

42.1 LUA Skript - Von der Problemstellung zur Realisierung Teil 1

wie ich in 42. Grundlagen der Programmierung - ohne LUA angekündigt hatte, werde ich nun Schritt für Schritt zeigen wie man eine Problemstellung in einer Anlage mit Einsatz von LUA löst.

Dieser Thread ist für absolute Anfänger gedacht. Es wird also etwas einfaches sein, was auch überschaubar ist.

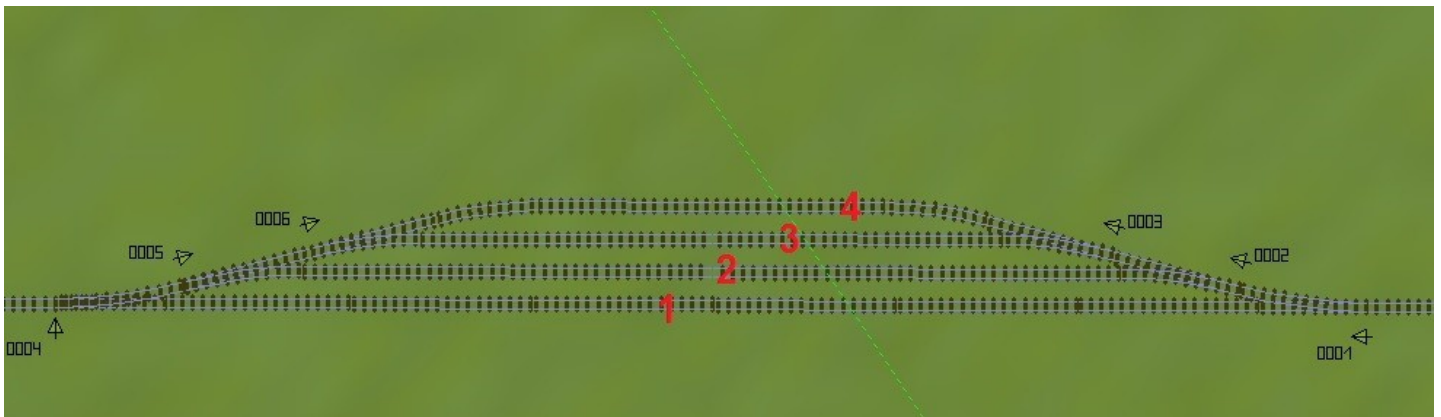
Dazu habe ich eine kleine Ausgangsanlage gebaut, welche ich euch hier zur Verfügung stelle, damit auch für alle die gleichen Ausgangsbedingungen vorhanden sind.

Zum Download:

[Ausgangsanlage](#)

Kommen wir nun zu einer einfachen Aufgabe. Wie ihr nach dem Laden der Anlage seht, steht eine Lok an einem Signal. Daraus werden dann am Ende 4. Sinnigerweise hat die Lok den Namen #Lok1. Diese soll immer auf das Gleis 1 fahren. Lok2 dann auf Gleis 2 usw.

Die Gleise habe ich wie folgt festgelegt.



Was muss also passieren, damit die Lok1 auf Gleis 1 fährt. Die Weiche mit der ID 1 muss auf Fahrt gestellt werden. Da nichts weiter auf dem Weg dahin gemacht werden muss, war es dann auch schon. Auch wenn es jetzt lapidar ist, habe ich trotzdem dazu ein Struktogramm erstellt. Es ist nur eine Anweisung, die aber so formuliert ist, dass sie wohl jeder versteht, der noch nichts mit programmieren zu tun hatte.

Struktogramm Funktion Gleis1()

Setze Weiche 1 auf Fahrt

Wie im Struktogramm zu sehen ist, soll die Funktion "Gleis1" heißen.

Wenn wir jetzt den LUA-Skript-Editor öffnen steht dort bereits folgendes:

```

I=0
clearlog()

print("Hey let's start, EEP Version is: ", EEPVer)

function EEPMain()
  print("Counter: ",I)
  I=I+1
  return 1
end

```

Wir kümmern uns hier nicht weiter über diesen schon vorhandenen Teil und lassen ihn einfach mal so.

