout-mainadress when using the light-function and this subadress is not avalaible at the turnout data; it shows the status of the light-function. But LOK offers access to 648 turnout-adresses so that loosing 80 adresses do not care.

Additional parameter setting by modification of LOK.INI

Following entries in LOK.INI are (possibly) of interest and can be modified only by editing the file LOK.INI:

```
  18 = lokkurz
   9 = weichkurz
 132 = loklang
  66 = weichlang
1278 = lokpause
 634 = weichpause
  14 = interrupt
   4 = request
   1 = delay
```

Pulse generation

The values for "lokkurz", "weichkurz", "loklang", "weichlang", "lokpause" and "weichpause" determine the pulse generation. If a correct control should not be possible and a oscilloscope analysis should give evidence that the timing is inaccurate the values above have to be modified accordingly. "lokkurz/loklang" defines the short (26 µs) / long (182 µs) pulse duration for locos and "lokpause" is the pause duration between two data packets of 1.7 ms. The values for turnouts ("weich...") are analogous though only half as long - thus 13 µs, 91 µs and 850 µs.

Response time

The "interrupt" value determines the duration of data transmission, thus indirectly the pause duration between two data packets. A minimum of 14 is valid resulting in a pause of 2.8 ms - a little more than twice as long as Märklin permits for the new data format. But this is too low for slow computers because computing time will run short for the evaluation of the handheld status. The slower the PC is the higher "interrupt" should be set. Note, however, that the variable "request" determines how fast the keyboard data are retrieved as it furthermore defines how often the handheld is polled. A value of 4 means that the handheld is polled after every fourth data transmission. A minimum of 2 is valid although too low for fast computers and keyboard control. The pauses should be increases with higher "interrupt" values for slow PC to gain more time for calculations. This results in longer intervals between the polling of the handheld which can be compensated by setting "request" to the minimum of 2. For extremely slow computers some try and error will be necessary to find the optimum parameter combination.

Delay for keyboard control

The loco control by keyboard can be hampered by rapid changes in speed levels with fast computers. In this case set "delay" to "1", if that does not suffice increase "request" adequately.

Translation

With exceptions of a few parts was this translation made by Dipl.Ing. Manfred Roehrig from Germany (roehrig@gem.uni-hannover.d400.de). Many thanks to him for this grateful work. Some mistakes you probably find are my work as result of trying to correct some misunderstandings.

