



# SOFTWARE RC CAR MKR1010

OKTOBER 2020

VORTRAGENDE: ARMIN FISCHER  
MARTIN SCHUBERT

E-MAIL: [FIA@HTLWRN.AC.AT](mailto:FIA@HTLWRN.AC.AT)  
[SUM@HTLWRN.AC.AT](mailto:SUM@HTLWRN.AC.AT)

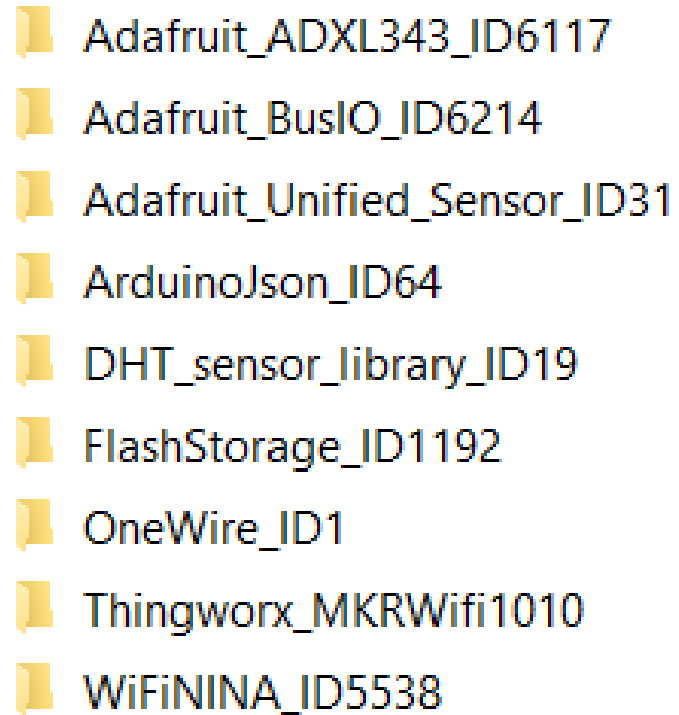
# INHALTSVERZEICHNIS

- Libraries einbinden
- Arduino verbinden / RC Car Example Code aufrufen
- Aufbau RC Car Arduino Code
- Thingworx Wiederholung
  - Erstellen eines Things
  - Erstellen von Properties
  - AppKey erstellen
- Wichtige Einstellungen in der Library
- Arduino Programm hochladen
  - Neue Funktion: Eingabe per serieller Schnittstelle (Software Putty)
- Statusausgabe der Kommunikation




# LIBRARIES EINBINDEN

- Zur Verfügung gestellte Libraries müssen in den Pfad \Documents\Arduino\libraries kopiert werden.
- GitHub Link: [https://github.com/Snorp84/DigiPro\\_IOT\\_Arduino\\_Schulung](https://github.com/Snorp84/DigiPro_IOT_Arduino_Schulung)

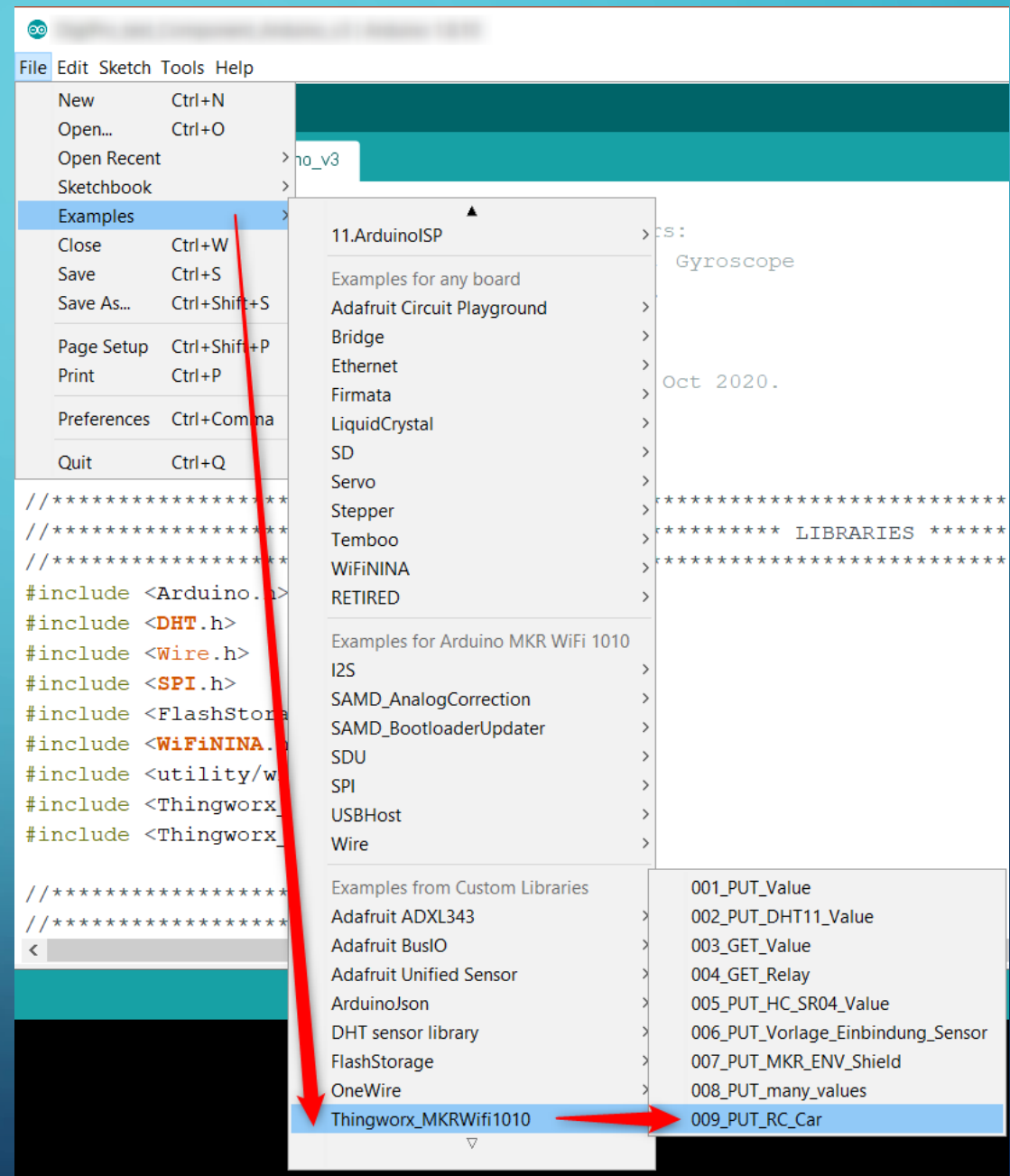


- Adafruit\_ADXL343\_ID6117
- Adafruit\_BusIO\_ID6214
- Adafruit\_Unified\_Sensor\_ID31
- ArduinoJson\_ID64
- DHT\_sensor\_library\_ID19
- FlashStorage\_ID1192
- OneWire\_ID1
- Thingworx\_MKRWifi1010
- WiFiNINA\_ID5538