Setzen wir nun unsere Funktion zum Schalten von Gleis 1 in dieses Skript. Beginnen wir mit dem Funktionsrumpf (achtet auf den roten Text, so wird ein Kommentar im Skript geschrieben):

```

function Gleis1()      -- Beginn der Funktion mit dem Namen Gleis1
end                    -- Hier endet die Funktion Gleis1

```

Nach der Eingabe sieht das Skript dann so aus:

```

I=0
clearlog()

print("Hey let's start, EEP Version is: ", EEPVer)

function EEPMain()
  print("Counter: ",I)
  I=I+1
  return 1
end

function Gleis1()
end

```

Jetzt müssen wir unsere Anweisung aus dem Struktogramm in unsere Funktion einfügen und zwar zwischen diesen bisher vorhandenen Zeilen. Jetzt geht es also darum den richtigen Befehl zu finden.

Wer mal im Internet nach LUA sucht, wird keinen LUA-Befehl finden, der eine Weiche oder ein Signal auf einen bestimmten Zustand setzt. Dies gibt es in LUA nicht. Das was uns in EEPX zur Verfügung steht, ist ein "EEPX-LUA". Es gibt also für EEPX bereits fertige Funktionen, die z.B Weichen stellen und Signale auf Fahrt oder Halt setzen. Des Rätsels Lösung befindet sich in EEPX im Menü Hilfe, LUA Handbuch. Öffnen wir es mal.

Dort finden wir auf S. 22 eine Übersicht über EEPX-spezifische Befehle. Diese Übersicht hat 3 Spalten. In der ersten steht der Befehl, in der zweiten eine Beschreibung und in der dritten ein Beispiel. Interessant für uns ist erst mal die Beschreibung.

Als ersten Eintrag lesen wir dort: "Der Inhalt des EEP-Ereignis-Fensters wird gelöscht". Ist irgendwie nicht das was wir brauchen. Also weiter nachgeschaut. Irgendwann lesen wir: "Setzt die Weiche x auf die Stellung y."

Wie war noch mal die Anweisung im Struktogramm? Setze Weiche 1 auf Fahrt. Genau das brauchen wir. In der Spalte 1 können wir nun den Befehl lesen: EEPSetSwitch(x,y,z) Der Parameter z muss nicht sein (deswegen der Hinweis optional)

Jetzt können wir unsere Funktion fertig machen. Der Befehl lautet: EEPSetSwitch(1,1)

Schauen wir uns nun die Parameter an, die wir dem Befehl (eigentlich ist es eine Funktion) in den Klammern übergeben. Die erste 1 ist die ID der Weiche. Also Weiche 1. Der zweite Parameter ist auch eine 1 und gibt an wie die Weiche zu stellen ist. 1 = Fahrt und 2 = Abzweig. Beachtet bitte, dass es hier eine 2-Wege Weiche ist. Bei einer 3-Wege-Weiche wäre auch noch der Wert 3 möglich. Ich überlasse es nun euch zu wissen was die 3 dann bedeutet.

Wenn wir mit Kenntnis der Hintergründe nun unseren Befehl in die Funktion schreiben, sieht es dann so aus:

