DCC Locomotive Decoder

Lokommander II User Manual

by TEHNO LOGISTIC

This document is subject to technical modification

by

Contents

1. Important information 6
2. Abbreviations 7
3. User manual contents 8
4. Main features 9
5. Technical specifications 10
6. General description of Lokommander II decoders 11
7. Decoder installation 18
8. Setting up the decoder 21
9. Decoder address 22
10. Speed settings 24
    10.1. Linear speed adjustment in 3 points 25
    10.2. Tabular speed adjustment in 28 steps 26
11. Motor Control 28
12. Controlled stops 36

12.1. Constant braking distance (CBD) 36
12.1.1. Fixed decelerations stop 36
12.1.2. Variable decelerations stop 37
12.2. Detecting asymmetric DCC signal (Lenz ABC) 37
12.3. Shuttle train operation (Penduling) 39
12.3.1. Without intermediate stops 39
12.3.2. With intermediate stops 40
12.4. Special brake functions 41
13. Function outputs 43
14. Analogue operation (DC) 50
14.1. Analogue mode 1 51
14.2. Analogue mode 2 52
14.3. Controlled stop on DC sector 52
15. Bidirectional communication (RailCom) 53
16. Special functions 54
17. Electrical Coupler Configuration 57

18. SUSI 60
18.1. Programming SUSI modules 60
19. Using external capacitors or a power pack 62
20. Resetting the decoder 65
21. Secondary address (decoder lock) 66
22. Firmware update 68
23. Accessories 69
24. Technical support 69
25. The decoder CV table 70
26. Annex Bits and Bytes 124

1. Important information

Please read below information before setting up to work

- Lokommander II decoders are exclusively designed for model trains. Any other use is not supported.
- Completely remove power (transformer or power supply) when connecting or disconnecting the decoder.
- Avoid mechanical stress or compressed air blows to the decoder board.
- Do not remove the heat shrink tube form the board (if present).
- Avoid electrical contact of the decoder board and wires (including unused ones) to the locomotive chassis. Strip the ends of unused cables.
- Do not solder any extension cables to the decoder circuit board unless this is absolutely necessary (connections to sound modules, power supply unit / power pack).
- Do not wrap the board in any material (like insulating tape). This will cause board overheating.
- Connect the wires as described in the user manual. Misusage / misconnection can cause malfunction of the decoder or can even damage the decoder.

2. Abbreviations

ABC	- Automatic Brake Control	n.c.	- not connected
AC	- Alternate Current	NMRA	- National Model Railroad Association
BEMF	- Back Electro-Motive Force	PID	- Proportional–Integral–Derivative
CBD	- Constant Braking Distance	PoM	- Programming On the Main
Clk	- Clock	PT	- Programming Track
CV	- Configuration Variable	RailCom	- Bidirectional Communication Protocol
DC	- Direct Current	REV	- Reverse
DCC	- Digital Command Control	RL	- Rear Light
FL	- Front Light	SPP	- Smart Power Pack
FWD	- Forward	SUSI	- Serial User Standard Interface
GND	- Ground, Negative Voltage Supply, V-	V+	- Positive Supply Voltage, (+) common
LSB	- Least Significant Bit (or Byte)	Vmax	- Maximum Speed
MI	- Maintenance Interval	Vmid	- Medium Speed
MSB	- Most Significant Bit (or Byte)	Vmin	- Minimum Speed

3. User manual contents

We thank you for purchasing a Lokommander II decoder. We hope it will add an extra enjoyment to your model train experience.

This manual is divided into several chapters, which show you step-by-step how to install and customize a Lokommander II decoder.

- General description of Lokommander II decoders contain details about the various form factors of the Lokommander II decoders together with the description of the available connections.
- Decoder installation and Setting up the decoder provides details about the installation of the decoders. Lokommander II can control most of the electrical motors available on the market. It is recommended that information about the electrical motor that will be used is available when reading this chapter.
- The next chapters contain useful information about the Decoder address / Secondary address (decoder lock), motor control, start, stop and speed adjustments, setting up functions and digital and analog operation.
- The complete list of the CVs with factory default values and value ranges is listed in the Table *9*: CV table at the end of the manual.
- Annex Bits and Bytes contain information related to the decimal-binary systems.

4. Main features

- Generic DCC mobile decoder, NMRA, NEM and RCN RailCommunity standards compatible[1]
- Support PT or PoM programming modes
- Support analogue (DC) operation with configurable functions
- User configurable addresses: short (1-127) and long (128-9999)
- User configurable speed steps:14, 28 or 128
- User configurable speed adjustment in 3 points (V_{min}, V_{mid}, V_{max}) or tabular (28 points)
- User configurable shunting speed (using CV114) Acceleration / Deceleration inhibition (using CV115)
- Constant distance braking activated on ABC / DC sector or at zero speed.
- Reduced speed drive on ABC Slow Speed sector
- Load compensation trough BEMF
- Shuttle train operation (Penduling / Push-Pull)
- Up to 10 dimmable auxiliary outputs, maximum current 300mA
- Output Mapping to functions F0, F1-F12
- Short-circuit and overcurrent protection on all outputs (motor and auxiliary)
- RAILCOM ® Bidirectional communication

[1] See Table 4 and Table 5

- SUSI© interface
- Outputs for Smart Power Pack (SPP ©)
- Electromagnetic coupler drive capability
- Firmware upgradeable via tOm Programmer even with the decoder mounted in the locomotive
- Small form factor useable in N, TT, H0, H0e scale models
- Maximum motor current 1000mA

5. Technical specifications

- Supply voltage range (DCC track voltage): 4÷24 V
- Stand by current (all outputs off): < 10 mA
- Maximum current for each output: 200 mA
- Maximum total current for the decoder: 400 mA
- Dimensions (without cables and connectors): see Table 1
- Weight: 4÷6 g
- Protection class: IP00
- Operating temperature: 0 ºC ÷ +60 ºC
- Storage temperature: -20 ºC ÷ +60 ºC
- Humidity: max 85 % non-condensing

6. General description of Lokommander II decoders

Lokommander II decoders are designed to be used in N, TT, H0, H0e scale models. The difference between decoders consists of the physical size, the connector type, the maximum output current and the number of available auxiliary outputs. As programming and usage they are identical. Table 1 contains the Lokommander II decoder models with corresponding ordering code and size. The decoders name coding is as follows:

- N18 = NEXT 18 female connector, 18 pins
- 6P = NEM 651 male connector, 6 pins. The white paint drop is showing pin 1 (Motor Right)
- 8P = NEM 652 male connector, 8 pins
- P12, P16, P22 = PLUX 12, PLUX 16, PLUX 22 connector; all versions have one index pin
- M21 = MTC 21 female connector, 21 pins
- S = SUSI male connector
- 90(R) = 90° connector; R stands for reverse 90°
- W = wired decoder. Connector can be present at the end of the wires.
- P at the end of the code stands for high power output on AUX3 and AUX4.
- M at the end of the code of MINI W22 means that wires are soldered also for AUX3 ÷ AUX7.

Only the continuous colored lines represent already soldered wires; the dashed colored lines are for reference only and show where additional wires can be soldered by the user.

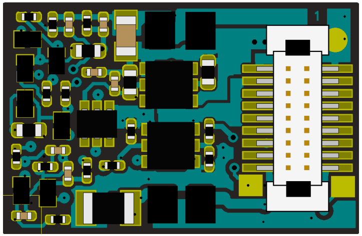
The drawings show the real wire colors (according to NMRA standard), and they are 100mm ±5% long.

Table 1: Lokommander II series cross reference

Model		Connector	Order code	Size mm	Size inch
MICRO	N18	NEXT 18	2010216	14.2x9.2x3	0.56x0.36x0.12
	6P	NEM 651	2010220	12.9x9.2x3	0.50x0.36x0.12
	6P90	NEM 651	2010221	12.9x9.2x3	0.50x0.36x0.12
	W6P	NEM 651 & Wires	2010222	12.9x9.2x3	0.50x0.36x0.12
	W	Wires	2010223	12.9x9.2x3	0.50x0.36x0.12
	6P90R	NEM 651	2010227	12.9x9.2x3	0.50x0.36x0.12
MINI	P12	PLUX 12	2010210	19.5x11x3.3	0.76x0.43x0.13
	P16	PLUX 16	2010211	19.5x11x3.3	0.76x0.43x0.13
	W6P	NEM 651 & Wires	2010207	20x11.6x3.8	0.78x0.46x0.15
	W8P	NEM 652 & Wires	2010212	20x11.6x3.8	0.78x0.46x0.15
	M21	MTC 21	2010208	20x15.5x4	0.79x0.61x0.16
	M21S	MTC 21 & SUSI	2010209	20x15.5x5	0.79x0.61x0.20
	M21SP	MTC 21 & SUSI	2010228	20x15.5x5	0.79x0.61x0.20
	P22	PLUX 22	2010217	20.5x15x3.3	0.81x0.59x0.13
	W22	Wires	2010218	20.5x15x3.3	0.81x0.59x0.13
	W22M	Wires	2010229	20.5x15x3.3	0.81x0.59x0.13

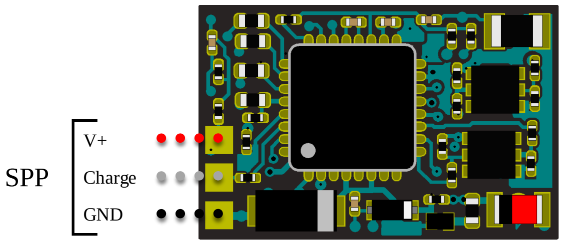
The Lokommander II MICRO N18 has a NEXT 18 connector and SPP wires soldering pads. The pins and soldering pads description are shown in Figure 1 below.

Figure 1: Lokommander II Micro N18



Left Rail	Left Rail
Motor Left	FL
AUX2	AUX6
SUSI-Data/AUX4	V+
GND	GND
V+	SUSI-Clk/AUX3
AUX5	AUX1
RL	Motor Right
Right Rail	Right Rail

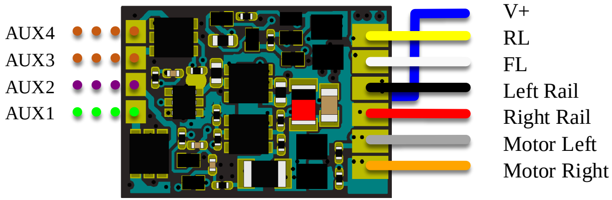
Bottom view



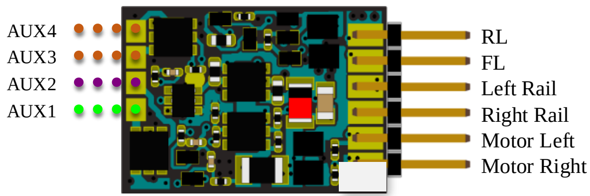
Top view

The Lokommander II MICRO 6P, 6P90, 6P90R, W6P and W are based on the same printed circuit board (PCB). The pins, wires and soldering pads description are shown in Figure 2 below.

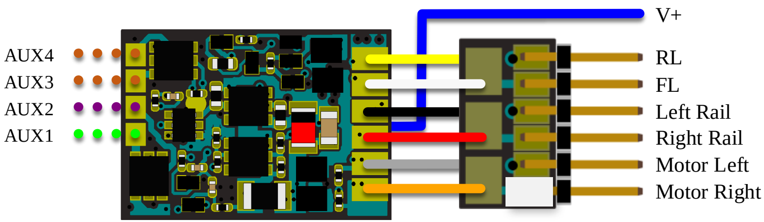
Figure 2: Lokommander II Micro 6P, 6P90, 6P90R, W6P and W



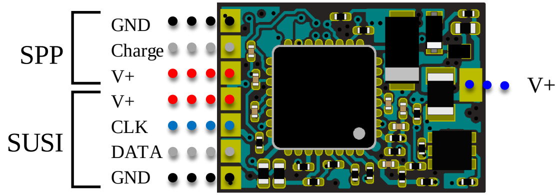
Micro W – Bottom view



Micro 6P, 6P90 and 6P90R – Bottom view



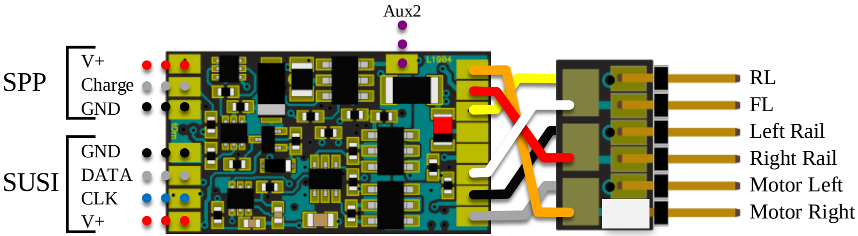
Micro W6P– Bottom view



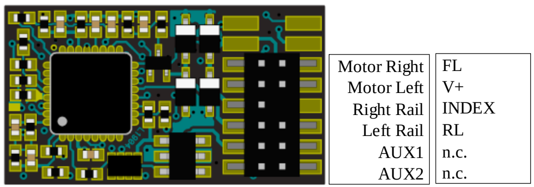
Top view – solder pads description

The Lokommander II MINI P12, P16, W6P and W8P are based on the same PCB. The pins, wires and soldering pads description are shown in Figure 3 below.

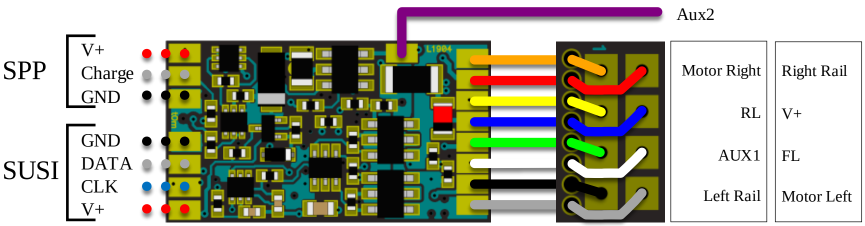
Figure 3: Lokommander II Mini P12, P16, W6P and W8P



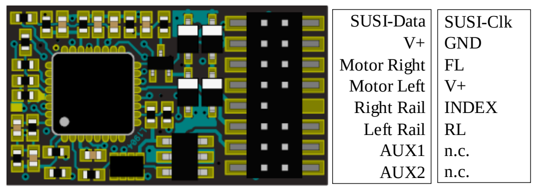
Mini W6P– Bottom view



Mini P12 – Top view



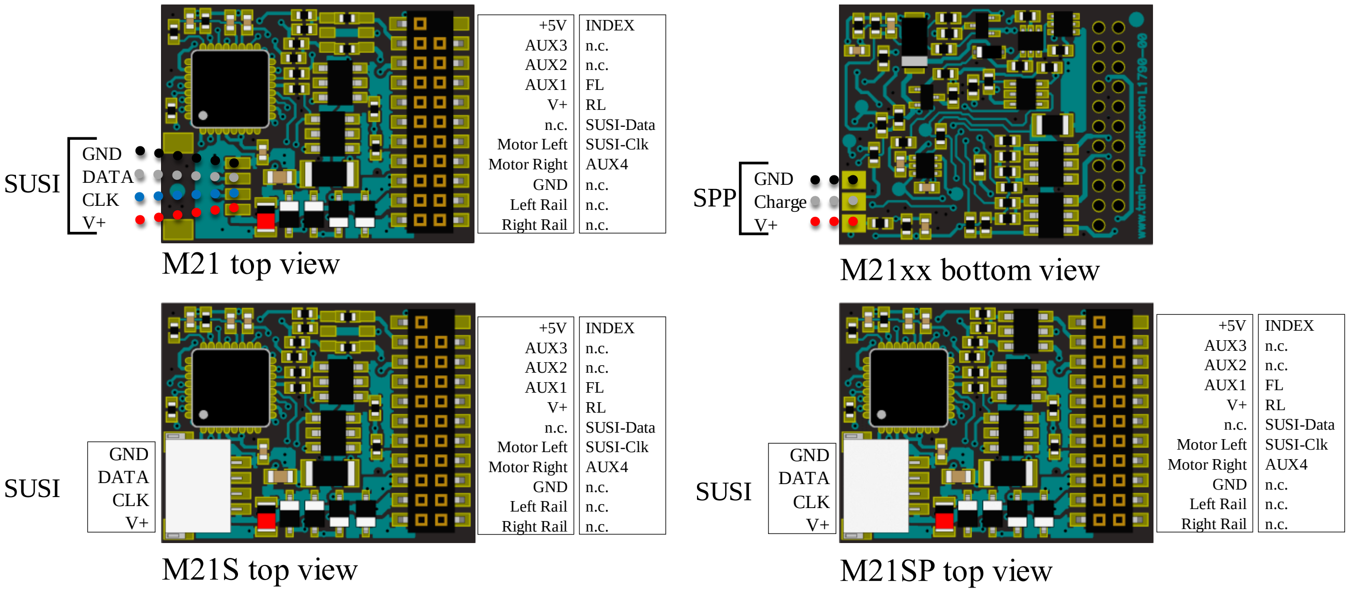
Mini W8P– Bottom view



Mini P16 – Top view

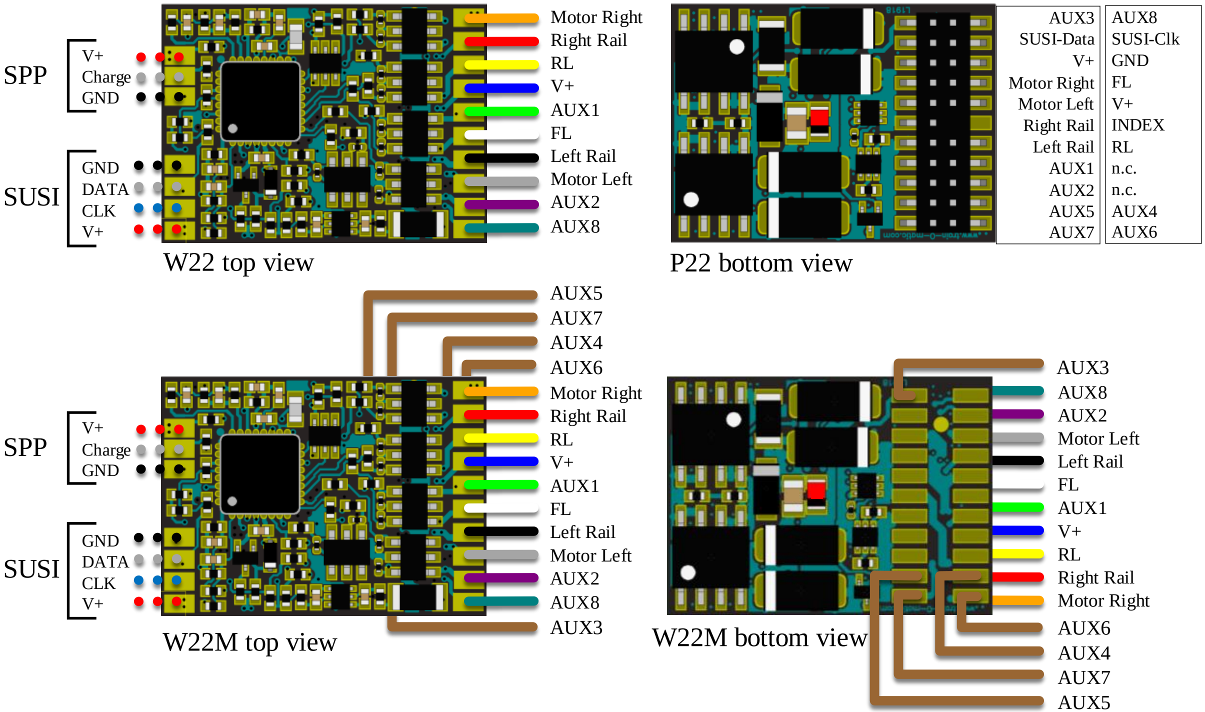
The Lokommander II MINI M21, M21S and M21SP are based on the same PCB. The pins and soldering pads description are shown in Figure 4 below.