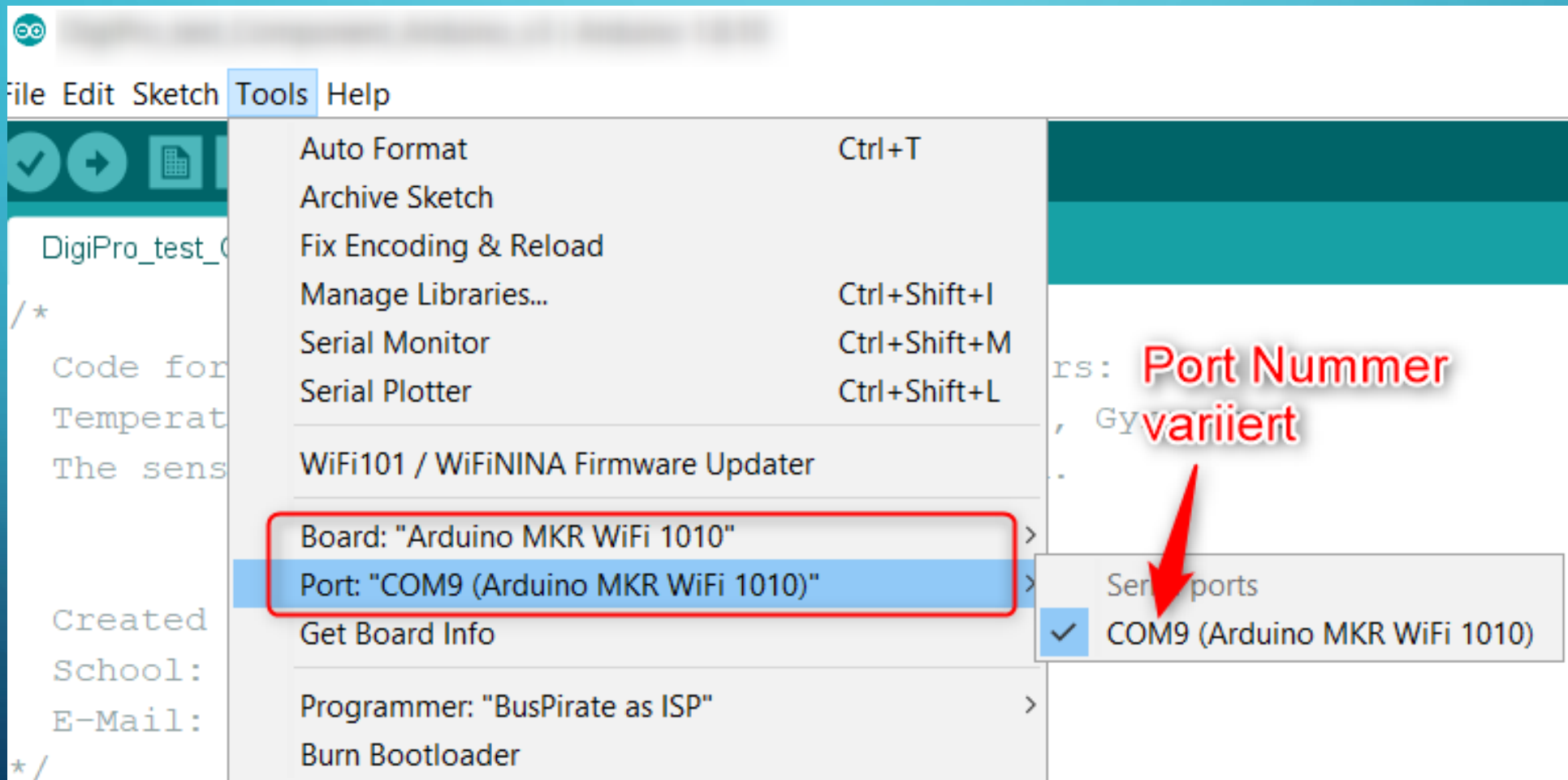
A decorative graphic on the left side of the slide, consisting of a network of light blue lines and small circles, resembling a circuit board or a stylized tree structure, set against a dark blue gradient background.

# ARDUINO VERBINDEN / WIEDERHOLUNG ARDUINO IDE OBERFLÄCHE

- Öffnen des RC Car Examples



# VERBINDEN ARDUINO

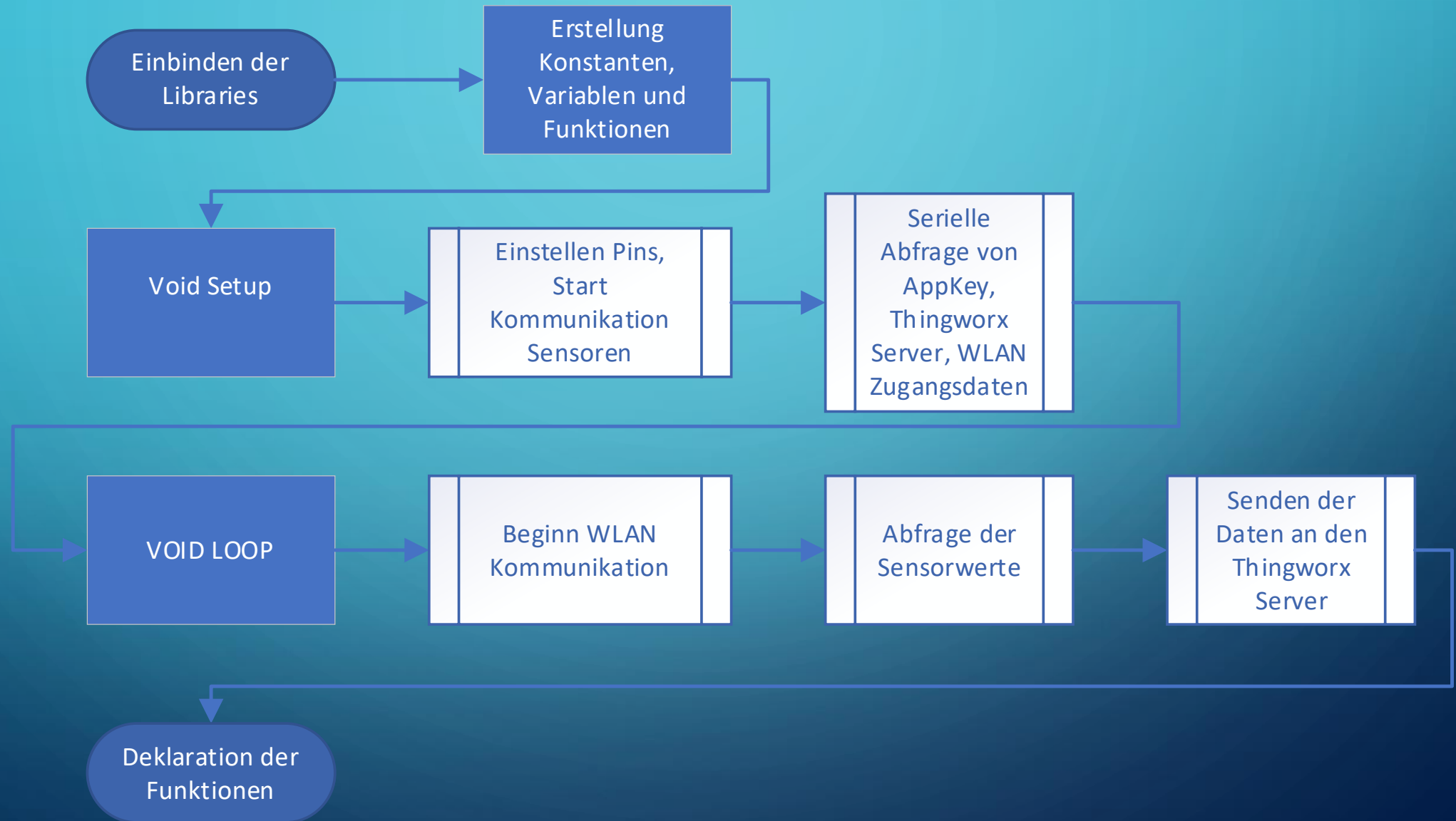


A decorative graphic on the left side of the slide, consisting of white lines and circles on a blue gradient background, resembling a circuit board or a stylized tree structure.