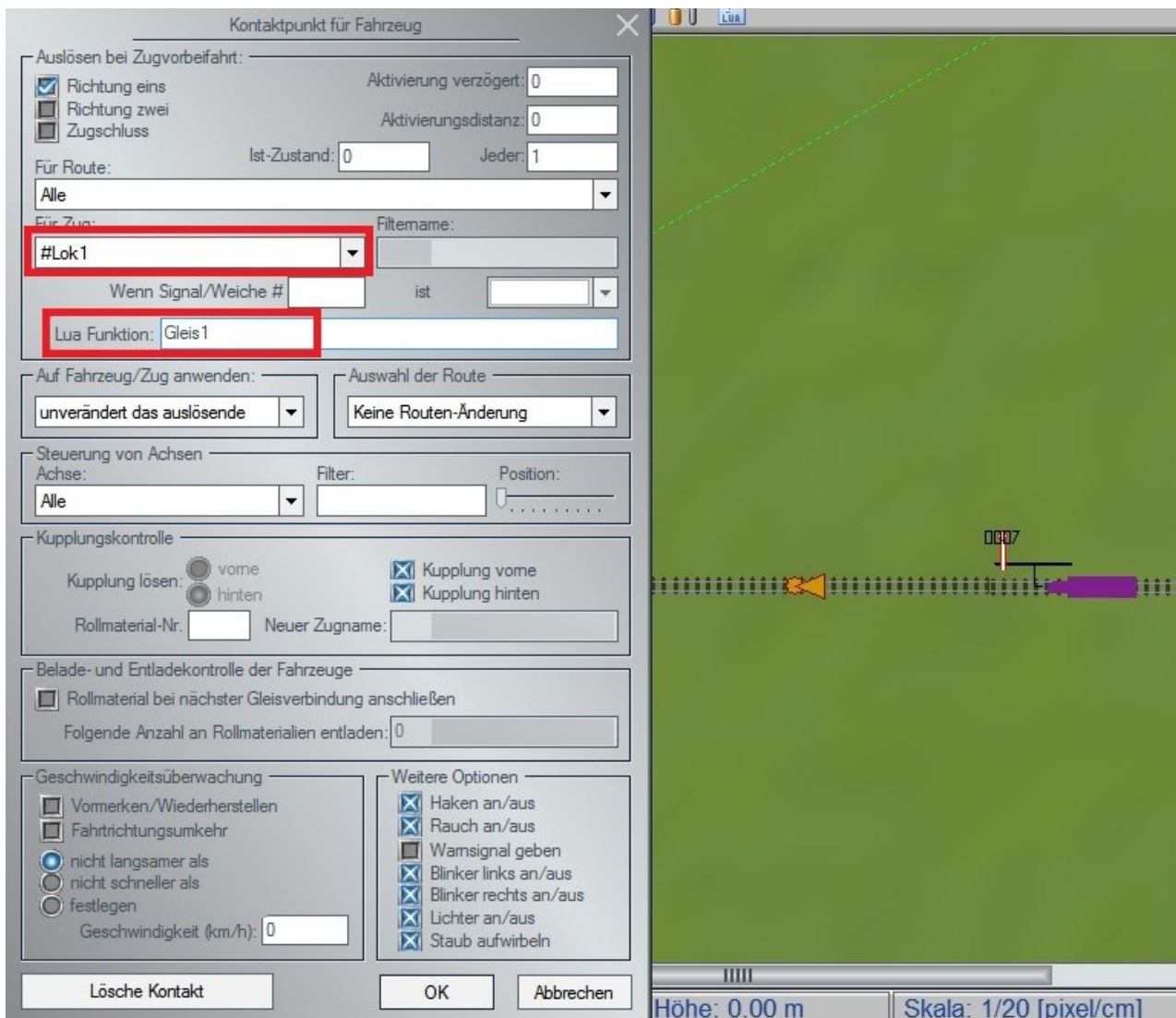
```
I=0
clearlog()

print("Hey let's start, EEP Version is: ", EEPVer)

function EEPMain()
  print("Counter: ",I)
  I=I+1
  return 1
end

function Gleis1()
  EEPSetSwitch(1,1) -- Setzt die Weiche 1 auf Fahrt
end
```