Figure 4: Lokommander II Mini M21, M21S and M21SP



The Lokommander II MINI P22, W22 and W22M are based on the same PCB. The pins, wires and soldering pads description are shown in Figure 5: Lokommander II Mini P22, W22 and W22M below.

Figure 5: Lokommander II Mini P22, W22 and W22M



7. Decoder installation

Before installing a digital decoder, especially in earlier models, it is recommended to make sure that the locomotive is operating properly in DC mode. To do this, perform the following operations:

- Clean the wheels and the blades
- Check the motor condition, measure the idle current of the motor powered by 5-10V, which should not exceed 200÷300mA, if necessary clean the brushes and the collector.
- Check the transmission system, if necessary, clean and lubricate the axes and the sprockets.
- If the locomotive is equipped with light bulb, check if they are rated to 16V. Replace them if necessary.

In the case of locomotives prepared for digitization, the installation of the decoders equipped with the standard connectors (PLUX, MTC, NEXT18, MICRO-6, NEM652) is done by extracting the dummy module for analog operation from the connector on the motherboard. In the thus released connector, insert the decoder by following the key (INDEX) or follow the instructions received with the locomotive.

The NEXT18, NEM 651 and NEM652 decoders can be inserted in wrong position (180º rotated). None of the decoders will get defective, but they will malfunction or not function at all as shown below:

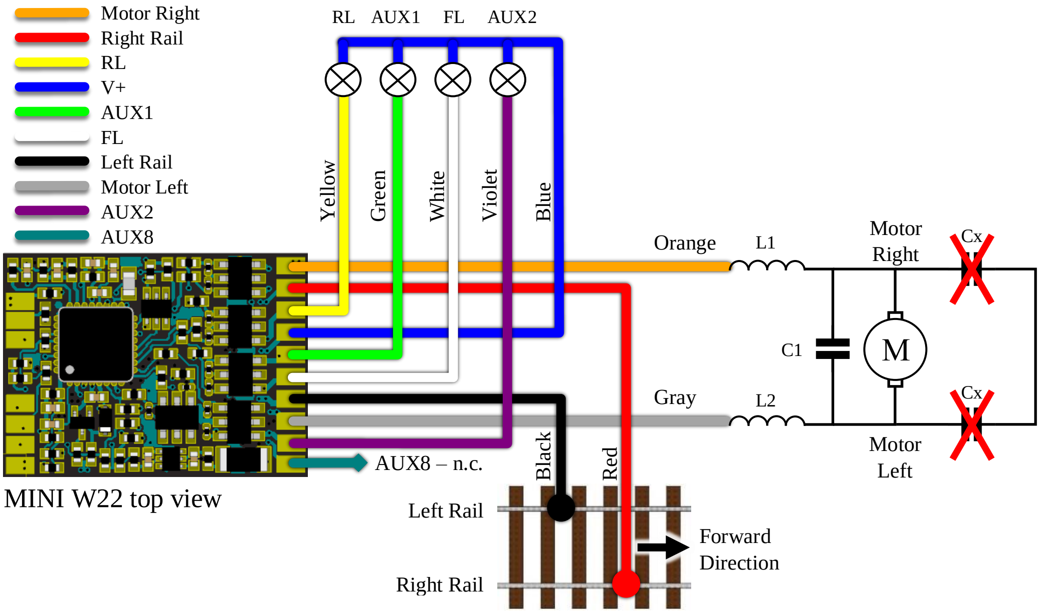
- NEXT18 – Will work but travel direction and directional lights will be reversed
- NEM651 – Will not work at all
- NEM652 – The travel direction will be reversed, and the directional lights will not work.

In the DC locomotives that are not prepared for digitization, a wired decoder (which has only connection wires) must be chosen and installed. The wires are color coded (as in the NMRA standard) and a connection example with MINI W22 is shown in Figure 6:

First determine the front of the locomotive (usually marked with F or 1). It is likely that certain capacitors and/or coils are soldered to the motor as in Figure 6: The Cx capacitors must be removed if present and C1 should be checked that it is not exceeding 47nF. If it is, it must be changed with one up to 47nF capacitors. L1 and L2 (if they are present) do not have a printed value on their case and depending on their value they can make the decoder malfunction. Still, using a filter on the motor is always a good practice, so we recommended to make it work in the optimum way to remove all the capacitors and inductors and place a train-O-matic Motor Filter (order code: tOm 02020301).

Lighting or other auxiliary loads will be connected between the blue (V+) wire and the corresponding wire to the desired output (FL, RL, AUX1, AUX2, etc.). If these loads are polarized (like LEDs), they must be connected respecting their polarity. The LEDs Anode must be connected to the V+ and the Cathode to the desired output with a current limiting resistor. To be able to accept various loads, the Lokommander II power outputs do not have current limiting resistor. So, if the load is an LED than a current limiting resistor must be used. The value should be chosen between 1kΩ-33kΩ depending on the desired maximum brightness (lower resistor value will determine brighter light).

Figure 6: Connecting a wired decoder



8. Setting up the decoder

Before connecting to the digital Command Station please make sure that:

- All connections were made in the right way
- There are no short circuits or loose/poor connections
- The wiring is not touching moving parts

If the decoder that is to be programmed is already installed in a locomotive, it is recommended to power it on placed on a programming track assuring that it is the only one connected.

The first action after powering on is to perform a reset (write any value but not 128 in CV8) to make sure that factory default values are loaded and to set the desired new address in CV1. The decoder comes with default address 3 that can be changed also to extended address (see Decoder address).

During the read/write process the Command Station is sending the requests and the decoder is sending back an acknowledge pulse that must be >100mA. In very few cases the 100mA is not reached so the Command Station cannot receive the confirmation. CV132 is used to increase the acknowledgement current pulse. Switching bits to 1 will turn on the specific output, so more current will be drawn from the locomotive.

9. Decoder address

The Lokommander II decoder supports either short (1÷127) or long (1÷9999) addresses. The factory default address is short (CV29 factory default value is 10 so bit 5 = "0") and the address is 3 (CV1 = 3). The decoder address is stored in CV1 and can be changed with the Command Station. To change the decoder address to long format, set Bit 5 of CV29 to "1". In this way the decoder will have the long address stored in CV17 and CV18.

The long addresses will be calculated with the following algorithm (in our example we will consider the long address 2000)

- Divide the desired long address with 256 (in our example 2000 / 256 = 7, remaining = 208)
- Add 192 to the result and program it in CV17 (7 + 192 = 199; program the value of 199 in CV17)
- Program the value of the remaining of the division in CV18 (program the value of 208 in CV18)

Program CV29 after the long address is stored in CV17 and CV18. After programming the 3 CVs as described above, the decoder can be accessed with the address 2000. Change bit 5 of CV29 to "0" to switch back to short address mode.

When a value is written in CV1, the consist address is automatically cleared, and the extended address will be automatically disabled!

Consists address is used for trains with more than one motor decoder (and motors). The Command Station must be able to send individual commands as well as global commands to the decoders in the multi-motor trains.

The Lokommander II decoders support the Advanced Consist functions. To activate this feature, the consist address must be set in CV19. When the content of CV19 is different from 0, the decoder will perform functions that are defined in CV21 and CV22 if they are transmitted to the consist address. All other functions will be performed while they are sent to the base address (defined in CV1 or CV17/CV18).

Functions in CV21 (F8-F1), CV22 (F12-F9, F0R, F0F) will not be performed if they are transmitted to the base address. For bit value “0” the function will only be enabled with the individual address, for value 1 the function will only be enabled with the consist address (see Chapter 25).

Example: if we want to use F0F, F0R, F3 and F4 with consist address, the following values are to be written in CV21 = 12 (00001100) and in CV22 = 3 (00000011). Speed and direction commands will be sent to all decoders within the same consist. In this way the headlights (of locomotives) and taillight of carriages can be turned on and off, based on the direction commands sent to the consist addresses, while the interior lights in different carriages can be turned on and off based on their individual base addresses.

Only functions F0, F1-F12 can be used in consist mode. The speed steps setting in CV29 must match the speed step setting of the Command Station for both base and consist addresses.

10. Speed settings

This chapter contains information related to the setting of the minimum, medium and maximum speed, acceleration, and deceleration rates of the locomotive.

The following CVs are used to define the speed related parameters:

- CV2: Motor start speed (the lowest speed step) - V_{min}
- CV5: Motor maximum speed (the highest speed step) - V_{max}
- CV6: Motor medium speed - V_{mid}
- CV3: acceleration rate
- CV4: deceleration rate

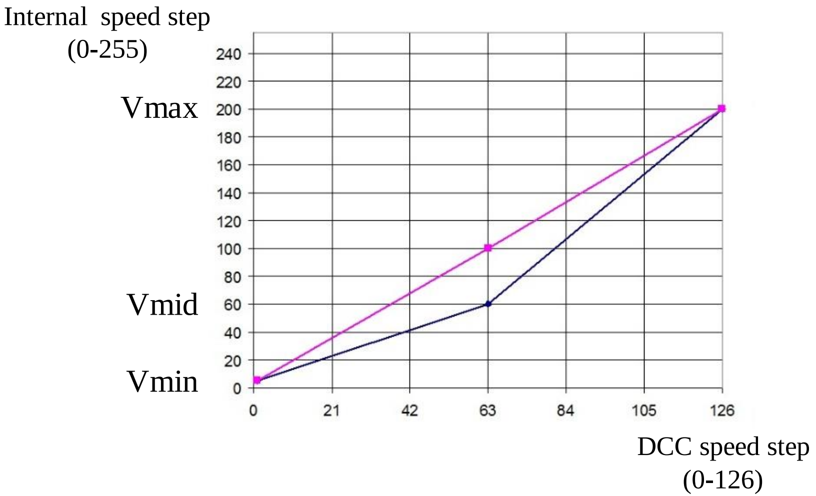
The motor speed can be controlled in 14, 28 or 128 speed steps. Using 128 speed steps is recommended to achieve a smooth speed change.

The speed can be adjusted using 3 points or using a speed table. Both ways are detailed below.

10.1. Linear speed adjustment in 3 points

When bit 4 of CV29 is set to “0” then 3 points speed adjustment mode is selected.

Figure 7: 3 points speed adjustment



V_{min} (CV2) and V_{max} (CV5) are defining the motor speed limits, the first and the last DCC speed step.

V_{mid} (CV6) DCC step position is at the middle between V_{min} and V_{max} as shown in Figure 7: 3 points speed adjustment.

It is preferred that during speed-up or slowing down, the slope between V_{min} and V_{mid} to be smaller so that near slow speeds the changes are not very fast. That means that $V_{mid} < (V_{min} + V_{max}) / 2$, the blue trace in Figure 7: 3 points speed adjustment.

Setting V_{mid} to 0 is like setting $V_{mid} = (V_{min} + V_{max}) / 2$ so the motor speed will change with a linear slope from V_{min} to V_{max} or V_{max} to V_{min} – the purple trace in Figure 7: 3 points speed adjustment.

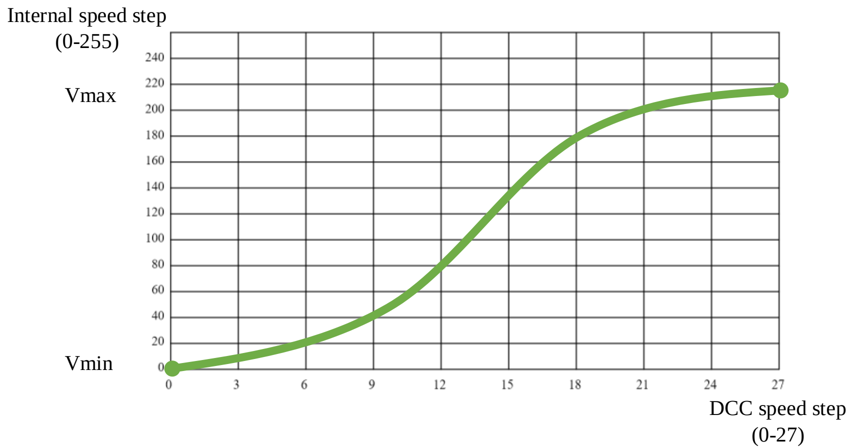
10.2. Tabular speed adjustment in 28 steps

When bit 4 of CV29 is set to “1” than tabular speed adjustment mode is selected (Figure 8: Tabular speed adjustment).

The motor speed steps are defined and stored in CV67÷CV94. Any speed curve shape can be defined.

CV66 for the forward trim and CV95 for the reverse trim are to be used for fine tuning and speed differentiation by the travel direction. With the default 0 values, these CVs have no effect. For other values, the speed is weighted (multiplied) by the CV value/128. If CV66 (95) = 128 the speed will not be changed. For values below 128, the actual speed will decrease, for higher values than 128 the actual speed will increase.

Figure 8: Tabular speed adjustment



To achieve a more realistic behavior of the railway models, the motor acceleration and deceleration can be differentiated by the travel direction using CV152 and CV153. These CV-s have the factory default value of "0" that means that CV3 and CV4 values will be used also for reverse direction. If CV152 and/or CV153 are not "0" then that value will be used for the reverse direction.

11. Motor Control

The Lokommander II decoders have Proportional–Integral–Derivative (PID) motor control loop that is using the Back Electro-Motive Force (BEMF). This is commonly known as "load compensation". PID controller can be enabled or disabled with Bit0 of CV60. The factory default value is “1” and that means PID controller is enabled.

The motor is connected to the horizontal branch of a H-bridge build with 4 FETs (like in Figure 9) and the power feed is through one of the diagonals (M1-M4 or M2-M3). The decoder microcontroller is driving the FETs with fixed frequency and using pulse width modulation (PWM). The PWM frequency is set by Bit7 of CV60. Factory default value is “0” and frequency is 32kHz. For “1” the frequency is 16kHz.

Figure 9: Generic H bridge

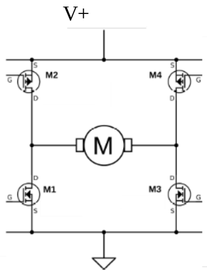
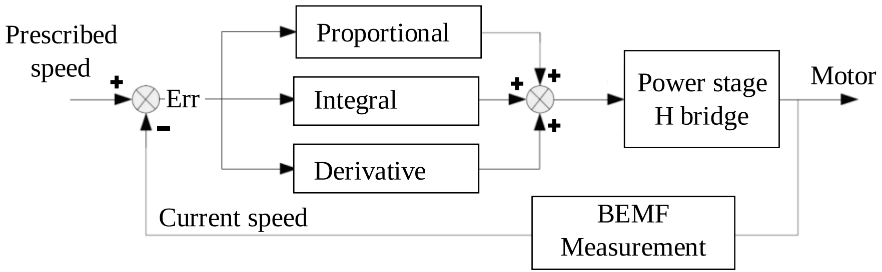


Figure 10: Speed control loop



The PID controller is implemented according to the block diagram above:

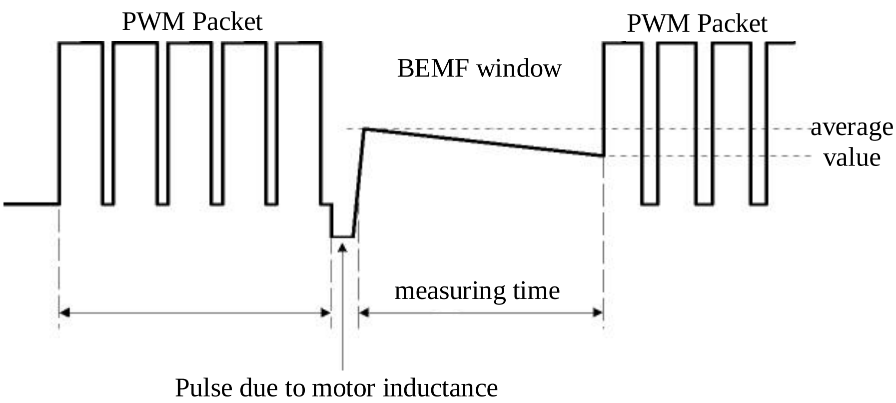
The reference signal (Prescribed Speed) is permanently compared to the current speed, and the resulting error signal (Err) is processed by the PID controller and will adjust the motor speed by changing the PWM duty cycle so that the error (the difference between Prescribed and Current speed) will be minimal.