# AUFBAU RC CAR ARDUINO CODE



## • Flussdiagramm



- Dokumentation im Arduino Code



```
//*****  
//***** LIBRARIES *****  
//*****  
#include <Arduino.h>  
#include <DHT.h>  
#include <Wire.h>  
#include <SPI.h>  
#include <FlashStorage.h>  
#include <Wi-Fi-NINA.h>  
#include <utility/wifi_drv.h>  
#include <Thingworx_MKRWifi1010.h>  
#include <Thingworx_MKRWifi1010_Variable.h>
```

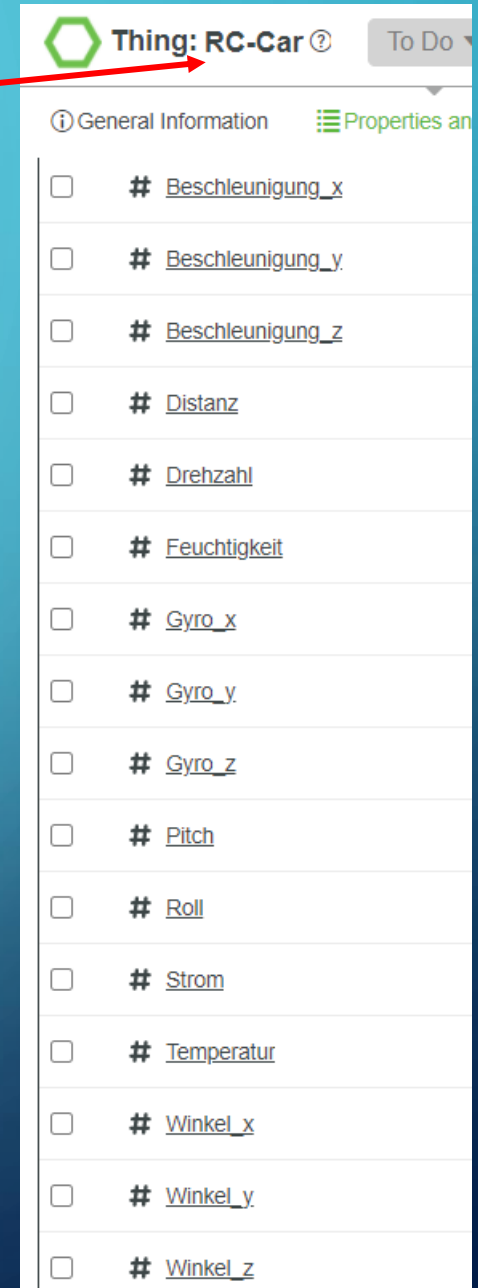
Abgrenzung der einzelnen  
Blöcke durch Kommentare



A decorative graphic on the left side of the slide, consisting of a network of white lines and small circles on a blue gradient background, resembling a circuit board or a neural network.

# THINGWORX WIEDERHOLUNG

- Erstellung Thing und Property siehe beigefügtes pdf
- Name Thing : RC-Car
- Namen der Properties für RC-Car
  - Temperatur
  - Feuchtigkeit
  - Drehzahl
  - Strom
  - Distanz
  - Pitch und Roll
  - Beschleunigung\_x , Beschleunigung\_y und Beschleunigung\_z
  - Gyro\_x , Gyro\_y und Gyro\_z



- Erstellung App-Key → Passwort für die Kommunikation zum Thingworx Server

The screenshot shows the Thingworx web interface. On the left, the 'Application Keys' option is highlighted in the sidebar. In the main content area, the 'Application Keys' list is visible, and the '+ New' button is highlighted with a red box. A red arrow points from this button to the 'New Application Key' form on the right.

The screenshot shows the 'New Application Key' form. Red boxes and arrows highlight the following fields and elements:

- Name (required):** Set to 'Arduino\_RC\_Car'.
- User Name Reference (required):** Set to 'Administrator'.
- Key ID:** Set to 'e44bc03b-ef2b-4581-a138-5f7cd8174e81'.
- Expiration Date:** Set to '01/20/2021 12:00 am'.
- Save button:** A green button at the top right of the form.

A red arrow points from the 'Expiration Date' field to a warning box:

**Wichtig!**  
Expiration Date einstellen, sonst  
ist der Schlüssel nur 1 Tag gültig!



# WICHTIGE EINSTELLUNGEN IN DER LIBRARY

- Datei : libraries/Thingworx\_MKRWifi1010/Thingworx\_MKRWifi1010\_Variable

```

Thingworx_MKRWifi1010_Variable.h
1  /*****
2  * Thingworx_MKRWifi1010_Variable.h - Definition of variables, which are used with the "Thingworx_MKRWifi1010.h" library
3  * Created by Armin Fischer, Oct 2020
4  * Version 3.0
5  * School: HTBLuVA Wiener Neustadt
6  * E-Mail: fia@htlwn.ac.at
7  *****/
8
9  #ifndef Thingworx_MKRWifi1010_Variable_H
10 #define Thingworx_MKRWifi1010_Variable_H
11
12  /*****
13  /*****TIMING VARIABLES*****/
14  /*****
15  const unsigned long TPOST = 2000; //Time between requests to TwX server (every 2 sec)
16  unsigned long lastConnectionTime = 0; //Last connection ms time between server requests
17
18  /*****
19  /*****WIFI CONNECTION*****/
20  /*****
21
22  char* ssid = ""; //WiFi SSID
23  char* password = ""; //WiFi Pass
24
25
26  /*****
27  /*****HOST THINGWORX*****/
28  /*****
29  char* host = "xxxxx.twx.htl.schule"; //TwX Host for HTL Austria twx.htl.schule (without http at beginning)
30  unsigned int port = 443; //TwX host port for https
31
32  /*****
33  /*****THINGWORX VARIABLES*****/
34  /*****
35  char appKey[] = "";
36  char thingName[] = "RC-Car"; //Thing name from TwX
37  String property_list[]={"Temperatur","Feuchtigkeit","Drehzahl","Strom","Distanz","Pitch","Roll","Beschleunigung_x","Beschleunigung_y","Beschleunigung_z","Gyro_x","Gyro_y","G
38
39  #endif

```

Sendefrequenz in ms

Thingname und Properties siehe vorige Folie

Schon voreinaestellt

- Andere Variablen (ssid, password, host und appKey) müssen bei den anderen Examples der Library eingestellt werden.