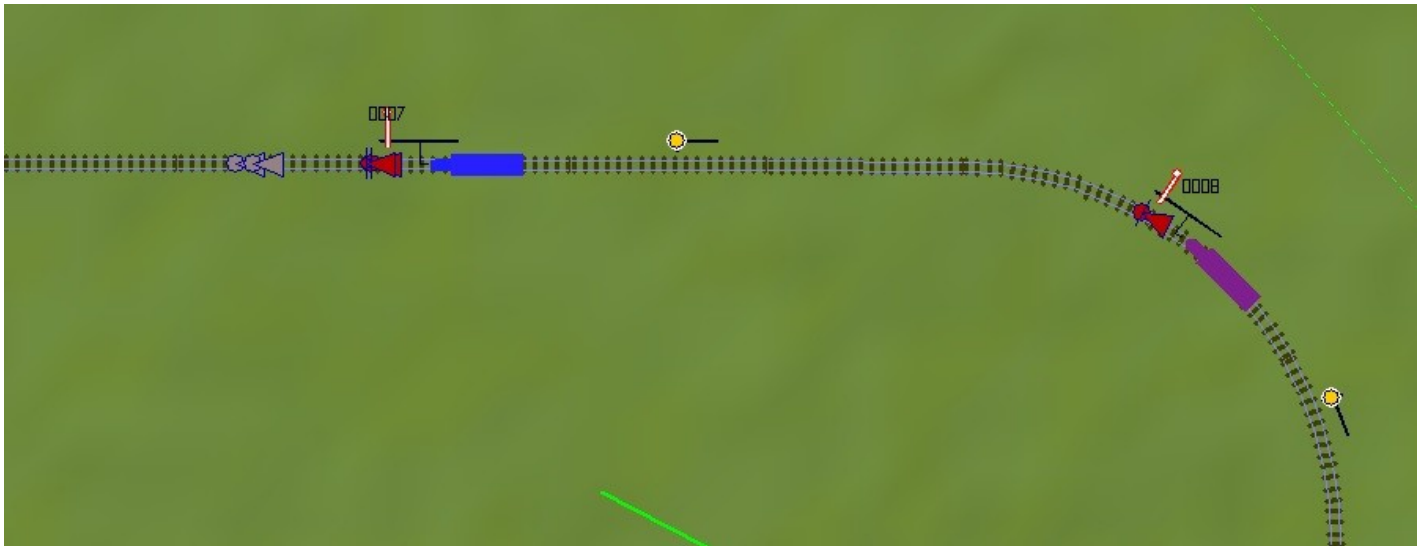
Damit ist unsere Funktion fertig. Mit einem Klick auf die Schaltfläche "Skript neu laden" schließen wir den LUA-Editor. Damit wir unser Skript testen können brauchen wir noch einen Fahrzeugkontaktpunkt den wir wie folgt setzen:



Achtet bitte darauf, dass vor und hinter dem Funktionsnamen im Kontaktpunkt kein Leerzeichen steht. Dies darf auch nicht im Funktionsnamen sein (z.B. zwischen Gleis und 1). Die Klammern wie im Skript dürfen ebenfalls nicht sein.

Setzen wir nun noch einen Signalkontaktpunkt links vom Signal, der das Signal auf Halt stellt und stellen die Weiche 1 per Hand auf Abzweig, können wir nun testen ob alles funktioniert.

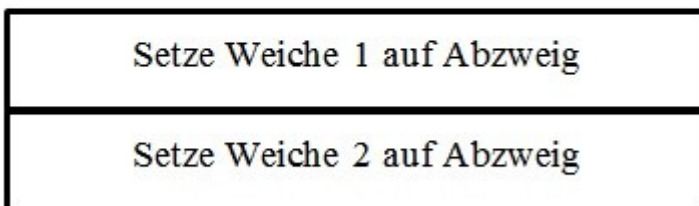
Wenn alles so funktioniert hat wie gedacht (zumindest bei mir war es so), können wir weiter machen. Hierfür stelle ich ein weiteres Signal auf die Anlage (siehe Bild) und eine weitere Lok (#Lok2).



Um vom Signal 7 zum Gleis 2 zu kommen, müssen wir die Weichen 1 und 2 entsprechend stellen. Nachfolgend das Struktogramm, in dem ich bereits die nötigen Befehle eingetragen habe. Dies soll zeigen wie ich solch ein Struktogramm dann in den Quelltext übertrage. Natürlich wird sich für solch Kleinigkeit später keiner ein Struktogramm oder einen anderen Plan skizzieren.

Struktogramm Funktion Gleis2()

function Gleis1()



EEPSetSwitch(1,2)

EEPSetSwitch(2,2)