To determine the current speed, the motor power is interrupted for short periods of time and the BEMF voltage is being measured. This voltage is directly proportional to the speed of the motor and is compared to the prescribed speed so the error signal will be equal to the difference between the two values.

The motor power interruption is named BEMF window (see Figure 11: BEMF window). When speed measurement is performed the motor is not powered on, so his speed is very likely to decrease. For this reason, it is recommended to limit the amount of BEMF windows/time and try to make the window as narrow as possible. Since the electrical motor is mainly inductive, when the power is removed the resulting voltage pulse can alter the BEMF measurement. That is why the BEMF measurement will start with a certain delay after the motor power is off (as in Figure 11: BEMF window). The more poles the motor has the shorter the voltage pulse will be.

Newer 5 pole motors will have shorter motor inductance pulses so the BEMF window size can be decreased, and the vehicle will lose less speed due to power interruption.

Figure 11: BEMF window



In Figure 11: BEMF window we can see that the BEMF voltage value is decreasing in the measuring time window. The reason is that the BEMF voltage value is proportional with the motor speed and since the motor is not powered on the speed will decrease in time. The speed decrease is also proportional with the motor total load that include gears, locomotive itself, attached wagons, etc. The BEMF voltage is measured several times and the current speed value will be obtained after averaging the measured values.

Factory default settings ensure proper operation in most applications, but fine tuning on the setup is recommended. The way to fine tune the settings is described below.

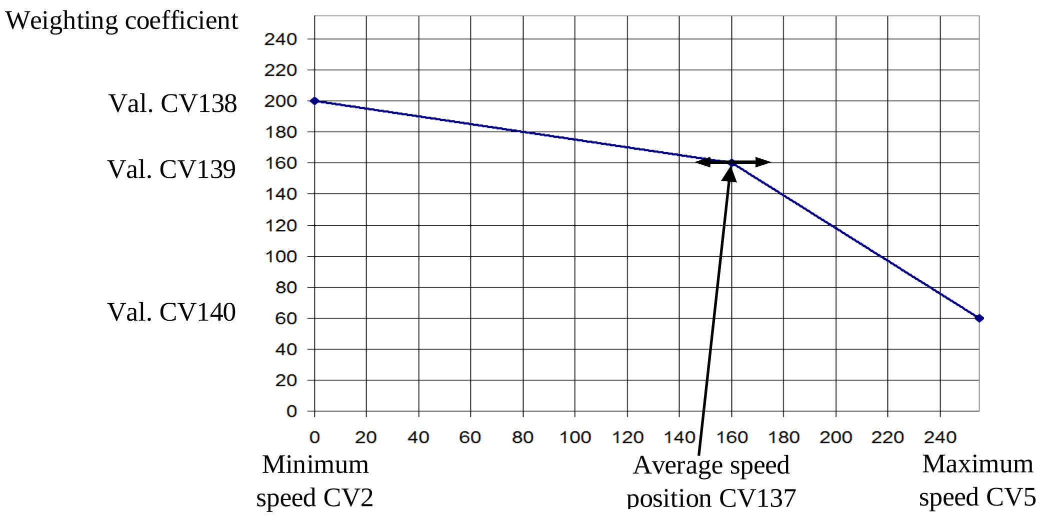
Lokommander II decoders motor control algorithm can be selected from CV9. The factory default value is 3. This value recommended to be used with most locomotive models, ensuring smooth running without leaps for all speed steps (using 128 speed steps is recommend for optimal BEMF / load compensation). For each value of CV9 (0÷8) corresponds a set of internal control parameters which are not accessible to be changed by the user.

CV9 values vs. locomotive type:

- 0÷2 for low inertia locomotives: Faulhaber motors, small locomotives, etc.
- 3÷5 for generic motors
- 6÷8 for high inertia motors
- 9 manual mode

By selecting a standard set, the user can access the coefficients of the PID controller (CV61, 62, 63) and a new set of parameters introduced in Lokommander II decoder generation: load compensation weight coefficients (CV137, 138, 139, 140). Practically, these weighting coefficients can determine how strong the load compensation is, depending on the speed of the locomotive. The characteristic of the load compensation is determined by two segments with negative slope, the first between V_{min} (CV2) and V_{mid} (CV137), the second between V_{mid} (CV137) and V_{max} (CV5) as shown in Figure 12: Load compensation.

Figure 12: Load compensation



CV138 sets the load compensation coefficient at the minimum speed (defined in CV2), and CV140 at the maximum speed (defined in CV5). At the medium speed of CV137 (which is different from the medium speed in CV6) the weighting is set by CV139. The maximum weight is when CVs 138/139/140 value is 255.

Load compensation is mostly effective at low speeds and at high speed it is mostly ineffective. Therefore, CV140 can be decreased without causing trouble running the motors. Experimenting with different values in CV138 (load compensation coefficient), CV137 (medium speed) and CV139 (weighting coefficient) can deliver good results even for problem motors. The above is recommended keeping CV9 and PID CVs default values.

Setting CV9 to “9” will switch to manual access and the following parameters can be tuned:

- CV64: PID error limit, ensures the limitation of integral term in the PID loop without reducing its response time. Value range is 1÷10. Jerky driving and low motor power mean CV64 value is too low. Excessive motor noise means CV64 value is too high.
- CV128: the number of PWM packets after which a BEMF window is inserted. The factory default value is “1”. Higher value is recommended for large, high inertia motors. The range of numeric values is limited to max “4”. The duration of a PWM packet is about 8ms.
- CV129: The averaging amount during BEMF measuring. The factory default is 6. For better quality motors with multiple poles, the averaging amount may be decreased. Value >10 is not practical.
- CV130: the BEMF measurement delay. The purpose is to avoid measurement errors due to the pulse generated by the motor inductance after power-off. The factory default is 6. This must be reduced

carefully because at a certain moment the measuring time can start during the motor pulse that will alter the measurement accuracy. It is recommended to experiment lowering the value only for high quality multi-pole motors (such as the Faulhaber or Maxon). For lower quality motors (such as the 3-pole Piko, Hobby category locomotives) increase of the value might be required.

Table 2 contain the values range of CV9. For CV9 = “9” (manual mode) the CV128/129/130 value range is mentioned. Setting the values outside the range will not cause decoder malfunction because he is limiting the values entered in the CVs only at the specified range.

The motor driver circuit is equipped with short circuit and overcurrent protection. The protection is triggered by the first current pulse which exceeds the detection threshold. False short circuit detections may happen due to current spikes on some motors. A current spike counter can be enabled, which counts the number of short spikes before protection is triggered. This number can be set in CV211. The default value is zero, so protection is triggered after the first spike. The recommended range is 0-10. A much higher value can lead to destruction of motor driver in case of short circuit.

Table 2: Parameter set values for different CV9 values

CV9	CV128 PWM packets number	CV129 BEMF averaging number	CV130 BEMF measurement delay	Note
0	1	4	1	
1	1	4	2	
2	1	6	2	
3	1	6	2	Default
4	2	4	1	
5	2	4	2	
6	2	6	2	
7	2	6	4	
8	2	8	6	
9	1÷4	1÷10	1÷12	

12. Controlled stops

12.1. Constant braking distance (CBD)

Constant Braking Distance stopping allows the locomotive to stop when a command is received on a fixed distance regardless of the travel speed. Stopping can be triggered by 3 factors:

- entering in a sector with asymmetric DCC signal (ABC)
- entering in a DC sector (Analogue operation (DC))
- receiving a zero-speed command

Stopping with CBD when receiving a zero-speed command is activated from CV27 Bit7 = 1.

There are two ways to stop on controlled distance:

12.1.1. Fixed decelerations stop

After receiving the stop command, the locomotive traverses a distance calculated at the current speed, then stops with the deceleration set in CV145 and CV146. Runtime with initial speed can be supplemented with a variable delay set in CV148 and CV149 by the formula: Delay = CV148 * 8ms.

12.1.2. Variable decelerations stop

After receiving the stop command, the locomotive will stop with the calculated deceleration based on the speed at the time of receiving the stop command and the stopping distance set by CV150 (CV151). This is a relative distance, being the multiple of the minimum braking distance from the maximum speed obtained with deceleration = 1

If the CV150 is zero (default value), the fixed deceleration stop from CV146 is selected. If CV146 is also zero (default value is 65), constant distance braking will be disabled. If both CVs are different from zero, the variable deceleration stop set in CV150 (CV151) has priority .

All stopping parameters can be differentiated for both travel direction. Thus, there are two sets of CVs, one for each direction. If the CV for reverse direction is zero, the forward CV value will be used for both directions.

Controlled braking distance stopping is inhibited by "Shunting" (default F3) or CBD-OFF (default F5). Please see the special functions.

12.2. Detecting asymmetric DCC signal (Lenz ABC)

The asymmetric DCC signal allows precise stopping in front of the signals or in the stations and then passing in the opposite direction. The locomotive decoder receives information about the status of the signal depending on the travel direction via the braking modules, which supply the braking section in front of the stop. Two different information can be transmitted: "Stop" or "Slow Approach".

Upon receiving the "Stop" command, the locomotive will initiate the controlled distance stopping procedure (Constant braking distance (CBD)), or if it is disabled, the locomotive will stop with the CV4 (CV153) deceleration. When receiving the "Slow Approach" command, the speed will be reduced to the value set in CV143 (CV144). If the value of this CV is greater than the actual internal speed, no speed change will occur.

ABC activation is made from CV27:

- Bit0 = 1: Allows ABC signal detection when the right track is more positive
- Bit1 = 1: Allows ABC signal detection when the left track is more positive

ABC typically only works in one direction, but activation for both directions is permitted (except Shuttle train operation).

The sensitivity of the ABC voltage difference detection between the two rails can be changed from CV141 (turn on threshold) and CV192 (turn off threshold). Additionally, they are two delays: ABC turn on delay CV193 and ABC turn off delay CV194. The ABC break condition is detected if the track voltage difference between the two rails is greater than the turn on threshold (CV141) for more than the ABC turn on delay time (CV193). The ABC break condition is terminated if the track voltage

difference between the two rails is smaller than the turn off threshold (CV192) for more than ABC turn off delay (CV194).

The two delays, similarly with other delays, can be set in 8ms steps. The default value for both is 25, meaning 0,2s. The two threshold values are set in CV141=10 (12/decoder type dependent) and CV192=8. It's mandatory that CV141 to be greater than CV192.

If the initial value does not provide good results at ABC detection, the optimal value of thresholds can be changed experimental in the range 8-16. A too low value causes undesired erroneous detection, and too high will make the detection difficult or even impossible.

12.3. Shuttle train operation (Penduling)

Shuttle train operation (also known as "Penduling" or "Push-pull") allows you to cycle on a route between two terminal stations repeatedly. Stopping and changing the direction of travel is done when receiving ABC commands at terminal stations. DCC commands only determine the travel speed and the enabled functions. There are two variants described below.

12.3.1. Without intermediate stops

The Shuttle train operation without intermediate stops requires two separate sections at the ends of the route that generate an ABC "Stop" signal corresponding to the direction the locomotive approaches (the more positive right track). The locomotive arriving in the terminal section stops, inverts the direction (including directional lights) and, after the waiting time, starts in the new direction. Activation is made by Bit4 (CV122)

= 1. CV142 will set the waiting time, in steps of 1 second. On the way may coexist ABC "Slow Approach" sectors, where the locomotive will slow down.

12.3.2. With intermediate stops

The Shuttle train operation with intermediate stops requires two separate sections at the ends of the route that generate an ABC "Slow approach" signal corresponding to the direction the locomotive approaches. In the intermediate sectors where the stop is desired, the ABC "Stop" signal will be activated corresponding to the direction from which the locomotive approaches. The intermediate stop will last until the ABC "Stop" signal will disappear. Activation is made by Bit5 (CV122) = 1. CV142 will set the waiting time (in terminal stations), in steps of 1 second.

For Shuttle train operation, ABC signal detection must be activated in CV27 for one direction.

ABC activation is not allowed for both directions, this will lead to erroneous operation of " Shuttle train operation! Simultaneous activation of Bit4 and Bit5 (CV122) is not allowed!

It is recommended to activate one of the constant braking distance methods to ensure that the locomotive stops every time in the same place, regardless of travel speed.

12.4. Special brake functions

The locomotive can be stopped, only slowed down, or the deceleration rate can be altered, activating one or more of the configurable three special brake functions. The **functions number** can be set in CV197, CV198 and CV199. By default, this CVs have the value 255, meaning that no function key will activate this special brake functions. The possible range is F0 … F28 (CV values: 0-28).

Activating one of the configured special brake functions, will have the following results:

- The deceleration rate from CV4 will be reduced with a percentage configured in CV200 … CV202, corresponding for each brake function. The default value of these CVs is zero. Value zero means no reduction and value 255 means 100% reduction. If more brake functions are activated simultaneously the corresponding reduction percentage will be cumulated, without exceeding 100%.
- The locomotive speed will be reduced to a maximal value set in CV203 … CV205 corresponding for each brake function. The default value for these CVs is zero, meaning that locomotive will stop if any brake function is activated. If the actual speed is lower than the activated brake speed, no speed change will occur. If more brake functions are activated, the lower speed will be considered.
-

If consist mode is used, the brake functions will be active for the decoder individual address.

13. Function outputs

Function outputs can command different loads such as LEDs, bulbs, smoke generator, electromagnetic couplers, etc. Lokommander II decoders have 2 kinds of outputs: power outputs and logic outputs.

- The power outputs have a transistor that connects the output to the ground (-) when the output is enabled. The loads will be connected between the power output and V+ (common +).
- The logic outputs supply a voltage of about + 5V when enabled, otherwise they are connected to the ground. The logic outputs cannot exceed the maximum current of 5mA, otherwise the decoder can get defective. A logic output can be used to command 1-2 LEDs with current limiting resistors, or via an external transistor to command larger loads. To supplement the number of outputs, the SUSI interface can be disabled (CV122 Bit0 = 0) and the corresponding pins can be used as 2 logical outputs. By factory default, they are configured as logical outputs. To use them for the SUSI interface, bit 0 and 1 of CV122 must be set to 1.

Some decoders have outputs that are not available on the connectors. General description of Lokommander II decoders and Table 3: Output configuration is showing the available auxiliary outputs (dotted lines in the photos) that can be used with soldered wires.

Table 3: Output configuration

Format / Name/ Connector			FL	RL	AUX 1	AUX 2	AUX 3	AUX 4	AUX 5	AUX 6	AUX 7	AUX 8	AUX 9	AUX 10
MICRO	N18	NEXT 18	P	P	P	P	L, S	L, S	L	L				
MICRO	6P	NEM 651	P	P	O, P	O, P	O, P	O, P	O, L, S	O, L, S				
MICRO	6P90	NEM 651	P	P	O, P	O, P	O, P	O, P	O, L, S	O, L, S				
MICRO	W6P	NEM 651 & Wires	P	P	O, P	O, P	O, P	O, P	O, L, S	O, L, S				
MICRO	W	Wires	P	P	O, P	O, P	O, P	O, P	O, L, S	O, L, S				
MICRO	6P90R	NEM 651	P	P	O, P	O, P	O, P	O, P	O, L, S	O, L, S				
MINI	P12	PLUX 12	P	P	P	P	O, L, S	O, L, S						
MINI	P16	PLUX 16	P	P	P	P	L, S	L, S						
MINI	W6P	NEM 651 & Wires	P	P	O, P	O, P	O, L, S	O, L, S						
MINI	W8P	NEM 652 & Wires	P	P	P	P	O, L, S	O, L, S						
MINI	M21	MTC 21	P	P	P	P	L	L	L, S	L, S				
MINI	M21S	MTC 21 & SUSI	P	P	P	P	L	L	L, S	L, S				
MINI	M21SP	MTC 21 & SUSI	P	P	P	P	P	P	L, S	L, S				
MINI	P22	PLUX 22	P	P	P	P	P	P	P	P	P	P	L, S	L, S
MINI	W22	Wires	P	P	P	P	O, P	O, P	O, P	O, P	O, P	P	O, L, S	O, L, S
MINI	W22M	Wires	P	P	P	P	P	P	P	P	P	P	O, L, S	O, L, S

P – Power output; L – Logical output; O – Optional output, accessible by soldering an additional wire; S – Output shared with SUSI.

For decoders with a maximum 8 outputs, we use a simplified mapping slightly different from the NMRA standard, which offers higher flexibility (any function can control any output).

Table 4: Output mapping

Function	CV nr.	Default value	AUX6	AUX5	AUX4	AUX3	AUX2	AUX1	RL	FL
F0f	33	1	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F0r	34	2	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F1f	35	1	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F1r	36	1	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F2	37	2	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F3	38	4	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F4	39	8	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F5	40	16	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F6	41	32	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F7	42	64	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F8	43	128	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F9	44	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F10	45	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F11	46	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F12	47	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)

NMRA standard mapping is used for 10 outputs MINI P22, W22 and W22M decoders.

