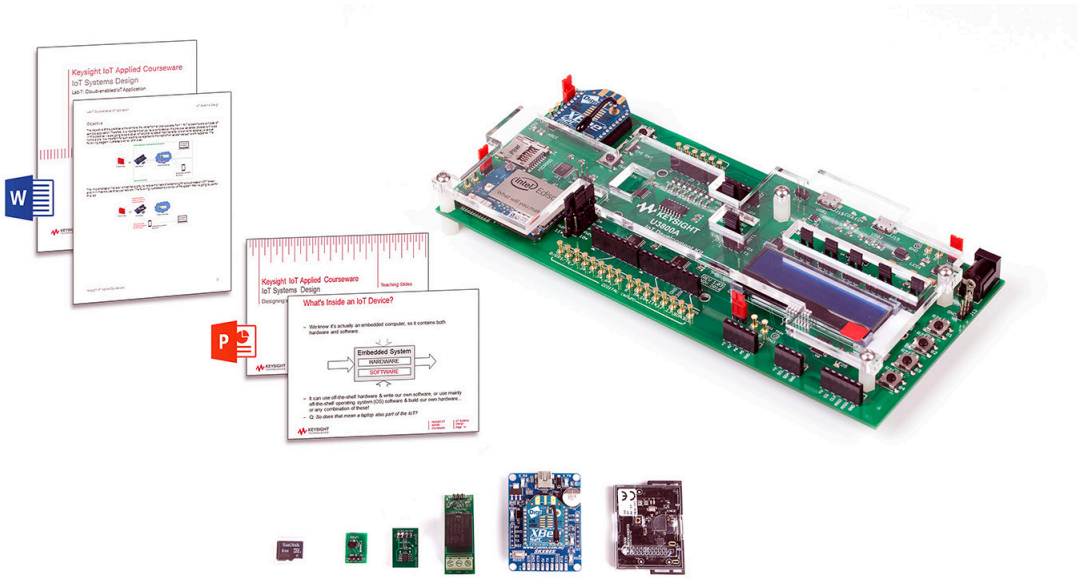
Keysight Technologies

U3803A/04A IoT Systems Design Applied Courseware

Data Sheet



KEYSIGHT TECHNOLOGIES

Unlocking Measurement Insights

Introduction

The Internet of Things (IoT) is the next mega trend that will change the way we live and work, and it is predicted to touch almost every consumer and industrial application. The core technologies that enable the IoT are wireless communication and sensor developments, and ongoing advances in these technologies result in unique challenges. These challenges include new communications standards, increased sensor integrations and power consumption management. This puts a heavy stress on an IoT device's design and validation cycle, and designers must constantly innovate to quickly and successfully develop and deploy IoT devices in the market.

The next generation of engineers will play a key role in the development of the IoT, and it is important that students graduate from an undergraduate engineering program prepared for the electronic design, test and measurement challenges needed to be solved in the industry. To fulfill that goal, education institutes must not only teach students the basics of testing and designing an IoT system, they must provide students with an understanding of the entire IoT ecosystem and also relate these experiences to real-world applications.

With more than 75 years of test and measurement expertise, Keysight Technologies, Inc. is ready to help you to nurture the next generation of IoT-ready professionals. Keysight's ready-to-teach U3800 Series IoT applied courseware focuses on teaching practical design and test techniques and is designed to give students the opportunity to work with industry-grade test and measurement instruments in the lab – the same instruments that they would use when they are out in the industry.

There are four IoT applied courseware:

1. IoT Fundamentals – Introduces the fundamentals of IoT. Students who complete this course will have an overview of the IoT's architecture, technologies and ecosystem.
2. IoT Systems Design – Introduces IoT system design techniques, leveraging embedded systems focusing on specific IoT examples. Students will be able to design and develop an IoT system targeting IoT gateway and sensor network.
3. IoT Wireless Communications – Allows students to develop typical IoT applications with various types of wireless connectivity. Students will be able to perform quick verification and design validation on these IoT applications. This courseware will be available in Fall 2017.
4. IoT Sensors and Power Management – Teaches students how to characterize the power consumption of the IoT device's on-board controller, sensors and wireless modules. Students will understand the principles of power management and will be able to characterize micro electro-mechanical systems (MEMS) devices. This courseware will be available in Fall 2017.

