NXP Semiconductors

User's Guide

Document identifier: SLN-LOCAL2-IOT-UG

Rev. 0, 19 April 2021

# SLN-LOCAL2-IOT User’s Guide

# Contents

# Figures

# Tables

# Chapter 1
# System Requirements and Prerequisites

The MCU Local Voice Control SDK requires an up-to-date computer which runs MCUXpresso IDE. It also requires a terminal program to communicate with the device via USB.

Table 1. Tested computer configurations

| Computer type | OS version | Serial terminal application |
|---|---|---|
| PC | Windows 10 | TeraTerm, PuTTY |
| Mac | macOS | Serial, CoolTerm, goSerial |
| PC | Linux | PuTTY |

Below are listed development tools using MCU Local Voice Control SDK.

Table 2. Software tools and versions

| Software tool | Version | Description |
|---|---|---|
| Segger | JLink_v6.98 or later | Tool to program the flash. |
| MCUXpresso IDE | Version 11.3.0 | Eclipse base IDE for development environment |

# Chapter 2
# Usage Conditions

The following information is provided per Article 10.8 of the Radio Equipment Directive 2014/53/EU:

- Frequency bands in which the equipment operates
- The maximum RF power transmitted

Table 3.  Usage conditions

| PN | RF technology | (a) Frequency range | (b) Max transmitted power |
|---|---|---|---|
| SLN-LOCAL2-IOT | WiFi | 2412MHz – 2472MHz | 17.9dBm |

**EUROPEAN DECLARATION OF CONFORMITY (Simplified DoC per Article 10.9 of the Radio Equipment Directive 2014/53/EU)**

This apparatus, namely SLN-LOCAL2-IOT, conforms to the Radio Equipment Directive 2014/53/EU. The full EU Declaration of Conformity for this apparatus can be found at this location: https://www.nxp.com/

The product is expected to be used laying flat on a table, microphone output pointing up.

The data mode of the USB bus is not covered by the CE certification as this mode is used exceptionally to reprogram the device.

# Chapter 3 Introduction

NXP's MCU Local Voice Control 2$^{nd}$ generation development kit (part number: SLN-LOCAL2-IOT) is a comprehensive, secure and cost-optimized turnkey solution with a widely-adopted development environment that enables customers to quickly get to market with a production ready end-to-end software application.

SLN-LOCAL2-IOT embeds all the components required to produce a secure and *edge-computing* voice-control product without need of Wi-Fi or Cloud connectivity. The architecture is built upon a single core:

- i.MX RT106S or RT105S for the main application, powered by an ARM® Cortex®-M7 core

SLN-LOCAL2-IOT hardware highlights:

- Up to 600 MHz (528 MHz default) Cortex-M7 MCU Core
- 1MB of On-chip RAM (512 KB TCM)
- 32 MB Hyper Flash Memory for Fast XiP (eXecute In Place)
- Three PDM MEMS Microphones
- TFA9894 Class-D Amplifier
- WiFi/Bluetooth Combo Chip
- Integrated Speaker
- GPIO Expansion Headers

SLN-LOCAL2-IOT software highlights:

- Two-Stage Bootstrap and Bootloader Allowing Flexibility in Customer's Implementation
- Secure Boot Flow with High Assurance Booting (HAB)
- Over-the-Air (OTA) Update via WiFi
- Over-the-Wire (OTW) Update via UART
- Automated Manufacturing / Reprogramming Tools
- Speech Recognition Engine by Deep Learning
- Audio Front End (AFE) for Far-Field Automatic Speech Recognition (ASR)

SLN-LOCAL2-IOT is supported by a comprehensive and free-of-charge enablement suite from NXP and its partners including:

- MCUXpresso Development Tools
- Hardware Design Files
- Local Voice Application Software Source Code
- Software Audio Tuning Tools
- Documentation
- Training Material

# Chapter 4
# Getting Started with MCU Local Voice Control

## 4.1 Package and Collateral Content

The MCU Local Voice Control Kit arrives as shown in the following figure. Make sure to check for damage or marks and if found, contact your NXP representative.

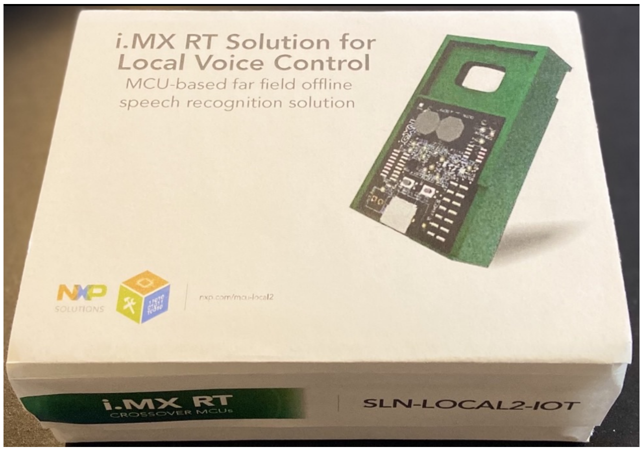


Figure 1. MCU Local Voice Control Kit package

When opening the box, the SLN-LOCAL2-IOT kit comes with a printed Quick Start Guide and a USB-C cable.

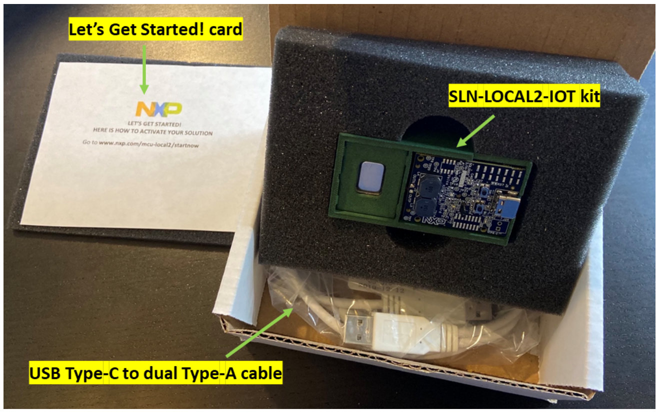


Figure 2. SLN-LOCAL2-IOT Kit content

## 4.2 Power on

Plug the USB Type-C connector into the SLN-LOCAL2-IOT kit and the dual Type-A connectors into your computer. Figure 3 shows how to connect the kit using a USB cable.

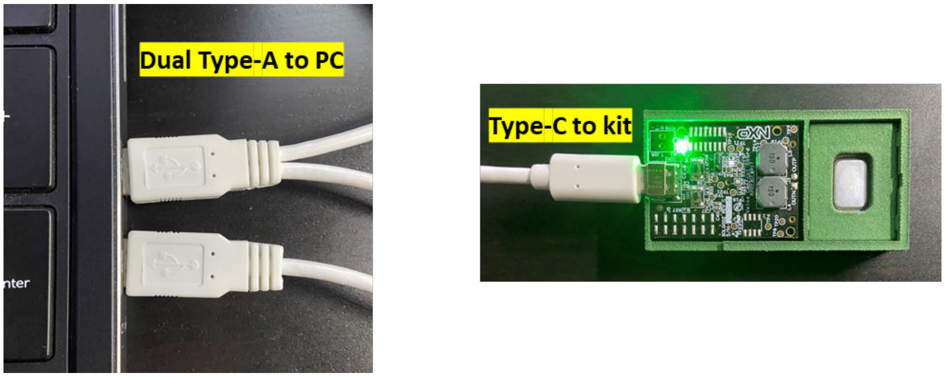


Figure 3. Connecting the SLN-LOCAL2-IOT Kit to the computer using a USB cable

When you power on the kit for the first time, the device turns the **LED green** for about 2 seconds before turning off. Once the LED is off, a chime is played from the onboard speaker indicating that the demo is running and listening to the wake word **Hey, NXP**. Figure 4 shows the LED status while booting and being ready for listening to wake word and commands.

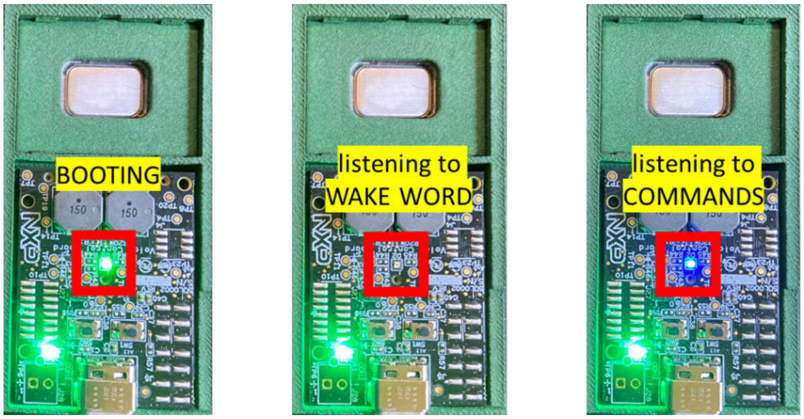


Figure 4. RGB LED status Indicator

## 4.3 Out-of-the-Box Demos

Three types of SLN-LOCAL2-IOT out-of-the-box demos are included:

- Demo #1: LED voice control
  - Language: English
  - Two-stage (wake word and command) ASR
- Demo #2: Smart Home (IoT) / Elevator / Audio / Washing-machine voice control in multiple languages
  - Language: selectable combinations of English, Chinese, German, and French
  - Two-stage ASR
- Demo #3: Oven voice control
  - Language: English
  - Multiturn (4-way) dialog-style ASR

### 4.3.1 Demo #1: LED Voice Control

Everything is ready out-of-the-box. Simply connect the kit to the computer and say **Hey, NXP**. The SLN-LOCAL2-IOT development kit responds to your voice with the sound bite **Can I help you?** and turns the **LED blue** while waiting for your LED control command.

Say one of the following commands:

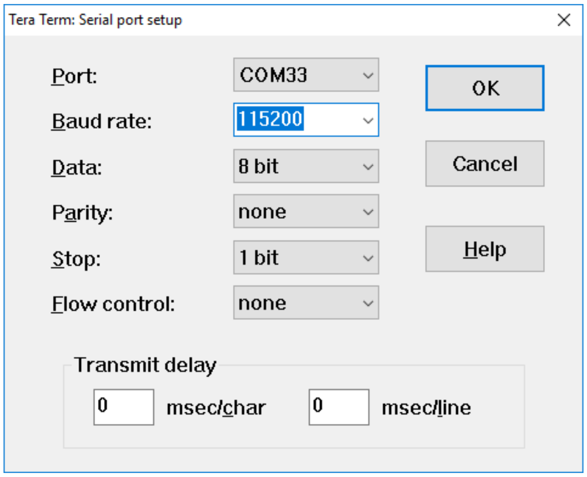
- L, E, D, Red
- L, E, D, Green
- L, E, D, Blue
- Cycle Fast
- Cycle Slow

If the kit detects your voice command, it changes the LED color accordingly and plays the sound bite **OK!**.