Table 5: Output mapping for MINI P22, W22 and W22M

Function	CV nr.	Default value	AUX 10	AUX 9	AUX 8	AUX 7	AUX 6	AUX 5	AUX 4	AUX 3	AUX 2	AUX 1	RL	FL
F0f	33	1					Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F0r	34	2					Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F1f	35	4					Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F2	36	8					Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F3	37	16					Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F4	38	4		Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)			
F5	39	8		Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)			
F6	40	16		Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)			
F7	41	32		Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)			
F8	42	64		Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)			
F9	43	16	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)						
F10	44	32	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)						
F11	45	64	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)						
F12	46	126	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)						
F1r	47	4					Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)

Mapping in grey fields is impossible. All the other mappings are possible. The factory default mapping is colored in green.

So, for the 10-output decoder the mappings are limited (as for NMRA) as follows:

- On F0f, F0r, F1f, F1r, F2 and F3 the FL, FR and AUX1-AUX6 outputs can be mapped, where "f" and "r" of F0 and F1 represent f=FWD/forward and r=REV/reverse
- On F4-F8 the AUX2-AUX9 outputs can be mapped
- On F9-F12 the AUX5-AUX10 outputs can be mapped

The PWM factor of the 12 outputs is set in CV48-59 (see Table *9*: CV table).

CV112 is controlling the start-up time (Fade-In), respectively CV113, the fall-off time (Fade-Out) of the PWM signal applied to outputs. These times can be set in steps of 8ms and represent the time in which the output PWM fill factor rise from 0 to 255, or vice versa. If CV48-59 establishes a fill factor lower than the maximum value 255, the rise and fall times decrease proportionally. These two parameters are common to all outputs. This function is useful when we want to simulate the slow turn on of incandescent bulbs.

If we want any output to be commanded with a continuous signal (without variable fill factor PWM) in CV117(CV185) we can set to value 1 the bit corresponding to the desired output(s). On Lokommander II versions with more than 8 outputs, the continuous command of outputs 9-12 can be set from CV185 bits 0-3.

Starting with software version 3.5.207, functions F0 (f / r), F1 (f / r) and F2-F12 can be configured to inhibit one or more output(s) FL, RL, AUX1, ... AUX 6.

Table 6: Inhibit function mapping

Function	CV nr.	Default value	AUX6	AUX5	AUX4	AUX3	AUX2	AUX1	RL	FL
F0f	166	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F0r	167	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F1f	168	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F2	169	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F3	170	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F4	171	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F5	172	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F6	173	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F7	174	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F8	175	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F9	176	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F10	177	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F11	178	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F12	179	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F1r	180	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)

According to Table 6, if we want that a function to inhibit one of the outputs, the corresponding output bit must be set to 1 in the CV corresponding to the function. Functions F0 and F1 can inhibit FL, FR, AUX1, ... AUX6 outputs depending on travel direction. CVs 166/168 set the inhibition of some outputs if the locomotive moves in the forward direction, respectively in CVs 167/180 sets the inhibition of some outputs if the locomotive moves in the reverse direction.

All outputs can be conditioned with the turn on of function F0. This can be done setting the corresponding bit of CV195/CV196. The association between bit position and the corresponding output is given in Table 7. By default, all these bits are zero, no conditioning is turned on. If any of these bits is set to 1, the corresponding output will turn on only if the (previously) mapped function is turned on together with the F0 function.

All outputs can be turned on and/or off with a delay specified in CV186 (turn on delay) and CV187 (turn off delay). These delays are common for all outputs and can be set in 8ms steps. The maximal possible delay value is 8*255 = 2040ms, approximately 2 second. To establish which output will use delays the following CVs are available: CV188, CV190 for turn on delay and CV189, CV191 for turn off delay. The meaning of bits from these CVs can be found in Table 7.

Each output can have direction dependency using some inhibit CVs. In CV206 and CV207 can be set which output will be inhibited in forward direction and in CV208 and CV209 can be set which output will be inhibited in reverse direction. The association between the bit position and the corresponding output is given in Table 7. For example, if we want AUX5 to be active only in forward direction, we must set bit6 in CV208 to inhibit AUX5 in reverse direction .

Table 7: Special mapping bit meaning

Bit position	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
CV195, CV188, CV189, CV206, CV208	AUX6	AUX5	AUX4	AUX3	AUX2	AUX1	RL	FL
CV196, CV190, CV191, CV207, CV209	-	-	-	-	AUX10	AUX9	AUX8	AUX7

14. Analogue operation (DC)

The decoder allows the locomotive to run even with classical speed controllers providing continuous power (DC). They can be two types: filtered and pulsed (PWM).

Set Bit 2 in CV29 to “1” to enable DC operation.

CV13 and CV14 are defining the available functions in DC mode. A value of “1” means the function is enabled in DC mode. Table 8 is showing the bit meaning.

Table 8: Analog function mapping

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CV13	F8	F7	F6	F5	F4	F3	F2	F1
CV14	F14	F13	F12	F11	F10	F9	RL	FL

There are two modes of analogue (DC) operation:

14.1. Analogue mode 1

Mode 1 can be used with DC controllers, which provides filtered continuous voltage. Depending on the rails voltage, the desired speed is set and motor control via the PID loop is provided. You can get a smooth run even at very low speeds, as in DCC mode. For instance, if nominal voltage is applied to the tracks, the motor will speed-up with the settings in CV3 (CV152).

The correlation between rail voltage and travel speed is linear as in the 3 CVs below:

- CV161: starting threshold: the motor starts when the rails voltage reaches this value.
- CV162: turn-off threshold: the motor stops when the rail voltage drops below this value, the value can be lower than the starting threshold.
- CV163: maximum speed: at this rail voltage will reach the maximum speed.

The value written in these CVs is calculated by multiplying the value of the desired voltage by 10. Example: for the maximum voltage of 14V, CV162 value will be 140.

This analogue mode will not work properly with pulsed (PWM) speed controllers!

To enable analogue mode 1, CV164 value must be “0”.

14.2. Analogue mode 2

In this mode, the motor is controlled by a high frequency pulsed (PWM) voltage. The PWM duty cycle is fixed and set by CV164. For the maximum value of 255, virtually all rail voltage is applied to the motor. If a lower value is set, the voltage applied to the motor will be less than that in the rails (allows the use of motors with lower rated voltage). To enable analogue mode 2, CV164 value must be different of “0”.

14.3. Controlled stop on DC sector

Continuous current can also be used in conjunction with DCC to supply DC brake sections. Thus, if a DCC-powered locomotive reaches a DC sector, it will stop if the following conditions are met: Bit4 or Bit5 in CV27 are "1", Bit 2 in CV29 is “0” and the voltage in the rails is higher than the threshold set in CV165.

The threshold set in CV165 (default value 100 => 10V) is useful when using a power pack simultaneously with the DC-brake function. So, if the supply voltage is below the threshold, we are in SPP mode, and the locomotive will stop after the time set in CV123. If the voltage exceeds the threshold, the DC brake function is activated, and the locomotive will stop at a controlled distance (see Constant braking distance (CBD))

15. Bidirectional communication (RailCom)

"Bidirectional" means that the data transfer is two ways, not only from the Command Station. The decoder will send data to the Command Station if is requested. The decoder can send messages such as confirmation of receipt of commands, address, actual speed, internal temperature, load and other status information.

The RailCom operating principle is based on the introduction of a window ("break") by the Command Station at the end of each DCC package where the power supply is interrupted and the two lines are short-circuited. In these windows the decoders send a few bytes of data that are received by a detector connected between locomotive and Command Station or by Command Station itself.

The data packet is divided into two channels. On the first channel, the address (short, long, or consist) of the decoder is transmitted. On the second channel, CV handling POM responses are delivered (reading, writing result).

RailCom communication can be deactivated from CV29-Bit3 ("0" - RailCom disabled, "1" - RailCom the enabled). Channels 1 and 2 are enabled in CV28 Bit0 and Bit1.

16. Special functions

By calling our special functions we can get information about:

- The internal temperature of the decoder
- The quality of the received DCC signal.
- Number of hours and minutes of operation
- The time stamp (hour) at which the last locomotive maintenance was performed.

To save the values of these parameters in the nonvolatile memory (EEPROM) of the decoder, the saving function, set in CV213, must be activated. The default value is 28, meaning that the save operation will be performed in the moment when F28 is activated. To save again (overwriting the previous values) F28 must be deactivated and activated again. The saving function can be modified from CV213 in the range F0 – F28. If value greater than 28 is configured in CV213, no saving will be performed for any function activation.

Without activating F28 (On, then Off), the values in the corresponding CVs are not updated!

The internal (saved) temperature of the decoder can be read from CV133. The temperature is given in Celsius degrees.

The DCC Signal Quality Indicator (QoS = Quality of Service) is stored in CV135 as percentage (in the range 0-100%). CV135 will store the minimum value of the QoS. To reset the value, 100 [%] is to be written in CV136. After that call the save function via F28 On, F28 Off to get the latest QoS value.

The operating hours and minutes are stored in CV156, 157 and 158 as below:

- **The number of operating minutes is stored in CV156**
- **The number of operating hours is the sum of the values stored in CV157 multiplied with 256 and the value of CV158 (call the save function with F28 On, F28 Off before reading the values).**

Maintenance period:

The decoder may retain the time stamp of the locomotive maintenance and may indicate if the set number of hours since the last maintenance is exceeded.

This function can be activated and configured in CV154 (see Table *9*: CV table). The maintenance interval is specified in hours in CV155. The factory default value is 40 hours. The value can be changed by the user in the range 0-255. After resetting the decoder, the value of CV155 will be 40 (hours).

The last maintenance time is stored in CV159 and CV160. The value can be calculated as below:

Hours = (CV159 Value) + 256 * (CV160 Value)

To confirm the maintenance, the so-called pseudo programming is used: the value 128 is entered in CV8 (it is not equivalent to a decoder reset!). As a result of this operation, the maintenance time mark is saved, and the new maintenance interval will be calculated from this time stamp.

If the exceeded maintenance interval has been signaled by setting CV30 bit 3, after maintenance confirmation the CV30 must be reset (to “0”). The CV30 is not automatically erased by the maintenance confirmation procedure.

The function number which switches On and off the Shunting Speed is set in CV114. The value range is 0...28. Default is F3 (CV114 = 3). The value 255 is a neutral value (no function will be allocated to the Shunting Speed).

The function number which switches off the Acceleration/Deceleration, is set in CV115. The value range is 0...28. Default is F4 (CV115 = 4). The value 255 is a neutral value (no function will be allocated to switch off the Acceleration/Deceleration).

The function number which disables the Constant Braking, is set in CV116. The value range is 0...28. Default is F5 (CV116 = 5). The value 255 is a neutral value (no function will be allocated to disable the Constant Braking).

17. Electrical Coupler Configuration

The Lokommander II decoder allows the use of any physical output for the action of electromagnetic couplers. If a logical output is chosen it is necessary to use an external transistor, the output supplying an insufficient current for actuating the coupler. The Krois® and Roco® couplers require a high-frequency PWM signal supply to avoid burning of the coils of the couplings. The automatic decoupling function of the decoder provides this command signal.

The automatic decoupling function can only be activated with the stationary locomotive.

The automatic decoupling function is a physical function (not logical, such as shunting speed, inactivation of acceleration and deceleration, etc.) and must be configured accordingly. Follow these guidelines for the configuration:

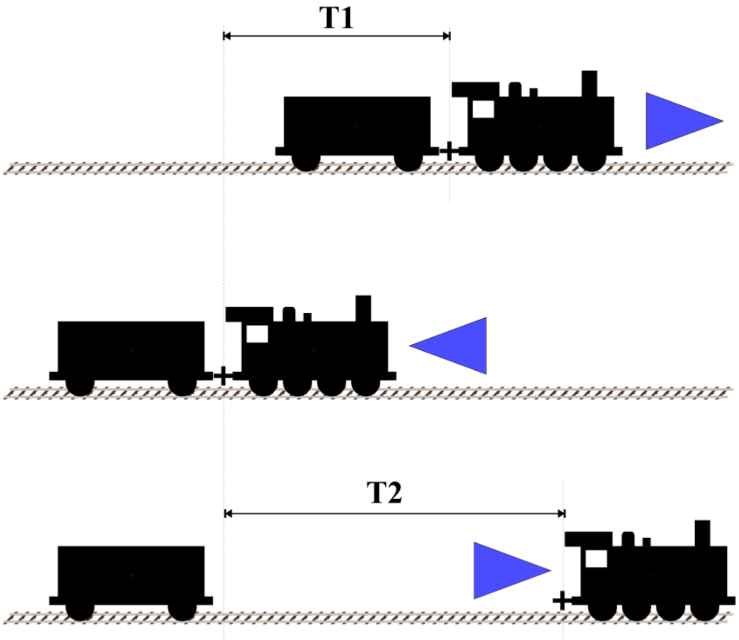
Choose an F function to be used for the automatic decoupling function (it can be a function used for other commands, for example sound).

For the selected function (CV33÷CV47) the mapping of the physical output has to be made (the output where the coupler is physically connected). As an example, if we choose the F8 function for automatic decoupling and the coupler electromagnets are connected to the physical output AUX2, CV43 must have the value 8, meaning that when the F8 function is activated (see Table 4: Output mapping). To activate AUX2 on F8, CV118 must have the value 4. CV118 stores the output number as follows: 1 for FL, 2 for RL, 3 for AUX1, 4 for AUX2 ... and 10 for AUX8.

CV131 is storing the coupling type (DC or PWM) as follows: Bit0 = “0” output HF PWM; Bit0 = “1” continuous output. Bit1 of CV131 is defining the motor control mode during uncoupling. If Bit1 = “0” the motor will be turned on instantly, making a sudden movement, if Bit1 = “1” the motor will be controlled by the PID loop with deceleration acceleration according to CV3 and CV4.

CV119÷CV121 are used to fine tune the decoupling process. CV121 is storing the maximum travel speed during the decoupling process. If CV121 = “0” then the locomotive will not move, only the electromagnetic coupling will be activated. When activating the uncoupling function, the locomotive will actuate the electromagnetic coupler and it will travel in opposite direction as before, for duration of T1. T1 is defined by CV119 (reverse motion time). So, the travel distance will be defined by CV121 (speed) and CV119 (duration). After T1 the locomotive will stop and reverse the direction again and will travel for a period T2 that is defined in CV120. Again, the travel distance will be dependent of the speed setting (CV121) and travel duration (CV120). All the functions that were activated before performing the decoupling function remain enabled during decoupling. The decoupling function will be automatically turned off after the cycle is complete.

Figure 13: Decoupling



The polarity of the electromagnetic couplers is important. If they are not properly connected, the movement (lift) is reversed!

18. SUSI

All the Lokommander II decoders have SUSI interface available on connector or soldering pads (see General description of Lokommander II decoders). The interface is used for sound or function decoders and the pinout is according to the standards (if connector is placed on the decoder). It is strongly recommended to use the standard wire colors as in General description of Lokommander II decoders to avoid wrong connections.

Attention! Improper connection of the SUSI module may cause damage to the SUSI module

18.1. Programming SUSI modules

Like locomotive decoders, SUSI sound decoders can be personalized by changing some operating parameters. The values of these parameters are stored in CV897 to CV1024. The SUSI sound decoder is programmed via the Lokommander II decoder. Depending on the CV number, the Lokommander II decoder will identify whether this CV should be written or read from a SUSI module connected to the decoder interface. Please refer to the SUSI decoder user manual before programming it.

The SUSI modules CVs can be written either in PT or PoM mode. Because some digital systems allow writing and reading of CVs only up to CV255, a special mechanism for these digital systems has been implemented in the Lokommander II decoder. Two CVs are dedicated to provide access to the higher level of the SUSI modules CVs. CV126 is used as index, and CV127 is used as transport CV. The target SUSI decoder CV number is composed of the value of CV126 + 800. CV127 is the container of the value that is to be written or read to / from the CV126 + 800. Below are 2 examples of read/write procedure.

Examples:

- If you want to write value “1” in CV897 of the SUSI module, you must write “97” (897 - 800 = 97) in CV126 and value “1” in CV127. After entering value “1” in CV127, the Lokommander II decoder will transmit a command on the SUSI interface to the sound module (or the function decoder) to write the value “1” in CV897.
- If you want to read the content of CV 902 from the SUSI module that is connected to the Lokommander II decoder interface, enter the value 102 (902 - 800 = 102) in CV126, and read the CV127 value. This value is equal to the value contained in CVC 902 of the sound module (or function decoder) connected to the Lokommander II decoder.

Bit1 of CV122 will enable/disable the SUSI interface. The factory default setting is SUSI enabled (Bit1 of CV122 = “1”)

19. Using external capacitors or a power pack

It is very likely that the rails and locomotive wheels can become dirty because of dust, grease or other insulating substances that will interrupt the power to get to the decoder and then to the motor. In these cases, the locomotive will stutter or even stop. To avoid this unwanted behavior, uninterruptible power supplies SPP or buffer capacitors (220µF / 25V or for better results higher but not exceeding 2200µF) can be used.

All the Lokommander II decoder have 3 contacts on one side of the PCB to connect a SPP or buffer capacitor. The exact position of the 3 contacts for each decoder type is shown in General description of Lokommander II decoders.

Installing these devices requires quality soldering equipment and experience. Our warranty does not cover defects due to inappropriate interventions or soldering.

SPP (with supercapacitor)

The SPP modules allow locomotives to run for up to 4 seconds without DCC power from rails (depending on locomotive consumption and capacitor charge status). To connect the SPP modules, use the 3 soldering pads located on one of the Lokommander II sides (see General description of Lokommander II decoders). Please refer to SPP user manual for installation instruction.

SPP sources work only in digital mode (bit2 of CV29 = “0”), in combined analogue/digital mode (Bit2 of CV29 = “1”).

SPP startup delay is configurable with CV124. This setting is useful when several decoders with SPPs are present on the track and if all SPPs will start charging immediately after power on, there will be a high current peak that can be interpreted as a short circuit on the rails. The delay is expressed in seconds. The factory default delay setting is 10s.