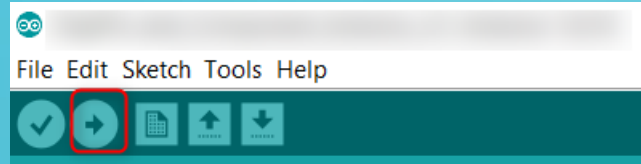


A decorative graphic on the left side of the slide, consisting of white lines and circles on a blue background, resembling a circuit board or a stylized tree structure.

# ARDUINO PROGRAMM HOCHLADEN



- Öffnen des Examples 009\_PUT\_RC\_Car
- Hochladen:



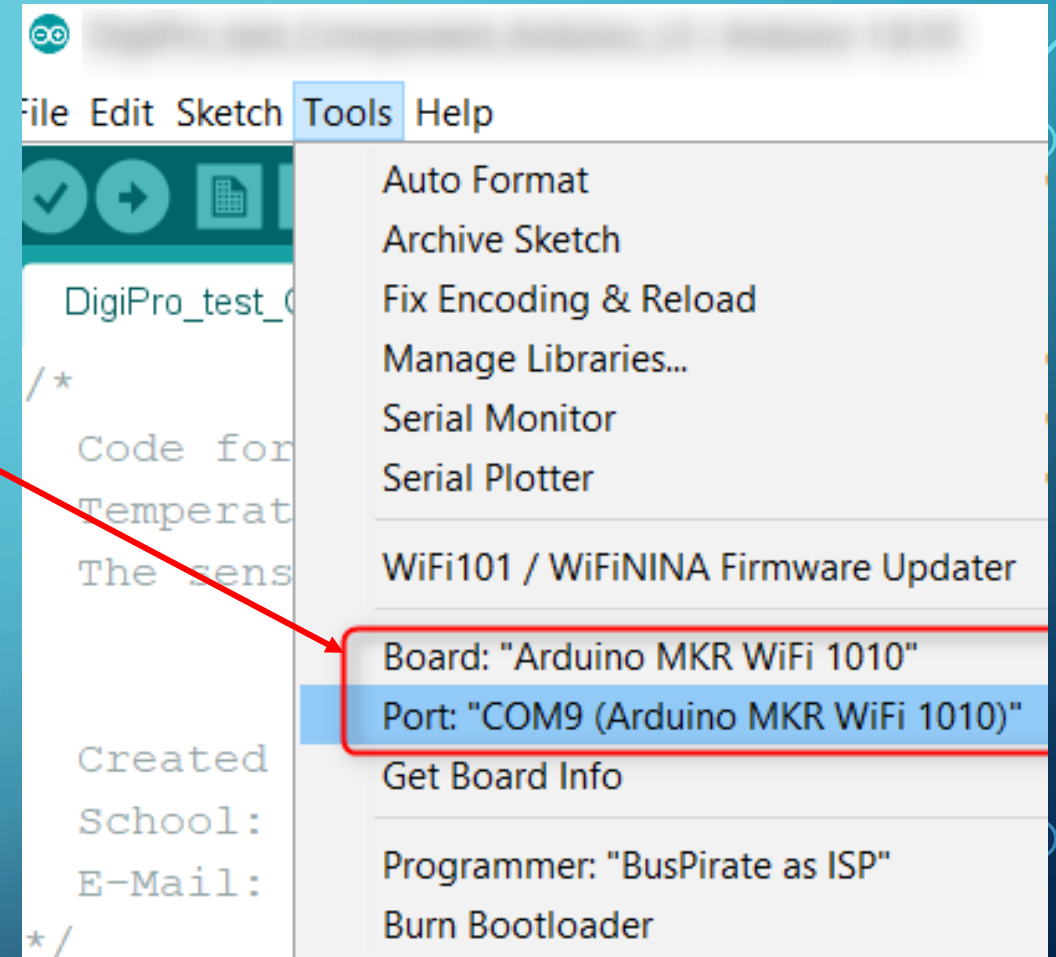
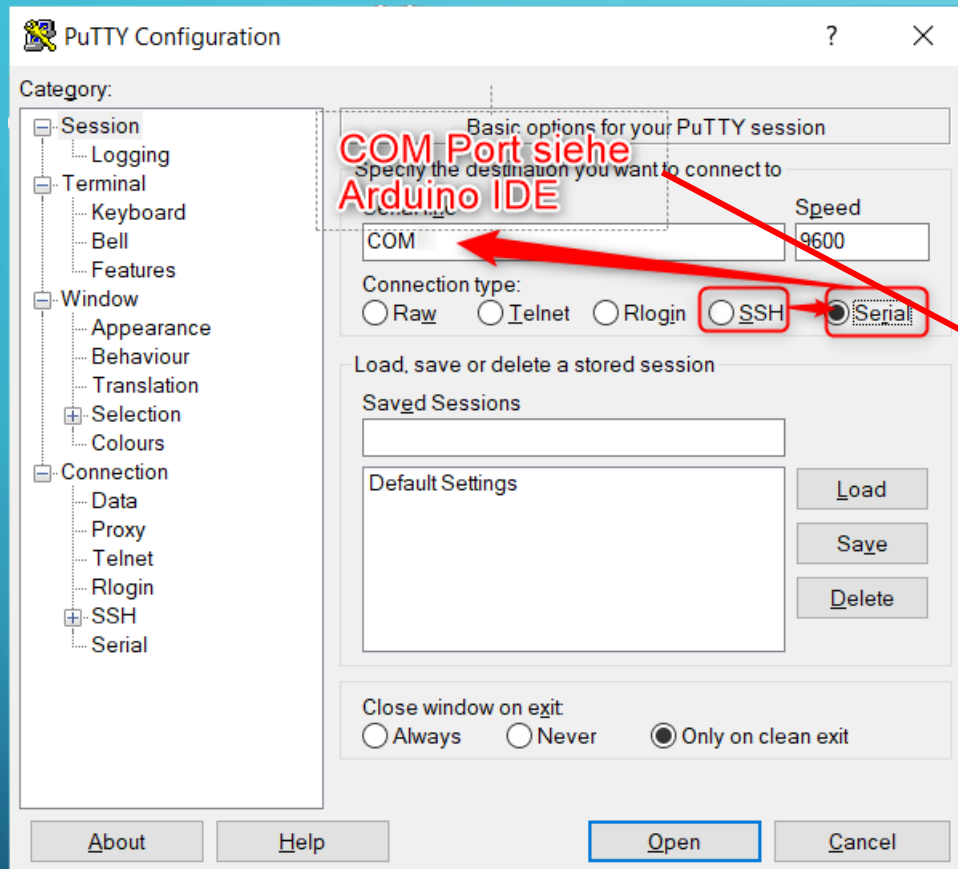
- Hochladen fertig, wenn auf der unteren Arduino Oberfläche folgendes Bild zu sehen ist

```
#include <DHT h>
<

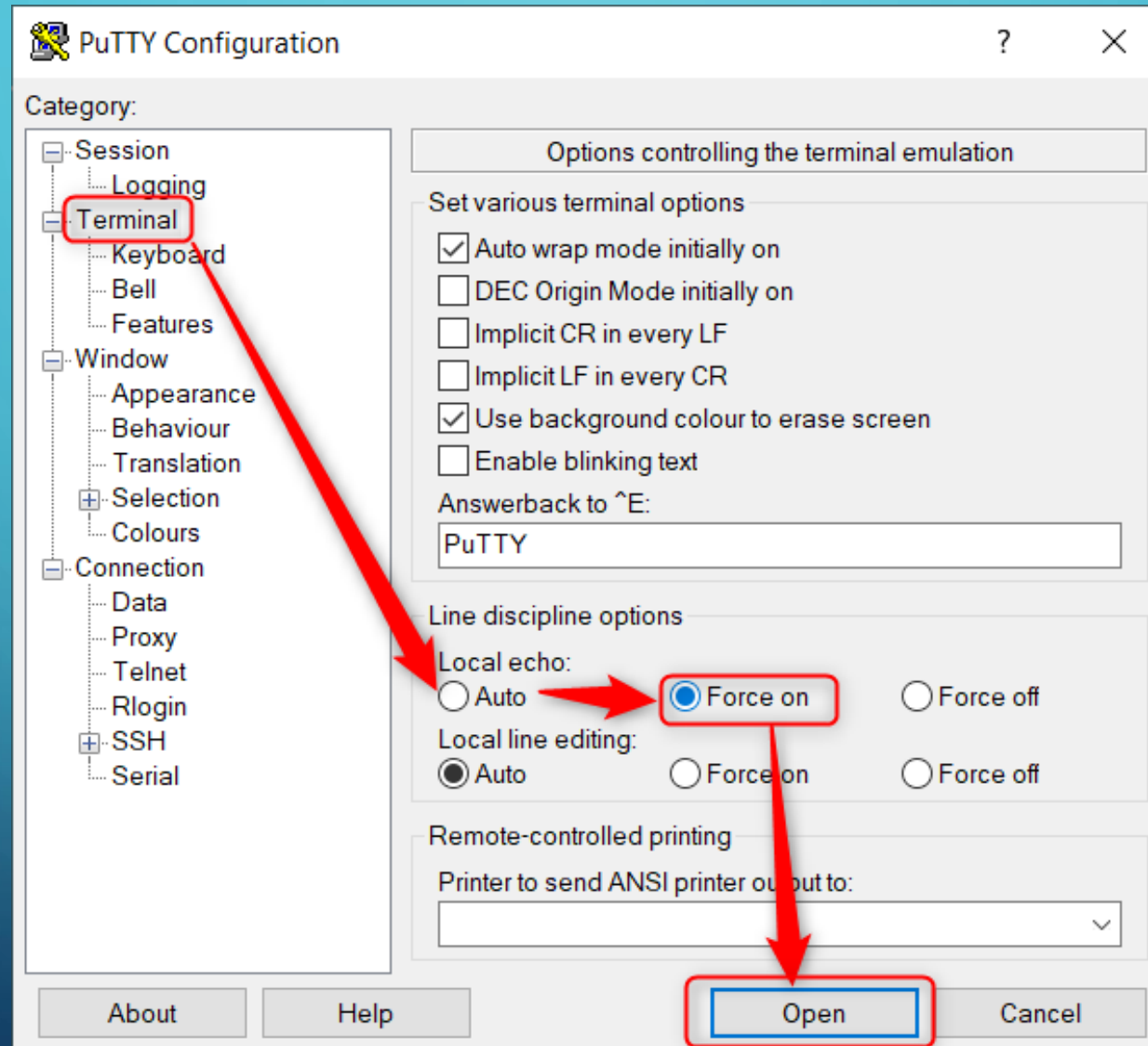
Done uploading.
[=====] 26% (192/713 pages)
[=====] 35% (256/713 pages)
[=====] 44% (320/713 pages)
[=====] 53% (384/713 pages)
[=====] 62% (448/713 pages)
[=====] 71% (512/713 pages)
[=====] 80% (576/713 pages)
[=====] 89% (640/713 pages)
[=====] 98% (704/713 pages)
[=====] 100% (713/713 pages)
done in 0.291 seconds

Verify 45572 bytes of flash with checksum.
Verify successful
done in 0.041 seconds
CPU reset.
```

