

KAS

Interface App mittels http Request PLC
Variablen Fernzugriff
Programmiert mit Android Studio



Bisher erschienene Ausgaben

<i>Ausgabe</i>	<i>Bemerkungen</i>
A 03/2017	Erste Version

IMPORTANT NOTICE

Registrierte Warenzeichen

PDMM und KAS ist ein eingetragenes Warenzeichen der Kollmorgen Corporation

Android ist ein eingetragenes Warenzeichen der Google Inc.

Java ist ein eingetragenes Warenzeichen der Sun Microsystems Inc.

Inhalt

KAS_Iface.apk	Applikationsinstallationsdatei
HttpRequestTask.java	Funktion für die Asynchrone Kommunikation
MainActivity.java	Programmbeispielcode
activity_main.xml	Grafische Oberfläche Programmbeispiel
MN_KASIfaceApp_0A0_de	Handbuch in Deutsch
MN_KASIfaceApp_0A0_en	Handbuch in Englisch

Inhaltsverzeichnis

Einleitung	4
<i>Syntax</i>	5
Installation	6
<i>KAS</i>	6
<i>KASIface</i>	7
Java Code	8
<i>MainActivity.java</i>	8
<i>activity_main.xml</i>	8
<i>HttpRequestTask.java</i>	9

Einleitung

Das *KASIface* Beispiel soll aufzeigen wie ein direkter Zugriff auf KAS-Variablen von einem mobilen Gerät realisiert werden kann. Dabei werden die Variablen als http Anfragen im Text Format angesprochen. Die Programmierung erfolgte mit Android Studio in Java.

KASIface

Kollmorgen Automation Suite

Control Panel

☐ Automatic

☐ Manual

☐ Off

E-STOP

Manual Mode Functions

TravelSpeed

[deg/s]

ABS MOVE

[deg]

REL MOVE

[deg]

Auto Mode Functions

Machine Speed

[deg/s]



User defined Variable access

Variable name

READ VALUE

Variable value

Haftungsausschluss:

	Alle Programmteile die mit dieser Version kommen, werden zur Verfügung gestellt ohne Mangelgewähr oder Funktionsgarantien. Es können technische als auch redaktionelle Mangel in der Spezifikation, wie auch der Ausführung vorhanden sein. Jeder Nutzer übernimmt die Verantwortung für die Verwendung von zur Verfügung gestellten Beispielen. Änderungen können jederzeit und ohne Benachrichtigung erfolgen.
	KOLLMORGEN srl haftet nicht für Schäden und schliesst alle Ansprüche aus, die sich durch den Einsatz des „KASIface“ oder Programmteilen daraus ergeben könnten.

Syntax

Die Anfrage um eine Variable zu lesen kann direkt in einem Web-Browser erfolgen.

Für die Variable *Axis1Status* sieht das dann wie folgt aus:

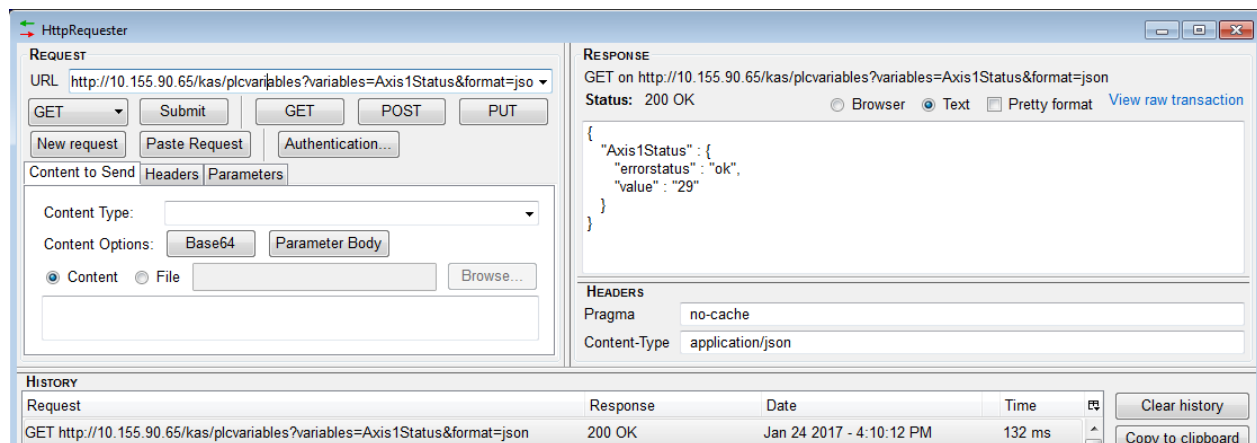
http://10.155.90.64//kas/plcvariables?variables=Axis1Status&format=text

Um die Variable *bEstop* auf 1 zu setzen ist folgende Abfrage zu tätigen:

*http://10.155.90.64//kas/plcvariables?&format=text
bEstop=1*

Ein weiteres Beispiel wie dies mit Excel oder Visual Basic aussieht, sowie Hilfen zu den verwendeten Funktionen findet sich in der Kollmorgen Wissensdatenbank KDN.