If the kit does not detect any of the LED control commands within a period of time, the device turns the **LED purple** and play a **chime** to indicate the response waiting time has ended. By default the response waiting time is **8 seconds** but you can change the value with the Shell command **timeout N** where `N` is the time value in milliseconds.

### 4.3.2 Connecting to a Serial Terminal

The out-of-the-box demos in the following sections require a connection to a serial terminal in order to display detected wake words and commands. Connect a serial terminal application to the USB serial device interface that enumerates (115200-8-N-1) as shown Figure 5.



Tera Term: Serial port setup

Port: COM33

Baud rate: 115200

Data: 8 bit

Parity: none

Stop: 1 bit

Flow control: none

Transmit delay

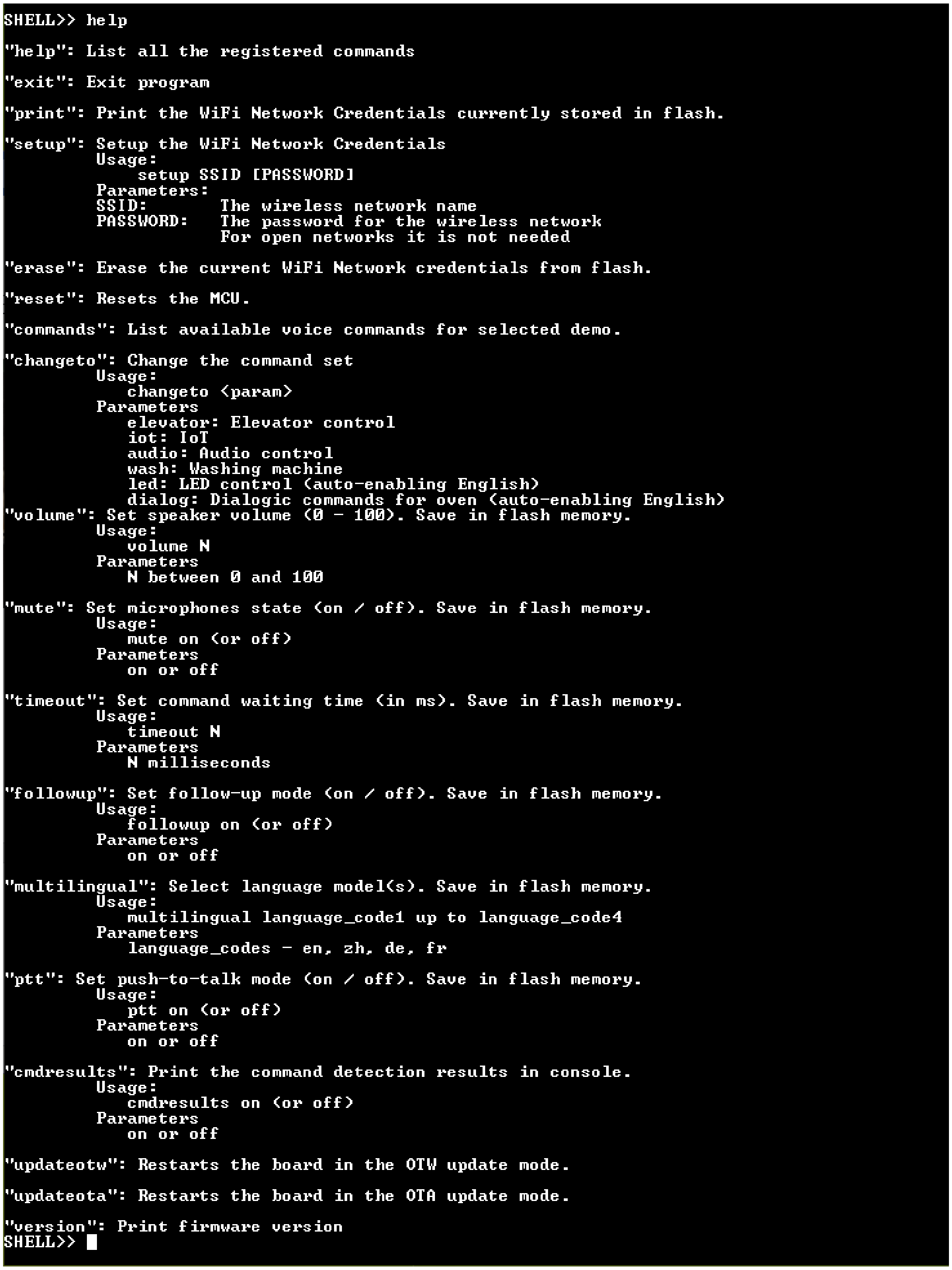
0 msec/char

0 msec/line

Figure 5. Serial terminal settings

Press **Enter** on the keyboard. The **SHELL>>** prompt appears.

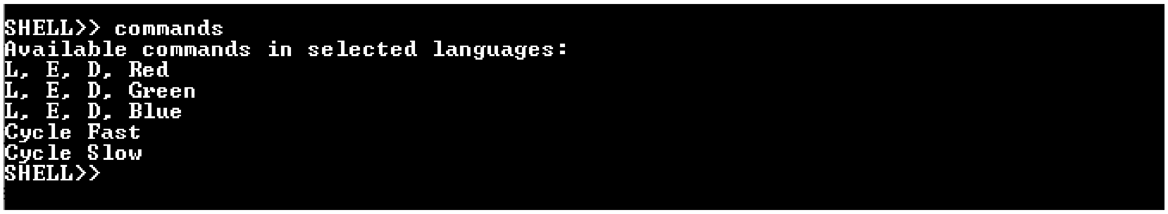
Type **help** to show the available commands at the Shell. Figure 6 shows the available Shell commands with a description of each.



```
SHELL>> help

"help": List all the registered commands

"exit": Exit program

"print": Print the WiFi Network Credentials currently stored in flash.

"setup": Setup the WiFi Network Credentials
         Usage:
             setup SSID [PASSWORD]
         Parameters:
         SSID:        The wireless network name
         PASSWORD:    The password for the wireless network
                      For open networks it is not needed

"erase": Erase the current WiFi Network credentials from flash.

"reset": Resets the MCU.

"commands": List available voice commands for selected demo.

"changeto": Change the command set
         Usage:
            changeto <param>
         Parameters
            elevator: Elevator control
            iot: IoT
            audio: Audio control
            wash: Washing machine
            led: LED control (auto-enabling English)
            dialog: Dialogic commands for oven (auto-enabling English)
"volume": Set speaker volume (0 - 100). Save in flash memory.
         Usage:
            volume N
         Parameters
            N between 0 and 100

"mute": Set microphones state (on / off). Save in flash memory.
         Usage:
            mute on (or off)
         Parameters
            on or off

"timeout": Set command waiting time (in ms). Save in flash memory.
         Usage:
            timeout N
         Parameters
            N milliseconds

"followup": Set follow-up mode (on / off). Save in flash memory.
         Usage:
            followup on (or off)
         Parameters
            on or off

"multilingual": Select language model(s). Save in flash memory.
         Usage:
            multilingual language_code1 up to language_code4
         Parameters
            language_codes - en, zh, de, fr

"ptt": Set push-to-talk mode (on / off). Save in flash memory.
         Usage:
            ptt on (or off)
         Parameters
            on or off

"cmdresults": Print the command detection results in console.
         Usage:
            cmdresults on (or off)
         Parameters
            on or off

"updateotw": Restarts the board in the OTW update mode.

"updateota": Restarts the board in the OTA update mode.

"version": Print firmware version
SHELL>> █
```

Figure 6. Available shell commands

Type commands to ensure which demo is set in selected languages. Figure 7 indicates that the current demo is set to the LED voice control demo.



```
SHELL>> commands
Available commands in selected languages:
L. E. D. Red
L. E. D. Green
L. E. D. Blue
Cycle Fast
Cycle Slow
SHELL>>
```

Figure 7. Commands for Demo 1: LED Voice Control

### 4.3.3 Switching Between the Out-of-the-Box Demos

Switching between the out-of-the-box demos can be done with Shell commands **changeto** and **multilingual** using the serial terminal. Figure 8 shows how to switch between the demos.

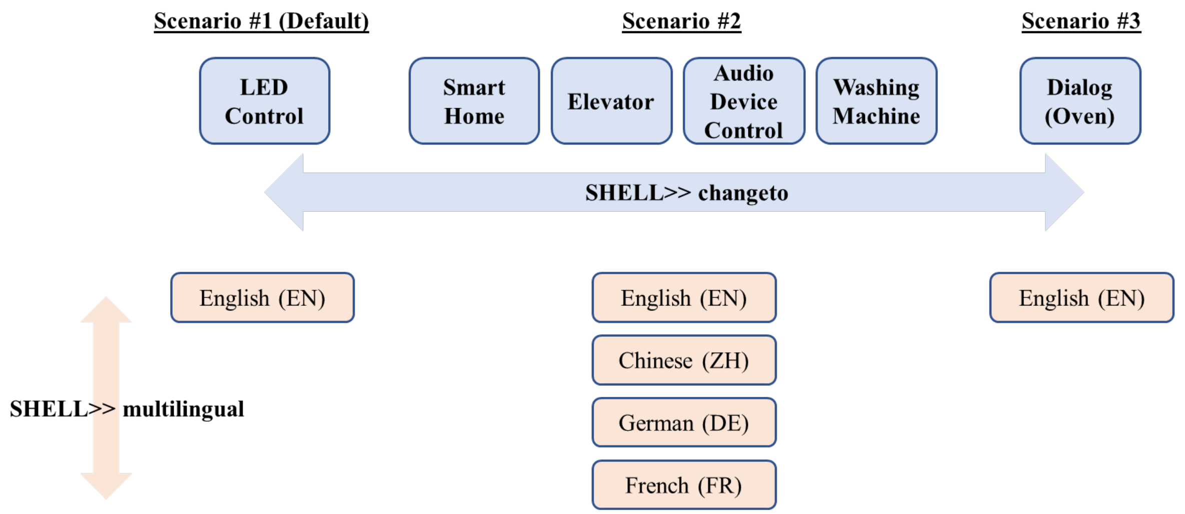
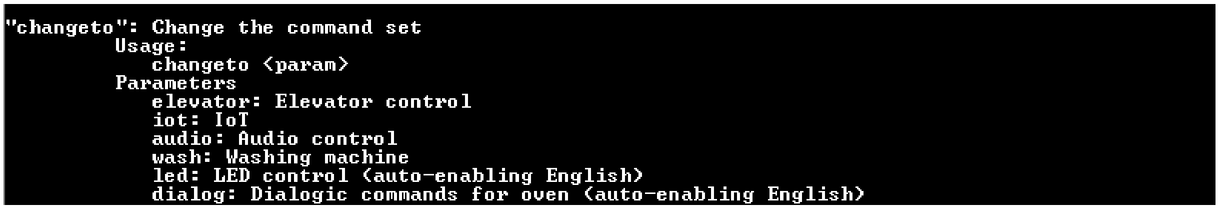