For safety reasons, this duration of the SPP can be defined in CV123 as a multiple of 16ms. The factory default value is 16 that means that the locomotive will perform an emergency stop after 0.25s (16 * 16ms = 256ms) after the DCC power is not present on the rails. The emergency stop will be performed even if the SPP capacitor is not completely discharged. The locomotive will resume the motion only after the DCC signal is present again.

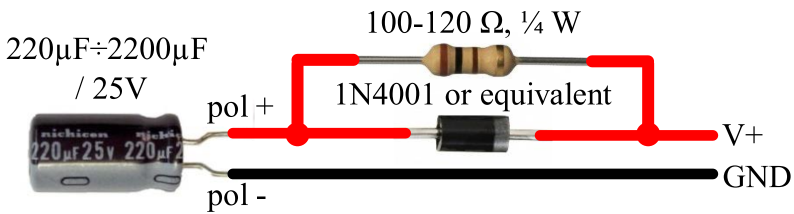
Note that the SPP will require about 300mA to fully charge. The full charging time is up to 2 minutes if the capacitor is completely discharged. Refer to the SPP user manual for more details.

Buffer capacitor

A simple buffer capacitor can be used instead of the SPP. Be aware that this will give way less extra power (depending of the size and amount of capacitors). However, expect fractions of a second.

Two additional components are needed together with the capacitor: a current limiting resistor connected in parallel with 1A rectifying diode as in **Hiba! A hivatkozási forrás nem található.**. Please note that the capacitors and diodes are voltage polarity sensitive.

Hiba! A hivatkozási forrás nem található.



The connections must be done according to the images from General description of Lokommander II decoders. The black wire will be soldered to the GND and the red wire to the V+ (the two extremes from the group of 3 pads reserved to SPP). It is recommended to place the components in heat shrinkable tube or insulating tape for insulation. All the 3 components (capacitor, resistor and diode) are not included in the Lokommander II decoder package, and they must be purchased separately.

The buffer capacitor will prevent CVs to be written to the decoder. Programming CVs will be possible only after removing the buffer capacitor.

By using SPP modules (Smart Power Pack or equivalent) both write and read of CVs is possible without having to remove them. Switching off the SPP source during programming is done automatically by the Lokommander II through the Charge connection.

20. Resetting the decoder

You can reset the decoder to the factory default settings at any time.

When using any DCC Command Station, it is sufficient to enter any numeric value in CV8 (except the value 128). As a result of this reset, all CVs will have the default value again (see the Default Value column in the Table *9*: CV table). Decoders can also be reset using tOm Programmer. For the same result, in the firmware TAB, press the Reset CV-s option. There are 2 CVs that are an exception. Your content will not be erased on reset. These are CV105 and CV106 which are intended to store user specific information (serial number, tag, inventory number, etc.). Your content will be modified by direct writing. Resetting the decoder does not change the content of these CVs.

A firmware upgrade will enter the default values in CV105 and CV106. To preserve their value, make a backup of the CVs before firmware upgrade (using the tOm Programmer)

Resetting the Lokommander II decoder will not reset the CVs of the connected SUSI modules.

21. Secondary address (decoder lock)

When using multiple decoders in the same model or trainset it is useful to use a secondary address so that the decoder can be selected during programming. In this way, any decoder that is in the same model can be programmed on the programming rail without having to be dismantled from the model. The secondary addresses are programmed in CV16 before the decoder is mounted in the model. The value range of the secondary addresses is 1 to 254 (the value 0 means that no secondary address is used). By assigning a different secondary address to each decoder belonging to the trainset, only the decoder for which CV15 = CV16 will be programmed. In this way, by writing each of the individual addresses in CV15 one after the other, we can program several decoders independently, even if they are on the programming rail at the same time. The decoders for which the CV15 ≠ CV16 will ignore any CV changes or readings.

⚠ The value of 255 for CV15 has a special function. Please see below.

⚠ WARNING: CV16 can be programmed only if the correct value is programmed in CV15.

When using the secondary addresses, it is important to know that the only CV that can be read or written without knowing the secondary address is CV15.

If you accidentally block the decoder by entering an unknown value in CV16, you must enter 255 in CV15. This special function will remove the LOCKED state and will program CV15 with the value of CV16. As a result of writing CV15 with the value 255, the decoder will be UNLOCKED without losing the possibility to LOCK it again by changing the value of CV15.

The value 0 for CV16 means an UNLOCKED decoder, whatever value is written in CV15.

This way of accessing/programming the CVs of the decoder is useful on permanently connected wagons or with permanently connected sets equipped with several decoders and it would be very impractical to program them in a traditional way (on the programming track would be all the existing decoders programmed with the same CV values, which is more of an undesirable thing).

22. Firmware update

You can update the Lokommander II decoder operating software (called firmware) at any time. New firmware versions are used to eliminate errors (bugs) when operating decoders or to implement new functions.

The firmware update can be performed with the tOm Programmer without having to remove the decoder from the locomotive.

The tOm Programmer's operating software and firmware upgrade files can be downloaded from the train-O-matic site. For the firmware upgrade procedure, please refer to the tOm Programmer's user manual.

The current firmware version can be read from the following CVs:

CV253 Firmware version

CV254 Firmware subversion

CV254 Build version, upper byte

CV256 Build version, lower byte

23. Accessories

- tOm SPP can optionally be connected to a decoder and offers up to 4 seconds of energy independence
- The tOm programmer is a PC interface for programming DCC locomotive decoders
- Shine FDT, Shine LT, Shine MICRO are LED modules for lighting locomotives and wagons
- Shine mini/midi/maxi digi/ana are LED circuit boards for the interior lighting of wagons
- TD Maxi, TD Roco are switching decoders for turnouts

For details on accessories and a complete list of railroad products, visit the page:

www.train-o-matic.com

24. Technical support

If you have any questions or suggestions about train-o-matic products, you can write to us at:

support@train-o-matic.com

25. The decoder CV table

The table below contains the full list of available CVs of the Lokommander II decoders with explanations and examples if available. We recommend that you change your CVs only if you are sure of their function and the impact of your action. Incorrect CV settings can negatively affect the performance of the decoder or cause incorrect responses to the commands transmitted to the decoder. The “CV” column contains the CVs number, the “Default Value” column contains the factory default value of the CVs (after a decoder reset, all CVs will have the appropriate value in this column), the column "Value Range" contains the range of usable values for each CV and the "Description" column contains the name and information about the CV function as well as the reference to the related chapter.

In the table below in the description column the bits value is shown in binary mode and, between brackets, in decimal mode. So always when the bit binary value is 0, the decimal value will be also 0.

Table 9: CV table

CV	Default Value	Value Range	Description
1	3	0-127	Decoder Short Address, 7 bits
2	1	1-255	V_{min}
3	5	0-63	Acceleration Rate, 0 = Fastest acceleration
4	5	0-63	Deceleration Rate, 0 = Fastest deceleration
5	200	1-255	V_{max}
6	68	0-255	V_{mid} = [25% - 75%] V_{max}
7	5	-	Software Version - Read-only
8	78	-	Manufacturer ID, decoder reset For each value written, the decoder is reset to the factory setting, except for the value 128, which indicates maintenance, if enabled (see Ch. 20)
9	3	0-9	Motor Control Algorithm, 0-8 User defined = 9 (see Ch. 11 and CV128 ÷ CV130)
11	12	0-255	Maximum time without data reception 12 * 8 = 96ms
12	53	0-255	Power Source Conversion

CV	Default Value	Value Range	Description
13	0	0-255	Analogue Mode, Alternate Mode Function Status F1-F8 Bit 0 = 0(0): F1 disabled in Analogue mode = 1(1): F1 enabled in Analogue mode Bit 1 = 0(0): F2 disabled in Analogue mode = 1(2): F2 enabled in Analogue mode Bit 2 = 0(0): F3 disabled in Analogue mode = 1(4): F3 enabled in Analogue mode Bit 3 = 0(0): F4 disabled in Analogue mode = 1(8): F4 enabled in Analogue mode Bit 4 = 0(0): F5 disabled in Analogue mode = 1(16): F5 enabled in Analogue mode Bit 5 = 0(0): F6 disabled in Analogue mode = 1(32): F6 enabled in Analogue mode Bit 6 = 0(0): F7 disabled in Analogue mode = 1(64): F7 enabled in Analogue mode Bit 7 = 0(0): F8 disabled in Analogue mode = 1(128): F8 enabled in Analogue mode

CV	Default Value	Value Range	Description
14	3 = 1 + 2	0-255	Analogue Mode, Alternate Mode Function. Status F0f, F0r, F9-F14 Bit 0 = 0(0): F0f disabled in Analogue mode = 1(1): F0f enabled in Analogue mode Bit 1 = 0(0): F0r disabled in Analogue mode = 1(2): F0r enabled in Analogue mode Bit 2 = 0(0): F9 disabled in Analogue mode = 1(4): F9 enabled in Analogue mowde Bit 3 = 0(0): F10 disabled in Analogue mode = 1(8): F10 enabled in Analogue mode Bit 4 = 0(0): F11 disabled in Analogue mode = 1(16): F11 enabled in Analogue mode Bit 5 = 0(0): F12 disabled in Analogue mode = 1(32): F12 enabled in Analogue mode Bit 6 = 0(0): F13 disabled in Analogue mode = 1(64): F13 enabled in Analogue mode Bit 7 = 0(0): F14 disabled in Analogue mode = 1(128): F14 enabled in Analogue mode
15	0	0-7	LockCV (secondary address): The programming of the decoder (CV change) is only allowed if CV15 = CV16. CV15 can be written in any situation. See Ch. 21

CV	Default Value	Value Range	Description
16	0	0-7	LockID: To prevent accidental programming use unique ID number for decoders with same address (placed in the same locomotive). Example: 1 - loco decoder, 2 - sound decoder, 3 – function decoder, … See Ch. 21
17	192	192-255	Extended Address, MSB (High Byte)
18	3	0-255	Extended Address, LSB (Low Byte)
19	0	0-127	Consist Address If CV19 > 0: Speed and direction is controlled by this consist address (not the individual address in CV1 or extended address CV17 and CV18); functions are controlled by either the consist address or individual address. (see CV21, CV22 and Ch. 9

CV	Default Value	Value Range	Description
21	0	0-255	Functions defined here will be controlled by the consist address. Bit 0 = 0(0): F1 controlled by individual address = 1(1): F1 controlled by consist address Bit 1 = 0(0): F2 controlled by individual address = 1(2): F2 controlled by consist address Bit 2 = 0(0): F3 controlled by individual address = 1(4): F3 controlled by consist address Bit 3 = 0(0): F4 controlled by individual address = 1(8): F4 controlled by consist address Bit 4 = 0(0): F5 controlled by individual address = 1(16): F5 controlled by consist address Bit 5 = 0(0): F6 controlled by individual address = 1(32): F6 controlled by consist address Bit 6 = 0(0): F7 controlled by individual address = 1(64): F7 controlled by consist address Bit 7 = 0(0): F8 controlled by individual address = 1(128): F8 controlled by consist address

CV	Default Value	Value Range	Description
22	0	0-63	Functions defined here will be controlled by the consist address. Bit 0 = 0(0): F0f controlled by individual address = 1(1): F0f controlled by consist address Bit 1 = 0 (0): F0r controlled by individual address = 1(2): F0r controlled by consist address Bit 2 = 0(0): F9 controlled by individual address = 1(4): F9 controlled by consist address Bit 3 = 0(0): F10 controlled by individual address = 1(8): F10 controlled by consist address Bit 4 = 0(0): F11 controlled by individual address = 1(16): F11 controlled by consist address Bit 5 = 0(0): F12 controlled by individual address = 1(32): F12 controlled by consist address

CV	Default Value	Value Range	Description
27	0	0-7	Automatic Brake Control Settings (ABC) (see Ch. 12) Bit0 = 0(0) Disable ABC detection when right rail is more positive = 1(1) Enable ABC detection when the right rail is more positive Bit1 = 0(0) Disable ABC detection when left rail is more positive = 1(2) Enable ABC detection when the left rail is more positive Bit4 = 0(0) Disable Auto Stop in the presence of reverse polarity DC = 1(16) Enable Auto Stop in the presence of reverse polarity DC Bit5 = 0(0) Disable Auto Stop in the presence forward polarity DC = 1(32) Enable Auto Stop in the presence forward polarity DC Bit7 = 0(0) Disable Auto Stop in the presence of zero speed brake = 1(128) Enable Auto Stop in the presence of zero speed brake
28	3	0-3	RailCom® configuration Bit 0 = 0(0): Chanel 1 Off (Address Broadcast) = 1(1): Chanel 1 On (Address Broadcast) Bit 1 = 0(0): Chanel 2 Off (Data Transmission) = 1(2): Chanel 2 On (Data Transmission)

CV	Default Value	Value Range	Description
29	14 = 2 + 4 + 8	0-63	Decoder Configuration Data Bit 0 = 0(0): Normal travel direction = 1(1): Opposite travel direction Bit 1 = 0(0): 14 speed steps (F0f is controlled by Bit4 in the DCC instructions for speed and direction) = 1(2): 28/128 speed steps (F0f is controlled by Bit4 in the DCC Group 1 instruction) Bit 2 = 0(0): Digital operation only = 1(4): Automatic analogue/digital switching Bit 3 = 0(0): RailCom® switched off = 1(8): RailCom® switched on Bit 4 = 0(0): 3 points peed steps by CV2, Cv5, and CV6 = 1(16): Speed steps from tables CV66÷CV95 Bit 5 = 0(0): Short address (CV1) = 1(32): Long address (CV17, CV18)

CV	Default Value	Value Range	Description
30	0	0-15	Error records for function outputs, motor and temperature monitoring: 0 - No error recorded. 1 (Bit0) - Motor overcurrent error occurred 2 (Bit1) - AUX Output overcurrent error occurred 4 (Bit2) - Overtemperature error occurred 8 (Bit3) - The maintenance period has been exceeded Saved errors can be cleared by programming “0” in CV30

CV	Default Value	Value Range	Description
33	1 = 1	0-255	F0, Forward move mapping. Bit0 = 0(0): FL disabled on F0 (forward) = 1(1): FL enabled on F0 (forward) Bit1 = 0(0): RL disabled on F0 (forward) = 1(2): RL enabled on F0 (forward) Bit2 = 0(0): AUX1 disabled on F0 (forward) = 1(4): AUX1 enabled on F0 (forward) Bit3 = 0(0): AUX2 disabled on F0 (forward) = 1(8): AUX2 enabled on F0 (forward) Bit4 = 0(0): AUX3 disabled on F0 (forward) = 1(16): AUX3 enabled on F0 (forward) Bit5 = 0(0): AUX4 disabled on F0 (forward) = 1(32): AUX4 enabled on F0 (forward) Bit6 = 0(0): AUX5 disabled on F0 (forward) = 1(64): AUX5 enabled on F0 (forward) Bit7 = 0(0): AUX6 disabled on F0 (forward) = 1(128): AUX6 enabled on F0 (forward)

CV	Default Value	Value Range	Description
34	2 = 2	0-255	F0, Reverse move mapping Bit0 = 0(0): FL disabled on F0 (reverse) = 1(1): FL enabled on F0 (reverse) Bit1 = 0(0): RL disabled on F0 (reverse) = 1(2): RL enabled on F0 (reverse) Bit2 = 0(0): AUX1 disabled on F0 (reverse) = 1(4): AUX1 enabled on F0 (reverse) Bit3 = 0(0): AUX2 disabled on F0 (reverse) = 1(8): AUX2 enabled on F0 (reverse) Bit4 = 0(0): AUX3 disabled on F0 (reverse) = 1(16): AUX3 enabled on F0 (reverse) Bit5 = 0(0): AUX4 disabled on F0 (reverse) = 1(32): AUX4 enabled on F0 (reverse) Bit6 = 0(0): AUX5 disabled on F0 (reverse) = 1(64): AUX5 enabled on F0 (reverse) Bit7 = 0(0): AUX6 disabled on F0 (reverse) = 1(128): AUX6 enabled on F0 (reverse)

CV	Default Value	Value Range	Description	
35	1 4 for PLUX22	0-255	F1, Forward move mapping. Bit0 = 0(0): FL disabled on F1fwd = 1(1): FL enabled on F1fwd Bit1 = 0(0): RL disabled on F1fwd = 1(2): RL enabled on F1fwd Bit2 = 0(0): AUX1 disabled on F1fwd = 1(4): AUX1 enabled on F1fwd Bit3 = 0(0): AUX2 disabled on F1fwd = 1(8): AUX2 enabled on F1fwd Bit4 = 0(0): AUX3 disabled on F1fwd = 1(16): AUX3 enabled on F1fwd Bit5 = 0(0): AUX4 disabled on F1fwd = 1(32): AUX4 enabled on F1fwd Bit6 = 0(0): AUX5 disabled on F1fwd = 1(64): AUX5 enabled on F1fwd Bit7 = 0(0): AUX6 disabled on F1fwd = 1(128):AUX6 enabled on F1fwd	F1 mapping for PLUX22 Bit0 = 0(0): FL disabled on F1 = 1(1): FL enabled on F1 Bit1 = 0(0): RL disabled on F1 = 1(2): RL enabled on F1 Bit2 = 0(0): AUX1 disabled on F1 = 1(4): AUX1 enabled on F1 Bit3 = 0(0): AUX2 disabled on F1 = 1(8): AUX2 enabled on F1 Bit4 = 0(0): AUX3 disabled on F1 = 1(16): AUX3 enabled on F1 Bit5 = 0(0): AUX4 disabled on F1 = 1(32): AUX4 enabled on F1 Bit6 = 0(0): AUX5 disabled on F1 = 1(64): AUX5 enabled on F1 Bit7 = 0(0): AUX6 disabled on F1 = 1(128): AUX6 enabled on F1

