

Anwendungsbeispiel • 05/2016

Erstellen einer $f(x)$ -Kurvenanzeige

WinCC Comfort V13 SP1 / WinCC Advanced V13 SP1



<https://support.industry.siemens.com/cs/ww/de/view/109481693>

Gewährleistung und Haftung

Hinweis

Die Anwendungsbeispiele sind unverbindlich und erheben keinen Anspruch auf Vollständigkeit hinsichtlich Konfiguration und Ausstattung sowie jeglicher Eventualitäten. Die Anwendungsbeispiele stellen keine kundenspezifischen Lösungen dar, sondern sollen lediglich Hilfestellung bieten bei typischen Aufgabenstellungen. Sie sind für den sachgemäßen Betrieb der beschriebenen Produkte selbst verantwortlich. Diese Anwendungsbeispiele entheben Sie nicht der Verpflichtung zu sicherem Umgang bei Anwendung, Installation, Betrieb und Wartung. Durch Nutzung dieser Anwendungsbeispiele erkennen Sie an, dass wir über die beschriebene Haftungsregelung hinaus nicht für etwaige Schäden haftbar gemacht werden können. Wir behalten uns das Recht vor, Änderungen an diesen Anwendungsbeispiele jederzeit ohne Ankündigung durchzuführen. Bei Abweichungen zwischen den Vorschlägen in diesem Anwendungsbeispiel und anderen Siemens Publikationen, wie z. B. Katalogen, hat der Inhalt der anderen Dokumentation Vorrang.

Für die in diesem Dokument enthaltenen Informationen übernehmen wir keine Gewähr.

Unsere Haftung, gleich aus welchem Rechtsgrund, für durch die Verwendung der in diesem Applikationsbeispiel beschriebenen Beispiele, Hinweise, Programme, Projektierungs- und Leistungsdaten usw. verursachte Schäden ist ausgeschlossen, soweit nicht z. B. nach dem Produkthaftungsgesetz in Fällen des Vorsatzes, der groben Fahrlässigkeit, wegen der Verletzung des Lebens, des Körpers oder der Gesundheit, wegen einer Übernahme der Garantie für die Beschaffenheit einer Sache, wegen des arglistigen Verschweigens eines Mangels oder wegen Verletzung wesentlicher Vertragspflichten zwingend gehaftet wird. Der Schadensersatz wegen Verletzung wesentlicher Vertragspflichten ist jedoch auf den vertragstypischen, vorhersehbaren Schaden begrenzt, soweit nicht Vorsatz oder grobe Fahrlässigkeit vorliegt oder wegen der Verletzung des Lebens, des Körpers oder der Gesundheit zwingend gehaftet wird. Eine Änderung der Beweislast zu Ihrem Nachteil ist hiermit nicht verbunden.

Weitergabe oder Vervielfältigung dieser Anwendungsbeispiele oder Auszüge daraus sind nicht gestattet, soweit nicht ausdrücklich von der Siemens AG zugestanden.

Security-hinweise

Siemens bietet Produkte und Lösungen mit Industrial Security-Funktionen an, die den sicheren Betrieb von Anlagen, Systemen, Maschinen und Netzwerken unterstützen.

Um Anlagen, Systeme, Maschinen und Netzwerke gegen Cyber-Bedrohungen zu sichern, ist es erforderlich, ein ganzheitliches Industrial Security-Konzept zu implementieren (und kontinuierlich aufrechtzuerhalten), das dem aktuellen Stand der Technik entspricht. Die Produkte und Lösungen von Siemens formen nur einen Bestandteil eines solchen Konzepts.

Der Kunde ist dafür verantwortlich, unbefugten Zugriff auf seine Anlagen, Systeme, Maschinen und Netzwerke zu verhindern. Systeme, Maschinen und Komponenten sollten nur mit dem Unternehmensnetzwerk oder dem Internet verbunden werden, wenn und soweit dies notwendig ist und entsprechende Schutzmaßnahmen (z.B. Nutzung von Firewalls und Netzwerksegmentierung) ergriffen wurden.

Zusätzlich sollten die Empfehlungen von Siemens zu entsprechenden Schutzmaßnahmen beachtet werden. Weiterführende Informationen über Industrial Security finden Sie unter <http://www.siemens.com/industrialsecurity>.

Die Produkte und Lösungen von Siemens werden ständig weiterentwickelt, um sie noch sicherer zu machen. Siemens empfiehlt ausdrücklich, Aktualisierungen durchzuführen, sobald die entsprechenden Updates zur Verfügung stehen und immer nur die aktuellen Produktversionen zu verwenden. Die Verwendung veralteter oder nicht mehr unterstützter Versionen kann das Risiko von Cyber-Bedrohungen erhöhen.

Um stets über Produkt-Updates informiert zu sein, abonnieren Sie den Siemens Industrial Security RSS Feed unter <http://www.siemens.com/industrialsecurity>.

Inhaltsverzeichnis

Gewährleistung und Haftung.....	2
1 Aufgabe.....	4
1.1 Übersicht	4
2 Lösung.....	6
2.1 Übersicht	6
2.2 Hard- und Software-Komponenten	7
2.2.1 Gültigkeit.....	7
2.2.2 Verwendete Komponenten.....	7
3 Grundlagen.....	8
4 Funktionsweise.....	9
4.1 Gesamtübersicht	9
4.2 SPS-Programm	10
4.3 HMI-Programm.....	12
4.3.1 TrendControl	12
4.3.2 TrendControlSQL	14
4.3.3 Variablenhaushalt.....	16
5 Installation und Inbetriebnahme	17
5.1 Installation der Hardware	17
5.2 Installation der Software (Download)	17
5.3 Inbetriebnahme	17
6 Bedienung der Applikation.....	20
6.1 Startbildschirm.....	21
6.2 f(x)-Kurvenanzeige – CSV-Datei.....	22
6.3 f(x)-Kurvenanzeige – SQL Datenbank	23
7 Literaturhinweise	24
8 Historie.....	24

1 Aufgabe

1.1 Übersicht

Einleitung

Oft ist es erforderlich, technische Vorgänge über eine Reihe von Wertepaaren (z.B. Druck und Temperatur in thermodynamischen Prozessen) in Kurvenform darzustellen.

Voraussetzung hierfür ist die Möglichkeit, einen funktionellen Zusammenhang zeitlich unabhängig darstellen zu können.

Eine $f(x)$ -Kurvenanzeige in WinCC Comfort/Advanced bietet grundsätzlich diese Möglichkeit. Allerdings ist diese in ihrer technischen Natur nicht zeitlich unabhängig, da sie Wertepaare von x und $f(x)$ anhand der aufgenommenen Zeitstempel zuordnet.