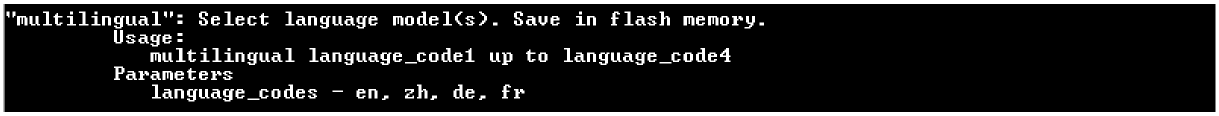
Figure 8. Demo settings by Shell commands - changeto and multilingual

Figure 9 and Figure 10 shows the usage of **changeto** and **multilingual** respectively.



```
"changeto": Change the command set
         Usage:
            changeto <param>
         Parameters
            elevator: Elevator control
            iot: IoT
            audio: Audio control
            wash: Washing machine
            led: LED control (auto-enabling English)
            dialog: Dialogic commands for oven (auto-enabling English)
```

Figure 9. Demo selection command

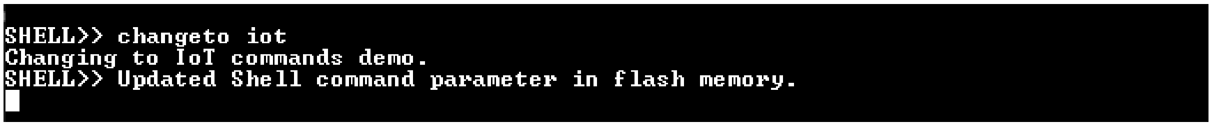


```
"multilingual": Select language model(s). Save in flash memory.
         Usage:
            multilingual language_code1 up to language_code4
         Parameters
            language_codes - en, zh, de, fr
```

Figure 10. Language selection command

### 4.3.4 Demo #2: Smart Home in Multiple Languages

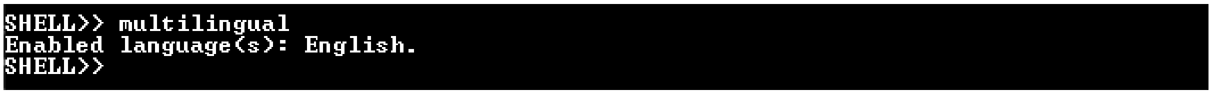
Type **changeto iot** to switch the demo to **Smart Home**, as shown in Figure 11.



```
SHELL>> changeto iot
Changing to IoT commands demo.
SHELL>> Updated Shell command parameter in flash memory.
```

Figure 11.  Result of changeto iot Shell command

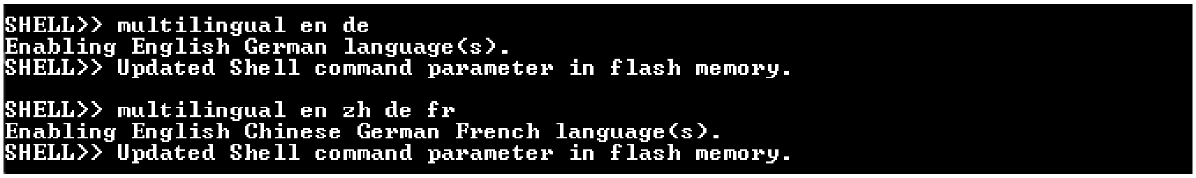
Type **multilingual** to ensure which languages are selected, as shown in Figure 12.



```
SHELL>> multilingual
Enabled language(s): English.
SHELL>>
```

Figure 12.  Result of multilingual Shell command

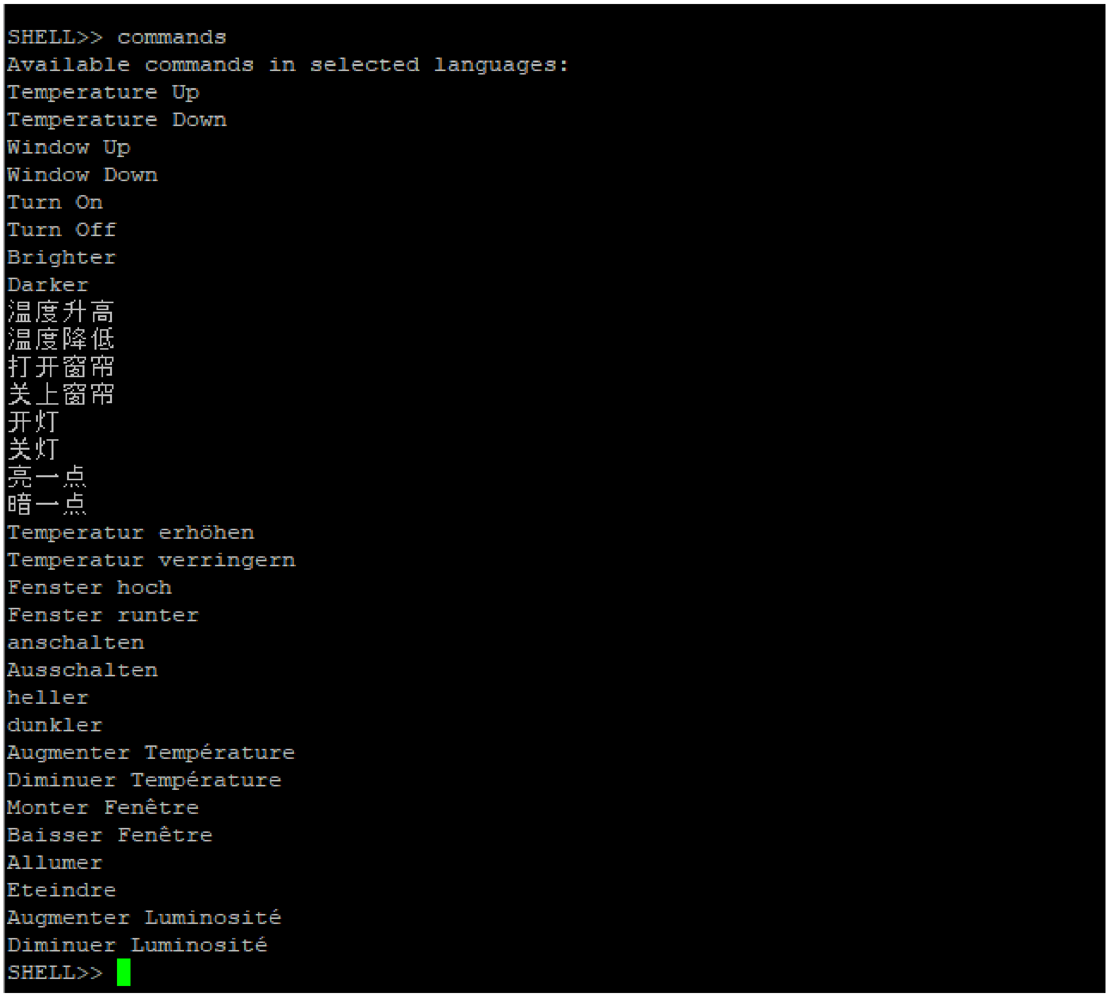
You can select any combinations of four different languages. For example, if you want to enable English and German, then enter **multilingual en de** where **en** and **de** are the language codes of English and German respectively. If you want to enable four languages, then enter **multilingual en zh de fr** where **zh** and **fr** are the language codes of Chinese and French respectively. You can also enable one language only. Simply enter **multilingual** followed by the **<language_code>** you want to enable. Figure 13 shows the examples. All the language selection is saved in flash memory, so it will remain in place even after the kit is rebooted.



```
SHELL>> multilingual en de
Enabling English German language(s).
SHELL>> Updated Shell command parameter in flash memory.

SHELL>> multilingual en zh de fr
Enabling English Chinese German French language(s).
SHELL>> Updated Shell command parameter in flash memory.
```

Figure 13.  Examples of selecting multiple languages

Type **commands**, to check that the current demo and the available voice commands are set in the selected languages. Figure 14 shows the commands of **Smart Home** in the selected four languages.



```
SHELL>> commands
Available commands in selected languages:
Temperature Up
Temperature Down
Window Up
Window Down
Turn On
Turn Off
Brighter
Darker
温度升高
温度降低
打开窗帘
关上窗帘
开灯
关灯
亮一点
暗一点
Temperatur erhöhen
Temperatur verringern
Fenster hoch
Fenster runter
anschalten
Ausschalten
heller
dunkler
Augmenter Température
Diminuer Température
Monter Fenêtre
Baisser Fenêtre
Allumer
Eteindre
Augmenter Luminosité
Diminuer Luminosité
SHELL>>
```

Figure 14. Available voice commands for Smart Home (IoT) demo in English, Chinese, German, and French

Say a wake word in one of the selected languages. Four wake words are available:

- Hey, NXP (English)
- *你好, 恩智浦* (Chinese)
- Hallo, NXP (German)
- Salut, NXP (French)

If the SLN-LOCAL2-IOT kit is triggered by a wake word, it will respond to your voice with a sound bite in the **detected language** and turns the **LED blue**, as shown in Figure 15, while waiting for your voice control command. The sound bite responses in the four languages are:

- Can I help you? (English)
- 我可以帮你吗? (Chinese)
- Kann ich Ihnen helfen? (German)
- Puis-je vous aider? (French)

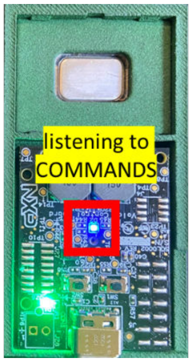
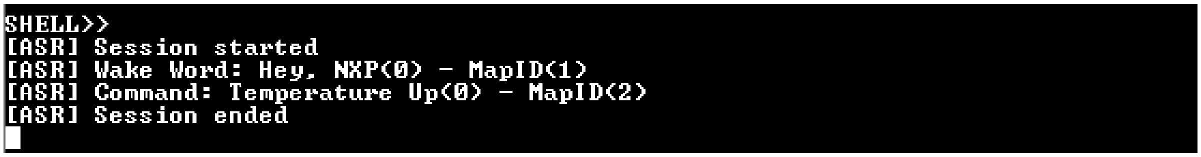


Figure 15.  LED Blue while listening to a voice command, same LED color for all Languages

Say any command from the list shown in Figure 14. If the voice command is detected, the kit responds with the sound bite **OK! in the detected language accent**.

Figure 16 shows a snapshot when detecting the English wake word **Hey, NXP** and Smart Home command **Temperature Up**.

