

PKW-Auto Turm

Einspursystem

Funktionsmodell

V16NSO10013

Bitte lesen Sie sich diese Anleitung erst einmal komplett durch, bevor sie mit dem Zusammenbau beginnen.

- **Einsetzen des Turm's :**

Turm einsetzen und auf die gewünschte Position bringen. Die Pos. X, Pos. Y und Pos. Z sowie die Immobilien ID können im Objekteigenschaften Fenster abgelesen werden und in der Excel-Tabelle eingetragen werden. **(Abb. 1)**

Der Wendepplatz ist am Pkw Turm anzudocken. Die Weichen ID's aufschreiben bzw. in der Tabelle eintragen. Auch diese müssen später in dem Lua Skript eingetragen werden.

Der Pkw Turm kann auf Anlagenhöhe abgesenkt eingebaut werden. Hierfür muss ein Ausschnitt gemacht werden. Es ist besser erst den Ausschnitt herzustellen und dann den Turm drauf zustellen.

Positionsberechnung für PKW-Turm										
In den gelben Feldern die Werte eintragen										
1	Positionen PKW-Turm		Zur Übernahme in den LUA-Script:							
2	Position X	19,49								
3	Position Y	-66,46								
4	Position Z	-0,74								
5	AT_BuהלID = ("R1" * "R5" * "R6" * "R7" * "R10" * "R9" * "R11" * "R12" * "R13" * "R14" * "R15")									
6	Positionen für Bühnen									
7			Turm 90 Grad							
8	Bühne 1L	Position X	19,49	16,82	Bühne 1R	Position X	19,49	PKW-Turm-Immobilien-ID	2	
9		Position Y	-66,46	-68,09		Position Y	-61,50	Abstieg ID direkt vor dem Turm	1	
10		Position Z	-0,39	-0,39		Position Z	-0,39			
11		Immobilien-ID	1			Immobilien-ID	4	Tableau links, Immobilien-ID	16	
12	Bühne 2L	Position X	19,49		Bühne 2R	Position X	19,49	Tableau mitte, Immobilien-ID	17	
13		Position Y	-66,46			Position Y	-61,50	Tableau rechts, Immobilien-ID	18	
14		Position Z	-2,13			Position Z	-2,13	Parkplatz/Schild	19	
15		Immobilien-ID	5			Immobilien-ID	6	Signal ID	3	
16	Bühne 3L	Position X	19,49		Bühne 3R	Position X	19,49			
17		Position Y	-66,46			Position Y	-61,50			
18		Position Z	4,03			Position Z	4,03			
19		Immobilien-ID	7			Immobilien-ID	8		2,87	
20									2,43	
21	Bühne 4L	Position X	19,49		Bühne 4R	Position X	19,49			
22		Position Y	-66,46			Position Y	-61,50			
23		Position Z	5,93			Position Z	5,93			
24		Immobilien-ID	19			Immobilien-ID	9			
25	Bühne 5L	Position X	19,49		Bühne 5R	Position X	19,49			
26		Position Y	-66,46			Position Y	-61,50			
27		Position Z	7,83			Position Z	7,83			
28		Immobilien-ID	11			Immobilien-ID	12			
29	Bühne 6L	Position X	19,49		Bühne 6R	Position X	19,49			
30		Position Y	-66,46			Position Y	-61,50			
31		Position Z	9,73			Position Z	9,73			
32		Immobilien-ID	13			Immobilien-ID	14			
33	Bühne 7L	Position X	19,49		Bühne 7R	Position X	19,49			
34		Position Y	-66,46			Position Y	-61,50			
35		Position Z	11,63			Position Z	11,63			
36		Immobilien-ID	0			Immobilien-ID	0			