Each courseware comes with a training kit and teaching slides. The training kit consists of a development kit, IoT sensor device, XBee ZigBee® kit, lab sheets, and problem-based assignments. Students can also use this kit to develop their own projects once they have completed the course.

Courseware Overview

The U3803A/04A IoT Systems Design applied courseware is a ready-to-teach package on the subject of the IoT, with the goal of providing students the ability to develop and embedded system with IoT capabilities. This courseware is designed as a resource for lecturers, and consists of teaching slides and a training kit.

- Targeted university subject: IoT systems, embedded systems
- Targeted year of study: Second to final year undergraduates
- Prerequisites(s): Basic programming

Teaching slides	Training kit
Editable Microsoft PowerPoint slides	IoT development kit
Covers 36+ hours of classroom sessions	IoT sensor device
	XBee ZigBee kit
	Lab sheets (Microsoft Word) and model answers
	Problem-based learning assignments
	Covers 18 hours of lab sessions

Learning outcomes

Students will be able to:

- Design an embedded IoT gateway and IoT devices
- Configure IoT end-to-end systems from IoT devices to the cloud
- Create the operations of various I/O devices
- Set up wireless local area network (WLAN) 802.11, Bluetooth LE and ZigBee wireless connectivity.
- Apply industry standard software tools in IoT development
- Evaluate I/O signals and troubleshoot IoT systems using industry-grade test and measurement instruments

Key features and benefits

- The IoT Systems Design courseware is designed for a full semester of teaching, and comes with teaching slides and a training kit. Educators can use this complete solution to accelerate the setup of a new IoT-focused course.
- The courseware integrates hands-on industry-relevant experiences and real-world applications in IoT design and testing. Students learn the design and test techniques practiced by the industry, and have the opportunity to work with industry-grade test and measurement instruments and software.
- The courseware material will be updated yearly for three years at no additional cost. This allows educators and students to keep pace with evolving IoT trends and technologies, which will change rapidly over the next several years.
- The IoT development kit is based on a carrier board with Arduino UNO form factor interface and an add-on ZigBee module. Various external sensor devices can be connected to the board, allowing students to implement different types of IoT applications.
- The development kit allows students to experiment with wireless local area network (WLAN) 802.11, *Bluetooth*® Low Energy (LE) and ZigBee wireless connectivity. Due to the kit's modular design, it can be easily expanded to include other wireless connectivity and sensors.
- The development kit provides various test points for troubleshooting, current-drain consumption measurement on sub-circuits, and sensor verification.
- The hardware building blocks are visible on the development board.
- The development kit is equipped with an Intel Edison compute module that runs on Yocto Linux and is compatible with Intel System Studio IoT Edition, which is an Eclipse-based integrated development environment (IDE). This allows students to compile C/C++ files or to run Python scripts.

Courseware Contents

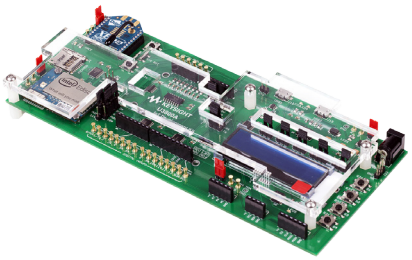
Teaching slides

The teaching slides are editable and cover 36+ hours of teaching for one full semester. The slides cover the following topics:

Essential elements of IoT systems	Introduction to an IoT-enabled embedded system, IoT building blocks, the past, present and future of IoT systems, and how IoT devices work.
Enabling technologies for IoT systems	Introduction to low-power embedded systems, Intel Atom and ARM-based CPUs, HDD and SDD, boot process, BIOS, GPU co-processors, and the challenges involved with IoT systems design.
Fundamentals of embedded systems for IoT	Introduction to embedded systems for IoT, including programming models and languages, shell programming, embedded operating systems and RTOS.
Connectivity for IoT	Introduction to various key wired and wireless technologies used in the implementation of IoT systems.
Designing IoT applications using embedded systems	Introduction to what a toolchain is, and how to compile and test Linux programs, communicate between programs, and multitask inside a program.
Introduction to cloud computing	Introduction to Internetworking, cloud computing and web services, and security and identity management
Case studies	Case studies covering smart automobile and disaster management applications.