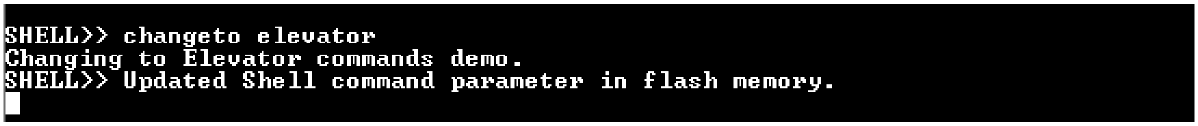


```
SHELL>>
[ASR] Session started
[ASR] Wake Word: Hey, NXP(0) - MapID(1)
[ASR] Command: Temperature Up(0) - MapID(2)
[ASR] Session ended
```

Figure 16.  Detecting English wake word followed by a Smart Home command

### 4.3.5   Demo #2: Elevator in Multiple Languages

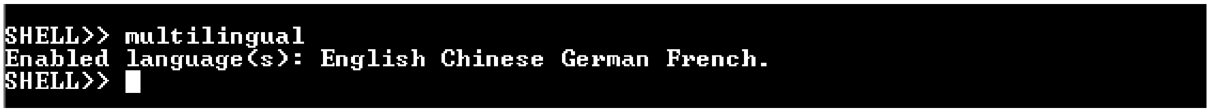
Type **changeto elevator** to switch the demo to **Elevator**, as shown in Figure 17.



```
SHELL>> changeto elevator
Changing to Elevator commands demo.
SHELL>> Updated Shell command parameter in flash memory.
```

Figure 17.  Result of changeto elevator Shell command

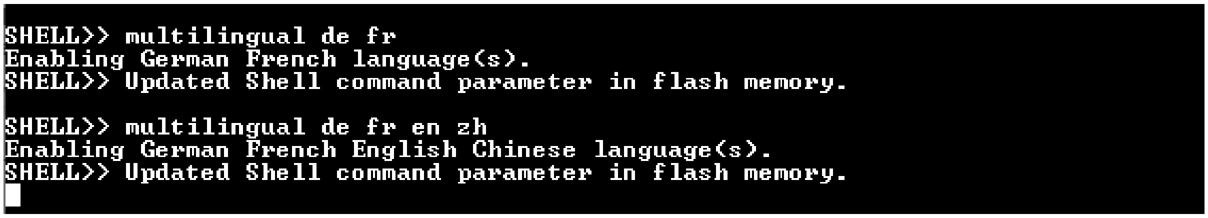
Type **multilingual** to check which languages are selected. Figure 18 shows the result of multilingual Shell command when English, Chinese, German, and French languages are selected,



```
SHELL>> multilingual
Enabled language(s): English Chinese German French.
SHELL>>
```

Figure 18.  Result of multilingual Shell command

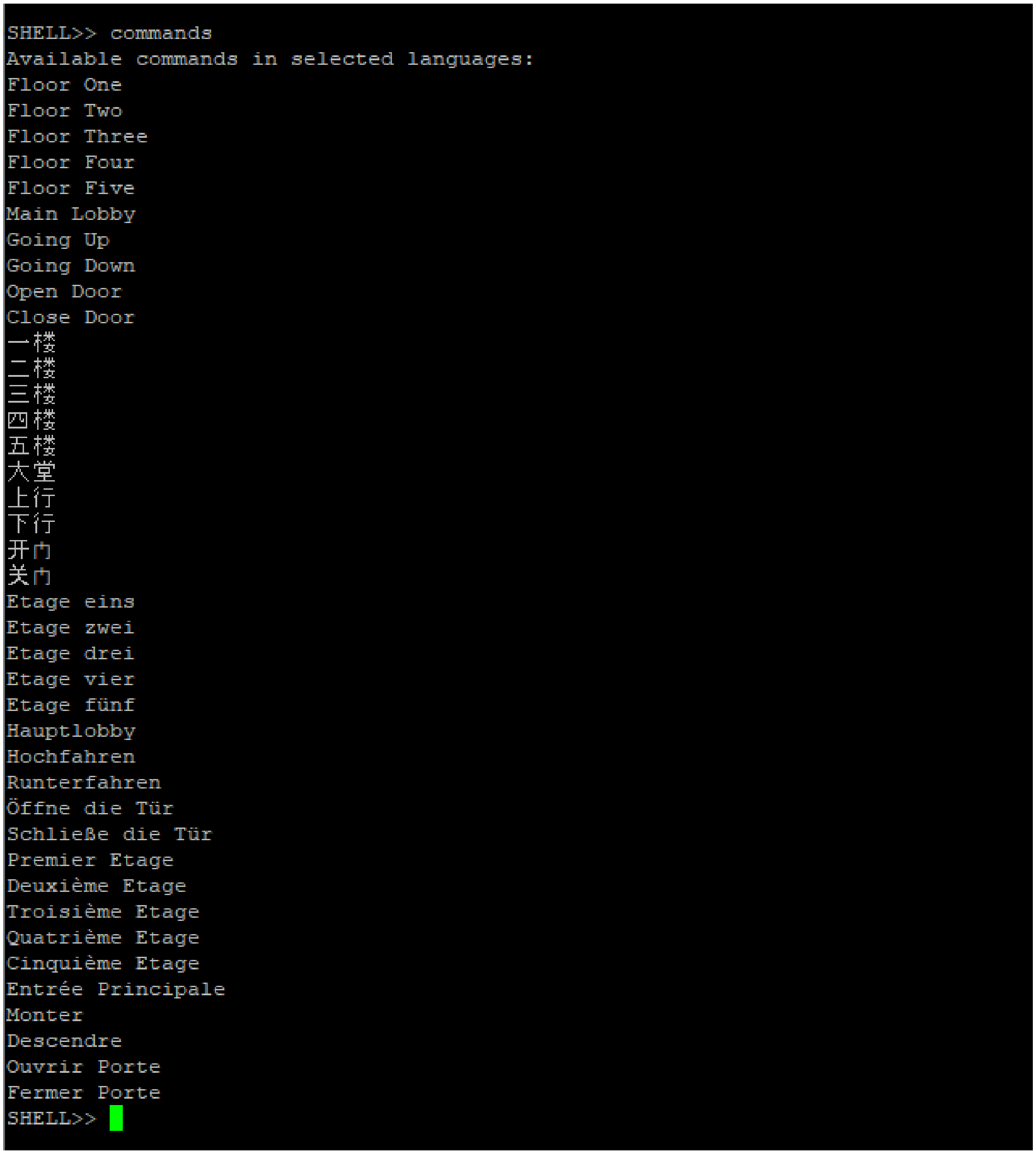
You can select any combinations of four different languages. For example, if you want to enable German and French, then enter **multilingual de fr** where **de** and **fr** are the language codes of German and French respectively. If you want to enable four languages, then type **multilingual en zh de fr** where **en** and **zh** are the language codes of English and Chinese respectively. You can also enable one language only. To enable one language, type **multilingual** followed by the **<language_code>** you want to enable. Figure 19 shows the examples. All the language selection is saved in flash memory. So it remains even after the kit is rebooted.



```
SHELL>> multilingual de fr
Enabling German French language(s).
SHELL>> Updated Shell command parameter in flash memory.

SHELL>> multilingual de fr en zh
Enabling German French English Chinese language(s).
SHELL>> Updated Shell command parameter in flash memory.
```

Figure 19. Examples of selecting multiple languages

Type **commands** to ensure that the current demo is set in selected languages. Figure 20 shows the commands of **Elevator** in the selected four languages.



```
SHELL>> commands
Available commands in selected languages:
Floor One
Floor Two
Floor Three
Floor Four
Floor Five
Main Lobby
Going Up
Going Down
Open Door
Close Door
一楼
二楼
三楼
四楼
五楼
大堂
上行
下行
开门
关门
Etage eins
Etage zwei
Etage drei
Etage vier
Etage fünf
Hauptlobby
Hochfahren
Runterfahren
Öffne die Tür
Schließe die Tür
Premier Etage
Deuxième Etage
Troisième Etage
Quatrième Etage
Cinquième Etage
Entrée Principale
Monter
Descendre
Ouvrir Porte
Fermer Porte
SHELL>>
```

Figure 20. Available voice commands for Elevator demo in English, Chinese, German, and French

Say a wake word of selected languages. Four wake words are available:

- Hey, NXP (English)
- 你好, 恩智浦 (Chinese)
- Hallo, NXP (German)
- Salut, NXP (French)

If the SLN-LOCAL2-IOT kit is triggered by a wake word, it responds to your voice with audio in the **detected language** and turns the **LED blue**, as shown in Figure 21, while waiting for your voice control command. The audio responses in the four languages are:

- Can I help you? (English)
- 我可以帮你吗? (Chinese)
- Kann ich Ihnen helfen? (German)
- Puis-je vous aider? (French)

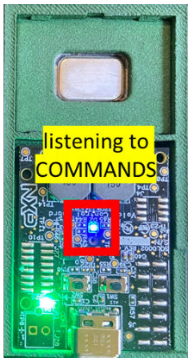
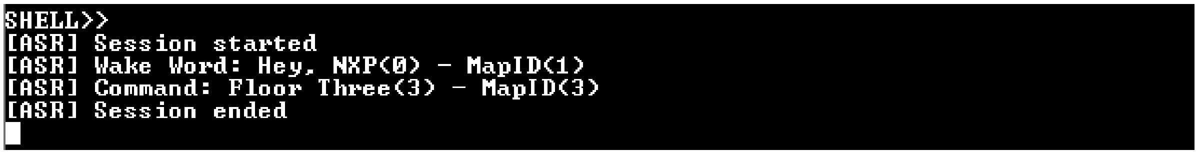


Figure 21. LED Blue while listening to a voice command, same LED color for all languages

Say any command from the list shown in Figure 20. If the voice command is detected, the kit responds with the sound bite **OK! in the detected language accent**.

Figure 22 shows the detection of the English wake word **Hey, NXP** and Elevator command **Floor Three**.

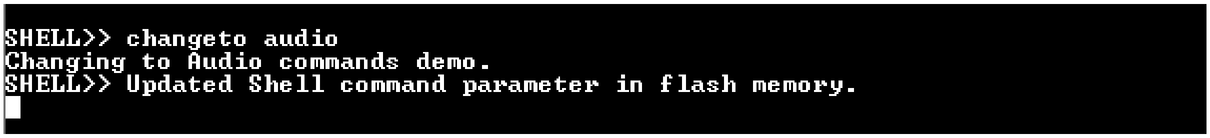


```
SHELL>>
[ASR] Session started
[ASR] Wake Word: Hey, NXP(0) - MapID(1)
[ASR] Command: Floor Three(3) - MapID(3)
[ASR] Session ended
```

Figure 22. Detecting English wake word followed by an Elevator command

### 4.3.6 Demo #2: Audio Device Control in Multiple Languages

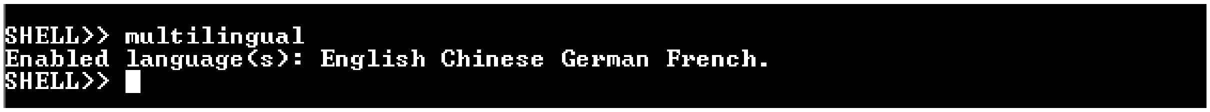
Type **changeto audio** to switch the demo to **Audio Device Control**, as shown in Figure 23.



```
SHELL>> changeto audio
Changing to Audio commands demo.
SHELL>> Updated Shell command parameter in flash memory.
```