Bei der Verwendung einer $f(x)$ -Kurvenanzeige gibt es daher folgende Einschränkungen:

- Die beiden Zeitstempel der Werte eines Paares müssen absolut identisch sein.
- Archivvariablen müssen „zyklisch kontinuierlich“ erfasst werden, bei der Archivierung „auf Anforderung“ (unterschiedlicher Zeitstempel) kann keine Anzeige erfolgen.
- Arrayvariablen werden nicht unterstützt.

1 Aufgabe

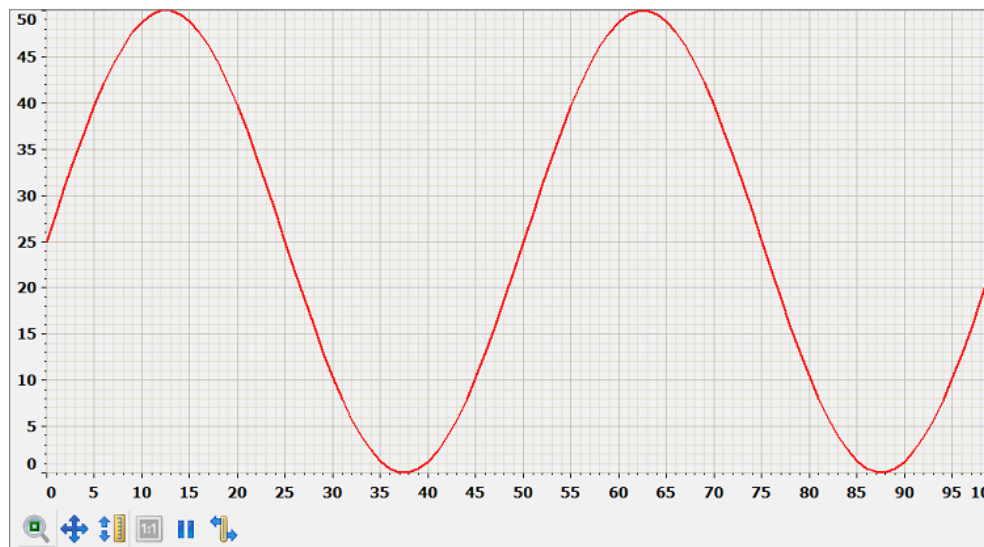
1.1 Übersicht

Überblick über die Automatisierungsaufgabe

Folgendes Bild gibt einen Überblick über die Automatisierungsaufgabe:

Abbildung 1-1

Index	Wertepaar Array [0]	Wertepaar Array [1]	Wertepaar Array [2]	Wertepaar Array [3]	Wertepaar Array [n]
X-Wert	0,0	5,0	12,5	20,0	...
Y-Wert	25,0	40,0	50,0	40,0	...



Beschreibung der Automatisierungsaufgabe

- Das Anwendungsbeispiel soll für Comfort Panel und RT Advanced umgesetzt werden.
- Zwei Arrays von Real-Werten (X-Wert und Y-Wert) in der Steuerung sollen in einer $f(x)$ -Kurvenanzeige abgebildet werden.
- Ein Array enthält dabei eine Reihe von Werten einer Variablen, beide Arrays zusammen stellen eine Reihe von Wertepaaren (Datenbereich) dar.
- Die Anzahl der abgebildeten Wertepaare wird bestimmt durch die Länge der beiden Arrays.
- Die Darstellung der Kurven soll in Form von parametrierbaren mathematischen Funktionen erfolgen, um die Leistungsfähigkeit der $f(x)$ -Kurvenanzeige zu verdeutlichen.

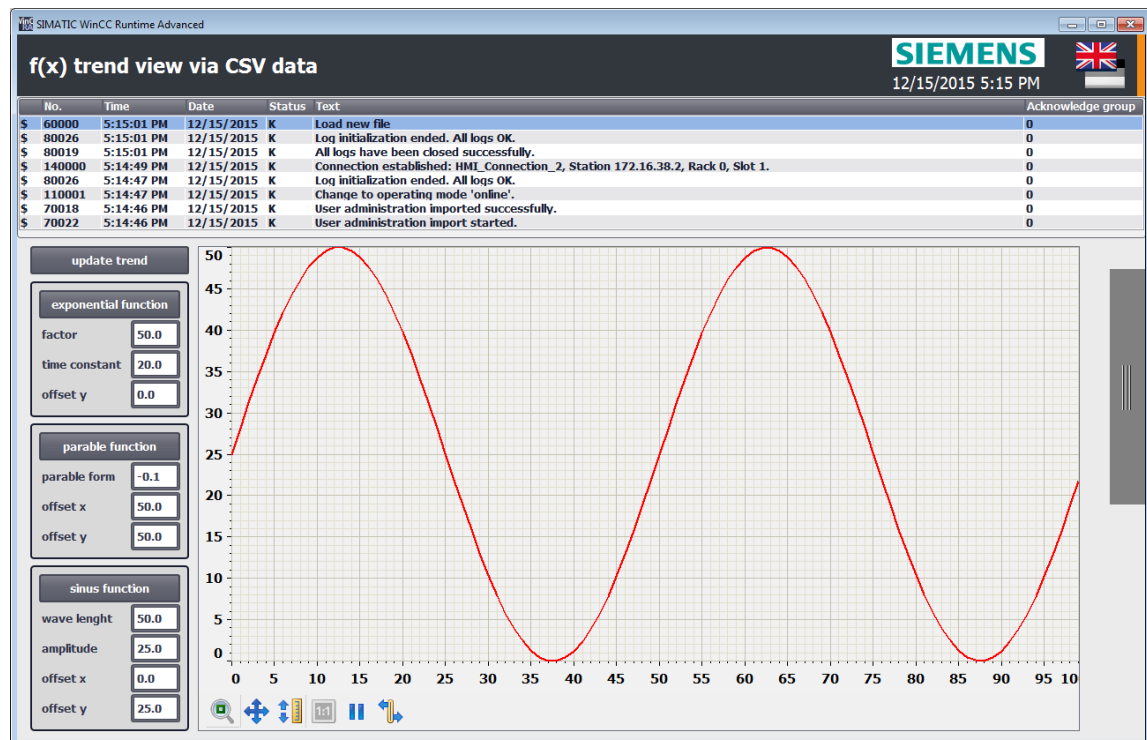
2 Lösung

2.1 Übersicht

Schema

Die folgende Abbildung zeigt ein Bild der Projektierung zur Laufzeit:

Abbildung 2-1



Vorteile

Die hier vorgestellte Lösung bietet Ihnen folgende Vorteile:

- Zeit- und Kostenersparnis durch ausprojektierte Funktionen
- Erweiterbarkeit bezüglich eigener Anforderungen
- Keine Einschränkungen der f(x)-Kurvenanzeige
- Verwendbar für Comfort Panel und WinCC Advanced RT

Abgrenzung

Diese Applikation enthält keine Beschreibung der grundlegenden Programmierung von Steuerung und HMI.

