Joy of innovation
nuvoTon

Thermostat HMI with Tuya IoT Template

www.nuvoton.com

Agenda

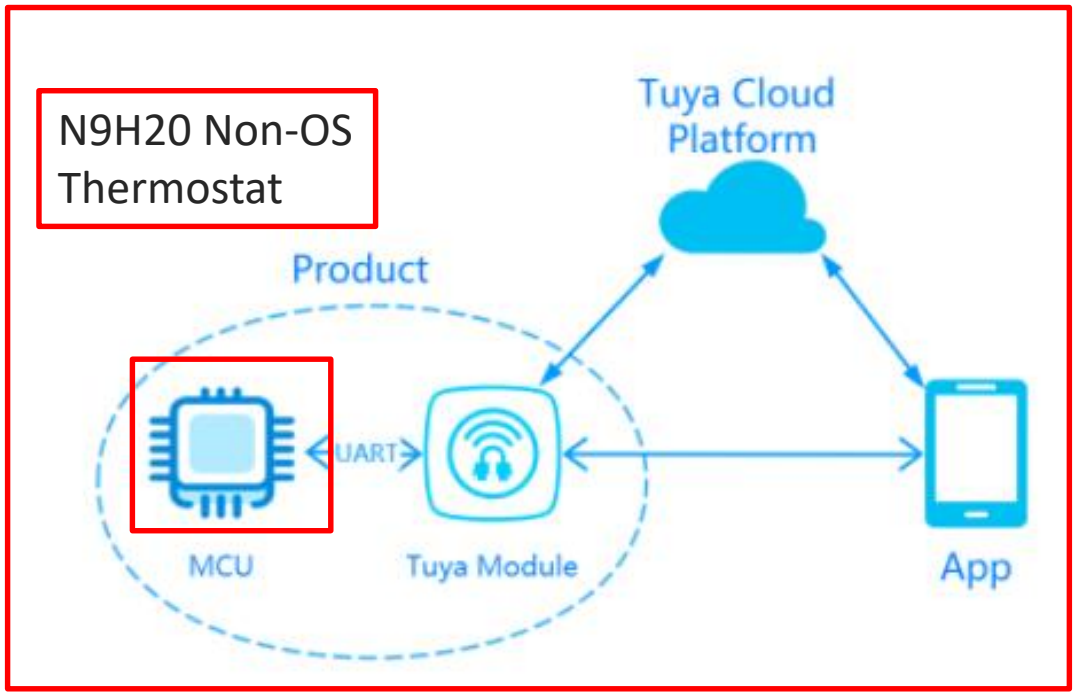
- Introduction
- Integrate Tuya IoT
- Integrate Thermostat HMI
- Q&A

Introduction

- 一站式物联网解决方案
  - https://developer.tuya.com/cn/docs/iot/guidelines-for-platform/introduction-of-tuya-iot-platform/introduction-of-tuya-iot-platform?id=K960olyp3qas9

CONT.

- **Utilize Tuya IoT Wi-Fi feature to apply N9H20 Non-OS Thermostat to cloud feature**



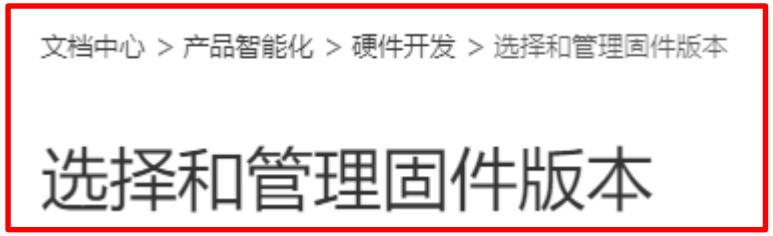
Integrate Tuya IoT

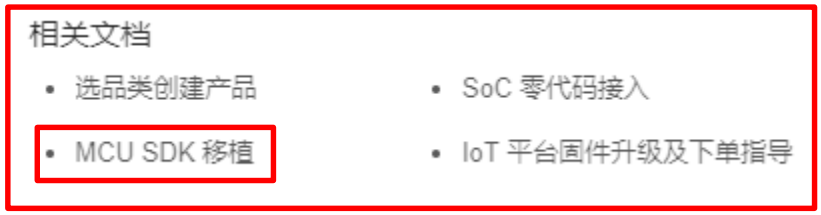
- **Start to develop**

You can find your PID here
Also please fill to
\N9H20_emWin_NonOS-master
\BSP\Driver\Source\TUYA_IoT\protocol.h
Please fill your PID to protocol.h and build the TUYA_IoT library for Thermostat HMI

CONT.