Figure 23. Result of changeto audio Shell command

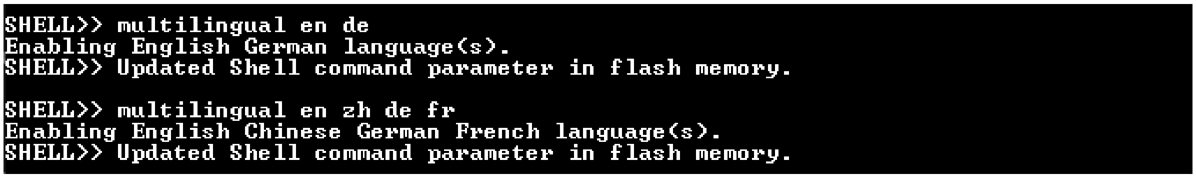
Type **multilingual** to check which languages are selected. Figure 24 shows the result of multilingual Shell command when English, Chinese, German, and French languages are selected.



```
SHELL>> multilingual
Enabled language(s): English Chinese German French.
SHELL>> █
```

Figure 24. Result of multilingual Shell command

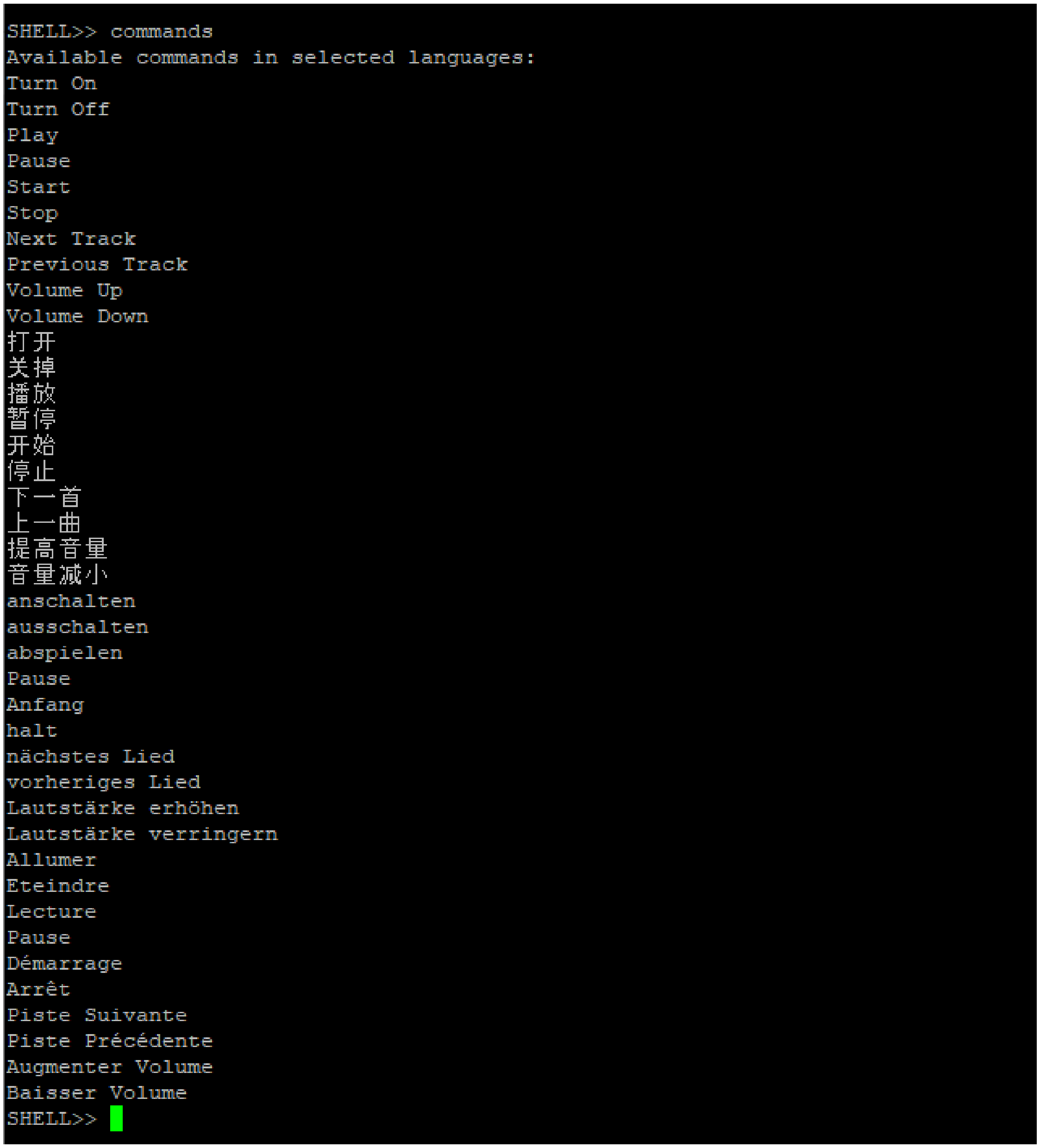
You can select any combinations of four different languages. For example, if you want to enable English and German, then enter **multilingual en de** where **en** and **de** are the language codes of English and German respectively. If you want to enable four languages, then enter **multilingual en zh de fr** where **zh** and **fr** are the language codes of Chinese and French respectively. You can also enable one language only. To enable one language, type **multilingual** followed by the **<language_code>** you want to enable. Figure 25 shows the examples. All the language selection is saved in flash memory, so it remains in place even after the kit is rebooted.



```
SHELL>> multilingual en de
Enabling English German language(s).
SHELL>> Updated Shell command parameter in flash memory.

SHELL>> multilingual en zh de fr
Enabling English Chinese German French language(s).
SHELL>> Updated Shell command parameter in flash memory.
```

Figure 25. Examples of selecting multiple languages

Type **commands**, to check the available voice commands for the current demo in the selected languages. Figure 26 shows the commands of **Audio Device Control** in the selected four languages.



```
SHELL>> commands
Available commands in selected languages:
Turn On
Turn Off
Play
Pause
Start
Stop
Next Track
Previous Track
Volume Up
Volume Down
打开
关掉
播放
暂停
开始
停止
下一首
上一曲
提高音量
音量减小
anschalten
ausschalten
abspielen
Pause
Anfang
halt
nächstes Lied
vorheriges Lied
Lautstärke erhöhen
Lautstärke verringern
Allumer
Eteindre
Lecture
Pause
Démarrage
Arrêt
Piste Suivante
Piste Précédente
Augmenter Volume
Baisser Volume
SHELL>>
```

Figure 26. Available voice commands for Audio Device Control demo in English, Chinese, German, and French

Say a wake word of selected languages. Four wake words are available:

- Hey, NXP (English)
- 你好, 恩智浦 (Chinese)
- Hallo, NXP (German)
- Salut, NXP (French)

If the SLN-LOCAL2-IOT kit is triggered by a wake word, it responds to your voice with a sound bite in the detected language and turns the **LED blue**, as shown in Figure 27, while waiting for your voice control command. The audio responses in the four languages are:

- Can I help you? (English)
- 我可以帮你吗? (Chinese)

- Kann ich Ihnen helfen? (German)
- Puis-je vous aider? (French)

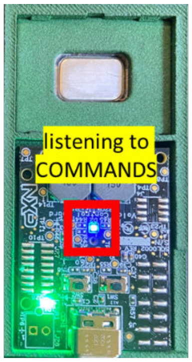
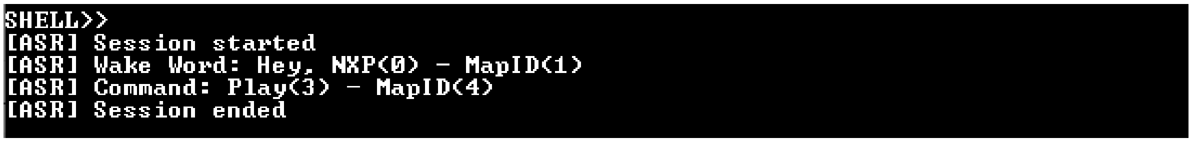


Figure 27. LED Blue while listening to a voice command, same LED color for all languages

Say a command from the list shown in Figure 26. If the voice command is detected, the kit responds with the sound bite **OK! in the detected language.**

Figure 28 shows a snapshot when detecting the English wake word **Hey, NXP** and Audio Device Control command **Play**.

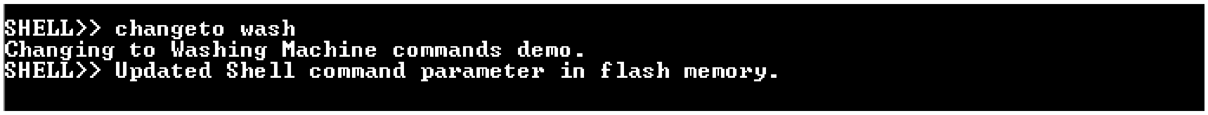


```
SHELL>>
[ASR] Session started
[ASR] Wake Word: Hey, NXP(0) - MapID(1)
[ASR] Command: Play(3) - MapID(4)
[ASR] Session ended
```

Figure 28. Detecting English wake word followed by an Audio Device Control command

### 4.3.7 Demo #2: Washing Machine in Multiple Languages

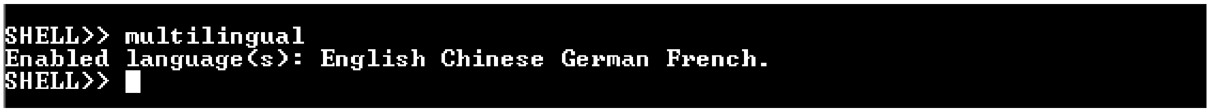
Type **changeto wash** to switch the demo to Washing Machine, as shown in Figure 29.



```
SHELL>> changeto wash
Changing to Washing Machine commands demo.
SHELL>> Updated Shell command parameter in flash memory.
```

Figure 29. Result of changeto wash Shell command

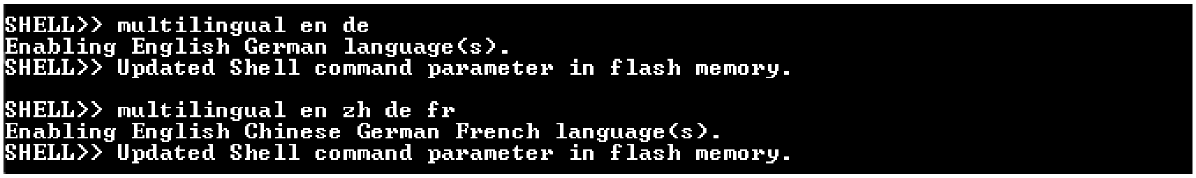
Type **multilingual** to ensure which languages are selected. Figure 30 shows the result of Multilingual shell command when English, Chinese, German, and French languages are selected.



```
SHELL>> multilingual
Enabled language(s): English Chinese German French.
SHELL>> █
```