- Einstellungen mit Software Putty



- Um die Eingaben per Tastatur am Bildschirm sehen zu können, muss noch folgende Einstellung getroffen werden:



- Nach dem Öffnen wird folgendes Fenster angezeigt:



```
COM9 - PuTTY
*****
*                                                                    *
*  DigiPro RC-Car Connection Setup                                  *
*                                                                    *
*  ATTENTION - When using PUTTY                                    *
*  --> entered chars are not displayed!!!                          *
*  --> confirm every value with <CTRL> + <J>                        *
*                                                                    *
*****

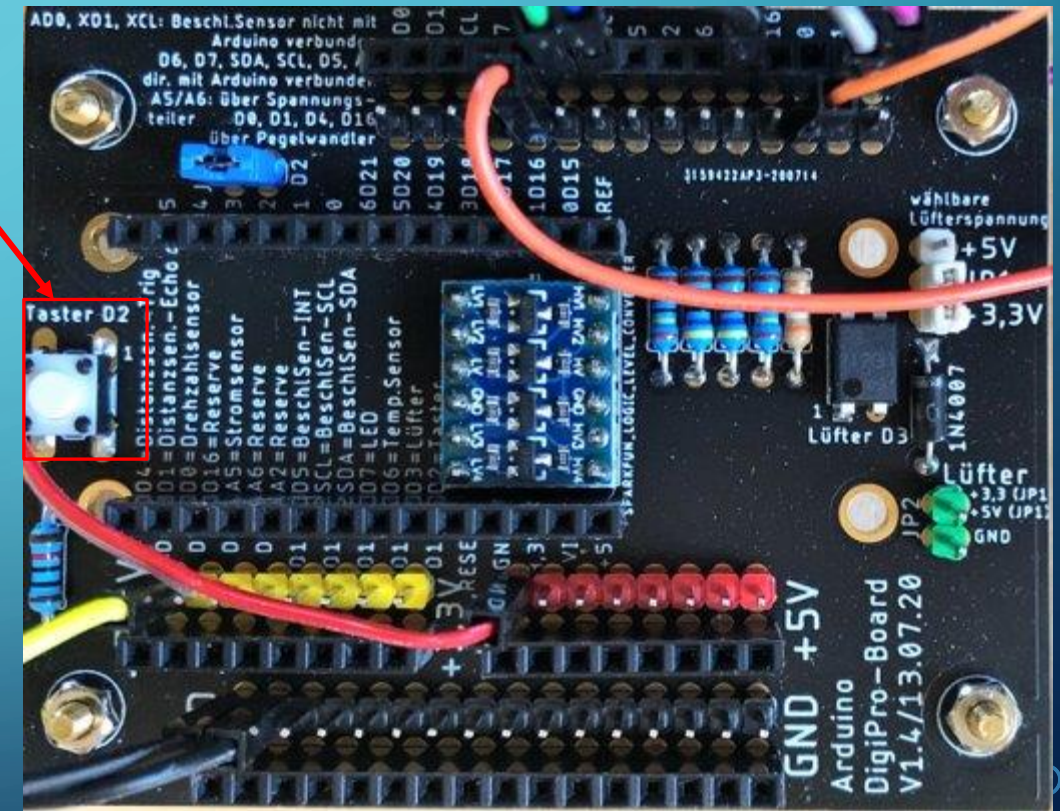
Insert AppKey:
Insert TWX-Server:
Insert SSID:
Insert Wifi Password:
```

- Es müssen der Reihe nach der AppKey, Thingworx Server, die SSID und das Passwort des WLANs eingegeben werden.
- Eine Eingabebestätigung wird mit STR+J durchgeführt (UND NICHT MIT ENTER!!!!!!).
- Es darf nicht die Backspace Taste zum Löschen von Buchstaben/Zahlen verwendet werden.
- Aus der Zwischenablage können Eingaben mit einem Rechtsklick in das Terminal Fenster kopiert werden.