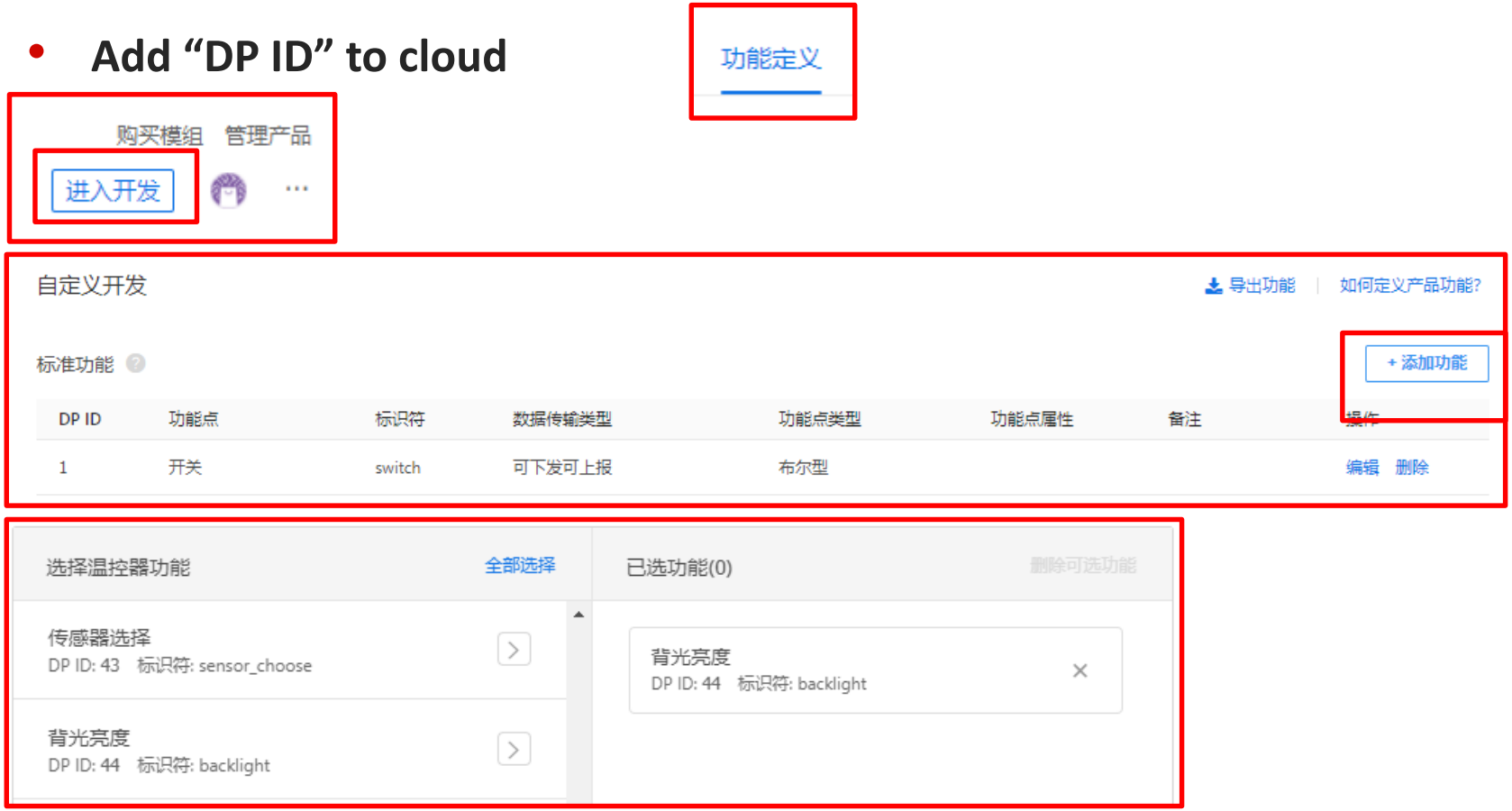
- **Utilize on-line official document**





CONT.

- **Add “DP ID” to cloud**



CONT.