Figure 30. Result of multilingual Shell command

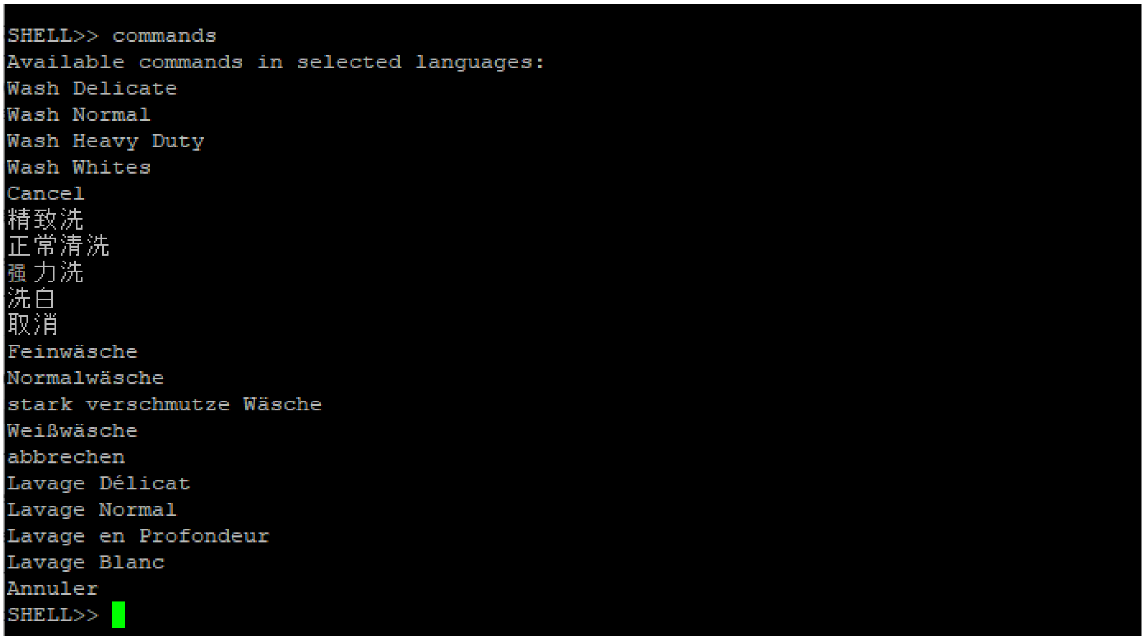
You can select any combinations of four different languages. For example, if you want to enable English and German, then enter **multilingual en de** where **en** and **de** are the language codes of English and German respectively. If you want to enable four languages, then enter **multilingual en zh de fr** where **zh** and **fr** are the language codes of Chinese and French respectively. You can also enable one language only. To enable one language, type **multilingual** followed by the **<language_code>** you want to enable.Figure 31 shows the examples. All the language selection is saved in flash memory, so it remains in place even after the kit is rebooted.



```
SHELL>> multilingual en de
Enabling English German language(s).
SHELL>> Updated Shell command parameter in flash memory.

SHELL>> multilingual en zh de fr
Enabling English Chinese German French language(s).
SHELL>> Updated Shell command parameter in flash memory.
```

Figure 31. Examples of selecting multiple languages

Type **commands** to ensure that the available voice commands for the current demo is set in the selected languages. Figure 32 shows the commands of **Washing Machine** in the selected four languages.



```
SHELL>> commands
Available commands in selected languages:
Wash Delicate
Wash Normal
Wash Heavy Duty
Wash Whites
Cancel
精致洗
正常清洗
强力洗
洗白
取消
Feinwäsche
Normalwäsche
stark verschmutze Wäsche
Weißwäsche
abbrechen
Lavage Délicat
Lavage Normal
Lavage en Profondeur
Lavage Blanc
Annuler
SHELL>>
```

Figure 32. Available voice commands for Washing Machine demo in English, Chinese, German, and French

Say a wake word of selected languages. Four wake words are available:

- Hey, NXP (English)
- 你好, 恩智浦 (Chinese)
- Hallo, NXP (German)
- Salut, NXP (French)

If the SLN-LOCAL2-IOT kit is triggered by a wake word, it responds to your voice with a sound bite in the **detected language** and turns the **LED blue**, as shown in Figure 33, while waiting for your voice control command. The audio responses in the four languages are:

- Can I help you? (English)
- 我可以帮你吗? (Chinese)
- Kann ich Ihnen helfen? (German)
- Puis-je vous aider? (French)

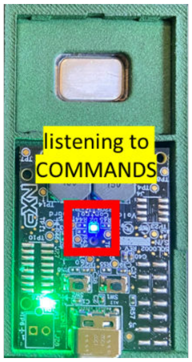
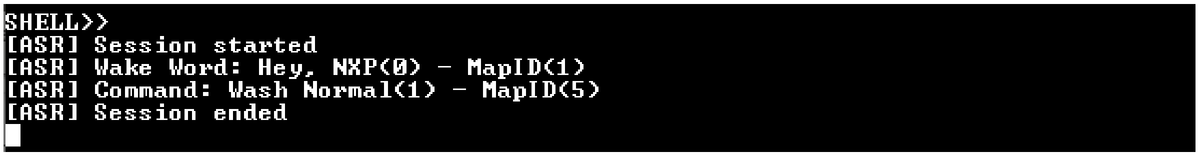


Figure 33. LED Blue while listening to a voice command, same LED color for all languages

Say a command from the list shown in Figure 32. If the voice command is detected, the kit will respond with the sound bite **OK! in the detected language.**

Figure 34 shows a snapshot when detecting the English wake word **Hey, NXP** and Audio Device Control command **Wash Normal**.

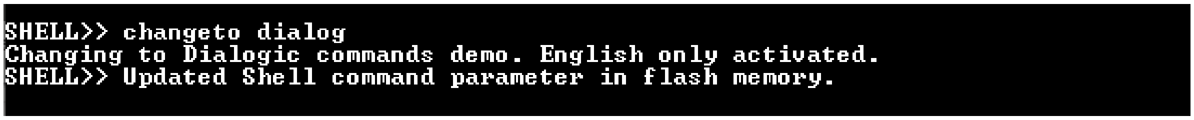


```
SHELL>>
[ASR] Session started
[ASR] Wake Word: Hey, NXP(0) - MapID(1)
[ASR] Command: Wash Normal(1) - MapID(5)
[ASR] Session ended
```

Figure 34. Detecting English wake word followed by a Washing Machine command

### 4.3.8 Demo #3: Dialog-style Voice Control

Type **changeto dialog** to switch the demo to Dialog, as shown Figure 35.

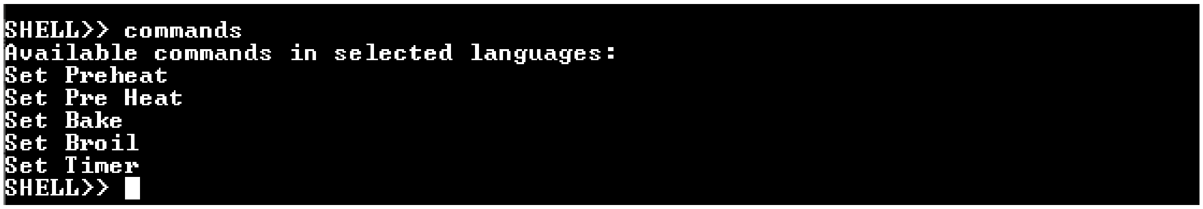


```
SHELL>> changeto dialog
Changing to Dialogic commands demo. English only activated.
SHELL>> Updated Shell command parameter in flash memory.
```

Figure 35. Result of changeto dialog Shell command

The **Dialog** demo supports English only. If your previous language settings included multiple languages, the shell command **changeto dialog** automatically changes the language selection to English, as shown in Figure 35.

Type **commands**, to ensure that the available voice commands for the current demo are set in the selected languages. Figure 36 shows the commands of the Dialog demo.



```
SHELL>> commands
Available commands in selected languages:
Set Preheat
Set Pre Heat
Set Bake
Set Broil
Set Timer
SHELL>>
```

Figure 36. Available voice commands for Dialog demo

Figure 37 shows the flow of the Dialog demo. After the wake word, if you say **Set Preheat** or **Set Bake** or **Set Broil**, you are prompted to specify the temperature value.

If you say **Set Timer**, you are prompted to specify the timer value. Say the temperature or timer value you want to set and the kit confirms with a sound bite.

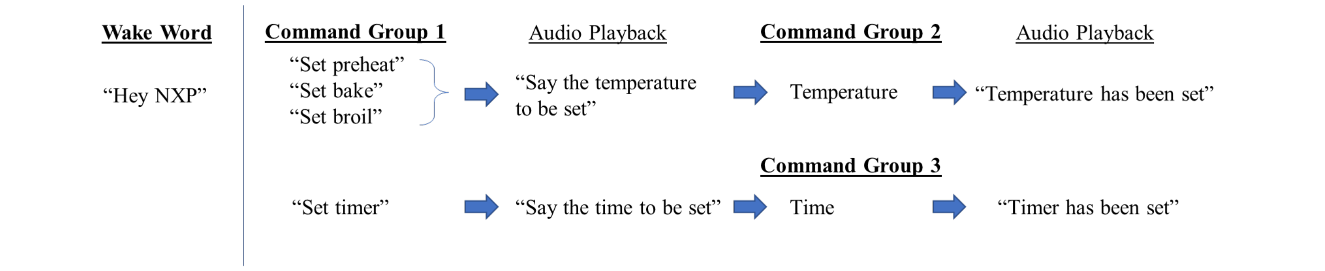


Figure 37. Dialog-type Voice Control with oven appliance use case

The following temperature and time values are accepted:

- Temperature
  - Value ranging from **two hundred degrees** to **five hundred degrees** with **10 degrees** interval.
- Timer
  - Value ranging from **one minute** to **ten minutes** with **1 minute** interval.
  - Value ranging from **ten minutes** to **sixty minutes** with **5 minutes** interval
  - **one hour**
  - **ninety minutes**
  - **two hours**
  - **three hours**

Say a wake word **Hey, NXP**. If the SLN-LOCAL2-IOT kit is triggered by a wake word, it responds to your voice with a sound bite **Can I help you?** and turns the **LED blue**, as shown in Figure 38, while waiting for your voice control command.