Abb. 2

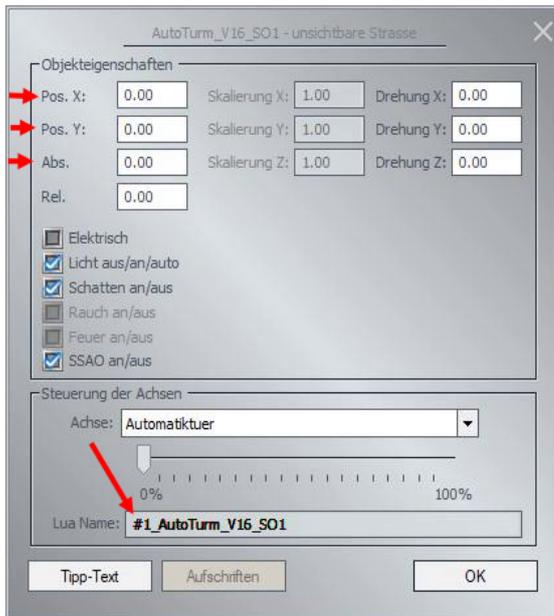


Abb. 1

- **Einsetzen der Bühnen in den Turm:**

Die Bühnen können im 3D oder 2D Modus neben dem Turm eingesetzt werden. Im 2D/3D Fenster die Immobilien ID ablesen und in die beigefügten Excel Tabelle eintragen. (Abb. 1, Abb. 2). Diese brauchen Sie später, um sie Im LUA Skript einzutragen.

Beginnend mit der ersten Bühne werden nun alle Bühnen der Reihe nach in den Turm eingesetzt. Die dementsprechenden Werte entnehmen Sie bitte aus der Excel Tabelle und tragen diese dann im Objekteigenschaftenfenster ein. (Abb. 1 und Abb. 2)

Ein Sperren des Objektes ist hilfreich um es gegen versehentliches anklicken zu sichern und so ein ungewolltes Springen zu vermeiden.

Achtung:

Wird der Turm in einer anderen Rotations- Position gebracht, müssen die Bühnen von Hand andockt werden.

Dazu die Bühne an der Mitte der hinteren Seite schieben, nun die Höhe aus der Excel Tabelle nehmen und in das Eigenschaftfenster eintragen, den Regler der Achse-Bühne auf 100%. Mit Ok bestätigen. Die Bühne in die Mitte schieben bis die Kette sichtbar wird, dann loslassen. Im 3D Spielemodus die Bühne mehrfach anklicken, sodass sie in ihre Position fährt (In Normaler Geschwindigkeit dauert es etwas). Ist die Bühne an ihrer Parkposition einen rechtsklick auf die Bühne und Sperren. Das solange wiederholen bis alle Bühnen eingebaut sind.

- **Tableau einbauen:**

Das Tableau besteht aus drei Teilen. Im Objekteigenschaftenfenster werden hier immer dieselben Werte eingetragen bei Pos. X, Pos. Y und Pos. Z.

So stehen alle Modellteile nebeneinander. Bitte die Immobilien ID notieren wie oben beschrieben. Wird auch im Lua Skript benötigt.

- **LUA:**

Um die Funktionen des Modells anwenden zu können benötigen Sie den Verweis auf das LUA-Skript (KFZ_Turm_Einspur_SO1.lua), das mit dem Set mit installiert wurde. Diese Datei befindet sich im EEP-Ordner „LUA“.

Damit diese Funktionen für Ihre selbst erstellte Anlage aktiv sind, müssen Sie diese Datei im LUA-Skript der Anlage mit einbinden. Außerdem müssen verschiedene Parameter definiert werden. Wenn Sie den mitgelieferten LUA-Skript im EEP-LUA-Ordner mit einem Texteditor öffnen finden Sie den Teil, den Sie übertragen müssen (Zeile 29 bis zur Zeile 56).

```
=====
-- Definitionen für KFZ-Turm - Entprechende Werte aus der Anlage hier eintragen
-- Definitions for car tower - enter the corresponding values from the EEP-layout here
-- Définitions pour tour de voiture - entrez ici les valeurs correspondantes du page EEP

KFZ_TurmD =
{
  -- Reihenfolge/sequence/séquence: 1R,1L,2R,2L,3R,3L,4R,4L,5R,5L,6R,6L,7R,7L
  BuehnelD = {"#2", "#3", "#4", "#5", "#6", "#7", "#8", "#9", "#10", "#11", "#12", "#13", "#14", "#15"}, -- ID's der Bühnen

  TurmID = "#1", -- ID vom Turm
  ParkplatzschildID = "#16", -- ID vom Parkplatzschild
  PanellID_Links = "#17", -- ID vom Panel links
  PanellID_Mitte = "#18", -- ID vom Panel mitte
  PanellID_Rechts = "#19", -- ID vom Panel rechts
  SignalID = 5, -- ID der Ampel vor dem Turm
  WeicheAusfahrt = -- Weiche direkt vor dem Turm
  {
    1, -- ID
    1, -- fahrt
    2, -- abzweig
  },
}

=====
-- Einbinden des Steuerungs-Moduls / Integration of the control module / Intégration du module de contrôle
dofile( ".\\LUA\\KFZ_Turm_SO1.lua")
=====
```