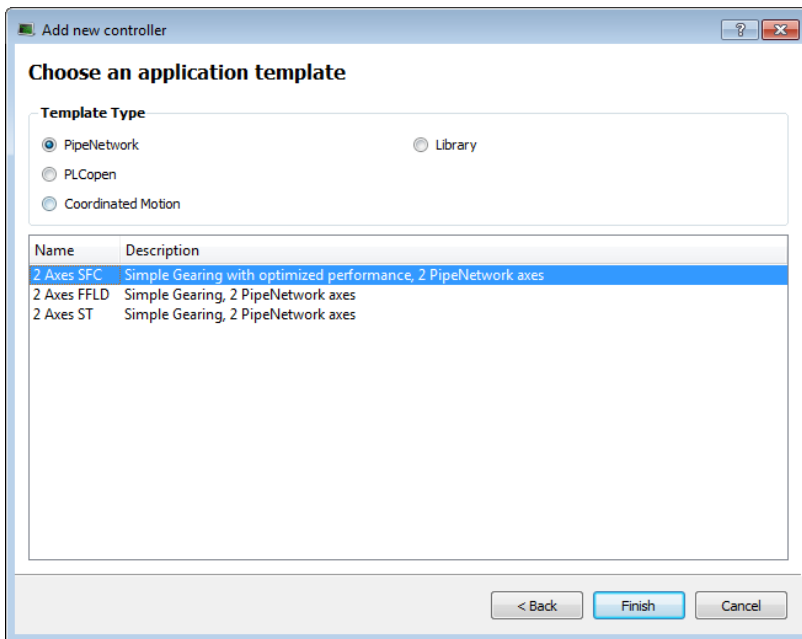
Firefox bietet mit dem HttpRequester add-on die Möglichkeit alle Befehle direkt im Browser zu testen.



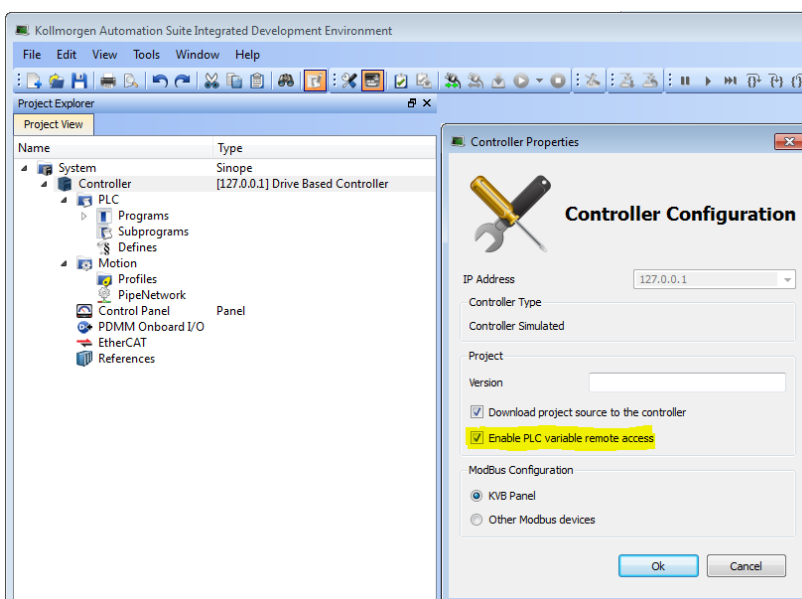
Installation

KAS

In der KAS Software kann das Standard Beispiel Projekt für PipeNetwork genutzt werden.
File -> New -> PDMM (oder entsprechenden Controller auswählen) -> Next -> PipeNetwork -> 2 Axis SFC -> Finish. Es sollte die Ausführung mit SFC genutzt werden, da der Funktionsumfang wie auch die Variablen Namen in den andern Versionen leicht variieren.



Um den Fernzugriff auf alle KAS Variablen zu ermöglichen muss die Funktion „Enable PLC variabel remote access“ aktiviert werden. Dies findet sich in den Einstellungen welche mit einem Rechtsklick auf Controller gemacht werden können.



Mit diesen Einstellungen kann das Projekt kompiliert und geladen werden.

KASIface

Die *KAS_Iface.apk* kann auf einem Android Betriebssystem installiert werden und ist danach direkt einsatzfähig. Das Icon kann danach bei den installierten Programmen gefunden werden.



Beim aufstarten muss die IP-Adresse des KAS Controllers eingegeben und mit OK bestätigt werden. Mit dieser Adresse werden nun alle Variablen gesucht. Sollte die Kommunikation nicht freigegeben, oder Verbindung unterbrochen sein, wird „no value“ zurückgegeben.

