

Anleitung LUX-Saug- und Messwagen by Matthias und Hardi

Funktionen und DCC-Adresse Messwagen

DCC-Adresse Standard = 601

(diese Adresse kann nur über den Sketch im Arduino geändert werden, ein direktes einspielen wie bei „normalen“ Decodern ist nicht möglich)

DCC Funktionen und deren Tasteneinrichtung

| | |
|------------|---|
| F0: On/Off | Licht ein aus |
| F1: On/Off | Pause Distanz Messung |
| F2: Impuls | Sauger Motor an/ aus |
| F3: Impuls | Löschen von Sauger Zählern |
| F4: Impuls | vorherige OLED Displayanzeige |
| F5: Impuls | nächste OLED Displayanzeige |
| F6: Impuls | Markierung setzen, um besondere Ereignisse zu kennzeichnen (Position auf der Strecke, ...) |
| F7: Impuls | Sauger-/ Motorgeschwindigkeit |
| F8: Impuls | Steigung und Neigungswinkel auf 0 kalibrieren |
| F9: Impuls | Rotation des Displays um es richtigerum lesen zu können |

Alle weiteren Funktionstasten sind derzeit unbelegt

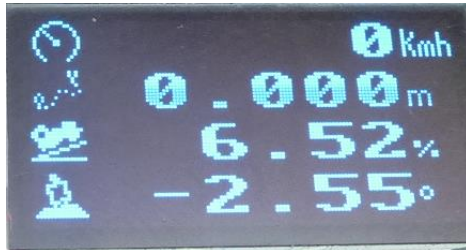
Tipp für Märklinfreunde:

Am besten einmal eine Einrichtung auf einer Zentrale vornehmen und dann eine Lok-Karte anlegen.
Damit wird die Einrichtung auf weiteren Zentralen schnell und einfach.

Anleitung LUX-Saug- und Messwagen by Matthias und Hardi

Verwendete Symbole auf dem OLED-Display

Anzeige des jeweiligen Status



Geschwindigkeit

Strecke

Steigung

Neigung



Strecke Sauger

Zeit Sauger

Gesamt Strecke

Gesamt Fahrzeit

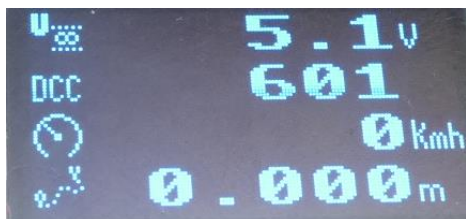


Gesamt Sauger

Gesamt Zeit Sauger

Spannung Schiene

Spannung externer Messpin



Spannung Arduino

DCC Adresse

Geschwindigkeit (Wiederholung)

Strecke (Wiederholung)

Die einzelnen Bildschirmanzeigen können entweder durch Tastendruck F4 und F5 gewechselt werden, bzw. einfacher durch einen „leichten Klopfer“ auf den Messwagen

Mit einem „leichten Doppelklopfer“ kann der Saugmodus ein oder ausgestellt werden oder natürlich auch durch die Taste F2

Anleitung LUX-Saug- und Messwagen by Matthias und Hardi

Bluetooth

Der LUX-Saug- und Messwagen kann auch Daten per Bluetooth übertragen.
Hierzu muss die Messwagen-App auf ein Android-Handy installiert werden.

Diese findet man hier: [Messwagen/messwagen.apk](https://www.messwagen.de/messwagen.apk)

Hier ist wichtig die Installation von „Unbekannten Quellen“ muss zugelassen werden.
Sonst blockt Android die Installation.

Verbindung mit Bluetooth

- Bluetooth auf dem Handy aktivieren
- „Messwagen1“ im Bluetooth-Menü suchen und verbinden
- Messwagen-App öffnen
- auf „PAIRED DEVICES“ tippen
- Messwagen1 aussuchen
- Verbindung wird gesucht/ aufgebaut
- wenn erfolgreich verbunden können über die App die Geschwindigkeit und die zurückgelegte Strecke angezeigt werden sobald sich der Wagen bewegt



Bluetooth Steuerung mit der Arduino IDE auf dem Rechner

Wenn der Laptop/ Rechner mit Bluetooth ausgestattet ist, kann man auch per Arduino IDE die Funktionen steuern.

Dazu öffnet man die Arduino IDE und richtet unter dem Punkt Werkzeuge den Arduino ein:

Board: Arduino AVR Boards – Arduino Pro oder Mini Pro
Prozessor: ATmega 328P (5V, 16MHz)
Com-Port: Port an dem der Arduino am Rechner angeschlossen ist
Programmer: Arduino ISP

Wie der Rechner mit dem Messwagen gekoppelt wird steht im Dokument
Bluetooth Messwagen Aufbau und Inbetriebnahme sehr ausführlich beschrieben.

Wenn das alles gemacht ist öffnet man unter Werkzeuge den „Seriellen Monitor“ oder den „Seriellen Plotter“. Die Baudrate muss auf 57600 eingestellt werden.

Anleitung LUX-Saug- und Messwagen

by Matthias und Hardi

Nun kann ganz unten in der Leiste das entsprechende Zeichen gefolgt von Enter als Kurzbefehl eingegeben werden. Somit lässt sich der Messwagen über die Kurzbefehle auch mit dem Rechner gesteuert werden.

| | |
|------|------------------------------------|
| '0': | Set_Enable_Distance_Measurement(0) |
| '1': | Set_Enable_Distance_Measurement(1) |
| '2': | Set_Enable_Distance_Measurement(2) |
| 'v': | Change_Vacuum(0) |
| 'V': | Change_Vacuum(1) |
| 'z': | MeasureFilteredAngleOffset |
| '+': | NextDisplay(1) |
| '-': | NextDisplay(-1) |
| 'c': | Clear_Vacuum_Counters() |
| 'U': | Read_Float(ee.Perimeter) |
| 'S': | Read_Float(ee.ScaleFact) |
| 'I': | Read_uint16(ee.Show_Interval) |

Wenn der Licht-Pin (Pin6) angeschlossen ist:

| | |
|------|--------------|
| 'I': | Set_Light(0) |
| 'L': | Set_Light(1) |

Will man am Sketch was ändern, wie z.B. die Adresse, so könnte man einfach im Sketch die Adresse (Standard DCC 601) auf den gewünschten Wert ändern und dann auf den „Hochladen“-Pfeil klicken, dann wird das Programm auf den Arduino geladen.

Achtung: nicht ungeduldig werden, das kann schon mal 2-3 Minuten dauern