Die **rot** markierten Werte müssen dann entsprechend mit den in Ihrer Anlage vorhandenen Werten angepasst werden.

Innerhalb der Funktion „EEPMain()“ müssen dann noch die Funktionen KTE_Check() und KTE_CheckPanel() aufgerufen werden. Siehe auch in der beiliegenden Demo-Anlage.

Am Ende dieser Anleitung finden Sie noch die Erklärungen/Dokumentation zu den einzelnen LUA-Funktionen.

- **Funktionsweise:**

Ist der Turm richtig zusammengebaut und das Lua Skript eingetragen steht einem Funktionstest nichts mehr im Wege. Über das Tableau lassen sich gezielt Autos aus

dem Turm holen. Mit dem Schalter „ Zufällige PKW ausfahrt „ wird ein beliebiges Fahrzeug aus dem Turm gefahren. Über die Beleuchtung lässt erkennen welche Plätze belegt sind. Das Pkw Schild zeigt zudem auch noch an, wie viele Plätze frei sind. Fährt ein Fahrzeug vor das Signal beginnt der Turm Automatisch mit dem Einparken.

- **Besonderheiten:**

1. Wird die Tageszeit von Hand verstellt gehen bei allen Autos die Lichter an, dies ist kein Fehler. Ist ein Parkvorgang abgeschlossen gehen an allen Autos die Lichter wieder aus, aber auch nicht mehr automatisch an.
2. Bei manchen Autos kommt es vor das sie bei bestimmten Betrachtungswinkeln hinter den Scheiben nicht mehr sichtbar sind bzw. plötzlich sichtbar werden.
3. Bei verschiedenen Autos ist es so, dass sie nicht mittig auf der Bühne stehen, das ist aber nur Optisch und hat keinen Einfluss auf die Funktion der Pkw Turm's.

- **Achtung:**

Die Bühnen dürfen im Betrieb nicht mehr mit der linken Maustaste angewählt werden da sie dann aus dem Turm herausspringen. Hier hilft nur noch ein neues Einbauen in den Turm. Danach muss aber auch die Immobilien ID geprüft werden.

Bei Abweichungen der eingetragenen Werte wird der Turm nicht mehr ordnungsgemäß funktionieren.

In der Demo Anlage sind alle im Set befindlichen Modelle verbaut.

Viel Spaß mit diesem Modellset

Weitere Modelle sind finden Sie unter

<https://www.skyline-eschwege.de>

<https://eepshopping.de/index.php>

Bei Fragen und Anregungen bitte eine kurze Email an:

Info@skyline-eschwege.de

• Dokumentation zur Datei „Demo_KFZ_Turm_Einspur_SO1.lua“

Diese Dokumentation soll dazu dienen, die einzelnen Funktionen im Script zu dokumentieren bzw. zu erklären, um dem User die Möglichkeit zu geben die Abläufe nach zu vollziehen.

Generell:

Es wurde darauf geachtet, dass Funktionen und Variablen eindeutige Namen erhalten, damit keine Konflikte mit bestehenden Lua-Scripten auftreten.