- In dieses Status Fenster gelangt man, wenn
  - Das Arduino Programm per Arduino IDE neu hochgeladen wird.
  - beim Start/Reset des Arduinos der SETUP Knopf gedrückt wird.

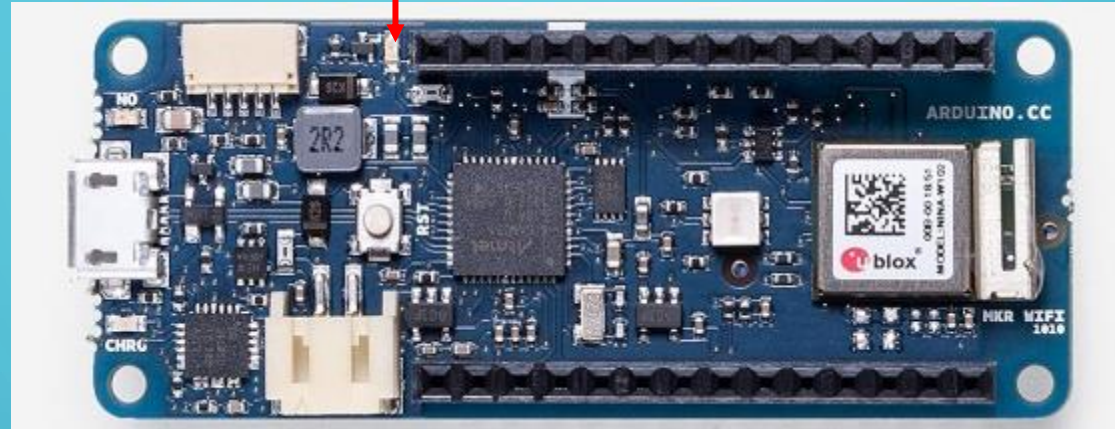
- Ziel: RC-Car schnell
  - bei Thingworx Server Änderungen
  - bei Wifi Änderungen konfigurieren zu können.



An abstract graphic on the left side of the slide, consisting of a network of light blue lines and small circles, resembling a circuit board or a neural network, set against a dark blue background.