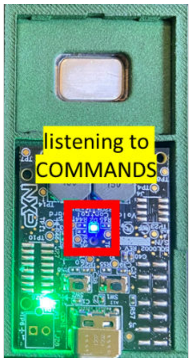
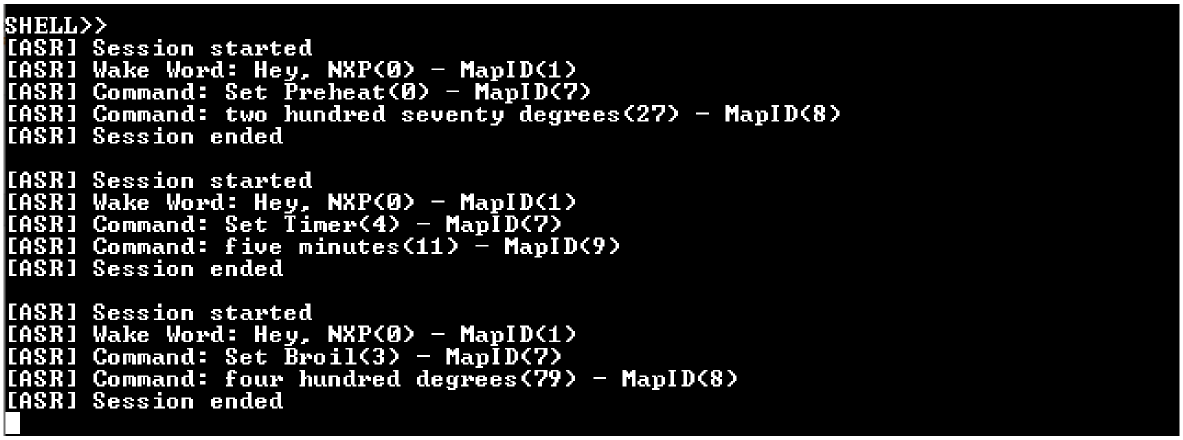


Figure 38. LED blue while listening to a voice command

Figure 39 shows a snapshot of Dialog demo examples.



```
SHELL>>
[ASR] Session started
[ASR] Wake Word: Hey, NXP(0) - MapID(1)
[ASR] Command: Set Preheat(0) - MapID(7)
[ASR] Command: two hundred seventy degrees(27) - MapID(8)
[ASR] Session ended

[ASR] Session started
[ASR] Wake Word: Hey, NXP(0) - MapID(1)
[ASR] Command: Set Timer(4) - MapID(7)
[ASR] Command: five minutes(11) - MapID(9)
[ASR] Session ended

[ASR] Session started
[ASR] Wake Word: Hey, NXP(0) - MapID(1)
[ASR] Command: Set Broil(3) - MapID(7)
[ASR] Command: four hundred degrees(79) - MapID(8)
[ASR] Session ended
```

Figure 39.  Dialog demo examples

# Chapter 5
# Controlling the Device

## 5.1 Physical Control Description

Table 4 describes the LED color behavior to help users understand what state the SLN-LOCAL2-IOT kit is in.

Table 4. Summary of LED color and behavior

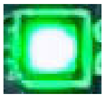
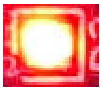
| Function | LED State (D2) | Color | Description |
|---|---|---|---|
| **Boot up** | Solid Green 2 seconds | | The device has powered on and is going through initialization |
| **Wake word detected** | Solid Blue | | The device detected the wake word and listens to a command. |
| **Command detected** | Green blink 200ms | <br> | The device detected a command. |
| **Timeout** | Purple blink 200ms | <br> | If no command is detected within a certain time, the device stops listening to a command. |
| **Microphone off** | Solid Orange | | Microphones are turned off. |
| **Push-to-Talk (PTT) mode** | Solid Cyan | | The device is on PTT mode. By pressing SW1, wake word detection phase is bypassed and the device listens to a command. |
| **Initialization Failed** | Solid Red |  | The device failed to initialize AFE or ASR. |
| **Audio stream error** | Solid Purple | | Audio stream after AFE is not transferred to ASR. |

*Table continues on the next page...*

Table 4. Summary of LED color and behavior (continued)

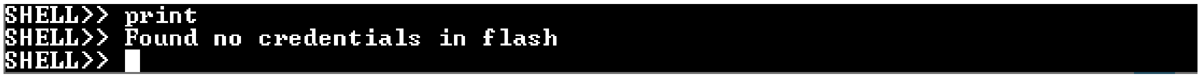
| Function | LED State (D2) | Color | Description |
|---|---|---|---|
| ASR memory error | Solid Orange | | During initialization or language or demo change, an error occurred in verifying memory pool size. |

## 5.2 Shell Command Interface

### 5.2.1 WiFi Credentials

Although the WiFi connectivity is not mandatory for the MCU-based local voice control solution, WiFi can be used for maintenance purpose such as an Over-the-Air (OTA) update. The WiFi credentials are saved in the flash memory and are managed by Shell command interface.

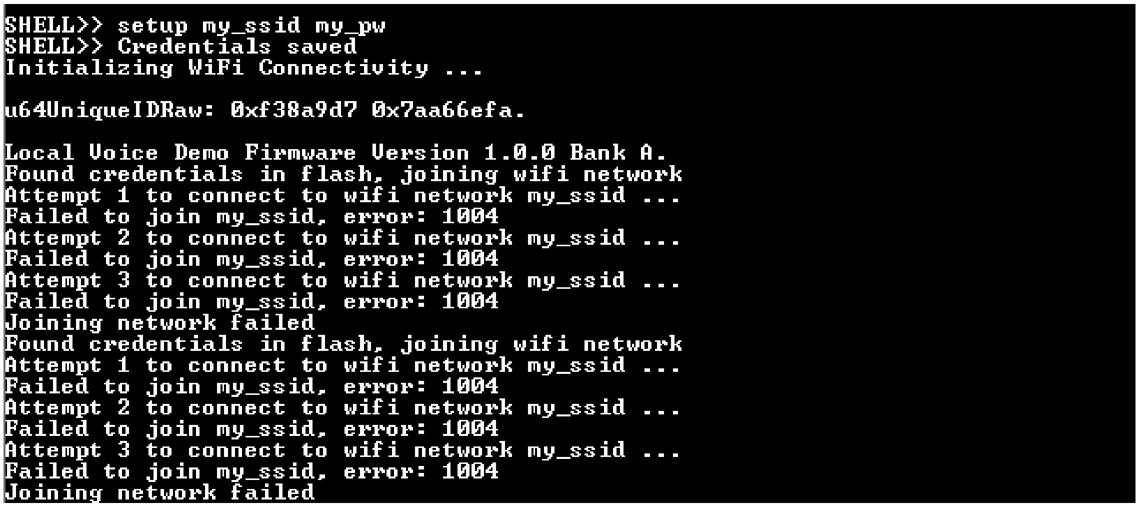
Type **print** to show the saved credentials. If no credentials are saved, the output appears as shown in Figure 40.



```
SHELL>> print
SHELL>> Found no credentials in flash
SHELL>> █
```

Figure 40. Shell command print to show the saved WiFi credentials

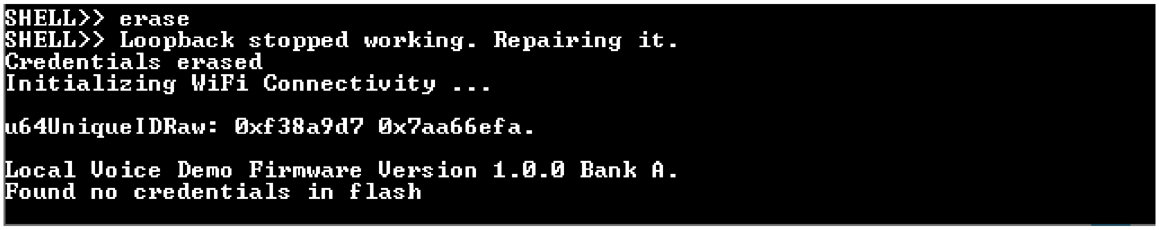
If you need WiFi, save your SSID and password in flash memory by entering **setup**. It will automatically restart the board and try connecting to your WiFi network. Figure 41 shows an example with incorrect SSID or password.



```
SHELL>> setup my_ssid my_pw
SHELL>> Credentials saved
Initializing WiFi Connectivity ...

u64UniqueIDRaw: 0xf38a9d7 0x7aa66efa.

Local Voice Demo Firmware Version 1.0.0 Bank A.
Found credentials in flash, joining wifi network
Attempt 1 to connect to wifi network my_ssid ...
Failed to join my_ssid, error: 1004
Attempt 2 to connect to wifi network my_ssid ...
Failed to join my_ssid, error: 1004
Attempt 3 to connect to wifi network my_ssid ...
Failed to join my_ssid, error: 1004
Joining network failed
Found credentials in flash, joining wifi network
Attempt 1 to connect to wifi network my_ssid ...
Failed to join my_ssid, error: 1004
Attempt 2 to connect to wifi network my_ssid ...
Failed to join my_ssid, error: 1004
Attempt 3 to connect to wifi network my_ssid ...
Failed to join my_ssid, error: 1004
Joining network failed
```

Figure 41. Connecting to WiFi network with incorrect credentials

You can erase your WiFi credentials by entering **erase**. The erasure is confirmed by the message shown in Figure 42.



```
SHELL>> erase
SHELL>> Loopback stopped working. Repairing it.
Credentials erased
Initializing WiFi Connectivity ...

u64UniqueIDRaw: 0xf38a9d7 0x7aa66efa.

Local Voice Demo Firmware Version 1.0.0 Bank A.
Found no credentials in flash
```

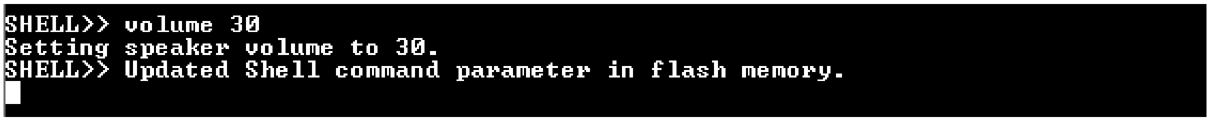
Figure 42. Erasing WiFi credentials

### 5.2.2 Switching Out-of-the-Box Demos

Shell commands **commands**, **changeto**, and **multilingual** are responsible for switching between demos and selecting demo languages. For more information and examples, see Out-of-the-Box Demos.

### 5.2.3 Controlling the Volume

You can control the speaker volume by entering **volume N** where **N** is an integer value ranging from *0* (mute) to *100* (max). The default volume is 55. Figure 43 shows an example of changing the speaker volume to 30.



```
SHELL>> volume 30
Setting speaker volume to 30.
SHELL>> Updated Shell command parameter in flash memory.
```

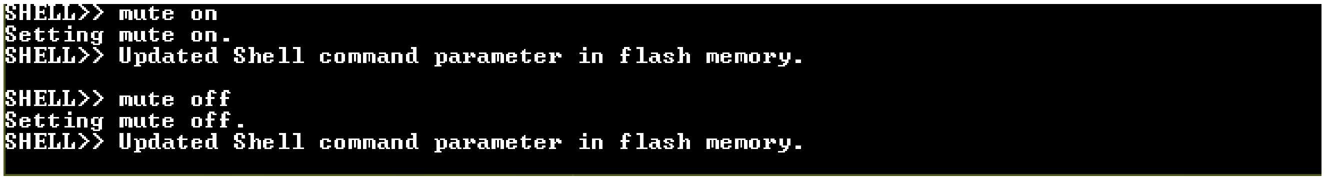
Figure 43. Setting speaker volume to 30

Speaker volume value is saved in flash memory and persists after reboot.

### 5.2.4 Muting the Microphone

You can mute or unmute your microphone by entering **mute on/off**. When muted, LED glows in solid Orange, as described in Physical Control Description.