Training kit

IoT development kit



This hardware kit is a customizable embedded system development kit that can be configured as a gateway or a sensor device. It incorporates an Intel Edison compute module that is designed for expert makers, entrepreneurs, and industrial IoT applications.

The system runs on Yocto Linux with open source software development compatible with Eclipse (C, C++, Python). Samples of start projects are also available to enhance the learning process and allow a wide range of potential applications.

All IoT applied courseware use the same development kit.

The development kit comes with the following features:

- Open source software development environment
- High performance, dual-core CPU and single core micro-controller support complex data collection in a low power package
- Integrated WLAN 802.11, *Bluetooth* LE and ZigBee wireless connectivity support
- 1 GB DDR and 4 GB flash memory, simplifying configuration and increasing scalability
- Arduino UNO and XBee form factor interfaces support
- UARTs, I^2C, SPI, 40 GPIO, SD card connector and LCD
- Micro USB (UART), micro USB OTG
- Flexible power supply options: AC power adapter or USB host
- Various test points for verification
- Sensor connectors for both analog and digital sensor signals

IoT sensor device

The TI SensorTag kit includes ten low-power sensors: ambient light, digital microphone, magnetic sensor, humidity, pressure, accelerometer, gyroscope, magnetometer, object temperature, and ambient temperature.

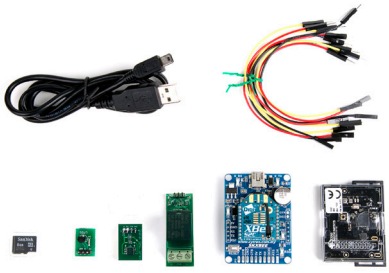
XBee ZigBee Kit

The XBee ZigBee starter kit is a compact platform that provides UART serial communication to an XBee ZigBee module. 5 V TTL logic interface offers a straightforward interface to microcontroller for embedded wireless development.

Accessories

The following accessories are included with the hardware kit:

Item	Quantity
Micro USB cable, 1 m	2
Mini USB cable, 1.2 m	1
TI SensorTag kit	1
XBee ZigBee kit	1
Analog temperature sensor	1
Digital temperature sensor	1
Relay actuator	1
Micro SD card	1



Lab sheets

The IoT Systems Design courseware includes seven editable lab sheets. Each lab requires 2-3 hours to complete. Model answers are provided with all lab sheets. The required instruments for this courseware are a digital multimeter (DMM) and an oscilloscope with built-in waveform generator. Refer to the Ordering Information section on page 9 for the recommended models.

Lab sheet topic	Need required instruments
1. Introduction to the IoT Development Kit – Use the IoT development kit to perform system setup, connect between host and target, and test run a simple program using Ellipse C/C++ IDE.	No
2. Introduction to the Peripherals of the IoT Development Kit – Explore various functions of the IoT development kit, and develop programs to interface with push-button, LCD, external mass storage, UART and GPIO.	No
3. Interfacing to IoT Devices – Set up the development kit to interface with external sensors and actuators, and learn to interface gateway to sensor devices and display the results on an LCD.	No
4. Digital Communication Protocols for IoT – Write applications to use I^2C and SPI for communication, and configure digital sensors with I^2C and SPI interfaces.	Yes, oscilloscope
5. Wireless Sensor Networks for IoT – Use wireless communication over *Bluetooth* LE and ZigBee by developing IoT node devices that communicate with each other.	Yes, DMM
6. Exploring Cloud Messaging Protocol – Learn to call and use cloud services, use HTTP and MQTT protocols to connect to the cloud, set up, and test with mobile devices.	No
7. Cloud-enabled IoT Operation – Deploy an IoT sensor node onto cloud and visualize the results on an end user client device, such as a wearable device for activity monitoring	No

