Application Note

Description: This document describes how to configure GENESIS64[™] to connect to an Azure IoT Hub.

General Requirement: MS Azure Portal account, Basic SQL Azure/IoT knowledge.

Introduction

This application note describes the steps necessary to connect on-premise GENESIS64 and an Azure cloud-based deployment platform. Adding IoT hub connectivity to an on-premise GENESIS64 system provides multiple benefits. The cloud system is able to aggregate data from multiple GENESIS64 onpremise systems. It enables visualization anywhere via the cloud to existing GENESIS64 systems. And you are able to add powerful analytics capabilities by connecting data to the cloud. The cloud communication path is highly secure and is based on Advanced Message Queuing Protocol (AMQP) and a publish/subscribe ("pub/sub") mechanism. Optionally integrate Microsoft Azure services such as Power BI and Machine Learning to provide greater depth of analyses.

Prerequisites and Use

Azure IoT Hub and IoT Device Setup

Create loT Hub in 1 an Azure Portal. Copy the iothubowner "Connection Stringprimary key". This will be used later in the Subscriber Configuration section.

P Search (Ctrl+/)	POUCY	PERMISSIONS	Access policy name
X Overview	iothubowner	registry write, service connect, device connect	Permissions
Activity log	service	service connect	Registry read
Access control (IAM)	device	device connect	Service connect @
SETTINGS	registryRead	registry read.	A news remease
E Properties	registryReadWrite	registry write	Shared access keys
🔒 Looks			Tenary tey 🖬
Automation script			Secondary key 0
GENERAL			N.
Shared access policies			Connection string-printing sey 0
Messaging			Connection string—secondary key 0
File upload			

Figure 1 - IoT Hub in Azure Portal

- Select your IoT Hub in the Azure Portal. Then click on IoT Device (under Device Management).
- 3. Create a new device in the Azure Portal. Assign a name to the device and use automatic generation of the keys.
- © 2018 ICONICS, Inc.



Home > - IoT Devices > Device Details	
A Device Details	* ×
🕞 Save 🛗 Device Twin 🖾 Message To Device 📣 Direct Method	
Device Id. 8	
MyOnPremDevice1	5
Primary key 🖲	_
	6
Secondary key 🖲	_
	l Ch
Connection strino—orimany key 🔍	_
HostName: azure-devices.net;DeviceId=MyOnPremDevice1;SharedAccessKey=	ΨC.
Connection string—secondary key 0	
HostName= azure-devices.net;DeviceId=MyOnPremDevice1;SharedAccessKey=	ß
Connect device to IoT Hub U	
Enable Utsable	



Local PC and Remote PC Setup

- 5. Install ICONICS Suite on the local PC.
- Install ICONICS Suite on a remote PC. This PC does not need 6 to be on the same network as the local PC, though ICONICS does recommend installing on an Azure VM within the same Azure resource group as the IoT Hub.

Publisher Configuration

- 1. On the local PC, open Workbench. Expand Internet of Thinas.
- 2. Create a new Real Time Publish List. Right-click on Publish Lists and select Add Publish List (Real Time). Name the publish list.
- In the **Published Points** tab, add points to be exposed over 3. the cloud by adding items to the list and using the data browser to navigate to the appropriate points. Give each point a short "Publish Name", as this reduces messages, minimizing Azure costs. Click Apply to save changes.

P P	L1 😤 🗙									
Publish List Name: PL1										
iene	ral Published Points									
Spe	cify the points that will be exposed by the publisher (<u>Clici</u>	c he	re to add multiple	tag	<u>a)</u>					-
	Point Name	Ţ	Publish Name	T	Send Timesta	T	Writable	T	Publish Group	Ŧ
÷	Click here to add new item									
	snmp:PC\;Heartbeat_default		heartbeat		~				None	
×	@sim64:Float.Random(Period[sec],Min,Max,Phase[deg]).Value		random		\checkmark				None	

Figure 4 - Published Points

Right-click Publisher Connections and create a new 4. Publisher Connection. Provide a valid name and fill in the Device Connection String. All other options may remain the same. For the Device Connection String, use the



Application Note

May 2018

"Connection String—primary key" from your IoT Device in the Azure Portal, as mentioned previously in this document.