Grundlegende Kenntnisse über diese Themen werden voraus gesetzt.

Vorausgesetzte Kenntnisse

Grundlegende Kenntnisse über die Programmierung mit VB-Script und SCL werden voraus gesetzt.

2.2 Hard- und Software-Komponenten

2.2.1 Gültigkeit

Diese Applikation ist gültig für:

- STEP 7 ab V13 SP1
- WinCC Comfort ab V13 SP1
- WinCC Advanced ab V13 SP1

2.2.2 Verwendete Komponenten

Die Applikation wurde mit den nachfolgenden Komponenten erstellt:

Hardware-Komponenten

Tabelle 2-1

Komponente	Anz.	Artikelnummer	Hinweis
TP1200 Comfort	1	6AV2124-0MC01-0AX0	Bei Verwendung von WinCC Advanced RT nicht erforderlich. Alternativ kann auch jedes andere Comfort Panel verwendet werden. Eine Migration (Gerätetausch) ist dann erforderlich.
CPU 1516-3PN/DP	1	6ES7516-3AN00-0AB0	Alternativ kann auch jede andere CPU aus der Familie S7-1200 oder S7-1500 verwendet werden.

Software-Komponenten

Tabelle 2-2

Komponente	Anz.	Artikelnummer	Hinweis
STEP 7 Professional V13 SP1	1	6ES7822-1...03-..	
WinCC Engineering V13 SP1	1	6AV210-....3-0	
WinCC Runtime Advanced V13 SP1	1	6AV2104-....3-0	Nur bei Verwendung einer WinCC Advanced RT erforderlich.

Beispieldateien und Projekte

Die folgende Liste enthält alle Dateien und Projekte, die in diesem Beispiel verwendet werden.

Tabelle 2-3

Komponente	Hinweis
109481693_CODE_v10.zip	Diese gepackte Datei enthält das STEP 7 und WinCC Projekt.
109481693_DOC_v10_de.pdf	Dieses Dokument.

3 Grundlagen

Bei der f(x)-Kurvenanzeige im TIA Portal gibt es eine Reihe von Besonderheiten, die zu beachten sind.

Die Anzeige der Datenpunkte eines Archivs in der f(x)-Kurvenanzeige erfolgt unter Verwendung eines Skriptes, um das richtige Format der Archivdaten zu gewährleisten.

Das Skript sorgt dafür, dass im Variablenarchiv

- zwei Arrays von Real-Werten (X-Wert und Y Wert) in der Steuerung als eine Reihe von Einzelwerten abgelegt werden.
- die einzelnen Wertepaare mit einem identischen Zeitstempel abgelegt werden.
- die Werte „auf Anforderung“ archiviert werden können.
- die Struktur der eines herkömmlichen Archivs entspricht.

In aller Regel wird für die Darstellung von Archivdaten in der f(x)-Kurvenanzeige ein CSV-Archiv verwendet, da dieses über Skript leicht beschrieben werden kann.

Wenn Sie statt einem Comfort Panel die WinCC Advanced RT einsetzen, haben Sie darüber hinaus die Möglichkeit, die Archivierung in eine SQL-Datenbank vorzunehmen. Im Beispielpjekt sind beide Lösungen implementiert.

Hinweis

Damit das Skript in das CSV-Archiv schreiben kann, müssen erst alle Archive geschlossen werden. Während diesem Zeitraum können keine weiteren Archivdaten abgelegt werden. Bei der Lösung mit der SQL-Datenbank kann direkt in die Datenbank geschrieben werden ohne andere Archive zu beeinflussen.

ACHTUNG

Drohender Datenverlust beim Beschreiben eines CSV-Archivs

Damit das Skript ein CSV-Archiv beschreiben kann, müssen für die Laufzeit alle Archive geschlossen werden. Während dieser Zeit müssen andere Archive gepuffert werden, die erst nach dem Öffnen der Archive nachgeschrieben werden.

Es können maximal 300 Werte gepuffert werden, bevor es zum Datenverlust kommt. Sollten Sie andere Archive in Ihrem Projekt verwenden, für die in der Zeit der Bearbeitung des Skriptes mehr als 300 Werte anfallen können, kommt es zu Datenverlust! Dies betrifft auch Meldearchive!

4 Funktionsweise

4.1 Gesamtübersicht

In diesem Beispiel werden Wertepaare, x und $f(x)$, einer mathematischen Funktion in der Steuerung (CPU 1516-3PN/DP) berechnet und in einen Datenbaustein geschrieben. Prinzipiell können jedoch auch andere Datenquellen visualisiert werden.

Das im Beispiel enthaltene Programm dient nur zum Erstellen beispielhafter Wertepaare von x und y . Über das HMI können drei mathematische Funktionen mit Parametern versorgt werden.

Durch Drücken der entsprechenden Schaltflächen im HMI (Exponential-, Parabel-, und Sinus-Funktion) werden die Funktionen im Funktionsbaustein „CreateDataPair“ berechnet und die Wertepaare im Instanz-Datenbaustein „CreateDataPair_DB“ abgelegt.

Die Länge des Datenbereiches (Array) gibt dabei die Anzahl der gezeigten Wertepaare vor. Soll eine andere Anzahl von Wertepaaren angezeigt werden, muss die Länge des Arrays angepasst werden:

- Länge des Arrays für x - und y -Werte („statX“ und „statY“) im Deklarationsteil des FB
- Anzahl der Schleifendurchläufe „#statIndex“ im Programmteil des FB
- Anzahl der Schleifendurchläufe „ObjLoopCount“ im Programmteil des VB-Skripts „TrendControl“
- Datensätze pro Archiv im Variablenarchiv (Anzahl = 2 x Anzahl der Elemente)
- $f(x)$ -Kurvenanzeige > X-Achse > Wertebereich
- $f(x)$ -Kurvenanzeige > Y-Achse > Wertebereich
- $f(x)$ -Kurvenanzeige > Kurve > Datenbereich > Messpunkte (Datum/Uhrzeit nicht verändern)

Hinweis

Beachten Sie, dass die Zählung innerhalb eines Arrays immer bei 0 beginnt. Bei z. B. 100 Werten hat das letzte Array-Element die Nummer 99 (n-1).

4.2 SPS-Programm

Das SPS-Programm besteht lediglich aus einem Funktionsbaustein „CreateDataPair“ und dem ihm zugewiesenen Instanz-Datenbaustein „CreateDataPair_DB“.