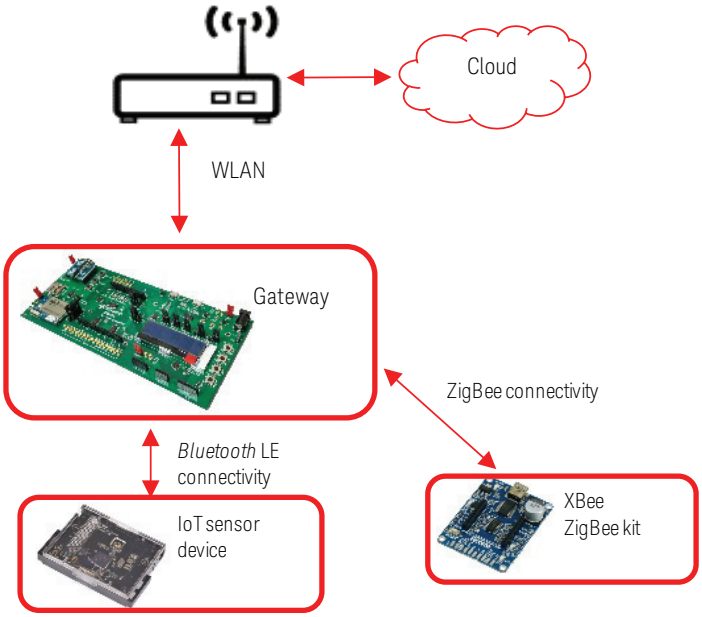


Figure 1. Typical lab setup.

Problem-based assignments

The problem-based assignments below allow students to enhance their problem-solving skills.

Smart street lamp	Develop a smart street lamp using the available sensors and actuators. The street lamp can be controlled over a network based on the light intensity of the surrounding environment.
Smart automobile	Develop a fitness tracker for cars using available sensors and logging the result in an SD card that can be retrieved with a smartphone for drivers to track the performance and safety of their cars.

IoT Development Kit Characteristics

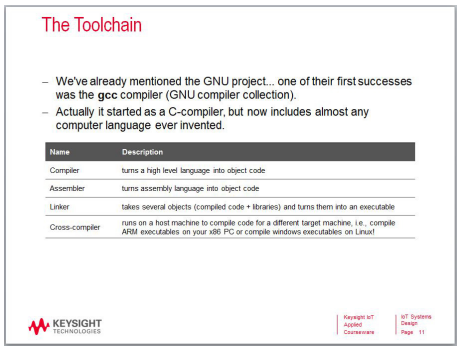
IoT development kit	
Dimensions	20 cm (w) x 8.5 cm (d) x 5 cm (h)
Compute module	Intel Edison (A dual-core, dual-threaded Intel Atom CPU at 500 MHz and a 32-bit Intel Quark microcontroller at 100 MHz)
RAM and flash storage	1 GB LPDDR3 PoP memory and 4 GB eMMC
Wireless communication	WLAN 802.11 a/b/g/n , *Bluetooth* LE (version 4.0) and ZigBee wireless connectivity
General	
Supply voltage	6 to 12 V AC adapter (2 mm DC jack) USB port
Warranty	1 year 3 months for accessories

System and installation requirements

PC operating system	Windows 8 and 10 (64-bit)
Interface	USB (3 ports)

Preview IoT Applied Courseware Contents

Take a look inside the contents of the IoT applied courseware. Samples of the teaching slides and lab sheets are available at **www.keysight.com/find/TeachIoT**



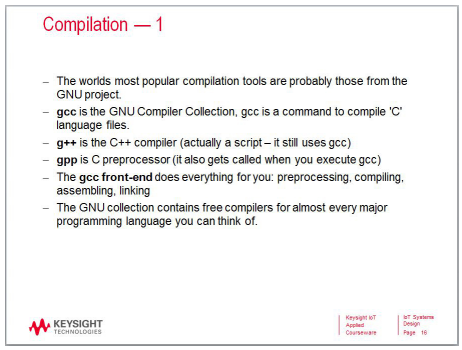
The Toolchain

- We've already mentioned the GNU project... one of their first successes was the **gcc** compiler (GNU compiler collection).
- Actually it started as a C-compiler, but now includes almost any computer language ever invented.