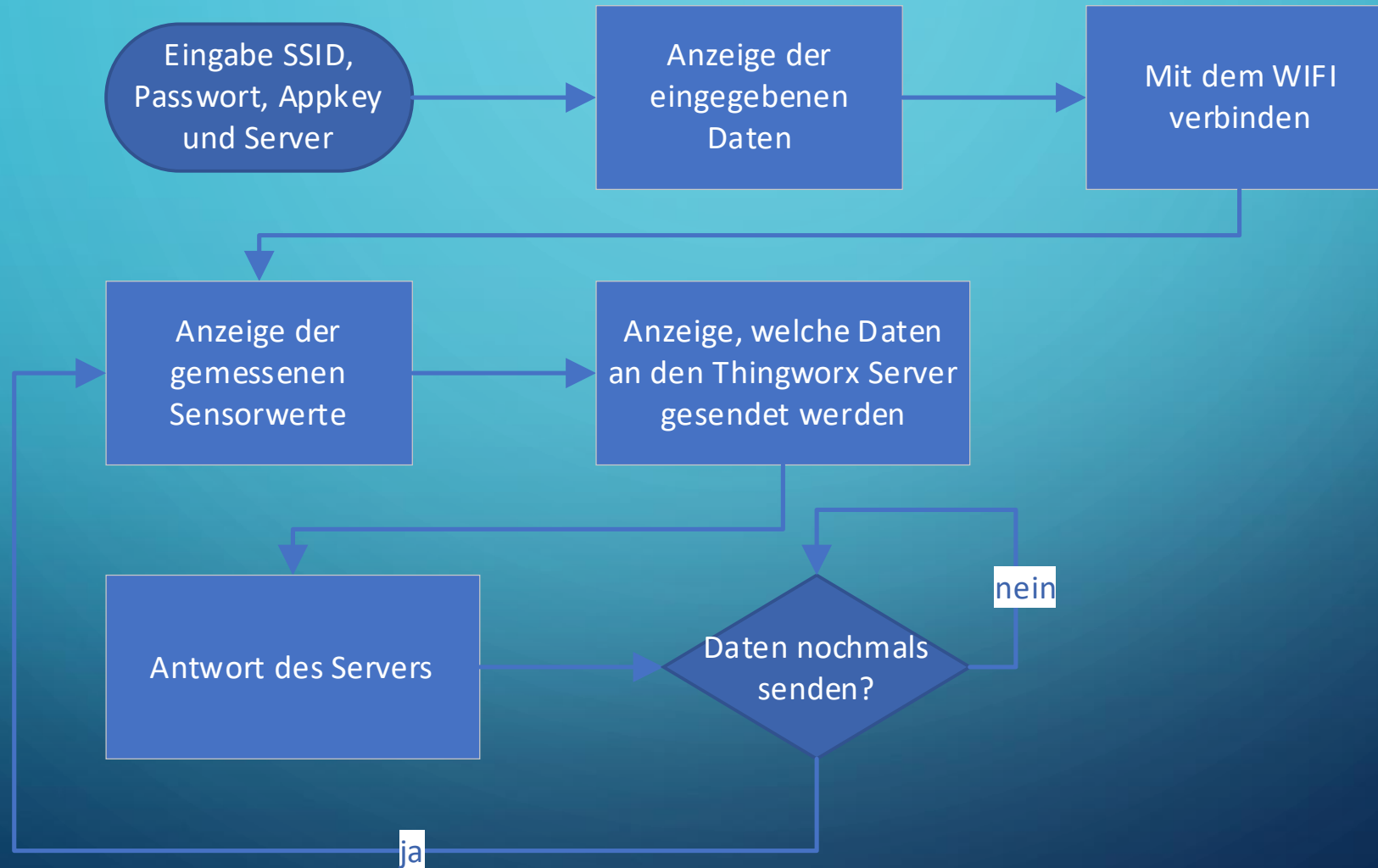
# STATUSAUSGABE DER KOMMUNIKATION

- Folgende Status können am Arduino LED abgelesen werden



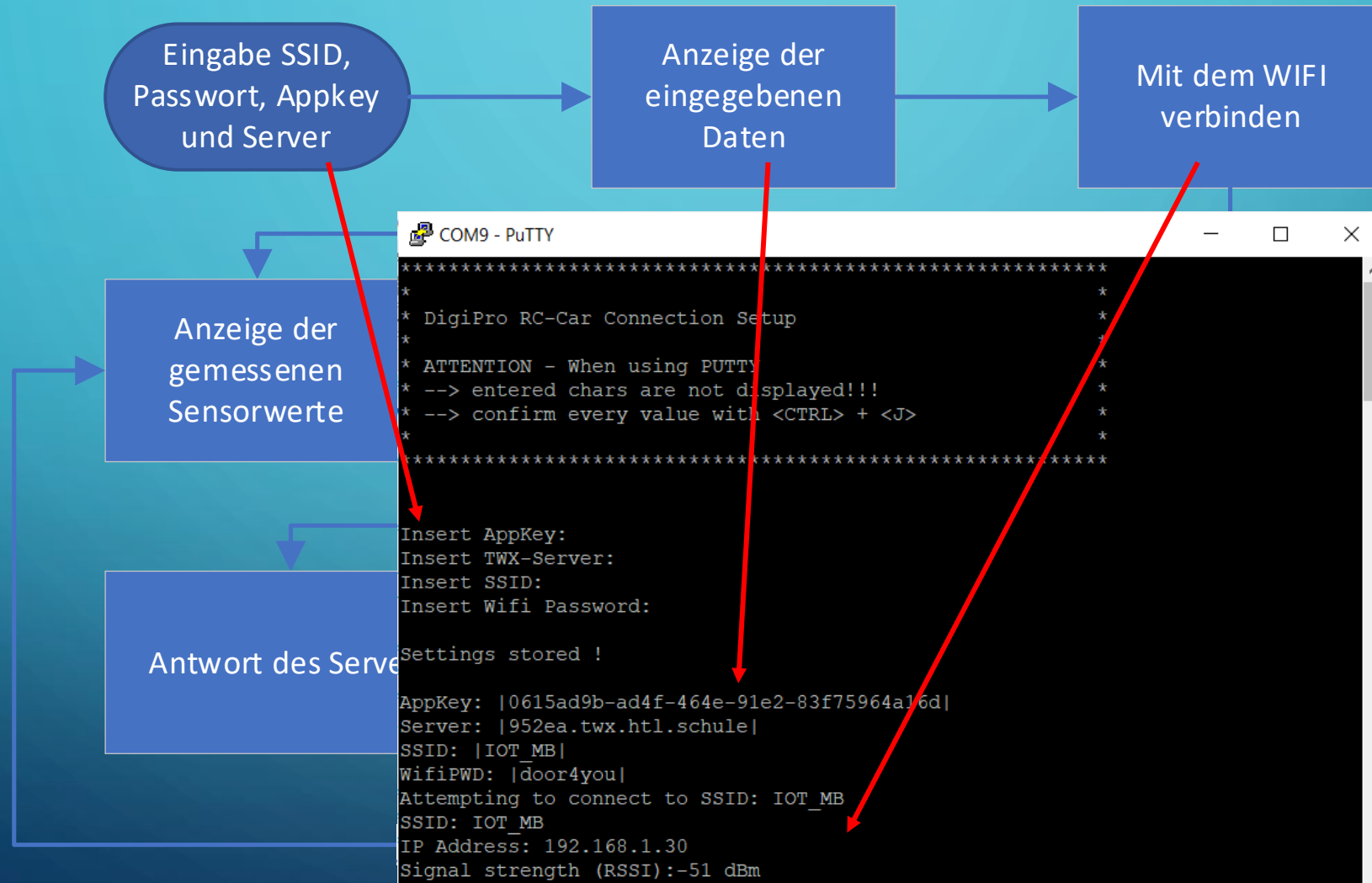
- Farbe rot: Connection Setup mit serieller Schnittstelle ist erforderlich (siehe Seite 23).
- Farbe blau: Daten werden an den Thingworx Server geschickt.
- Farbe grün: Code wird abgearbeitet.

- Wenn der Arduino mit dem Wifi verbunden ist und Daten zum Thingworx Server gesendet werden, dann ist das Status Fenster folgendermaßen aufgebaut.





- Wenn der Arduino mit dem Wifi verbunden ist und Daten zum Thingworx Server gesendet werden, dann ist das Status Fenster folgendermaßen aufgebaut.



- Wenn der Arduino mit dem Wifi verbunden ist und Daten zum Thingworx Server gesendet

```
COM9 - PuTTY
*****
[ACS712]
|   v_value = 1554.00
|   Current = -10.45

[HC-020K]
|   RPM = 0

[DHT22]
|   Temperature = nan
|   Humidity = nan

[HCSR04]
|   Distance = 0

[GY521]-Angle
|   Pitch = -35.28
|   Roll = -35.28

[GY521]-Acc
|   X = -0.00
|   Y = -0.00
|   Z = -0.00

[GY521]-Gyro
|   X = -0.02
|   Y = -0.02
|   Z = -0.02
```

E  
Pas

```
Connected to: 952ea.twx.htl.schule:443
PUT /Thingworx/Things/RC-Car/Properties/* HTTP/1.1
Host: 952ea.twx.htl.schule
Content-Type: application/json
Content-Length: 258
Connection: close
x-thingworx-session: false
appKey: 0615ad9b-ad4f-464e-91e2-83f75964a16d

{"Temperatur":null,"Feuchtigkeit":null,"Drehzahl":0,"Strom":-10.45,"Distanz":0,"Pitch":-35.28228,"Roll":-35.28228,"Beschleunigung_x":-0.000488,"Beschleunigung_y":-0.000488,"Beschleunigung_z":-0.000488,"Gyro_x":-0.015267,"Gyro_y":-0.015267,"Gyro_z":-0.015267}
```

Anzeige der  
gemessenen  
Sensorwerte

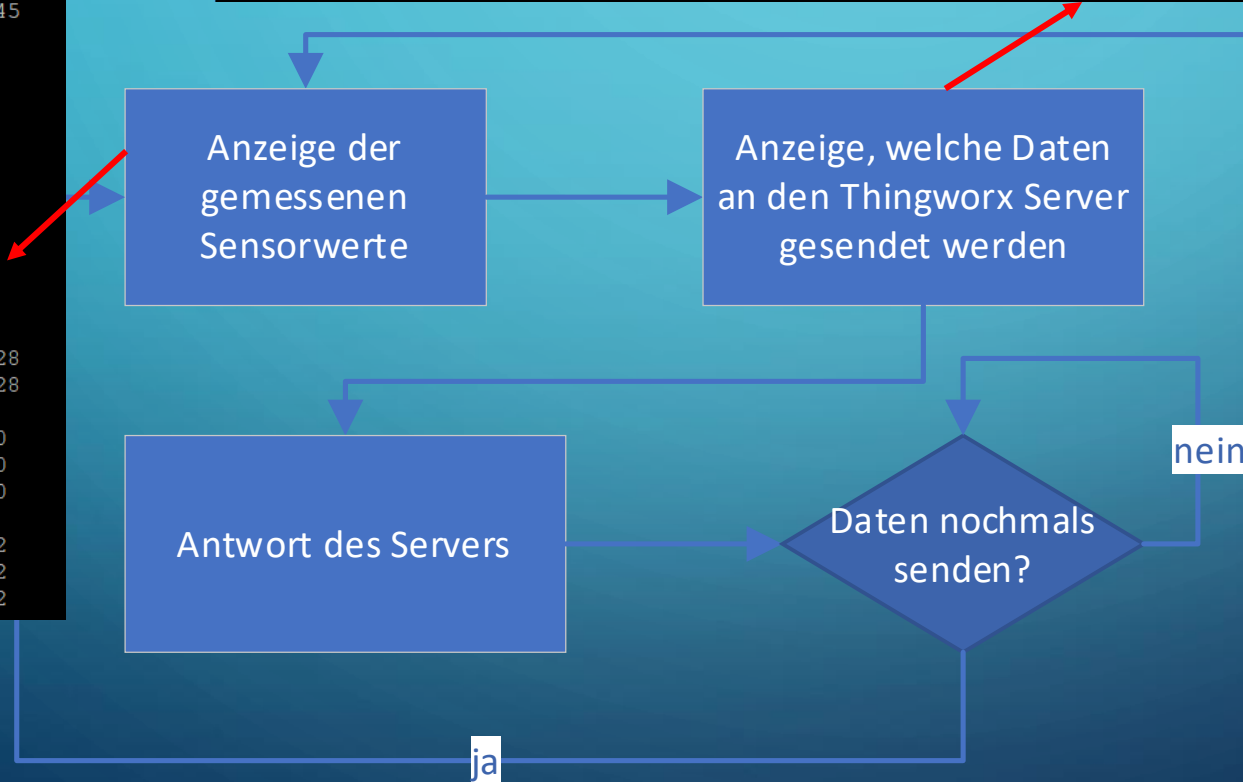
Anzeige, welche Daten  
an den Thingworx Server  
gesendet werden