Der Funktionsbaustein berechnet die Wertepaare, x und $f(x)$ von bis zu drei mathematischen Funktionen und legt diese in zwei Arrays innerhalb des Instanz-Datenbausteins ab. Über das HMI können diese Funktionen mit Parametern versorgt und die Berechnung angestoßen werden.

Deklaration

Abbildung 4-1

CreateDataPair				
		Name	Data type	Default value
1		▼ Input		
2		E_Factor	Real	50.0
3		E_TimeConstant	Real	20.0
4		E_OffsetY	Real	0.0
5		P_Form	Real	-0.1
6		P_OffsetX	Real	50.0
7		P_OffsetY	Real	50.0
8		S_Length	Real	50.0
9		S_Amplitude	Real	25.0
10		S_OffsetX	Real	0.0
11		S_OffsetY	Real	25.0
12		► Output		
13		▼ InOut		
14		TriggerFunction	Int	0
15		▼ Static		
16		statIndex	Int	0
17		statExponent	Real	0.0
18		► statX	Array[0..99] of Real	
19		► statY	Array[0..99] of Real	

Tabelle 4-1

Name	Funktion
E_Factor	Parameter Exponential-Funktion: Faktor
E_TimeConstant	Parameter Exponential-Funktion: Zeitkonstante
E_OffsetY	Parameter Exponential-Funktion: Versatz y
P_Form	Parameter Parabel-Funktion: Form
P_OffsetX	Parameter Parabel-Funktion: Versatz x
P_OffsetY	Parameter Parabel-Funktion: Versatz y
S_Length	Parameter Sinus-Funktion: Wellenlänge
S_Amplitude	Parameter Sinus-Funktion: Amplitude
S_offsetX	Parameter Sinus-Funktion: Versatz x
S_offsetY	Parameter Sinus-Funktion: Versatz y
TriggerFunction	Auslöseereignis zum Berechnen der Funktion und der Ablage der Wertepaare in den Arrays
statIndex	Zeiger für die Array-Elemente
statExponent	Element der Exponential-Funktion
statX	Array der x-Werte
statY	Array der y-Werte

4.3 HMI-Programm

Da die f(x)-Kurvenanzeige immer einen Zeitstempel zum referenzieren benötigt, wird in diesem Beispiel ein fiktiver Zeitstempel verwendet. Dieser Zeitstempel ist jedoch für den Anwender zur Laufzeit nicht mehr sichtbar.

Zur Anzeige wird ein Variablenarchiv verwendet, welches auch im TIA-Portal projiziert ist. Das Archiv selbst wird dann durch ein Skript beschrieben.

4.3.1 TrendControl

Mit dem VB-Skript „TrendControl“ werden die Werte in ein Archiv im CSV-Format geschrieben. Der Aufbau der CSV-Datei darf nicht verändert werden, da sonst die Kurvenanzeige diese nicht mehr als Archiv erkennt und die Werte nicht mehr darstellen kann.

```
Function TrendControl()
Dim timestring, ObjFile, ObjFileSystem, arrx, array, ObjFilePath,
ObjFileName, ObjLoopCount, ObjIndex, ObjTime, Objx, Objy

1 Const varNametagX = "LogValX"
Const varNametagY = "LogValY"
Const splitsign = ";"
Const ForAppending = 8

2 'Get current datetime
timestring = "" & Now & ""

Set ObjFileSystem = CreateObject("Scripting.FileSystemObject")
Set arrx = SmartTags("Input_X")
Set array = SmartTags("Input_Y")

3 ObjLoopCount = UBound(arrx)
ObjFilePath = "C:\Logs\" 'File location for tag log
ObjFileName = "TrendControl_f(x)0.csv" 'Name of tag log

'Closing logs
CloseAllLogs

Do While SmartTags("LogsAreClosed") = False 'Wait until all logs are closed
Loop

'Deleting old file
On Error Resume Next
ObjFileSystem.DeleteFile ObjFilePath & ObjFileName, True

If Err.Number <> 0 Then
ShowSystemAlarm "Error: Deleting old file not possible. " & Err.Number & " " &
Err.Description
Err.Clear
OpenAllLogs
TrendControl = 1 'RetVal: 1= error deleting old file
Exit Function
End If

'Create new archive as csv file
Set ObjFile = ObjFileSystem.OpenTextFile(ObjFilePath & ObjFileName,
ForAppending, True)

If Err.Number <> 0 Then
ShowSystemAlarm "Error: Open file not possible. " & Err.Number & " " &
Err.Description
Err.Clear
OpenAllLogs
TrendControl = 2 'RetVal: 2= error creating new file
Exit Function
End If

'Write header into data log
ObjFile.WriteLine("VarName" & splitsign & "TimeString" & splitsign &
"VarValue" & splitsign & "Validity" & splitsign & "Time_ms")
```

4 Funktionsweise

4.3 HMI-Programm

```

If Err.Number <> 0 Then
    ShowSystemAlarm "Error: Can not write to data log. " & Err.Number & " " &
Err.Description
    Err.Clear
    OpenAllLogs
    TrendControl = 3 'RetVal: 3= error writing to new file
    Exit Function
End If

'Write entries into data log
For ObjIndex = 0 To ObjLoopCount
    'Write x element to log
    Objx = arrx(ObjIndex)
    ObjFile.WriteLine(varNametagX & splitsign & timestring & splitsign & Objx &
splitsign & "1" & splitsign & ObjIndex)

    'Write y element to log
    Objy = array(ObjIndex)
    ObjFile.WriteLine(varNametagY & splitsign & timestring & splitsign & Objy &
splitsign & "1" & splitsign & ObjIndex)
Next
ObjFile.Close
OpenAllLogs
Do While SmartTags("LogsAreClosed") = True 'Wait until all logs are opened
Loop

'End function
ShowSystemAlarm "Load new file"
Set ObjFile = Nothing
Set ObjFileSystem = Nothing
TrendControl = 0 'RetVal: 0= no error occurred
End Function

```

4

Tabelle 4-2

Punkt	Skriptauszug	Beschreibung
1.	<pre> Const varNametagX = "LogValX" Const varNametagY = "LogValY" Const splitsign = "," </pre>	Die Variable "LogValX" und „LogValY“ wird als Archivvariable verwendet. Passen Sie Ihr Archiv oder das Skript entsprechend an. Das Trennzeichen „," wird im Archiv (CSV-Datei) verwendet und ist abhängig von den Ländereinstellungen des Betriebssystems.
2.	<pre> timestring = "" & Now & "" </pre>	Hier wird der aktuelle Datum/Uhrzeit-Stempel für die Archiveinträge gebildet. Passen Sie diesen bei Bedarf an.
3.	<pre> ObjFilePath = "C:\Logs\" 'File location for tag log ObjFileName = "TrendControl_f(x)0.csv" 'Name of tag log </pre>	Hier wird der Archivpfad und der Archivname bestimmt. Passen Sie Ihr Archiv oder das Skript entsprechend an.
4.	<pre> ShowSystemAlarm "Load new file" </pre>	Hier wird über die Meldeanzeige im Projekt der Hinweis ausgegeben, dass das Skript eine neue f(x)-Kurve gezeichnet hat. Passen Sie die Sprache und den Text bei Bedarf an.