Name	Description
Compiler	turns a high level language into object code
Assembler	turns assembly language into object code
Linker	takes several objects (compiled code + libraries) and turns them into an executable
Cross-compiler	runs on a host machine to compile code for a different target machine, i.e., compile ARM executables on your x86 PC or compile windows executables on Linux!

KEYSIGHT TECHNOLOGIES

Keysight IoT Applied Courseware | IoT Systems Design | Page 11



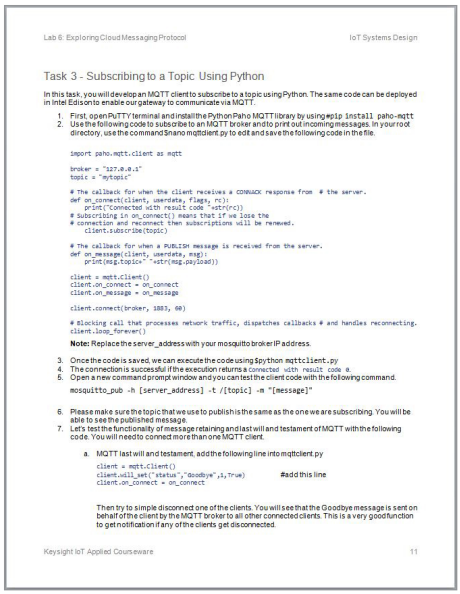
Compilation — 1

- The worlds most popular compilation tools are probably those from the GNU project.
- **gcc** is the GNU Compiler Collection, gcc is a command to compile 'C' language files.
- **g++** is the C++ compiler (actually a script – it still uses gcc)
- **gpp** is C preprocessor (it also gets called when you execute gcc)
- The **gcc front-end** does everything for you: preprocessing, compiling, assembling, linking
- The GNU collection contains free compilers for almost every major programming language you can think of.

KEYSIGHT TECHNOLOGIES

Keysight IoT Applied Courseware | IoT Systems Design | Page 16

Samples of the teaching slides – Chapter 5, Designing IoT Applications Using Embedded Systems. View more samples at the above link.



Lab 6: Exploring Cloud Messaging Protocol — IoT Systems Design

Task 3 - Subscribing to a Topic Using Python

In this task, you will develop an MQTT client to subscribe to a topic using Python. The same code can be deployed in Intel Edison to enable our gateway to communicate via MQTT.

1. First, open PuTTY terminal and install the Python Paho MQTT library by using `pip install paho-mqtt`
2. Use the following code to subscribe to an MQTT broker and to print out incoming messages. In your root directory, use the command $nano mqttclient.py to edit and save the following code in the file.

```
import paho.mqtt.client as mqtt

broker = "127.0.0.1"
topic = "mytopic"

# The callback for when the client receives a CONNACK response from  # the server.
def on_connect(client, userdata, flags, rc):
    print("Connected with result code "+str(rc))
# Subscribing in on_connect() means that if we lose the
# connection and reconnect then subscriptions will be renewed.
    client.subscribe(topic)

# The callback for when a PUBLISH message is received from the server.
def on_message(client, userdata, msg):
    print(msg.topic+" "+str(msg.payload))

client = mqtt.Client()
client.on_connect = on_connect
client.on_message = on_message

client.connect(broker, 1883, 60)

# Blocking call that processes network traffic, dispatches callbacks # and handles reconnecting.
client.loop_forever()
```

Note: Replace the server_address with your mosquitto broker IP address.