CV	Default Value	Value Range	Description	
36	1 8 for PLUX22	0-255	F1, reverse move mapping Bit0 = 0(0): FL disabled on F1rev = 1(1): FL enabled on F1rev Bit1 = 0(0): RL disabled on F1rev = 1(2): RL enabled on F1rev Bit2 = 0(0): AUX1 disabled on F1rev = 1(4): AUX1 enabled on F1rev Bit3 = 0(0): AUX2 disabled on F1rev = 1(8): AUX2 enabled on F1rev Bit4 = 0(0): AUX3 disabled on F1rev = 1(16): AUX3 enabled on F1rev Bit5 = 0(0): AUX4 disabled on F1rev = 1(32): AUX4 enabled on F1rev Bit6 = 0(0): AUX5 disabled on F1rev = 1(64): AUX5 enabled on F1rev Bit7 = 0(0): AUX6 disabled on F1rev = 1(128):AUX6 enabled on F1rev	F2 mapping for PLUX22 Bit0 = 0(0): FL disabled on F2 = 1(1): FL enabled on F2 Bit1 = 0(0): RL disabled on F2 = 1(2): RL enabled on F2 Bit2 = 0(0): AUX1 disabled on F2 = 1(4): AUX1 enabled on F2 Bit3 = 0(0): AUX2 disabled on F2 = 1(8): AUX2 enabled on F2 Bit4 = 0(0): AUX3 disabled on F2 = 1(16): AUX3 enabled on F2 Bit5 = 0(0): AUX4 disabled on F2 = 1(32): AUX4 enabled on F2 Bit6 = 0(0): AUX5 disabled on F2 = 1(64): AUX5 enabled on F2 Bit7 = 0(0): AUX6 disabled on F2 = 1(128): AUX6 enabled on F2

CV	Default Value	Value Range	Description	
37	2 16 for PLUX22	0-255	F2 mapping Bit0 = 0(0): FL disabled on F2 = 1(1): FL enabled on F2 Bit1 = 0(0): RL disabled on F2 = 1(2): RL enabled on F2 Bit2 = 0(0): AUX1 disabled on F2 = 1(4): AUX1 enabled on F2 Bit3 = 0(0): AUX2 disabled on F2 = 1(8): AUX2 enabled on F2 Bit4 = 0(0): AUX3 disabled on F2 = 1(16): AUX3 enabled on F2 Bit5 = 0(0): AUX4 disabled on F2 = 1(32): AUX4 enabled on F2 Bit6 = 0(0): AUX5 disabled on F2 = 1(64): AUX5 enabled on F2 Bit7 = 0(0): AUX6 disabled on F2 = 1(128): AUX6 enabled on F2	F3 mapping for PLUX22 Bit0 = 0(0): FL disabled on F3 = 1(1): FL enabled on F3 Bit1 = 0(0): RL disabled on F3 = 1(2): RL enabled on F3 Bit2 = 0(0): AUX1 disabled on F3 = 1(4): AUX1 enabled on F3 Bit3 = 0(0): AUX2 disabled on F3 = 1(8): AUX2 enabled on F3 Bit4 = 0(0): AUX3 disabled on F3 = 1(16): AUX3 enabled on F3 Bit5 = 0(0): AUX4 disabled on F3 = 1(32): AUX4 enabled on F3 Bit6 = 0(0): AUX5 disabled on F3 = 1(64): AUX5 enabled on F3 Bit7 = 0(0): AUX6 disabled on F3 = 1(128): AUX6 enabled on F3

CV	Default Value	Value Range	Description	
38	4 4 for PLUX22	0-255	F3 mapping Bit0 = 0(0): FL disabled on F3 = 1(1): FL enabled on F3 Bit1 = 0(0): RL disabled on F3 = 1(2): RL enabled on F3 Bit2 = 0(0): AUX1 disabled on F3 = 1(4): AUX1 enabled on F3 Bit3 = 0(0): AUX2 disabled on F3 = 1(8): AUX2 enabled on F3 Bit4 = 0(0): AUX3 disabled on F3 = 1(16): AUX3 enabled on F3 Bit5 = 0(0): AUX4 disabled on F3 = 1(32): AUX4 enabled on F3 Bit6 = 0(0): AUX5 disabled on F3 = 1(64): AUX5 enabled on F3 Bit7 = 0(0): AUX6 disabled on F3 = 1(128): AUX6 enabled on F3	F4 mapping for PLUX22 Bit0 = 0(0): FL disabled on F4 = 1(1): FL enabled on F4 Bit1 = 0(0): RL disabled on F4 = 1(2): RL enabled on F4 Bit2 = 0(0): AUX1 disabled on F4 = 1(4): AUX1 enabled on F4 Bit3 = 0(0): AUX2 disabled on F4 = 1(8): AUX2 enabled on F4 Bit4 = 0(0): AUX3 disabled on F4 = 1(16): AUX3 enabled on F4 Bit5 = 0(0): AUX4 disabled on F4 = 1(32): AUX4 enabled on F4 Bit6 = 0(0): AUX5 disabled on F4 = 1(64): AUX5 enabled on F4 Bit7 = 0(0): AUX6 disabled on F4 = 1(128): AUX6 enabled on F4

CV	Default Value	Value Range	Description	
39	8 8 for PLUX22	0-255	F4 mapping Bit0 = 0(0): FL disabled on F4 = 1(1): FL enabled on F4 Bit1 = 0(0): RL disabled on F4 = 1(2): RL enabled on F4 Bit2 = 0(0): AUX1 disabled on F4 = 1(4): AUX1 enabled on F4 Bit3 = 0(0): AUX2 disabled on F4 = 1(8): AUX2 enabled on F4 Bit4 = 0(0): AUX3 disabled on F4 = 1(16): AUX3 enabled on F4 Bit5 = 0(0): AUX4 disabled on F4 = 1(32): AUX4 enabled on F4 Bit6 = 0(0): AUX5 disabled on F4 = 1(64): AUX5 enabled on F4 Bit7 = 0(0): AUX6 disabled on F4 = 1(128): AUX6 enabled on F4	F5 mapping for PLUX22 Bit0 = 0(0): AUX2 disabled on F5 = 1(1): AUX2 enabled on F5 Bit1 = 0(0): AUX3 disabled on F5 = 1(2): AUX3 enabled on F5 Bit2 = 0(0): AUX4 disabled on F5 = 1(4): AUX4 enabled on F5 Bit3 = 0(0): AUX5 disabled on F5 = 1(8): AUX5 enabled on F5 Bit4 = 0(0): AUX6 disabled on F5 = 1(16): AUX6 enabled on F5 Bit5 = 0(0): AUX7 disabled on F5 = 1(32): AUX7 enabled on F5 Bit6 = 0(0): AUX 8 disabled on F5 = 1(64): AUX 8 enabled on F5 Bit7 = 0(0): AUX 9 disabled on F5 = 1(128): AUX 9 enabled on F5

CV	Default Value	Value Range	Description	
40	16 16 for PLUX22	0-255	F5 mapping Bit0 = 0(0): FL disabled on F5 = 1(1): FL enabled on F5 Bit1 = 0(0): RL disabled on F5 = 1(2): RL enabled on F5 Bit2 = 0(0): AUX1 disabled on F5 = 1(4): AUX1 enabled on F5 Bit3 = 0(0): AUX2 disabled on F5 = 1(8): AUX2 enabled on F5 Bit4 = 0(0): AUX3 disabled on F5 = 1(16): AUX3 enabled on F5 Bit5 = 0(0): AUX4 disabled on F5 = 1(32): AUX4 enabled on F5 Bit6 = 0(0): AUX5 disabled on F5 = 1(64): AUX5 enabled on F5 Bit7 = 0(0): AUX6 disabled on F5 = 1(128): AUX6 enabled on F5	F6 mapping for PLUX22 Bit0 = 0(0): AUX2 disabled on F6 = 1(1): AUX2 enabled on F6 Bit1 = 0(0): AUX3 disabled on F6 = 1(2): AUX3 enabled on F6 Bit2 = 0(0): AUX4 disabled on F6 = 1(4): AUX4 enabled on F6 Bit3 = 0(0): AUX5 disabled on F6 = 1(8): AUX5 enabled on F6 Bit4 = 0(0): AUX6 disabled on F6 = 1(16): AUX6 enabled on F6 Bit5 = 0(0): AUX7 disabled on F6 = 1(32): AUX7 enabled on F6 Bit6 = 0(0): AUX8 disabled on F6 = 1(64): AUX 8 enabled on F6 Bit7 = 0(0): AUX 9 disabled on F6 = 1(128): AUX 9 enabled on F6

CV	Default Value	Value Range	Description	
41	32 32 for PLUX22	0-255	F6 mapping Bit0 = 0(0): FL disabled on F6 = 1(1): FL enabled on F6 Bit1 = 0(0): RL disabled on F6 = 1(2): RL enabled on F6 Bit2 = 0(0): AUX1 disabled on F6 = 1(4): AUX1 enabled on F6 Bit3 = 0(0): AUX2 disabled on F6 = 1(8): AUX2 enabled on F6 Bit4 = 0(0): AUX3 disabled on F6 = 1(16): AUX3 enabled on F6 Bit5 = 0(0): AUX4 disabled on F6 = 1(32): AUX4 enabled on F6 Bit6 = 0(0): AUX5 disabled on F6 = 1(64): AUX5 enabled on F6 Bit7 = 0(0): AUX6 disabled on F6 = 1(128): AUX6 enabled on F6	F7 mapping for PLUX22 Bit0 = 0(0): AUX2 disabled on F7 = 1(1): AUX2 enabled on F7 Bit1 = 0(0): AUX3 disabled on F7 = 1(2): AUX3 enabled on F7 Bit2 = 0(0): AUX4 disabled on F7 = 1(4): AUX4 enabled on F7 Bit3 = 0(0): AUX5 disabled on F7 = 1(8): AUX5 enabled on F7 Bit4 = 0(0): AUX6 disabled on F7 = 1(16): AUX6 enabled on F7 Bit5 = 0(0): AUX7 disabled on F7 = 1(32): AUX7 enabled on F7 Bit6 = 0(0): AUX 8 disabled on F7 = 1(64): AUX 8 enabled on F7 Bit7 = 0(0): AUX 9 disabled on F7 = 1(128): AUX 9 enabled on F7

CV	Default Value	Value Range	Description	
42	64 64 for PLUX22	0-255	F7 mapping Bit0 = 0(0): FL disabled on F7 = 1(1): FL enabled on F7 Bit1 = 0(0): RL disabled on F7 = 1(2): RL enabled on F7 Bit2 = 0(0): AUX1 disabled on F7 = 1(4): AUX1 enabled on F7 Bit3 = 0(0): AUX2 disabled on F7 = 1(8): AUX2 enabled on F7 Bit4 = 0(0): AUX3 disabled on F7 = 1(16): AUX3 enabled on F7 Bit5 = 0(0): AUX4 disabled on F7 = 1(32): AUX4 enabled on F7 Bit6 = 0(0): AUX5 disabled on F7 = 1(64): AUX5 enabled on F7 Bit7 = 0(0): AUX6 disabled on F7 = 1(128): AUX6 enabled on F7	F8 mapping for PLUX22 Bit0 = 0(0): AUX2 disabled on F8 = 1(1): AUX2 enabled on F8 Bit1 = 0(0): AUX3 disabled on F8 = 1(2): AUX3 enabled on F8 Bit2 = 0(0): AUX4 disabled on F8 = 1(4): AUX4 enabled on F8 Bit3 = 0(0): AUX5 disabled on F8 = 1(8): AUX5 enabled on F8 Bit4 = 0(0): AUX6 disabled on F8 = 1(16): AUX6 enabled on F8 Bit5 = 0(0): AUX7 disabled on F8 = 1(32): AUX7 enabled on F8 Bit6 = 0(0): AUX 8 disabled on F8 = 1(64): AUX 8 enabled on F8 Bit7 = 0(0): AUX 9 disabled on F8 = 1(128): AUX 9 enabled on F8

CV	Default Value	Value Range	Description	
43	128 16 for PLUX22	0-255	F8 mapping Bit0 = 0(0): FL disabled on F8 = 1(1): FL enabled on F8 Bit1 = 0(0): RL disabled on F8 = 1(2): RL enabled on F8 Bit2 = 0(0): AUX1 disabled on F8 = 1(4): AUX1 enabled on F8 Bit3 = 0(0): AUX2 disabled on F8 = 1(8): AUX2 enabled on F8 Bit4 = 0(0): AUX3 disabled on F8 = 1(16): AUX3 enabled on F8 Bit5 = 0(0): AUX4 disabled on F8 = 1(32): AUX4 enabled on F8 Bit6 = 0(0): AUX5 disabled on F8 = 1(64): AUX5 enabled on F8 Bit7 = 0(0): AUX6 disabled on F8 = 1(128): AUX6 enabled on F8	F9 mapping for PLUX22 Bit0 = 0(0): AUX5 disabled on F9 = 1(1): AUX5 enabled on F9 Bit1 = 0(0): AUX6 disabled on F9 = 1(2): AUX6 enabled on F9 Bit2 = 0(0): AUX7 disabled on F9 = 1(4): AUX7 enabled on F9 Bit3 = 0(0): AUX 8 disabled on F9 = 1(8): AUX 8 enabled on F9 Bit4 = 0(0): AUX 9 disabled on F9 = 1(16): AUX 9 enabled on F9 Bit5 = 0(0): AUX 10 disabled on F9 = 1(32): AUX 10 enabled on F9

CV	Default Value	Value Range	Description	
44	0 32 for PLUX22	0-255	F9 mapping Bit0 = 0(0): FL disabled on F9 = 1(1): FL enabled on F9 Bit1 = 0(0): RL disabled on F9 = 1(2): RL enabled on F9 Bit2 = 0(0): AUX1 disabled on F9 = 1(4): AUX1 enabled on F9 Bit3 = 0(0): AUX2 disabled on F9 = 1(8): AUX2 enabled on F9 Bit4 = 0(0): AUX3 disabled on F9 = 1(16): AUX3 enabled on F9 Bit5 = 0(0): AUX4 disabled on F9 = 1(32): AUX4 enabled on F9 Bit6 = 0(0): AUX5 disabled on F9 = 1(64): AUX5 enabled on F9 Bit7 = 0(0): AUX6 disabled on F9 = 1(128): AUX6 enabled on F9	F10 mapping for PLUX22 Bit0 = 0(0): AUX5 disabled on F10 = 1(1): AUX5 enabled on F10 Bit1 = 0(0): AUX6 disabled on F10 = 1(2): AUX6 enabled on F10 Bit2 = 0(0): AUX7 disabled on F10 = 1(4): AUX7 enabled on F10 Bit3 = 0(0): AUX 8 disabled on F10 = 1(8): AUX 8 enabled on F10 Bit4 = 0(0): AUX 9 disabled on F10 = 1(16):AUX 9 enabled on F10 Bit5 = 0(0): AUX 10 disabled on F10 = 1(32):AUX 10 enabled on F10

CV	Default Value	Value Range	Description	
45	0 64 for PLUX22	0-255	F10 mapping Bit0 = 0(0): FL disabled on F10 = 1(1): FL enabled on F10 Bit1 = 0(0): RL disabled on F10 = 1(2): RL enabled on F10 Bit2 = 0(0): AUX1 disabled on F10 = 1(4): AUX1 enabled on F10 Bit3 = 0(0): AUX2 disabled on F10 = 1(8): AUX2 enabled on F10 Bit4 = 0(0): AUX3 disabled on F10 = 1(16): AUX3 enabled on F10 Bit5 = 0(0): AUX4 disabled on F10 = 1(32): AUX4 enabled on F10 Bit6 = 0(0): AUX5 disabled on F10 = 1(64): AUX5 enabled on F10 Bit7 = 0(0): AUX6 disabled on F10 = 1(128): AUX6 enabled on F10	F11 mapping for PLUX22 Bit0 = 0(0): AUX5 disabled on F11 = 1(1): AUX5 enabled on F11 Bit1 = 0(0): AUX6 disabled on F11 = 1(2): AUX6 enabled on F11 Bit2 = 0(0): AUX7 disabled on F11 = 1(4): AUX7 enabled on F11 Bit3 = 0(0): AUX8 disabled on F11 = 1(8): AUX8 enabled on F11 Bit4 = 0(0): AUX9 disabled on F11 = 1(16): AUX9 enabled on F11 Bit5 = 0(0): AUX10 disabled on F11 = 1(32): AUX10 enabled on F11

CV	Default Value	Value Range	Description	
46	0 128 for PLUX22	0-255	F11 mapping Bit0 = 0(0): FL disabled on F11 = 1(1): FL enabled on F11 Bit1 = 0(0): RL disabled on F11 = 1(2): RL enabled on F11 Bit2 = 0(0): AUX1 disabled on F11 = 1(4): AUX1 enabled on F11 Bit3 = 0(0): AUX2 disabled on F11 = 1(8): AUX2 enabled on F11 Bit4 = 0(0): AUX3 disabled on F11 = 1(16): AUX3 enabled on F11 Bit5 = 0(0): AUX4 disabled on F11 = 1(32): AUX4 enabled on F11 Bit6 = 0(0): AUX5 disabled on F11 = 1(64): AUX5 enabled on F11 Bit7 = 0(0): AUX6 disabled on F11 = 1(128): AUX6 enabled on F11	F12 mapping for PLUX22 Bit0 = 0(0): AUX5 disabled on F12 = 1(1): AUX5 enabled on F12 Bit1 = 0(0): AUX6 disabled on F12 = 1(2): AUX6 enabled on F12 Bit2 = 0(0): AUX7 disabled on F12 = 1(4): AUX7 enabled on F12 Bit3 = 0(0): AUX8 disabled on F12 = 1(8): AUX8 enabled on F12 Bit4 = 0(0): AUX9 disabled on F12 = 1(16): AUX9 enabled on F12 Bit5 = 0(0): AUX10 disabled on F12 = 1(32): AUX10 enabled on F12