Sulzbach, April 1998

Main-adress	145027-Pins: 1 2 3 4	Turnoutnumber: (EDITS/Selfbuilted)	Märklin-C-decoder-switches 1 - 8
00	o o o o	0 1 2 3	- - - - - - - -
01	+ - - -	4 5 6 7	- 2 3 - 5 - 7 -
02	o - - -	8 9 10 11	- - 3 - 5 - 7 -
03	- + - -	12 13 14 15	1 - - - 5 - 7 -
04	+ + - -	16 17 18 19	- 2 - 4 5 - 7 -
05	o + - -	20 21 22 23	- - - 4 5 - 7 -
06	- o - -	24 25 26 27	1 - - - 5 - 7 -
07	+ o - -	28 29 30 31	- 2 - - 5 - 7 -
08	o o - -	32 33 34 35	- - - - 5 - 7 -
09	- - + -	36 37 38 39	1 - 3 - - 6 7 -
10	+ - + -	40 41 42 43	- 2 3 - - 6 7 -
11	o - + -	44 45 46 47	- - 3 - - 6 7 -
12	- + + -	48 49 50 51	1 - - 4 - 6 7 -
13	+ + + -	52 53 54 55	- 2 - 4 6 - 7 -
14	o + + -	56 57 58 59	- - - 4 6 - 7 -
15	- o + -	60 61 62 63	1 - - - 6 - 7 -
16	+ o + -	64 65 66 67	- 2 - - 6 - 7 -
17	o o + -	68 69 70 71	- - - - 6 - 7 -
18	- - o -	72 73 74 75	1 - 3 - - - 7 -
19	+ - o -	76 77 78 79	- 2 3 - - - 7 -
20	o - o -	80 81 82 83	- - 3 - - - 7 -
21	- + o -	84 85 86 87	1 - - 4 - - 7 -
22	+ + o -	88 89 90 91	- 2 - 4 - - 7 -
23	o + o -	92 93 94 95	- - - 4 - - 7 -
24	- o o -	96 97 98 99	1 - - - - - 7 -
25	+ o o -	100 101 102 103	- 2 - - - - 7 -
26	o o o -	104 105 106 107	- - - - - - 7 -
27	- - - +	108 109 110 111	1 - 3 - 5 - - 8
28	+ - - +	112 113 114 115	- 2 3 - 5 - - 8
29	o - - +	116 117 118 119	- - 3 - 5 - - 8
30	- + - +	120 121 122 123	1 - - 4 5 - - 8
31	+ + - +	124 125 126 127	- 2 - 4 5 - - 8
32	o + - +	128 129 130 131	- - - 4 5 - - 8
33	- o - +	132 133 134 135	1 - - - 5 - - 8
34	+ o - +	136 137 138 139	- 2 - - 5 - - 8
35	o o - +	140 141 142 143	- - - - 5 - - 8
36	- - + +	144 145 146 147	1 - 3 - - 6 - 8
37	+ - + +	148 149 150 151	- 2 3 - - 6 - 8
38	o - + +	152 153 154 155	- - 3 - - 6 - 8
39	- + + +	156 157 158 159	1 - - 4 - 6 - 8
40	+ + + +	160 161 162 163	- 2 - 4 - 6 - 8
41	o + + +	164 165 166 167	- - - 4 - 6 - 8
42	- o + +	168 169 170 171	1 - - - - 6 - 8
43	+ o + +	172 173 174 175	- 2 - - - 6 - 8
44	o o + +	176 177 178 179	- - - - - 6 - 8
45	- - o +	180 181 182 183	1 - 3 - - - - 8
46	+ - o +	184 185 186 187	- 2 3 - - - - 8
47	o - o +	188 189 190 191	- - 3 - - - - 8
48	- + o +	192 193 194 195	1 - - 4 - - - 8
49	+ + o +	196 197 198 199	- 2 - 4 - - - 8
50	o + o +	200 201 202 203	- - - 4 - - - 8
51	- o o +	204 205 206 207	1 - - - - - - 8
52	+ o o +	208 209 210 211	- 2 - - - - - 8
53	o o o +	212 213 214 215	- - - - - - - 8
54	- - - o	216 217 218 219	1 - 3 - 5 - - -
55	+ - - o	220 221 222 223	- 2 3 - 5 - - -
56	o - - o	224 225 226 227	- - 3 - 5 - - -
57	- + - o	228 229 230 231	1 - - 4 5 - - -
58	+ + - o	232 233 234 235	- 2 - 4 5 - - -
59	o + - o	236 237 238 239	- - - 4 5 - - -
60	- o - o	240 241 242 243	1 - - - 5 - - -
61	+ o - o	244 245 246 247	- 2 - - 5 - - -
62	o o - o	248 249 250 251	- - - - 5 - - -
63	- - + o	252 253 254 255	1 - 3 - - 6 - -
64	+ - + o	256 257 258 259	- 2 3 - - 6 - -
65	o - + o	260 261 262 263	- - 3 - - 6 - -
66	- + + o	264 265 266 267	1 - - 4 - 6 - -
67	+ + + o	268 269 270 271	- 2 - 4 - 6 - -
68	o + + o	272 273 274 275	- - - 4 - 6 - -
69	- o + o	276 277 278 279	1 - - - - 6 - -
70	+ o + o	280 281 282 283	- 2 - - - 6 - -
71	o o + o	284 285 286 287	- - - - - 6 - -
72	- - o o	288 289 290 291	1 - 3 - - - - -
73	+ - o o	292 293 294 295	- 2 3 - - - - -
74	o - o o	296 297 298 299	- - 3 - - - - -
75	- + o o	300 301 302 303	1 - - 4 - - - -
76	+ + o o	304 305 306 307	- 2 - 4 - - - -
77	o + o o	308 309 310 311	- - - 4 - - - -
78	- o o o	312 313 314 315	1 - - - - - - -
79	+ o o o	316 317 318 319	- 2 - - - - - -
80	- - - -	320 321 322 323	1 - 3 - 5 - 7 -

+ = Plus
- = Minus
o = Offen

Märklin officially uses only the main-adresses 01 to 64. Märklin-turnout-no. = EDITS-turnoutnumber - 3