Es gibt 3 Variablen (jeweils als Tabellen definiert):

- KFZ_TurmEsD** Hier werden die Werte für Immobilien-ID's, Weichen-ID, Signal-ID und der Speicherslot in EEP für die jeweilige Anlage eingetragen.
Die Definition dieser Variablen muss im Anlagen-Lua-Script erfolgen, da diese Werte ja bei jeder Anlage variieren.
- KFZ_TurmEsK** Hier sind alle konstanten Werte definiert, die nicht verändert werden dürfen, da Sie den gesamten Ablauf und die einzelnen Funktionen steuern.
Eine Ausnahme: Der Wert der Variable `_debug` kann entsprechend den Wünschen, je nachdem was im EEP-Ereignisfenster angezeigt werden soll, verändert werden.
- KFZ_TurmEs** Hier speichert Lua Werte, die den Ablaufstatus definieren. Alle Werte in dieser Tabelle (ausser „Counter“, „Sprache“ und `PanellInstalled`) werden im angegebenen EEP-Speicherslot mit der Anlage gespeichert, damit bei Unterbrechungen, der Ablauf an der Stelle fortgesetzt wird, an der unterbrochen/gespeichert wurde.

Funktionen:

KTE_Check()

Diese Funktion muss bei jedem Durchlauf der `Main()` Funktion im Anlagenscript aufgerufen werden. Sie steuert die Abläufe für Einparken und Ausparken von Fahrzeugen.

KTE_CheckPanel()

Diese Funktion muss bei jedem Durchlauf der `Main()` Funktion im Anlagenscript aufgerufen werden, wenn das Bedienpanel installiert wird. Damit werden dann die „Funktionsknöpfe“ des Panels geprüft und entsprechende Aktionen gestartet.

KTE_AusparkenZufallPos()

Diese Funktion kann durch den Benutzer aufgerufen werden (durch Kontaktpunkt oder aus dem eigenen Lua-Script), um ein Fahrzeug aus dem Turm auszuparken. Es wird dabei eine zufällige Bühne, auf der ein PKW steht, ausgewählt.

KTE_IstParkhausFrei()

Diese Funktion kann der User benutzen, um zu erfahren, ob das Parkhaus noch freie Plätze hat. Der Rückgabewert ist:

- `true`, wenn mindestens eine Parkposition frei ist
- `false`, wenn das Parkhaus besetzt ist

GetEEPLanguage()

Diese Funktion ermittelt die EEP-Sprache. Dies ist notwendig, um die Achsensteuerung in allen drei Sprachversionen durchführen zu können. Da die Achsenbezeichnungen länderspezifisch sind muss beim Aufruf einer Achsensteuerung der entsprechende Achsenname benutzt werden.
z.B bei der Bühne:

"Buehne",	deutsche Bezeichnung
"Stage",	englische Bezeichnung
"Étape"	französische Bezeichnung

Interne Funktionen:

Diese sind für den Internen Ablauf bestimmt und sollten nicht „von aussen“ aufgerufen werden.

KTE_PrintStatus()

Diese Funktion wird nur beim Laden der Werte aus dem EEP-Speicherslot aufgerufen, wenn `_debug` den Wert 2 hat. Dabei werden dann alle relevanten Werte der Variablen im EEP-Ereignisfenster ausgegeben.

KTE_Init()

Diese Funktion wird beim Start (beim Laden des Scriptes) ausgeführt und stellt sicher, dass alle Variablen entsprechend gesetzt werden.

KTE_Reset()

Hier werden (fast) alle Variablen auf ihre Grundwerte zurück gesetzt, die Lifte und Bühnen in eine Grundposition gebracht und die Namen der Fahrzeuge im Turm aktualisiert.

KTE_SetAktion(wert)

Mit dieser Funktion wird der Wert für die Variable „KFZ_Turm.EsAktion“ gesetzt und die Werte an EEP übergeben (das geschieht in der Funktion „KTE_SaveData()“), um sicher zu stellen, dass der Status der Ablaufsteuerung immer aktuell mit der Anlage übereinstimmt.