3. Once the code is saved, we can execute the code using $python `mqttclient.py`
4. The connection is successful if the execution returns a `connected with result code 0`.
5. Open a new command prompt window and you can test the client code with the following command.

```
mosquitto_pub -h [server_address] -t /[topic] -m "[message]"
```

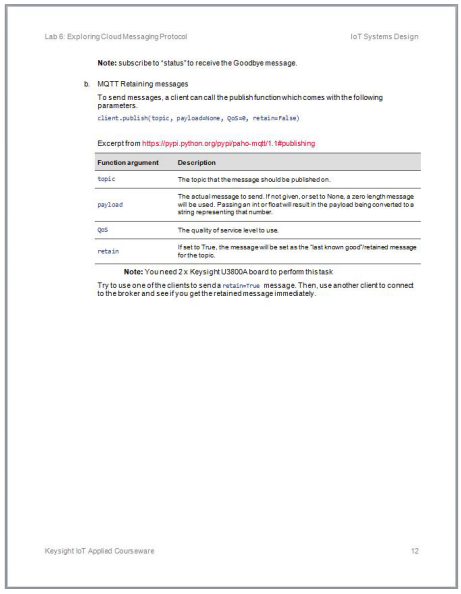
6. Please make sure the topic that we use to publish is the same as the one we are subscribing. You will be able to see the published message.
7. Let's test the functionality of message retaining and last will and testament of MQTT with the following code. You will need to connect more than one MQTT client.

a. MQTT last will and testament, add the following line into mqttclient.py

```
client = mqtt.Client()
client.will_set("status","Goodbye",1,True)          #add this line
client.on_connect = on_connect
```

Then try to simple disconnect one of the clients. You will see that the Goodbye message is sent on behalf of the client by the MQTT broker to all other connected clients. This is a very good function to get notification if any of the clients get disconnected.

Keysight IoT Applied Courseware — 11



Lab 6: Exploring Cloud Messaging Protocol — IoT Systems Design

Note: subscribe to "status" to receive the Goodbye message.

b. MQTT Retaining messages

To send messages, a client can call the publish function which comes with the following parameters.

```
client.publish(topic, payload=None, QoS=0, retain=False)
```

Excerpt from https://pypi.python.org/pypi/paho-mqtt/1.1#publishing

Function argument	Description
topic	The topic that the message should be published on.
payload	The actual message to send. If not given, or set to None, a zero length message will be used. Passing an int or float will result in the payload being converted to a string representing that number.
QoS	The quality of service level to use.
retain	If set to True, the message will be set as the "last known good"/retained message for the topic.

Note: You need 2 x Keysight U3800A board to perform this task

Try to use one of the clients to send a `retain=True` message. Then, use another client to connect to the broker and see if you get the retained message immediately.

Keysight IoT Applied Courseware — 12

Samples of the lab sheets – Lab 5, Exploring Cloud Messaging Protocol. View more samples at the above link.

Watch a video overview or live demonstration

Visit the Keysight Educators playlist on the Keysight YouTube network at **https://www.keysight.com/find/education-videos.** Watch an overview video to understand more about the IoT applied courseware, and take a look at how the training kit can be used in action within your teaching lab.

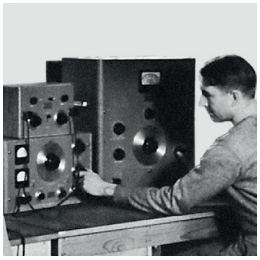
Ordering Information

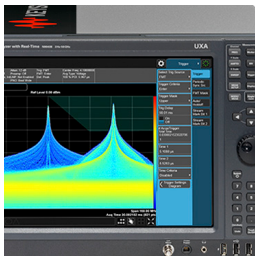
Product number	Description
IoT Systems Design Applied Courseware	
U3803A	IoT Systems Design applied courseware, with training kit only
U3804A	IoT Systems Design applied courseware, with training kit and teaching slides
Recommended instruments	
34465A-DIG[1]	6½ digit, performance True*volt* digital multimeter with high-speed digitizing and advanced triggering
EDUX1002G	InfiniiVision 1000 X-Series education oscilloscope with waveform generator, 50 MHz, 1 GS/s, 2 analog channels

1. Other 34460 Series Truevolt DMMs models may be used, but 34465A-DIG is recommended as this model comes with a digitizing option for use with the IoT Sensors and Power Management applied courseware (available Fall 2017).