4.3.2 TrendControlSQL

Mit dem VB-Skript „TrendControlSQL“ werden die Werte in ein Archiv einer SQL-Datenbank geschrieben. Der Aufbau der SQL-Datenbank darf nicht verändert werden, da sonst die Kurvenanzeige diese nicht mehr als Archiv erkennt und die Werte nicht mehr darstellen kann.

Hinweis

Zur Nutzung der SQL-Datenbank muss diese vorher im Betriebssystem angelegt werden. Weitere Hinweise dazu entnehmen Sie dem Kapitel „[Inbetriebnahme](#)“.

```

Function TrendControlSQL()
On Error Resume Next

'Define constants for the script, like logfilename, tagname,...
1 Const ObjLoopCount = 99
Const varNametagX = "LogValXSQL"
Const varNametagY = "LogValYSQL"
Const validity = 1
Const splitsign = ","

'tablename of logs in master database
Const tableX = "[master].[dbo].[LogXSQL0]"
Const tableY = "[master].[dbo].[LogYSQL0]"
Dim arrX, arrY, fso, conn, rst, count, timeString, sql, TimeMS

2 'get current datetime
timeString = "'" & Now & "'"

'clear all dataset
ClearLog hmiDataLog, "LogXSQL"
ClearLog hmiDataLog, "LogYSQL"
3 For count = 0 To 100000
Next

'set an instance of database
Set conn = CreateObject("ADODB.Connection")
Set rst = CreateObject("ADODB.Recordset")

4 'open the connection to database with DataSourceName "LogSQL"
conn.Open "Provider=MSDASQL;DSN=LogSQL"

'copy array to internal tag
Set arrX = SmartTags("Input_X")
Set arrY = SmartTags("Input_Y")

'write the arrays into database
For count = 0 To ObjLoopCount
    TimeMS = TimeMS + 1
    sql = "INSERT INTO " & tableX &
    "([VarName],[TimeString],[VarValue],[Validity],[Time_ms]) VALUES("_
    & varNametagX & splitsign & timeString & splitsign &
    Replace(arrX(count), ",", ".") & splitsign & validity & splitsign &
    TimeMS & ")"
    Set rst = conn.Execute(sql)

    sql = "INSERT INTO " & tableY &
    "([VarName],[TimeString],[VarValue],[Validity],[Time_ms]) VALUES("_
    & varNametagY & splitsign & timeString & splitsign &
    Replace(arrY(count), ",", ".") & splitsign & validity & splitsign &
    TimeMS & ")"
    Set rst = conn.Execute(sql)
Next

'close connection and return TimeMS value for next run
conn.Close

```

5

```

Set rst = Nothing
Set conn = Nothing

If Err.Number <> 0 Then
    ShowSystemAlarm "Problem writing to SQL. Maybe the logs are
closed?"
    Err.Clear
    TrendControlSQL = 1
Else
    TrendControlSQL = 0
End If
End Function

```

Tabelle 4-3

Punkt	Skriptauszug	Beschreibung
1.	<pre> Const ObjLoopCount = 99 Const varNametagX = "LogValXSQL" Const varNametagY = "LogValYSQL" Const splitsign = "," </pre>	Die Konstante „ObjLoopCount“ wird von der Größe der Arrays bestimmt und muss entsprechend angepasst werden. Die Variable "LogValXSQL" und „LogValYSQL“ wird als Archivvariable verwendet. Passen Sie Ihr Archiv oder das Skript entsprechend an. Das Trennzeichen „," wird im Archiv (SQL-Datenbank) verwendet und ist abhängig von den Ländereinstellungen des Betriebssystems.
2.	<pre> timestring = "" & Now & "" </pre>	Hier wird der aktuelle Datum/Uhrzeit-Stempel für die Archiveinträge gebildet. Passen Sie diesen bei Bedarf an.
3.	<pre> For count = 0 To 100000 Next </pre>	Die Schleife dient als Pause, bis das Löschen der alten Datenbankeinträge abgeschlossen ist. Im Prozess kann das Löschen meist zwischen zwei Messungen bewerkstelligt werden, so dass diese Schleife nicht notwendig ist.
4.	<pre> conn.Open "Provider=MSDASQL;DSN=LogSQL" </pre>	Hier wird der Datenbankname bestimmt. Passen Sie Ihren Datenbanknamen oder das Skript entsprechend an.
5.	<pre> ShowSystemAlarm "Problem writing to SQL. Maybe the logs are closed?" </pre>	Hier wird über die Meldeanzeige im Projekt der Hinweis ausgegeben, dass während der Laufzeit ein Fehler aufgetreten ist. Passen Sie die Sprache und den Text bei Bedarf an.

4.3.3 Variablenhaushalt

Abbildung 4-2

TrendControl_f(x)				
	Name ▲	Data type	Connection	PLC name
	CreateArchiveRetVal	Int	<Internal tag>	
	E_Factor	Real	HMI_Connection_2	PLC_1
	E_OffsetY	Real	HMI_Connection_2	PLC_1
	E_TimeConstant	Real	HMI_Connection_2	PLC_1
	Input_X	Array [0..99] of Real	HMI_Connection_2	PLC_1
	Input_Y	Array [0..99] of Real	HMI_Connection_2	PLC_1
	LogsAreClosed	Bool	<Internal tag>	
	LogValX	Real	<Internal tag>	
	LogValXSQL	Real	<Internal tag>	
	LogValY	Real	<Internal tag>	
	LogValYSQL	Real	<Internal tag>	
	P_Form	Real	HMI_Connection_2	PLC_1
	P_OffsetX	Real	HMI_Connection_2	PLC_1
	P_OffsetY	Real	HMI_Connection_2	PLC_1
	S_Amplitude	Real	HMI_Connection_2	PLC_1
	S_Length	Real	HMI_Connection_2	PLC_1
	S_OffsetX	Real	HMI_Connection_2	PLC_1
	S_OffsetY	Real	HMI_Connection_2	PLC_1
	TimeMSVal	UDInt	<Internal tag>	
	TriggerFunction	Int	HMI_Connection_2	PLC_1