Antwort des Servers

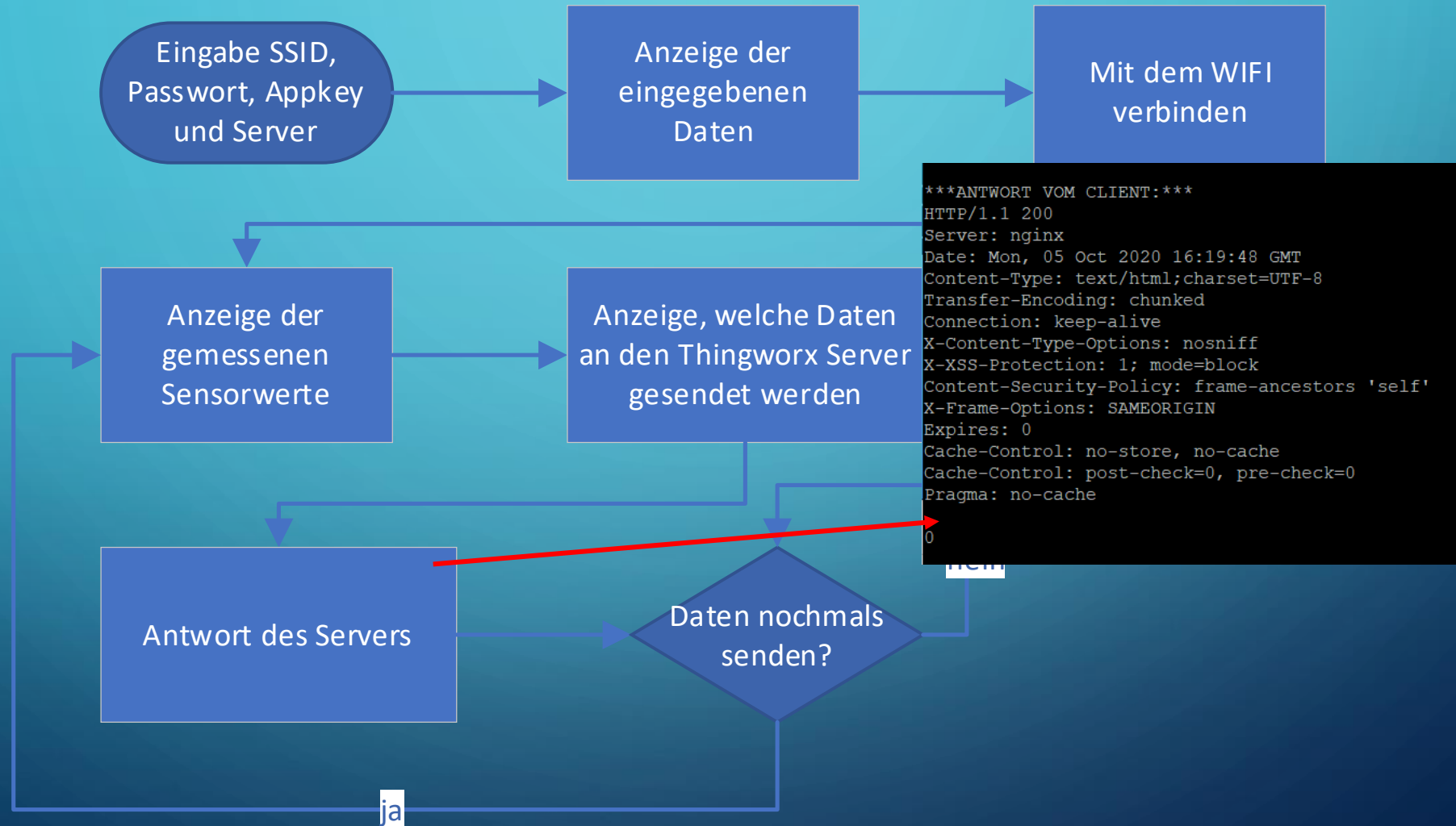
Daten nochmals  
senden?

nein

ja



- Wenn der Arduino mit dem Wifi verbunden ist und Daten zum Thingworx Server gesendet werden, dann ist das Status Fenster folgendermaßen aufgebaut.



- Wiederholung: Fehler beim Kommunikationsaufbau

```
***ANTWORT VOM CLIENT:***  
HTTP/1.1 200  
Server: nginx  
Date: Mon, 05 Oct 2020 16:19:48 GMT  
Content-Type: text/html;charset=UTF-8  
Transfer-Encoding: chunked  
Connection: keep-alive  
X-Content-Type-Options: nosniff  
X-XSS-Protection: 1; mode=block  
Content-Security-Policy: frame-ancestors 'self'  
X-Frame-Options: SAMEORIGIN  
Expires: 0  
Cache-Control: no-store, no-cache  
Cache-Control: post-check=0, pre-check=0  
Pragma: no-cache
```

HTTP- Code

0 Antwort des Server bei Fehler

- Falscher Propertyname

```
***ANTWORT VOM CLIENT:***  
HTTP/1.1 404
```

```
6b  
Unable To Write HC_SR05 on Arduino_MKR_1010_Wifi - Invalid Property &#x3a; HC_SR05 in Arduino_MKR_1010_Wifi  
0
```

- Falscher Server

```
***ANTWORT VOM CLIENT:***  
HTTP/1.1 503 Service Unavailable
```

```
<html><body><b>Http/1.1 Service Unavailable</b></body> </html>*****
```

- Falscher Sicherheitsschlüssel

```
***ANTWORT VOM CLIENT:***  
HTTP/1.1 401
```

- Link für HTTP Status-Tabelle: <https://de.wikipedia.org/wiki/HTTP-Statuscode>