NOTE: Be sure to use the *Device* connection string, not the *iothubowner* connection string

Also fill in the Real Time Publish list that you just created. Here you can also change the encoder used to send data to the Azure IoT Hub.

- 5. Apply your changes.
- 6. Select Internet of Things in the Project Explorer in Workbench. Click the Publisher traffic light to start the IoT Publisher Service.

NOTE: Ensure the connection is enabled by checking the **The connection is enabled** box.

Publisher Connection Name:	PublisherConnection	
General Settings		
The connection is enable	ed	
 Enable compatibility with 	h Mitsubishi Electric clients	
Connection Type:	Azure IoT Hub	Ŧ
Encoder:	Binary	• 🛛 🕻
Heartheat Rate:	20° (seconds $0 = no timeout)$	
Publish Lists		-
Publish Lists Real Time Publish List:	PL1	-
Publish Lists Real Time Publish List: Historical Publish List:	PL1 None	• 🗘 C •
Publish Lists Real Time Publish List: Historical Publish List: Analyzer Publish List:	PL1 None None	- 2 C +
Publish Lists Real Time Publish List: Historical Publish List: Analyzer Publish List:	PL1 None None	• 0 [> + • 0 [> + • 0 [> +
Publish Lists Real Time Publish List: Historical Publish List: Analyzer Publish List: IoT Hub Settings	PL1 None None	- 0 D + - 0 D + - 0 D +
Publish Lists Real Time Publish List: Historical Publish List: Analyzer Publish List: IoT Hub Settings Connection String:	PL1 None None HostName tostName	
Publish Lists Real Time Publish List: Historical Publish List: Analyzer Publish List: IoT Hub Settlings Connection String: Protocol:	PL1 None None None HostName gzure-devices.net.DeviceId=MyOnPremDevice1;SharedAccessKey= Automatic	



Subscriber Configuration

- 1. On the remote PC with the ICONICS Suite installed, open Workbench and expand Internet of Things. Right-click on **Subscriber Connections** and add a new Subscriber Connection.
- 2. Fill out the required fields. Use the same IoT Hub Connection String from the first step of the prerequisite process. Provide a connection name and click **Apply**.

ubscriber Connection Name: Sul	bscriberConnection		
General Settings			
The connection is enabled			
 Enable compatibility with Mits 	ubishi Electric clients		
Connection Type:	Azure IoT Hub		
Early Start:	0 ‡	(minutes)	
Default Decoder:	Binary		• Ø [
Dynamic Subscription Life Time:	5 🗘	(minutes)	
Keep Alive Timeout:	60 🗘	(seconds, 0 = no timeout)	
Io1 Hub Settings			
Connection String:	HostName:	azure-devices.net;SharedAccessKeyName=iothubowner;SharedAccessKey=••••	

Figure 6 – Subscriber Connection Setup

3. Now open GraphWorX64. In the Data Browser on the left, expand **My Computer** and select **Internet of Things**. The IoT Subscription you created appears. If you expand this further, the points you entered in the Published Points tab become visible.

Data Browser 7 ×
Search: 🔹 🔎 🔁
 Published Data via Static List Image: Transform Image: Transform
★ Favorites
▲ 💭 My Computer
Assets
▲ K Internet of Things
▲ 📑 SubscriberConnection
▲ 💻 MyOnPremDevice1
All Available Data
📋 Published Data via Dynamic List
Published Data via Static List
▶ [i0i] heartbeat
▶ 🔛 random
Statistics

Figure 7 - Browsing IoT Subscription

- 4. Open the Symbol Library in the left-hand pane by clicking **Symbols** and search for "KPI Gauge". Drag KPI Gauge 1 into the canvas and set the gauge's DataSource to your IoT tag using the Data Browser. Enter Runtime to activate your tag.
- 5. You can also see the "All Available Data" folder within the Date Browser. This folder exposes the entire address space and is referred to as the "Dynamic Publish List". If you go back to the Publish List we created in Workbench, you can use tags from this dynamic publish list to add to the list we defined earlier (known as the "Static Publish List").