Tabelle 4-4

Name	Funktion
CreateArchiveRetVal	Rückgabewert der VBS-Funktion
E_Factor	Parameter Exponential-Funktion: Faktor
E_OffsetY	Parameter Exponential-Funktion: Versatz y
E_TimeConstant	Parameter Exponential-Funktion: Zeitkonstante
Input_X	Array der x-Werte
Input_Y	Array der y-Werte
LogsAreClosed	Ereignis Archive geschlossen Achtung Diese Variable benötigt die Systemmeldungen des HMI
LogValX	Archivvariable für den x-Wert (nur bei CSV-Archiv)
LogValXSQL	Archivvariable für den x-Wert (nur bei SQL-Datenbank)
LogValY	Archivvariable für den y-Wert (nur bei CSV-Archiv)
LogValYSQL	Archivvariable für den y-Wert (nur bei SQL-Datenbank)
P_Form	Parameter Parabel-Funktion: Form
P_OffsetX	Parameter Parabel-Funktion: Versatz x
P_OffsetY	Parameter Parabel-Funktion: Versatz y
S_Amplitude	Parameter Sinus-Funktion: Amplitude
S_Length	Parameter Sinus-Funktion: Wellenlänge
S_offsetX	Parameter Sinus-Funktion: Versatz x
S_offsetY	Parameter Sinus-Funktion: Versatz y
TriggerFunction	Auslöseereignis zum Berechnen der Funktion und der Ablage der Wertepaare in den Arrays

5 Installation und Inbetriebnahme

Nachfolgend finden Sie die erforderlichen Schritte für die Installation und Inbetriebnahme des Anwendungsbeispiels.

5.1 Installation der Hardware

Für den Hardwareaufbau wurden die im Kapitel „[Hard- und Software-Komponenten](#)“ aufgeführten Baugruppen gewählt. Dabei muss lediglich ein TP1200 Comfort mit einer CPU 1516-3PN/DP über Ethernet verbunden werden.

Sollten Sie die Baugruppen nicht vorrätig haben, können Sie das Anwendungsbeispiel auch simulieren. Weitere Hinweise dazu finden Sie im Kapitel „[Inbetriebnahme](#)“.

5.2 Installation der Software (Download)

Tabelle 5-1

Schritt	Beschreibung
1.	Entpacken Sie die Datei „109481693_CODE_v10.zip“ in ein Verzeichnis Ihrer Wahl.
2.	Öffnen Sie das Projekt mit dem TIA Portal V13 SP1, oder höher.
3.	Passen Sie die projektierte Hardware unter „Geräte & Netze“ Ihren Anforderungen an.
4.	Öffnen Sie das STEP 7-Projekt und generieren Sie es neu.
5.	Laden Sie die Bausteine in Ihre Steuerung und starten Sie diese.
6.	Öffnen Sie das WinCC-Projekt und generieren Sie es neu.
7.	Laden Sie das WinCC-Projekt auf Ihr Comfort-Panel oder starten Sie die Runtime.

5.3 Inbetriebnahme

Simulation der Baugruppen

Sollten Sie das Anwendungsbeispiel ohne Hardware testen wollen, so verwenden Sie statt der vorgesehenen Baugruppen folgende Konfiguration:

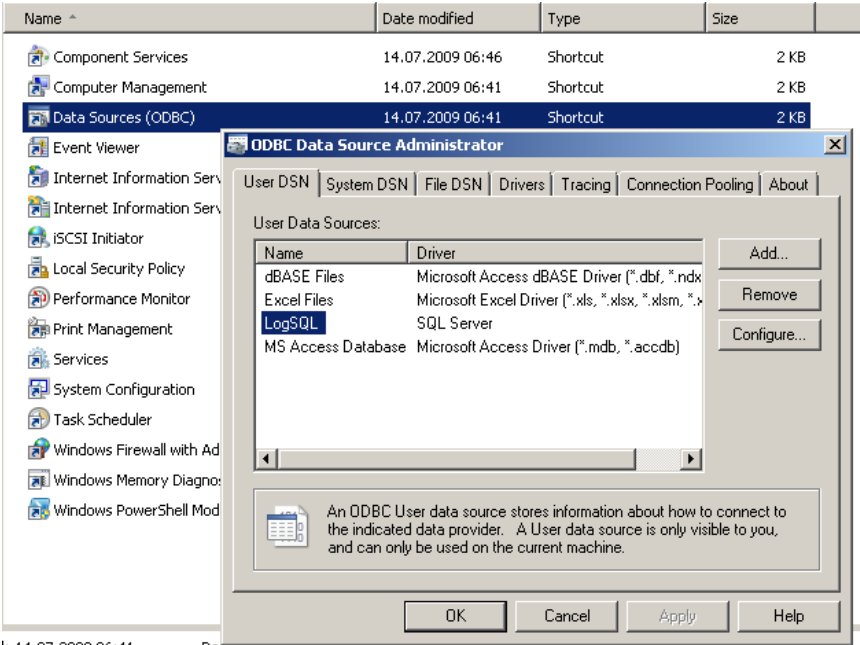
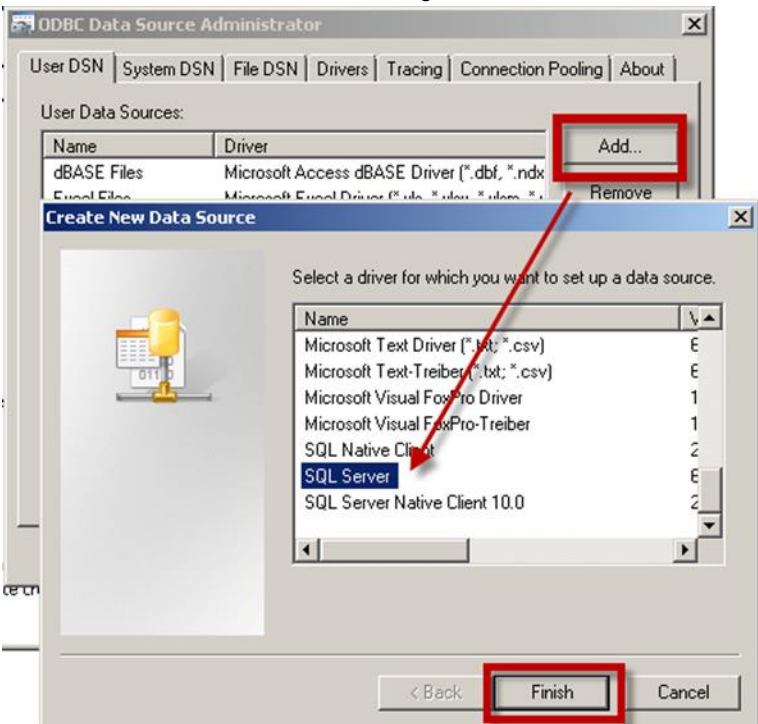
Tabelle 5-2