end|

Das LUA-Skript sollte dann so aussehen.

```

I=0
clearlog()

print("Hey let's start, EEP Version is: ", EEPVer)

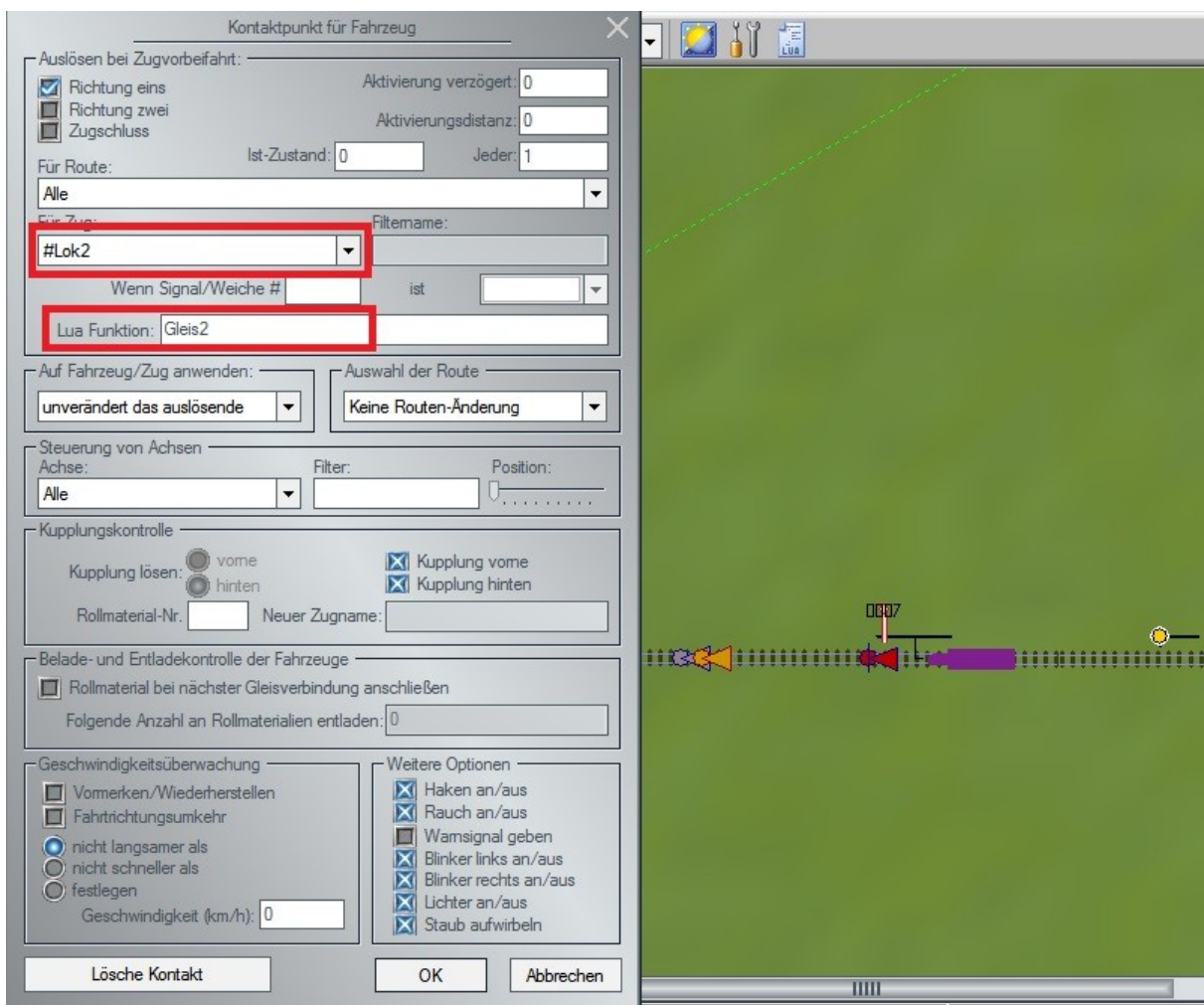
function EEPMain()
  print("Counter: ",I)
  I=I+1
  return 1
end

function Gleis1()
  EEPSetSwitch(1,1) -- Setzt die Weiche 1 auf Fahrt
end

function Gleis2()
  EEPSetSwitch(1,2)
  EEPSetSwitch(2,2)
end

```

Jetzt noch ein weiterer Fahrzeugkontaktpunkt wie im nachfolgenden Bild und wir können testen.



Die Programmierung für die Gleise 3 und 4 überlasse ich euch jetzt selber. Es ist nichts neues an Informationen nötig, um dies hinzubekommen. Lediglich das Schalten einer weiteren Weiche muss mit eingeplant werden und wie das geht wisst ihr ja nun.

Zum Download:

[Anlage](#)