KTE_SetBetriebsArt (wert)

Mit dieser Funktion wird der Wert für die Variable „KFZ_TurmEs.BetriebsArt“ gesetzt und die Werte an EEP übergeben (das geschieht in der Funktion „KTE_SaveData()“), um sicher zu stellen, dass der Status der Ablaufsteuerung immer aktuell mit der Anlage übereinstimmt.

KTE_SetBuehneAktiv (wert)

Mit dieser Funktion wird der Wert für die Variable „KFZ_TurmEs.BuehneAktiv“ gesetzt und die Werte an EEP übergeben (das geschieht in der Funktion „KTE_SaveData()“), um sicher zu stellen, dass der Status der Ablaufsteuerung immer aktuell mit der Anlage übereinstimmt.

KTE_SetZeitschleifeEinf (wert)

Mit dieser Funktion wird der Wert für die Variable „KFZ_TurmEs.ZeitschleifeEinf“ gesetzt und die Werte an EEP übergeben (das geschieht in der Funktion „KTE_SaveData()“), um sicher zu stellen, dass der Status der Ablaufsteuerung immer aktuell mit der Anlage übereinstimmt.

KTE_SetZeitschleifeAusf (wert)

Mit dieser Funktion wird der Wert für die Variable „KFZEs_Turm.ZeitschleifeAusf“ gesetzt und die Werte an EEP übergeben (das geschieht in der Funktion „KTE_SaveData()“), um sicher zu stellen, dass der Status der Ablaufsteuerung immer aktuell mit der Anlage übereinstimmt.

KTE_SetBuehnePKW (wert)

Mit dieser Funktion wird der Wert für die Variable „KFZ_TurmEs.BuehnePKW“ gesetzt und die Werte an EEP übergeben (das geschieht in der Funktion „KTE_SaveData()“), um sicher zu stellen, dass der Status der Ablaufsteuerung immer aktuell mit der Anlage übereinstimmt.

KTE_SetEtage (wert)

Mit dieser Funktion wird der Wert für die Variable „KFZ_TurmEs.Etage“ gesetzt und

die Werte an EEP übergeben (das geschieht in der Funktion „KTE_SaveData()“), um sicher zu stellen, dass der Status der Ablaufsteuerung immer aktuell mit der Anlage übereinstimmt.

KTE_LoadData()

Mit dieser Funktion werden die Relevanten Variablen aus der EEP-Anlage ausgelesen.

KTE_SaveData ()

Mit dieser Funktion werden die Relevanten Variablen in die EEP-Anlage geschrieben.

KTE_SteuerungParkhausSchild(korrektur)

Mit dieser Funktion werden die Anzeigen für das Parkhausschild gesteuert. Der Parameter „korrektur“ dient dazu bei einer begonnenen Einparkprozedur bereits einen freien Platz weniger auf dem Schild anzuzeigen.

KTE_SteuerungPanel()

Diese Funktion steuert die Anzeigen (Achsen) des Panelmodells (Lichter grün/rot, die Anzeige oben und die Schalter)

KTE_IstBuehneBesetzt(buehne)

Diese Funktion liefert die Information über die benannte Bühne (Parameter „buehne“).

true = Bühne ist belegt

false = Bühne ist frei

KTE_GetFreieBuehne()

Die Rückgabe dieser Funktion ist entweder die erste frei Bühne oder, wenn alle Bühnen belegt sind, den Wert 0

KTE_MoveLift(position)

Mit dieser Funktion wird die Bewegung des Lifts (welches konstruktionsbedingt in der Realität mehrere Lifte sind). Der Lift wird von der momentanen Position in die angegebene Position „gefahren“.

KTE_BuehneAufParkposition(buehne)

Diese Funktion „fährt“ die Bühne mit Lift von der Position am Eingang zur jeweiligen Parkposition.

KTE_BuehneAufAusfahrt(buehne)

Diese Funktion „fährt“ die Bühne mit Lift von der Parkposition zum Eingang.

KTE_AusparkenAusPos(buehne)

Diese Funktion startet das Ausparken von einer angegebenen Bühne.