Figure 44 shows the result of mute on / off (in bold) commands.

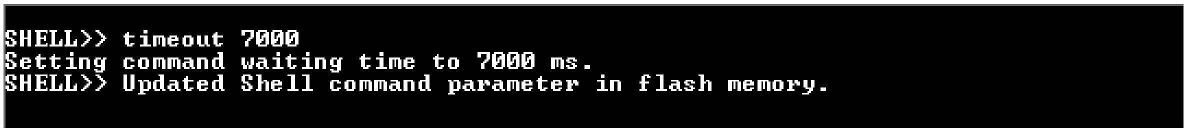


```
SHELL>> mute on
Setting mute on.
SHELL>> Updated Shell command parameter in flash memory.

SHELL>> mute off
Setting mute off.
SHELL>> Updated Shell command parameter in flash memory.
```

Figure 44. Muting the microphone

### 5.2.5 Setting the Timeout

You can set the command waiting time by entering **timeout N** where **N** is milliseconds. Figure 45 shows an example of setting the command waiting time to 7 seconds. The default timeout is 8 seconds.



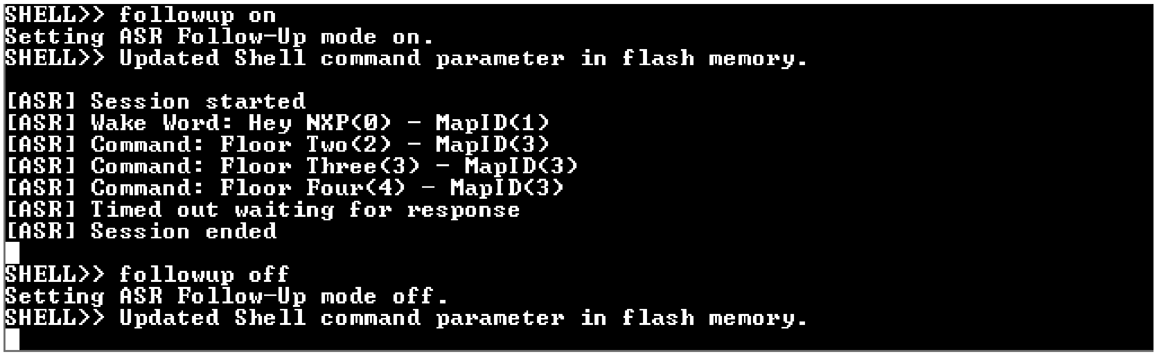
```
SHELL>> timeout 7000
Setting command waiting time to 7000 ms.
SHELL>> Updated Shell command parameter in flash memory.
```

Figure 45. Setting the waiting time to 7 seconds

You must say the voice command before the waiting time ends.

### 5.2.6 Enabling the Follow-up Mode

With the follow-up mode enabled, you can continue saying multiple commands after the kit is triggered by the wake word. Figure 46 shows an example of the wake word and commands – **Hey, NXP, Floor Two, Floor Three,** and **Floor Four**. Note that the wake word was said only once followed by three voice commands. After the last command, if no additional command follows for the duration of the waiting time, the ASR session ends.



```
SHELL>> followup on
Setting ASR Follow-Up mode on.
SHELL>> Updated Shell command parameter in flash memory.

[ASR] Session started
[ASR] Wake Word: Hey NXP(0) - MapID(1)
[ASR] Command: Floor Two(2) - MapID(3)
[ASR] Command: Floor Three(3) - MapID(3)
[ASR] Command: Floor Four(4) - MapID(3)
[ASR] Timed out waiting for response
[ASR] Session ended

SHELL>> followup off
Setting ASR Follow-Up mode off.
SHELL>> Updated Shell command parameter in flash memory.
```

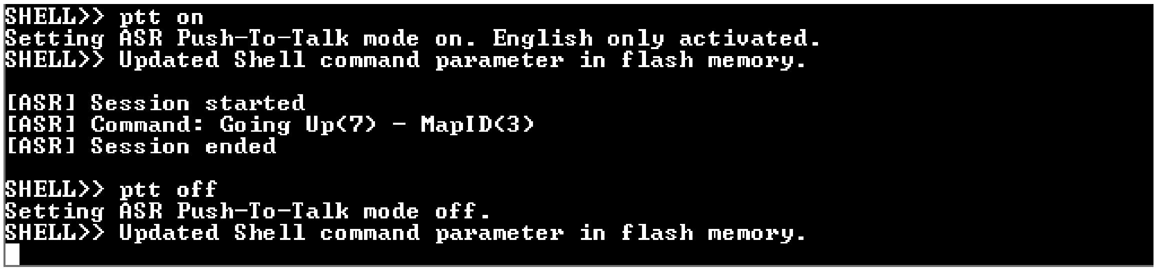
Figure 46. Follow-up mode use case

### 5.2.7 Enabling the Push-to-Talk (PTT) Mode

The PTT mode allows you to bypass the wake-word detection phase.

Enter **ptt on** to enable the ptt mode and **ptt off** to disable it. Cyan LED color indicates that the kit is currently in PTT mode, as described Physical Control Description. During PTT mode, press **SW1** to skip the wake word and continue saying a voice command.

Figure 47 shows an example of the PTT mode usage. Note that without a wake word, the voice command **Going Up** is detected.



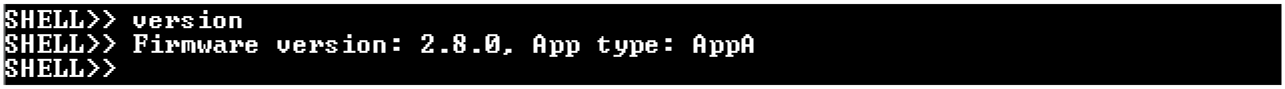
```
SHELL>> ptt on
Setting ASR Push-To-Talk mode on. English only activated.
SHELL>> Updated Shell command parameter in flash memory.

[ASR] Session started
[ASR] Command: Going Up(7) - MapID(3)
[ASR] Session ended

SHELL>> ptt off
Setting ASR Push-To-Talk mode off.
SHELL>> Updated Shell command parameter in flash memory.
```

Figure 47. Example of PTT mode usage

### 5.2.8 Checking the Software Version

The command **version** prints the firmware version and the current application type either App A or B. The type A or B is determined by the memory bank addresses, described in USB Mass Storage Device (MSD) Mode.

Figure 48 shows the result of version command when the application is located at bank A.



```
SHELL>> version
SHELL>> Firmware version: 2.8.0, App type: AppA
SHELL>>
```

Figure 48. Check the software version

## 5.3 USB Mass Storage Device (MSD) Mode

There are two application banks in the flash memory on the SLN-LOCAL2-IOT kit.

- Address for Application Bank A: 0x60300000
- Address for Application Bank B: 0x60D00000

For more information about generating an application binary for each bank, see Chapter 9 of the SLN-LOCAL2-IOT Developer's Guide (SLN-LOCAL2-IOT-DG).

ATTENTION

BYPASSING IMAGE VERIFICATION IS A SECURITY HOLE AND IT IS THE RESPONSIBILITY OF THE PRODUCT MAKER TO REMOVE THE VIOLATION IN PRODUCTION.

USB Mass Storage Device (MSD) allows you to re-flash the main application binary without a J-Link probe.

The MSD feature by default bypasses signature verification to allow an easier development flow as signing images can be a process not suitable for quick debugging and validation.

To put the device into MSD mode, hold down switch 2 (SW2) and power cycle the board until the pink LED (D2) lights up. The pink LED will turn on and off in 3 second intervals.

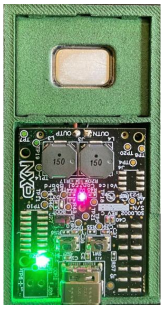
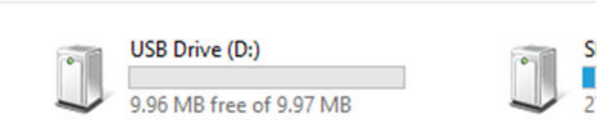


Figure 49. MSD update mode LED

Navigate to the computer's file explorer and confirm that the SLN-LOCAL2-IOT kit is mounted as a USB mass storage drive. A mounted kit is displayed on the file explorer as shown in the following figure:



USB Drive (D:)

9.96 MB free of 9.97 MB

Figure 50. SLN-LOCAL2-IOT Kit mounted as a USB Mass Storage Drive

Drag and drop the generated *.bin file for bank A or B onto the MSD drive. This starts the download process and write the *.bin file to flash. After the image is programmed into flash, it begins execution.

NOTE

Download the bank A binary, if the kit currently runs on bank B. Download the bank B binary, if the kit currently runs on bank A. You can find which bank is currently set by typing **version** at the Shell prompt.

# Chapter 6
# Product Specifications

Table 5. Product specifications

| Description | Specification |
|---|---|
| Electrical rating | DC supply via USB type C connector, 5.0 V +/-10 %, 2 A |
| Temperature rating | 10 °C to 40 °C |
| Wireless standards | Wi-Fi 2.4 GHz band (IEEE802.11b/g/n), Bluetooth, BLE |
| Radio frequency range | 2400 MHz – 2483.5 MHz |

# Chapter 7
# References

The following references are available to supplement this document:

- SLN-LOCAL2-IOT Developer’s Guide (SLN-LOCAL2-IOT-DG)
- Hardware files (Gerbers, schematics, BOM)

# Chapter 8
# Revision History

This table summarizes revisions to this document.

Table 6. Revision history

| Date | Version | Substantive changes | Reviewers |
|---|---|---|---|
| 19 April 2021 | 0 | Initial Draft | NXP |

# Chapter 9
# Acronyms

This table lists the acronyms used in this document.

| Acronym | Definition |
|---|---|
| AFE | Audio Front End |
| ASR | Automatic Speech Recognition |
| IOT | Internet Of Thing |
| JTAG | Joint Test Action Group |
| MCU | Microcontroller Unit |
| MEMS | Micro-Electro-Mechanical System |
| MSD | Mass Storage Device |
| OEM | Original Equipment Manufacturer |
| OTA | Over The Air |
| OTW | Over The Wire |
| PCM | Pulse-code modulation |
| PDM | Pulse-density modulation |
| PTT | Push-to-Talk |
| ROM | Read Only Memory |
| RTOS | Real-Time Operating System |
| SDK | Software Development Kit |
| UART | Universal asynchronous receiver-transmitter |

*How To Reach Us*

Home Page:

nxp.com

Web Support:

nxp.com/support

Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters that may be provided in NXP data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including “typicals,” must be validated for each customer application by customer's technical experts. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: nxp.com/SalesTermsandConditions.

While NXP has implemented advanced security features, all products may be subject to unidentified vulnerabilities. Customers are responsible for the design and operation of their applications and products to reduce the effect of these vulnerabilities on customer’s applications and products, and NXP accepts no liability for any vulnerability that is discovered. Customers should implement appropriate design and operating safeguards to minimize the risks associated with their applications and products.

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 19 April 2021
Document identifier: SLN-LOCAL2-IOT-UG