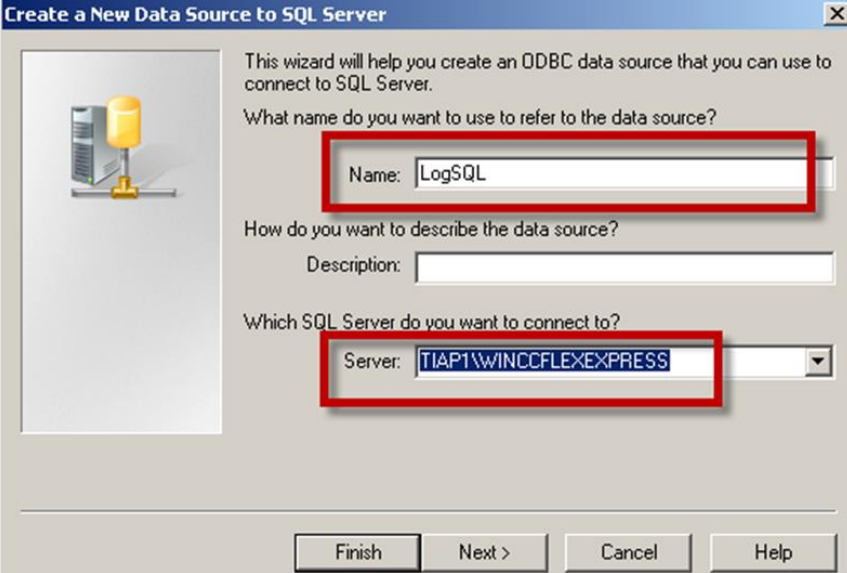
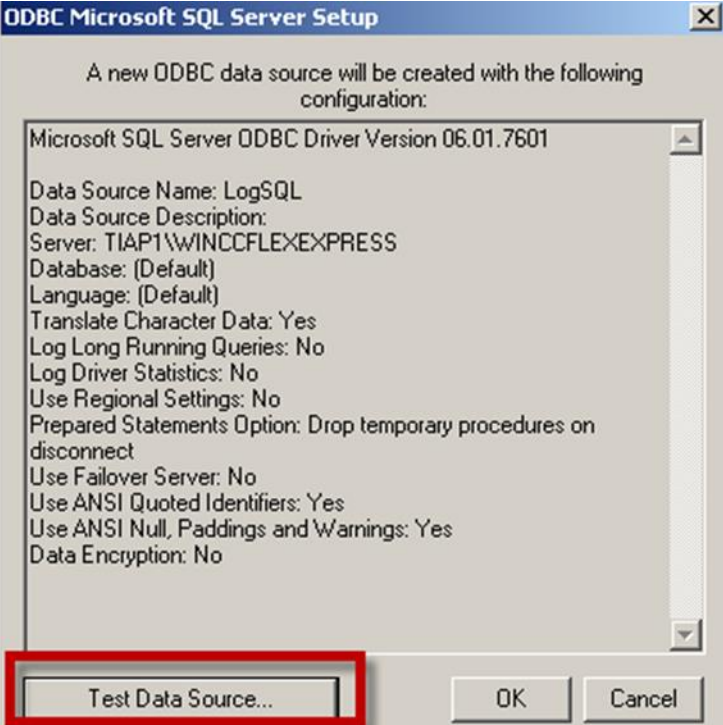
Schritt	Beschreibung
1.	Ändern Sie in der PG/PC-Schnittstelle den Zugangspunkt der Applikation auf „S7ONLINE (STEP7) -> PLCSIM S7-1200/S7-1500“.
2.	Starten Sie S7-PLCSIM und laden Sie dort die SPS-Bausteine mit "Laden in Gerät".
3.	Öffnen Sie die im Projekt enthaltene SIMATIC PC Station über "Simulation starten".

Archivierung mit SQL-Datenbank

Falls Sie die im Projekt enthaltene SIMATIC PC Station verwenden und dort die Archivierung in eine SQL-Datenbank wählen, muss hier noch zusätzlich im Betriebssystem die ODBC-Datenquelle angelegt werden:

Tabelle 5-3

Schritt	Beschreibung
1.	- Öffnen Sie in der Systemsteuerung die Kategorie „Verwaltung“. - Öffnen Sie den Dienst „Datenquellen (ODBC)“. - Wählen Sie das Register „Benutzer DSN“. 
2.	- Drücken Sie die Schaltfläche „Hinzufügen...“. - Wählen Sie den Treiber „SQL Server“ - Drücken Sie die Schaltfläche „Fertig stellen“. 

Schritt	Beschreibung
3.	- Wählen Sie als Datenquellennamen „LogSQL“. - Wählen Sie als SQL-Server „<Rechnername>WINCCFLEXEXPRESS“. 
4.	- Drücken Sie die Schaltfläche „Weiter“. - Übernehmen Sie in den folgenden Dialogen alle Voreinstellungen. - Im Zusammenfassungsdialog am Ende der Konfiguration können Sie die Verbindung testen. 
5.	Nach der Bestätigung über die Schaltfläche „OK“ ist die Datenquelle konfiguriert.

6 Bedienung der Applikation

Hier erfahren Sie, wie Sie das Beispielprojekt bedienen können.

- Voraussetzung zum Bedienen der Applikation ist das geöffnete bzw. geladene Projekt.
- Die folgende Beschreibung in diesem Kapitel bezieht sich auf WinCC RT Advanced, die Bedienung für ein Comfort Panel ist aber identisch.
- Unterschieden wird die Möglichkeit, als Archiv statt einer CSV-Datei eine SQL-Datenbank zu nutzen. Diese Option steht lediglich für WinCC RT Advanced zur Verfügung.

6.1 Startbildschirm

Abbildung 6-1



Tabelle 6-1

Position	Schaltfläche
1.	Start Der Startbildschirm des Anwendungsbeispiels.
2.	f(x)-Kurvenanzeige – CSV-Datei Hier verzweigen Sie in das Anwendungsbeispiel, welches zur Datenablage eine CSV-Datei verwendet (für Comfort Panel und WinCC RT Advanced).
3.	f(x)-Kurvenanzeige – SQL Datenbank Hier verzweigen Sie in das Anwendungsbeispiel, welches zur Datenablage eine SQL-Datenbank verwendet (nur für WinCC RT Advanced).
4.	System Hier können Sie Systemfunktionen anwählen, z. B. RT beenden.
5.	Support Hier erfahren Sie mehr über den Industry Online Support.
6.	Navigationsleiste auf- und zuklappen Die Navigationsleiste ist in jedem Bild erreichbar und befindet sich immer am rechten Bildschirmrand.