Evolving

Our unique combination of hardware, software, support, and people can help you reach your next breakthrough. We are unlocking the future of technology.





From Hewlett-Packard to Agilent to Keysight

myKeysight

myKeysight
www.keysight.com/find/mykeysight
A personalized view into the information most relevant to you.

KEYSIGHT SERVICES
Accelerate Technology Adoption.
Lower costs.

Keysight Services
www.keysight.com/find/service
Our deep offering in design, test, and measurement services deploys an industry-leading array of people, processes, and tools. The result? We help you implement new technologies and engineer improved processes that lower costs.

STANDARD 3 YR WARRANTY

Three-Year Warranty
www.keysight.com/find/ThreeYearWarranty
Keysight's committed to superior product quality and lower total cost of ownership. Keysight is the only test and measurement company with three-year warranty standard on all instruments, worldwide. And, we provide a one-year warranty on many accessories, calibration devices, systems and custom products.

CALIBRATION WARRANTY UP TO 10 YEARS ASSURANCE PLANS

Keysight Assurance Plans
www.keysight.com/find/AssurancePlans
Up to ten years of protection and no budgetary surprises to ensure your instruments are operating to specification, so you can rely on accurate measurements.

Keysight Channel Partners
www.keysight.com/find/channelpartners
Get the best of both worlds: Keysight's measurement expertise and product breadth, combined with channel partner convenience.

For more information on Keysight Technologies' products, applications or services, please contact your local Keysight office. The complete list is available at:
www.keysight.com/find/contactus

Americas

Canada	(877) 894 4414
Brazil	55 11 3351 7010
Mexico	001 800 254 2440
United States	(800) 829 4444

Asia Pacific

Australia	1 800 629 485
China	800 810 0189
Hong Kong	800 938 693
India	1 800 11 2626
Japan	0120 (421) 345
Korea	080 769 0800
Malaysia	1 800 888 848
Singapore	1 800 375 8100
Taiwan	0800 047 866
Other AP Countries	(65) 6375 8100

Europe & Middle East

Austria	0800 001122
Belgium	0800 58580
Finland	0800 523252
France	0805 980333
Germany	0800 6270999
Ireland	1800 832700
Israel	1 809 343051
Italy	800 599100
Luxembourg	+32 800 58580
Netherlands	0800 0233200
Russia	8800 5009286
Spain	800 000154
Sweden	0200 882255
Switzerland	0800 805353 Opt. 1 (DE) Opt. 2 (FR) Opt. 3 (IT)
United Kingdom	0800 0260637

For other unlisted countries:
www.keysight.com/find/contactus
(BP-10-3-16)

DEKRA Certified
ISO9001 Quality Management System

www.keysight.com/go/quality
Keysight Technologies, Inc.
DEKRA Certified ISO 9001:2015
Quality Management System

KEYSIGHT TECHNOLOGIES
Unlocking Measurement Insights

Published in USA, February 24, 2017
5992-2087EN
www.keysight.com

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for Keysight ***manufacturer:***

Other Similar products are found below :

34904A 34465A U5404A U3400A-1CM U1461A U1281A U1273A U1194A U1185A 10076C E3640A E3642A DSOX3034T N2781B N3307A U1271A U1452A U2722A U2921A-100 82350C 1147B N6731B N6774A N3302A N2744A U1273AX N2142A N3304A E3631A/0E3/902 U8002A/0EM/903 E3643A/0EM/903 DAQA194A 33522B/903 N2856A 33611A/C13SPWR-903 U8001A/0EM/903 33210A/903 DAQM905A E3640A/0E3/902 DAQM907A N6743B 34465A/240/C13SPWR-900/C13SPWR-PLG N4837A 34450A/120/903 E3648A/0E3/902 U2751A U1586B 33522B U1115A U1594A