CV	Default Value	Value Range	Description	
47	0 4 for PLUX22	0-255	F12 mapping Bit0 = 0(0): FL disabled on F12 = 1(1): FL enabled on F12 Bit1 = 0(0): RL disabled on F12 = 1(2): RL enabled on F12 Bit2 = 0(0): AUX1 disabled on F12 = 1(4): AUX1 enabled on F12 Bit3 = 0(0): AUX2 disabled on F12 = 1(8): AUX2 enabled on F12 Bit4 = 0(0): AUX3 disabled on F12 = 1(16): AUX3 enabled on F12 Bit5 = 0(0): AUX4 disabled on F12 = 1(32): AUX4 enabled on F12 Bit6 = 0(0): AUX5 disabled on F12 = 1(64): AUX5 enabled on F12 Bit7 = 0(0): AUX6 disabled on F12 = 1(128): AUX6 enabled on F12	F1, Reverse mapping for PLUX22 Bit0 = 0(0): FL disabled on F1rev = 1(1): FL enabled on F1rev Bit1 = 0(0): RL disabled on F1rev = 1(2): RL enabled on F1rev Bit2 = 0(0): AUX1 disabled on F1rev = 1(4): AUX1 enabled on F1rev Bit3 = 0(0): AUX2 disabled on F1rev = 1(8): AUX2 enabled on F1rev Bit4 = 0(0): AUX3 disabled on F1rev = 1(16): AUX3 enabled on F1rev Bit5 = 0(0): AUX4 disabled on F1rev = 1(32): AUX4 enabled on F1rev Bit6 = 0(0): AUX5 disabled on F1rev = 1(64): AUX5 enabled on F1rev Bit7 = 0(0): AUX6 disabled on F1rev = 1(128):AUX6 enabled on F1rev
48	255	0-255	FL Light intensity, [1-255]	
49	255	0-255	RL Light intensity, [1-255]	

CV	Default Value	Value Range	Description
50	255	0-255	AUX1 Light intensity, [1-255]
51	255	0-255	AUX2 Light intensity, [1-255]
52	255	0-255	AUX3 Light intensity, [1-255]
53	255	0-255	AUX4 Light intensity, [1-255]
54	255	0-255	AUX5 Light intensity, [1-255]
55	255	0-255	AUX6 Light intensity, [1-255]
56	255	0-255	AUX7 Light intensity, [1-255] (for Mini P22, W22 and W22M only)
57	255	0-255	AUX8 Light intensity, [1-255] (for Mini P22, W22 and W22M only)
58	255	0-255	AUX9 Light intensity, [1-255] (for Mini P22, W22 and W22M only)
59	255	0-255	AUX10 Light intensity, [1-255] (for Mini P22, W22 and W22M only)
60	1	0,1, 128, 129	Motor PID and PWM Control Bit0 = 0(0): PID Control Disabled = 1(1): PID Control Enabled Bit7 = 0(0): Motor PWM Frequency = 32kHz = 1(128): Motor PWM Frequency = 16kHz
61	80	0-255	PID P constant
62	120	0-255	PID I constant
63	40	0-255	PID D constant
64	1	1-10	PID error limit

CV	Default Value	Value Range	Description
66	0	0-255	Forward trim, multiply forward speed with (CV value/128), “0” - disabled
67	2	1-255	Speed Table, speed step 1
…..			…..
94	240	1-255	Speed Table, speed step 28
95	0	0-255	Revers trim, multiply revers speed with (CV value/128), “0” - disabled
105	0	0-255	USER data
106	0	0-255	USER data
112	50	1-127	Fade-in Light Effect (fade duration) for FR, FL, AUX Ex.:1 = 8ms, 15 = 120ms, 125 = 1000ms
113	25	1-127	Fade-out Light Effect (fade duration) for FR, FL, AUX ex.:1 = 8ms, 15 = 120ms, 125 = 1000ms
114	**3**	**0-255**	**Shunting speed, Function number F0÷F28 (0 – F0, 1 – F1, 2 – F2, 3 – F3, … 28 – F8). Default F3**
115	**4**	**0-255**	**Switch Off Acceleration Deceleration, Function number F0÷F28 (0 – F0, 1 – F1, 2 – F2, 3 – F3, … 28 – F8). Default F4**
116	**5**	**0-255**	**Disable Constant Braking, Function number F0÷F28 (0 – F0, 1 – F1, 2 – F2, 3 – F3, … 28 – F8). Default F5**

CV	Default Value	Value Range	Description
117	0	0-255	Continuous or PWM signal Output Mapping FL, RL, AUX1÷AUX6[1] Bit0 = 0(0): FL - PWM signal – Fade-in/out = 1(1): FL - Continuous signal, no diming Bit1 = 0(0): RL - PWM signal – Fade-in/out = 1(2): RL - Continuous signal, no diming Bit2 = 0(0): AUX1 - PWM signal – Fade-in/out = 1(4): AUX1 - Continuous signal, no diming Bit3 = 0(0): AUX2 - PWM signal – Fade-in/out = 1(8): AUX2 - Continuous signal, no diming Bit4 = 0(0): AUX3 - PWM signal – Fade-in/out = 1(16): AUX3 - Continuous signal, no diming Bit5 = 0(0): AUX4 - PWM signal – Fade-in/out = 1(32): AUX4 - Continuous signal, no diming Bit6 = 0(0): AUX5 - PWM signal – Fade-in/out = 1(64): AUX5 - Continuous signal, no diming Bit7 = 0(0): AUX6 - PWM signal – Fade-in/out = 1(128): AUX6 -Continuous signal, no diming

[1] Aux7÷Aux10 configured in CV185

CV	Default Value	Value Range	Description
118	0	0-12	Electrical Coupler Output mapping. Only one of the outputs can be configured for automatic decoupling CV118 = 0, No output configured for automatic decoupling CV118 = 1, FL configured for automatic decoupling CV118 = 2, RL configured for automatic decoupling CV118 = 3, AUX1 configured for automatic decoupling CV118 = 4, AUX2 configured for automatic decoupling ……. CV118 = 11, AUX9 configured for automatic decoupling CV118 = 12, AUX10 configured for automatic decoupling
119	50	0-255	Decoupling waiting time T1 (see Ch. 17) T1 = (CV119 value)*8ms Ex: 400ms = 50*8ms
120	50	0-255	Decoupling waiting time T2 (see Ch. 17) T1 = (CV120 value)*8ms Ex: 400ms = 50*8ms
121	50	0-255	Locomotive speed during automatic decoupling (see Ch. 17)

CV	Default Value	Value Range	Description
122	3 = 1 + 2 +	0-255	Second Decoder Configuration Data: Bit 0 = 0(0): SUSI pins used as auxiliary Outputs (see Table 3) = 1(1): SUSI pins used as SUSI CLK/DATA or Locowire Bit 1 = 0(0): Locowire interface active = 1(2): SUSI Interface enabled Bit 2 = 0(0): No motor loads are transmitted over SUSI = 1(4): Motor loads are transmitted over SUSI Bit 3 = 0(0): Motor PWM weighting OFF = 1(8): Motor PWM weighting with track voltage ON Bit 4,5= 00(0): No Shuttle train operation (see Ch. 12.3) = 10(16): Shuttle train operations without intermediate stop enabled = 01(32): Shuttle train operation with intermediate stop enabled = 11(48): Not permitted, must be avoided! Bit 6 = 0(0): FL/RL disabled during firmware update = 1(64): FL/RL outputs blinks during firmware update Bit 7 = 0(0): not used

CV	Default Value	Value Range	Description
123	16	0-255	SPP (Smart Power Pack) running time in 16ms steps (see Ch. 19) Duration = (value CV123) * 16ms; Ex: SPP run time = 16*16ms =256ms
124	1	0-255	SPP (Smart Power Pack) start delay in seconds, in 1s steps
125	1	0-255	SUSI wait time for the start of the sound decoder
126	102	0-255	SUSI CV
127	-	0-255	SUSI DATA read/write to address 800 + CV126
128	1	1-4	The number of PWM packets after which a BEMF window is entered (only if CV9 = 9) (see Ch. Motor Control)
129	6	1-10	BEMF measurement average (only if CV9 = 9)
130	6	1-12	BEMF measurement delay (only if CV9 = 9)
131	2 = 2	0-3	Configuring the output with automatic decoupling: Bit 0 = 0(0): The selected output receives an HF PWM signal 50% = 1(1): The selected output receives a LF PWM signal 0-100% Bit 1 = 0(0): Inactive PID controller during declutching = 1(2): Active PID controller during declutching
132	240	0-255	CV operations acknowledge mapping on outputs (see Ch. 8)
133	0	-	Decoder temperature read out (saved only if the special function is activated)

CV	Default Value	Value Range	Description
134	100	60-120	Maximum temperature protection triggering value
135	100	0-100%	QoS (Quality of Service) current value activated by CV122 bit 7 = 1 (The value is available only after special function is switched on/off)
136	100	0-100%	QoS (Quality of Service) minimum value activated by CV122 bit 7 = 1 (The value is available only after special function is switched on/off)
137	60	0-255	Medium speed for Load Compensation (different from CV6)
138	255	0-255	Load Compensation coefficient at the minimum speed (CV2)
139	100	0-255	Load Compensation at medium speed (CV137)
140	80	0-255	Load Compensation at maximum speed (CV5)
141	**10**	**0-50**	**ABC Sensitivity – Sets the ABC detection threshold in 0.1V increments** **Ex. value = 14 => ABC detection threshold = 1,4V**
142	10	0-255	Wait time when stationary in shuttle train mode, in seconds (see Ch. 12.3)
143	255	0-100	Maximum forward speed when ABC slow speed is triggered

CV	Default Value	Value Range	Description
144	255	0-100	Maximum reverse speed when ABC slow speed is triggered
146	0	0-15	Forward deceleration rate in fixed delay CBD mode (see Ch. 12.1) = 0: Constant braking distance disabled = 1-15: Braking distance = (value CV146) x (minimum braking distance)
147	0	0-15	Reverse deceleration rate in fixed delay CBD mode (see Ch. 12.1) = 0: the forward deceleration rate from CV146 is used = 1-15: Reverse Braking distance = (value CV147) x (minimum braking distance)
148	0	0-255	Braking distance Delay in CBD mode (with fixed deceleration) = 0: without delay To increase stopping distance in small amounts, increase the value of this CV. Braking is delayed by (CV148 value) * 8ms
149	0	0-255	Reverse braking distance Delay in CBD mode (with fixed deceleration) = 0: The braking distance deceleration from CV148 is used To increase stopping distance in small amounts, increase the value of this CV. Reverse braking is delayed by (CV149 value) * 8ms
150	0	0-255	CBD Brake distance forward in CBD mode with variable deceleration = 0: Fixed delay CBD mode is used

CV	Default Value	Value Range	Description
151	0	0-255	CBD Brake distance reverse in CBD mode with variable deceleration (complementary to CV150) = 0: The braking distance forward in CV150 is used
152	0	0-63	Acceleration Rate Reverse (complementary to CV3) = 0: the acceleration rate of CV3 is used in both directions
153	0	0-63	Deceleration rate backwards (complementary to CV4) = 0: the deceleration rate of CV4 is used in both directions

CV	Default Value	Value Range	Description
154	0	0-15	Maintenance Configuration (see Ch. 16) Bit0 = 0(0): Maintenance function disabled = 1(1): Maintenance function enabled Bit1 = 0(0): Maintenance Interval (MI) overrun is not signalled in CV30, bit3 = 1(2): MI overrun is signalled in CV30, bit3 Bit2 = 0(0): MI overrun is not signalled with FL/RL = 1(4): MI overrun is signalled by FL/RL Flashing with low frequency Bit3 = 0(0): MI exceeding by 50% is not signalled by FL/RL = 1(8): MI exceeding by 50% is signalled by FL/RL Flashing with high frequency
155	40	0-255	Maintenance Interval (MI) expressed in hours
156	-	0-59	Number of minutes since the last maintenance
157	-	0-255	Operating hours, lower byte (see Ch. 16)
158	-	0-255	Operating hours, upper byte (see Ch. 16)
159	-	0-255	The number of hours since the last maintenance hour, lower byte
160	-	0-255	The number of hours since the last maintenance hour, upper byte

CV	Default Value	Value Range	Description
161	85	0-255	The voltage threshold in the rails at which the motor starts in analogue mode 1, in 0,1V steps (see Ch. 14.1) Ex. (CV161 value)=85 => threshold=8.5V
162	65	0-255	The voltage threshold in the rails at which the motor stops in analogue mode 1, in 0.1 V steps (see Ch. 14.1) Ex. (CV161 value)=65 => threshold=6.5V
163	160	0-160	The voltage threshold in the rails at which the motor reaches maximum speed in analogue mode 1, in 0.1 V steps (see Ch. 14.1) Ex. (CV147 value=160) => threshold=16V
164	255	0-255	Motor PWM value for analogue mode 2 (see Ch. 14.2) = 0: analogue mode 1 selected = 1-255: analogue mode 2 selected
165	100	0-255	The threshold for CBD triggering in the DC sector when using SPP, in 0.1V steps (see Ch. 14.3) Voltage in the rails > threshold value => controlled DC braking Voltage in the rails < threshold value => SPP runtime

CV	Default Value	Value Range	Description
166	0	0-255	Inhibition of outputs with F0f (F0 forward) – see Ch. 13 Bit0 = 0(0): FL not inhibited by F0 (forward) = 1(1): FL inhibited by F0 (forward) Bit1 = 0(0): RL not inhibited by F0 (forward) = 1(2): RL inhibited by F0 (forward) Bit2 = 0(0): AUX1 not inhibited by F0 (forward) = 1(4): AUX1 inhibited by F0 (forward) Bit3 = 0(0): AUX2 not inhibited by F0 (forward) = 1(8): AUX2 inhibited by F0 (forward) Bit4 = 0(0): AUX3 not inhibited by F0 (forward) = 1(16): AUX3 inhibited by F0 (forward) Bit5 = 0(0): AUX4 not inhibited by F0 (forward) = 1(32): AUX4 inhibited by F0 (forward) Bit6 = 0(0): AUX5 not inhibited by F0 (forward) = 1(64): AUX5 inhibited by F0 (forward) Bit7 = 0(0): AUX6 not inhibited by F0 (forward) = 1(128): AUX6 inhibited by F0 (forward)

CV	Default Value	Value Range	Description
167	0	0-255	Inhibition of outputs with F0r (F0 reverse) Bit0 = 0(0): FL not inhibited by F0 (reverse) = 1(1): FL inhibited by F0 (reverse) Bit1 = 0(0): RL not inhibited by F0 (reverse) = 1(2): RL inhibited by F0 (reverse) Bit2 = 0(0): AUX1 not inhibited by F0 (reverse) = 1(4): AUX1 inhibited by F0 (reverse) Bit3 = 0(0): AUX2 not inhibited by F0 (reverse) = 1(8): AUX2 inhibited by F0 (reverse) Bit4 = 0(0): AUX3 not inhibited by F0 (reverse) = 1(16): AUX3 inhibited by F0 (reverse) Bit5 = 0(0): AUX4 not inhibited by F0 (reverse) = 1(32): AUX4 inhibited by F0 (reverse) Bit6 = 0(0): AUX5 not inhibited by F0 (reverse) = 1(64): AUX5 inhibited by F0 (reverse) Bit7 = 0(0): AUX6 not inhibited by F0 (reverse) = 1(128): AUX6 inhibited by F0 (reverse)

CV	Default Value	Value Range	Description
168	0	0-255	Inhibition of outputs with F1f (F1 forward movement) Bit0 = 0(0): FL not inhibited by F1 (forward) = 1(1): FL inhibited by F1 (forward) Bit1 = 0(0): RL not inhibited by F1 (forward) = 1(2): RL inhibited by F1 (forward) Bit2 = 0(0): AUX1 not inhibited by F1 (forward) = 1(4): AUX1 inhibited by F1 (forward) Bit3 = 0(0): AUX2 not inhibited by F1 (forward) = 1(8): AUX2 inhibited by F1 (forward) Bit4 = 0(0): AUX3 not inhibited by F1 (forward) = 1(16): AUX3 inhibited by F1 (forward) Bit5 = 0(0): AUX4 not inhibited by F1 (forward) = 1(32): AUX4 inhibited by F1 (forward) Bit6 = 0(0): AUX5 not inhibited by F1 (forward) = 1(64): AUX5 inhibited by F1 (forward) Bit7 = 0(0): AUX6 not inhibited by F1 (forward) = 1(128): AUX6 inhibited by F1 (forward)

CV	Default Value	Value Range	Description
169	0	0-255	Inhibition of outputs with F2 Bit0 = 0(0): FL not inhibited by F2 = 1(1): FL inhibited by F2 Bit1 = 0(0): RL not inhibited by F2 = 1(2): RL inhibited by F2 Bit2 = 0(0): AUX1 not inhibited by F2 = 1(4): AUX1 inhibited by F2 Bit3 = 0(0): AUX2 not inhibited by F2 = 1(8): AUX2 inhibited by F2 Bit4 = 0(0): AUX3 not inhibited by F2 = 1(16): AUX3 inhibited by F2 Bit5 = 0(0): AUX4 not inhibited by F2 = 1(32): AUX4 inhibited by F2 Bit6 = 0(0): AUX5 not inhibited by F2 = 1(64): AUX5 inhibited by F2 Bit7 = 0(0): AUX6 not inhibited by F2 = 1(128): AUX6 inhibited by F2