Hinweis Aus Gründen der Übersichtlichkeit wird nachfolgend lediglich auf die Positionen 2.) „f(x)-Kurvenanzeige – CSV-Datei“ und 3.) „f(x)-Kurvenanzeige – SQL Datenbank“ eingegangen.

6.2 f(x)-Kurvenanzeige – CSV-Datei

Abbildung 6-2

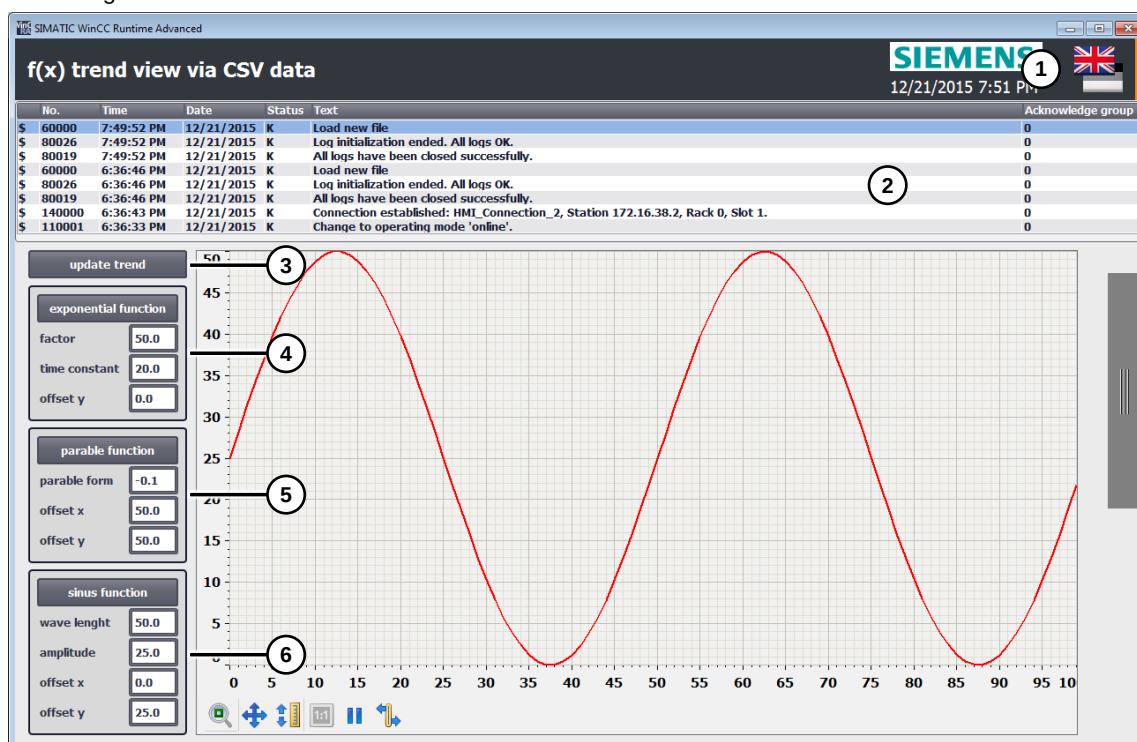


Tabelle 6-2

Position	Schaltfläche
1.	Sprachumschaltung Hier können Sie die Sprache wechseln.
2.	Meldeanzeige Hier können Sie alle aktuellen Meldungen einsehen. Im Fehlerfall wird eine entsprechende Meldung ausgegeben.
3.	Aktualisiere Kurve - Die Datenpunkte des Instanz-DB in der Steuerung werden mit einem Skript in die CSV-Datei (Archiv) geschrieben. - Das Bild wird neu angewählt. Dabei werden die Datenpunkte aus dem Archiv in die Kurve geschrieben und zur Anzeige gebracht.
4.	Exponential Funktion - Geben Sie hier die Parameter für die Exponentialfunktion vor: - Faktor - Zeitkonstante - Versatz y - Schreiben Sie diese Parameter mit der Schaltfläche „Exponential Funktion“ in die Steuerung. - In der Steuerung werden dann die einzelnen Datenpunkte für diese Funktion mit den angegebenen Parametern im Instanz-DB abgelegt.

Position	Schaltfläche
5.	Parabel Funktion - Geben Sie hier die Parameter für die Parabelfunktion vor: - Form - Versatz x - Versatz y - Schreiben Sie diese Parameter mit der Schaltfläche „Parabel Funktion“ in die Steuerung. - In der Steuerung werden dann die einzelnen Datenpunkte für diese Funktion mit den angegebenen Parametern im Instanz-DB abgelegt.
6.	Sinus Funktion - Geben Sie hier die Parameter für die Sinusfunktion vor: - Wellenlänge - Amplitude - Versatz x - Versatz y - Schreiben Sie diese Parameter mit der Schaltfläche „Sinus Funktion“ in die Steuerung. - In der Steuerung werden dann die einzelnen Datenpunkte für diese Funktion mit den angegebenen Parametern im Instanz-DB abgelegt.

6.3 f(x)-Kurvenanzeige – SQL Datenbank

Die Anzeige und die Bedienung in diesem Bild entsprechen dem des Bildes „f(x)-Kurvenanzeige – CSV-Datei“. Dabei gibt es folgende Unterschiede:

- Der Schaltfläche „Aktualisiere Kurve“ ist ein anderes Skript zugewiesen.
- Die Datenpunkte des Instanz-DB in der Steuerung werden mit diesem Skript in eine SQL-Datenbank (Archiv) geschrieben.
- Der Kurvenanzeige ist als Archiv statt einer CSV-Datei eine SQL-Datenbank als Datenquelle zugewiesen.

Hinweis

- Dieses Bild ist nur für WinCC RT Advanced verfügbar.
- Beachten Sie, dass für das in diesem Bild verwendete Skript eine entsprechende SQL-Datenbank angelegt sein muss.
- Das Anlegen der SQL-Datenbank wird im Kapitel „[Funktionsweise](#)“ beschrieben.

7 Literaturhinweise

Tabelle 7-1

	Thema
\1\	Siemens Industry Online Support https://support.industry.siemens.com
\2\	Downloadseite des Beitrages https://support.industry.siemens.com/cs/ww/de/view/109481693
\3\	Zugriff auf eine SQL-Datenbank über ein Skript https://support.industry.siemens.com/cs/ww/de/view/26283062

8 Historie

Tabelle 8-1

Version	Datum	Änderung
V1.0	05/2016	Erste Ausgabe