Enter the PxMM IP-Address

10.155.90.64

OK

Wenn die Kommunikation steht werden alle Variablen zyklisch (250ms) gelesen, geschrieben wird Event basiert.

Die Events sind wie folgt definiert:

Typ	Anwendung	Event
Button	E-Stop	Bei einem <i>Klick</i> wird jeweils zwischen 0 und 1 gewechselt
	AbsMove, RelMove	Bei einem <i>Klick</i> wird eine 1 direkt gefolgt von einer 0 geschrieben.
RadioButton	Mode	Bei eine Statuswechsel wird der entsprechende Modus (0, 1, 2) geschrieben
EditText	TravelSpeed, MasterAbsPos, MasterDeltaPos, MachineSpeed	Bei einem <i>Klick</i> wird das Lesen der entsprechenden Variable unterbrochen, bis der Fokus wieder verloren ist. Geschrieben wird mit <i>Enter</i> . Nachdem schreiben geht der Fokus immer verloren.
	User defined variable	Im Lesemodus wird der Wert zyklisch gelesen Im Schreibmodus wird der Wert mit <i>Enter</i> geschrieben

Die Anwendung wurde mit Android 6.0.1 auf einem Samsung A3 getestet, hat aber keine Minimumversion definiert. Andere Hard- oder Softwarekonstellationen können Änderungen benötigen, welche hier nicht aufgeführt sind.

Java Code

Der Angefügte Java Code besteht aus drei Teilen.

MainActivity.java

Haupt Programm, hier werden all Ein- wie auch Ausgaben abgehandelt.

protected void onCreate(Bundle)

- AlertDialog für die Eingabe der IP-Adresse des KAS
- Eventhandling aller Schreibeingaben
- Runnable Handler für den Zyklischen Aufruf von ReadKasVar()

public void WriteBoolTrueFalse(String, boolean)

- Der *String* gibt den Parameter an welcher mit *boolean* (WAHR) Beschrieben wird
- Die Funktion ruft sich nach erfolgreicher Ausführung nochmals mit FALSCH auf um den Parameter wieder auf null zusetzen.

public void ReadKasVar()

- Alle zu lesenden Parameter sind hier eingefügt
- Diese Funktion wird durch den Runnable Handler alle 250ms aufgerufen

activity_main.xml

XML Beschreibungsdatei der Grafischen Oberfläche. Die Darstellung ist relative und in mehrere GridLayout aufgeteilt.

HttpRequestTask.java

Erweiterung des AsyncTask für alle URL basierten Anfragen. Diese Funktion wird im MainActivity für das Lesen und Schreiben aufgerufen.

Die Funktion hat bis zu drei Parameter:

Als erstes die IP-Adresse als String, diese wird in der Software mit der Variable *strIpAddr* geführt.

Als zweites den Parameternamen als String.

Als drittes den zu Schreibenden Wert als String.

Mit der *AsyncResponse* kann die Text Antwort als String zurückgelesen werden. Wird eine Variable gelesen findet sich hier der Wert. Beim Schreiben kommt ein „ok“ oder ein entsprechender Fehler zurück.

Lesen:

Wird die Funktion mit zwei Argumenten aufgerufen, so wird der Parameter gelesen.

```
AsyncTask<String,Void,String> ReadKasVar_AbsMove = new HttpRequestTask(new HttpRequestTask.AsyncResponse() {  
    @Override  
    public void AsyncResponse(String str_Result) {  
        if (!bWriteAbsMovePos) {  
            TextAbsMove.setText(str_Result);  
        }  
    }  
}).execute(strIpAddr, "MasterAbsPos");
```

Schreiben:

Wird die Funktion mit drei Argumenten aufgerufen, so wird der Parameter geschrieben.

```
AsyncTask<String,Void,String> WriteKasVar_AbsPos = new HttpRequestTask(new HttpRequestTask.AsyncResponse() {  
    @Override  
    public void AsyncResponse(String str_Result) {  
    }  
}).execute(strIpAddr, "MasterAbsPos", TempWriteValue);
```

WISSENSWERTES ÜBER KOLLMORGEN

Kollmorgen ist ein führender Anbieter von Antriebssystemen und Komponenten für den Maschinenbau. Dank großem Know-how im Bereich Antriebssysteme, höchster Qualität und umfassender Fachkenntnisse bei der Verknüpfung und Integration von standardisierten und spezifischen Produkten liefert Kollmorgen optimale Lösungen, die mit Leistung, Zuverlässigkeit und Bedienerfreundlichkeit bestechen und Maschinenbauern einen wichtigen Wettbewerbsvorteil bieten.

Besuchen Sie www.kollmorgen.com für Unterstützung bei der Lösung Ihrer Applikationsaufgabe oder kontaktieren Sie uns unter:

KOLLMORGEN srl

Largo Brughetti 1/B2

20813 Bovisio Masciago, Italia

Web www.kollmorgen.com

Email mil-info@kollmorgen.com

Tel.: +39 - 0362 - 594260

Fax: +39 - 0362 - 594263