CV	Default Value	Value Range	Description
170	0	0-255	Inhibition of outputs with F3 Bit0 = 0(0): FL not inhibited by F3 = 1(1): FL inhibited by F3 Bit1 = 0(0): RL not inhibited by F3 = 1(2): RL inhibited by F3 Bit2 = 0(0): AUX1 not inhibited by F3 = 1(4): AUX1 inhibited by F3 Bit3 = 0(0): AUX2 not inhibited by F3 = 1(8): AUX2 inhibited by F3 Bit4 = 0(0): AUX3 not inhibited by F3 = 1(16): AUX3 inhibited by F3 Bit5 = 0(0): AUX4 not inhibited by F3 = 1(32): AUX4 inhibited by F3 Bit6 = 0(0): AUX5 not inhibited by F3 = 1(64): AUX5 inhibited by F3 Bit7 = 0(0): AUX6 not inhibited by F3 = 1(128): AUX6 inhibited by F3

CV	Default Value	Value Range	Description
171	0	0-255	Inhibition of outputs with F4 Bit0 = 0(0): FL not inhibited by F4 = 1(1): FL inhibited by F4 Bit1 = 0(0): RL not inhibited by F4 = 1(2): RL inhibited by F4 Bit2 = 0(0): AUX1 not inhibited by F4 = 1(4): AUX1 inhibited by F4 Bit3 = 0(0): AUX2 not inhibited by F4 = 1(8): AUX2 inhibited by F4 Bit4 = 0(0): AUX3 not inhibited by F4 = 1(16): AUX3 inhibited by F4 Bit5 = 0(0): AUX4 not inhibited by F4 = 1(32): AUX4 inhibited by F4 Bit6 = 0(0): AUX5 not inhibited by F4 = 1(64): AUX5 inhibited by F4 Bit7 = 0(0): AUX6 not inhibited by F4 = 1(128): AUX6 inhibited by F4

CV	Default Value	Value Range	Description
172	0	0-255	Inhibition of outputs with F5 Bit0 = 0(0): FL not inhibited by F5 = 1(1): FL inhibited by F5 Bit1 = 0(0): RL not inhibited by F5 = 1(2): RL inhibited by F5 Bit2 = 0(0): AUX1 not inhibited by F5 = 1(4): AUX1 inhibited by F5 Bit3 = 0(0): AUX2 not inhibited by F5 = 1(8): AUX2 inhibited by F5 Bit4 = 0(0): AUX3 not inhibited by F5 = 1(16): AUX3 inhibited by F5 Bit5 = 0(0): AUX4 not inhibited by F5 = 1(32): AUX4 inhibited by F5 Bit6 = 0(0): AUX5 not inhibited by F5 = 1(64): AUX5 inhibited by F5 Bit7 = 0(0): AUX6 not inhibited by F5 = 1(128): AUX6 inhibited by F5

CV	Default Value	Value Range	Description
173	0	0-255	Inhibition of outputs with F6 Bit0 = 0(0): FL not inhibited by F6 = 1(1): FL inhibited by F6 Bit1 = 0(0): RL not inhibited by F6 = 1(2): RL inhibited by F6 Bit2 = 0(0): AUX1 not inhibited by F6 = 1(4): AUX1 inhibited by F6 Bit3 = 0(0): AUX2 not inhibited by F6 = 1(8): AUX2 inhibited by F6 Bit4 = 0(0): AUX3 not inhibited by F6 = 1(16): AUX3 inhibited by F6 Bit5 = 0(0): AUX4 not inhibited by F6 = 1(32): AUX4 inhibited by F6 Bit6 = 0(0): AUX5 not inhibited by F6 = 1(64): AUX5 inhibited by F6 Bit7 = 0(0): AUX6 not inhibited by F6 = 1(128): AUX6 inhibited by F6

CV	Default Value	Value Range	Description
174	0	0-255	Inhibition of outputs with F7 Bit0 = 0(0): FL not inhibited by F7 = 1(1): FL inhibited by F7 Bit1 = 0(0): RL not inhibited by F7 = 1(2): RL inhibited by F7 Bit2 = 0(0): AUX1 not inhibited by F7 = 1(4): AUX1 inhibited by F7 Bit3 = 0(0): AUX2 not inhibited by F7 = 1(8): AUX2 inhibited by F7 Bit4 = 0(0): AUX3 not inhibited by F7 = 1(16): AUX3 inhibited by F7 Bit5 = 0(0): AUX4 not inhibited by F7 = 1(32): AUX4 inhibited by F7 Bit6 = 0(0): AUX5 not inhibited by F7 = 1(64): AUX5 inhibited by F7 Bit7 = 0(0): AUX6 not inhibited by F7 = 1(128): AUX6 inhibited by F7

CV	Default Value	Value Range	Description
175	0	0-255	Inhibition of outputs with F8 Bit0 = 0(0): FL not inhibited by F8 = 1(1): FL inhibited by F8 Bit1 = 0(0): RL not inhibited by F8 = 1(2): RL inhibited by F8 Bit2 = 0(0): AUX1 not inhibited by F8 = 1(4): AUX1 inhibited by F8 Bit3 = 0(0): AUX2 not inhibited by F8 = 1(8): AUX2 inhibited by F8 Bit4 = 0(0): AUX3 not inhibited by F8 = 1(16): AUX3 inhibited by F8 Bit5 = 0(0): AUX4 not inhibited by F8 = 1(32): AUX4 inhibited by F8 Bit6 = 0(0): AUX5 not inhibited by F8 = 1(64): AUX5 inhibited by F8 Bit7 = 0(0): AUX6 not inhibited by F8 = 1(128): AUX6 inhibited by F8

CV	Default Value	Value Range	Description
176	0	0-255	Inhibition of outputs with F9 Bit0 = 0(0): FL not inhibited by F9 = 1(1): FL inhibited by F9 Bit1 = 0(0): RL not inhibited by F9 = 1(2): RL inhibited by F9 Bit2 = 0(0): AUX1 not inhibited by F9 = 1(4): AUX1 inhibited by F9 Bit3 = 0(0): AUX2 not inhibited by F9 = 1(8): AUX2 inhibited by F9 Bit4 = 0(0): AUX3 not inhibited by F9 = 1(16): AUX3 inhibited by F9 Bit5 = 0(0): AUX4 not inhibited by F9 = 1(32): AUX4 inhibited by F9 Bit6 = 0(0): AUX5 not inhibited by F9 = 1(64): AUX5 inhibited by F9 Bit7 = 0(0): AUX6 not inhibited by F9 = 1(128): AUX6 inhibited by F9

CV	Default Value	Value Range	Description
177	0	0-255	Inhibition of outputs with F10 Bit0 = 0(0): FL not inhibited by F10 = 1(1): FL inhibited by F10 Bit1 = 0(0): RL not inhibited by F10 = 1(2): RL inhibited by F10 Bit2 = 0(0): AUX1 not inhibited by F10 = 1(4): AUX1 inhibited by F10 Bit3 = 0(0): AUX2 not inhibited by F10 = 1(8): AUX2 inhibited by F10 Bit4 = 0(0): AUX3 not inhibited by F10 = 1(16): AUX3 inhibited by F10 Bit5 = 0(0): AUX4 not inhibited by F10 = 1(32): AUX4 inhibited by F10 Bit6 = 0(0): AUX5 not inhibited by F10 = 1(64): AUX5 inhibited by F10 Bit7 = 0(0): AUX6 not inhibited by F10 = 1(128): AUX6 inhibited by F10

CV	Default Value	Value Range	Description
178	0	0-255	Inhibition of outputs with F11 Bit0 = 0(0): FL not inhibited by F11 = 1(1): FL inhibited by F11 Bit1 = 0(0): RL not inhibited by F11 = 1(2): RL inhibited by F11 Bit2 = 0(0): AUX1 not inhibited by F11 = 1(4): AUX1 inhibited by F11 Bit3 = 0(0): AUX2 not inhibited by F11 = 1(8): AUX2 inhibited by F11 Bit4 = 0(0): AUX3 not inhibited by F11 = 1(16): AUX3 inhibited by F11 Bit5 = 0(0): AUX4 not inhibited by F11 = 1(32): AUX4 inhibited by F11 Bit6 = 0(0): AUX5 not inhibited by F11 = 1(64): AUX5 inhibited by F11 Bit7 = 0(0): AUX6 not inhibited by F11 = 1(128): AUX6 inhibited by F11

CV	Default Value	Value Range	Description
179	0	0-255	Inhibition of outputs with F12 Bit0 = 0(0): FL not inhibited by F12 = 1(1): FL inhibited by F12 Bit1 = 0(0): RL not inhibited by F12 = 1(2): RL inhibited by F12 Bit2 = 0(0): AUX1 not inhibited by F12 = 1(4): AUX1 inhibited by F12 Bit3 = 0(0): AUX2 not inhibited by F12 = 1(8): AUX2 inhibited by F12 Bit4 = 0(0): AUX3 not inhibited by F12 = 1(16): AUX3 inhibited by F12 Bit5 = 0(0): AUX4 not inhibited by F12 = 1(32): AUX4 inhibited by F12 Bit6 = 0(0): AUX5 not inhibited by F12 = 1(64): AUX5 inhibited by F12 Bit7 = 0(0): AUX6 not inhibited by F12 = 1(128): AUX6 inhibited by F12

CV	Default Value	Value Range	Description
180	0	0-255	Inhibition of outputs with F1r (F1 reverse) Bit0 = 0(0): FL not inhibited by F1 (reverse) = 1(1): FL inhibited by F1 (reverse) Bit1 = 0(0): RL not inhibited by F1 (reverse) = 1(2): RL inhibited by F1 (reverse) Bit2 = 0(0): AUX1 not inhibited by F1 (reverse) = 1(4): AUX1 inhibited by F1 (reverse) Bit3 = 0(0): AUX2 not inhibited by F1 (reverse) = 1(8): AUX2 inhibited by F1 (reverse) Bit4 = 0(0): AUX3 not inhibited by F1 (reverse) = 1(16): AUX3 inhibited by F1 (reverse) Bit5 = 0(0): AUX4 not inhibited by F1 (reverse) = 1(32): AUX4 inhibited by F1 (reverse) Bit6 = 0(0): AUX5 not inhibited by F1 (reverse) = 1(64): AUX5 inhibited by F1 (reverse) Bit7 = 0(0): AUX6 not inhibited by F1 (reverse) = 1(128): AUX6 inhibited by F1 (reverse)
181	0	0, 1	Saving the last state of functions and outputs: 0 - disabled, 1 - enabled
182	-		Status saved FL, RL, F1..F4
183	-		Status saved F5÷F12

CV	Default Value	Value Range	Description
184	-		Status saved F13÷F20
185	0	0-15	Continuous or PWM (dimming) signal Output Mapping AUX7÷AUX10 Bit0 = 0(0): AUX7 - PWM signal with dimming = 1(1): AUX7 - Continuous signal with no dimming Bit1 = 0(0): AUX8 - PWM signal with dimming = 1(2): AUX8 - Continuous signal with no dimming Bit2 = 0(0): AUX9 - PWM signal with dimming = 1(4): AUX9 - Continuous signal with no dimming Bit3 = 0(0): AUX10 - PWM signal with dimming = 1(8): AUX10 - Continuous signal with no dimming
186	**20**	**0-255**	**Outputs turn on delay, (CV186 value)*8ms, default 160ms**
187	**20**	**0-255**	**Outputs turn off delay, (CV187 value)*8ms, default 160ms**
188	**0**	**0-255**	**Switch on delay mask for: AUX6, AUX5, AUX4, AUX3, AUX2, AUX1, RL, FL**
189	**0**	**0-255**	**Switch off delay mask for: AUX6, AUX5, AUX4, AUX3, AUX2, AUX1, RL, FL**
190	**0**	**0-255**	**Switch on delay mask for: -, -, -, -, AUX10, AUX9, AUX8, AUX7**

CV	Default Value	Value Range	Description
191	**0**	**0-255**	**Switch off delay mask for:** **-, -, -, -, AUX10, AUX9, AUX8, AUX7**
192	**8**	**0-40**	**ABC Sensitivity OFF** **sets ABC detection threshold in steps of 0.1V** **Ex.: value = 6 => ABC detection threshold = 0.6V**
193	**25**	**0-255**	**ABC state turn-on delay, 8ms steps, default 200ms**
194	**25**	**0-255**	**ABC state turn-off delay, 8ms steps, default 200ms**
195	**0**	**0-255**	**Bit mask of conditioned outputs with function key F0 activated:** **AUX6, AUX5, AUX4, AUX3, AUX2, AUX1, RL, FL**
196	**0**	**0-255**	**Bit mask of conditioned outputs with function key F0 activated:** **-, -, -, -, AUX10, AUX9, AUX8, AUX7**
197	**255**	**0-255**	**The number of the function key to which the break function 1 is assigned**
198	**255**	**0-255**	**The number of the function key to which the break function 2 is assigned**
199	**255**	**0-255**	**The number of the function key to which the break function 3 is assigned**
200	**0**	**0-255**	**Brake Function 1 deceleration rate reduction percentage.** **255 corresponds to 100%**

CV	Default Value	Value Range	Description
201	**0**	**0-255**	**Brake Function 2 deceleration rate reduction percentage.** **255 corresponds to 100%**
202	**0**	**0-255**	**Brake Function 3 deceleration rate reduction percentage.** **255 corresponds to 100%**
203	**0**	**0-255**	**Maximum speed with brake function 1 activated**
204	**0**	**0-255**	**Maximum speed with brake function 2 activated**
205	**0**	**0-255**	**Maximum speed with brake function 3 activated**
206	**0**	**0-255**	**Inhibit mask for outputs, in forward direction:** **AUX6, AUX5, AUX4, AUX3, AUX2, AUX1, RL, FL**
207	**0**	**0-255**	**Inhibit mask for outputs, in forward direction:** **-, -, -, -, AUX10, AUX9, AUX8, AUX7**
208	**0**	**0-255**	**Inhibit mask for outputs, in reverse direction:** **AUX6, AUX5, AUX4, AUX3, AUX2, AUX1, RL, FL**
209	**0**	**0-255**	**Inhibit mask for outputs, in reverse direction:** **-, -, -, -, AUX10, AUX9, AUX8, AUX7**
211	**0**	**0-40**	**Motor short circuit protection delay**
212	**0**	**-**	**NOT USED**
213	**28**	**0-255**	**Special saving function number**

26. Annex Bits and Bytes

If we want to modify the values of the configuration variables (CV), it is good to keep a few notions regarding the representation of numbers in binary format. In binary format we have only two digits “0” and “1”. A binary number is called a bit. An 8-bit group will call a byte, representing a binary number of 8 binary digits. Configuration variables, CV, are bytes stored in non-volatile memory of decoders. The bits of a byte are numbered from 0 to 7. Bit 0, it's the least significant (LSB), has the decimal value of 1 and bit (7) it's the most significant (MSB), has the decimal value of 128.

Table 10: Bits

	MSB							LSB
Bit Position	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Bit decimal value	128	64	32	16	8	4	2	1

If we know the bit configuration and we want to find out the decimal value, we use the following calculation formula:

Dec = B7*128 + B6*64 + B5*32 + B4*16 + B3*8 + B2*4 + B1*2 + B0

where B0 ... B7 represents the value of the respective bit (“0” or “1”).

Example: if B7 = “1”, B5 = “1”, B2 = “1”, and the rest is “0”, we will have:

Dec = 1*128 + 0*64 + 1*32 + 0*16 + 0*8 + 1*4 + 0*2 + 0 =

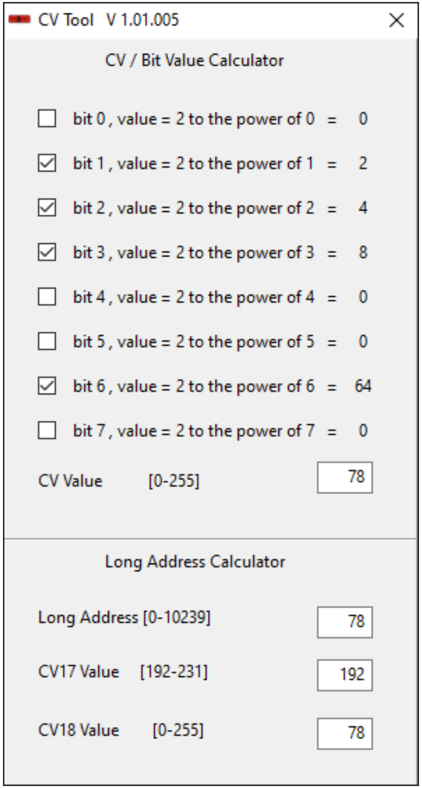
= 128 + 0 + 32 + 0 + 0 + 4 + 0 + 0 = 164

If we want to find the bits configuration from the decimal value, we do the opposite. We try to subtract from the decimal value the bits value begins with MSB and we keep the difference for the next subtractions until we get zero. For possible subtractions, with a positive result, the bit will have a value of 1. For the impossible subtractions, when the difference is negative, we abandon the operation (the value of the bit will be zero) and continue with the next decrease.

Example: we want to find the bits configuration for decimal value 73:

73 - 128 = -55	=> Bit7 = 0
73 - 64 = 9	=> Bit6 = 1
9 - 32 = -23	=> Bit5 = 0
9 - 16 = - 7	=> Bit4 = 0
9 - 8 = 1	=> Bit3 = 1
1 - 4 = - 3	=> Bit2 = 0
1 - 2 = - 1	=> Bit1 = 0
1 - 1 = 0	=> Bit0 = 1

Figure 14: CV tool main window



CV tool is a small utility program to convert the decimal value to binary and vice versa or to calculate the value of extended addresses.

It can be downloaded from the address below:

https://train-o-matic.com/downloads/software/cvTool.zip

Tehnologistic SRL
Str. Libertatii 35A
407035 Apahida
Romania