- **Utilize “\N9H20_emWin_NonOS-master\BSP\SampleCode\emWin\Thermostat_N9H20_NonOS\Doc\DP_ID_List.xlsx” to add DP ID**

自定义开发

导出功能 | 如何定义产品功

标准功能

+ 添加功

DP ID	功能点	标识符	数据传输类型	功能点类型	功能点属性	备注	操作
1	开关	switch	可下发可上报	布尔型			编辑 删除
44	背光亮度	backlight	可下发可上报	数值型	数值范围：0-100, 间距：1, 倍数：0, 单位：%		编辑 删除

When update “DP ID”
Please reset Tuya IoT module
And update TUYA_IoT library

CONT.

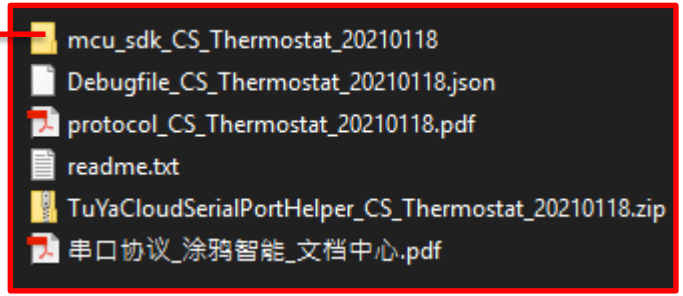
- **Utilize self-defined “DP ID”**
  - E. g., switching menu

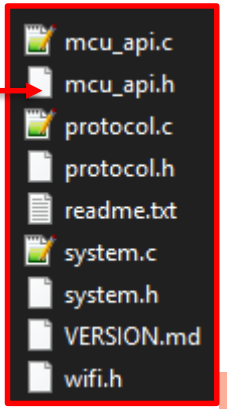
When update “DP ID”
Please reset Tuya IoT module
And update TUYA_IoT library

1	DP ID	功能点	标识符	数据传输类型	功能点类型	功能点属性	备注
12	107	Menu	Menu	可下发可上报	value	数值范围：1-3, 间距：1, 倍数：0,	单位：

CONT.

- **Download Tuya IoT SDK**





Please update “DP ID” and SDK file in TUYA_IoT library

CONT.

- **Tuya IoT SDK memory consumption**
  - ROM: 4KB
  - RAM: 100B

CONT.

- SDK
  - mcu_api.c (top level API: mcu_get_wifi_connect_status)
  - system.c (middle level API: wifi_uart_write_frame)
  - protocol.c (low lelve API: uart_transmit_output)

```c
void mcu_get_wifi_connect_status(void)
{
    wifi_uart_write_frame(GET_WIFI_STATUS_CMD, MCU_TX_VER, 0);
}
```

```c
void wifi_uart_write_frame(unsigned char fr_type, unsigned char fr_ver, unsigned short len)
{
    unsigned char check_sum = 0;

    wifi_uart_tx_buf[HEAD_FIRST] = 0x55;
    wifi_uart_tx_buf[HEAD_SECOND] = 0xaa;
    wifi_uart_tx_buf[PROTOCOL_VERSION] = fr_ver;
    wifi_uart_tx_buf[FRAME_TYPE] = fr_type;
    wifi_uart_tx_buf[LENGTH_HIGH] = len >> 8;
    wifi_uart_tx_buf[LENGTH_LOW] = len & 0xff;

    len += PROTOCOL_HEAD;
    check_sum = get_check_sum((unsigned char *)wifi_uart_tx_buf, len - 1);
    wifi_uart_tx_buf[len - 1] = check_sum;
```

```c
void uart_transmit_output(unsigned char value)
{
    #error "请将MCU串口发送函数填入该函数,并删除该行"
```

CONT.

- UART Rx

```
93  void UartDataValid_Handler(UINT8* buf, UINT32 u32Len)
94  {
95  //  UINT32 u32Idx;
96      INT32 i;
97
98      if(u32Len == 0)
99      {
100         sysprintf("uart rx Len = %d\n", u32Len);
101         return;
102     }
103 //  for(u32Idx = 0; u32Idx < u32Len; u32Idx++)
104 //  {
105 //      sysprintf("uart rx buf[%d] = 0x%x ",u32Idx, buf[u32Idx]);
106 //      sysprintf("\n");
107 //  }
108
109     for(i = 0; i < u32Len; i++)
110     {
111         uart_receive_input(buf[i]);
112     }
113     memset(buf, 0, u32Len);
114 }
```

CONT.

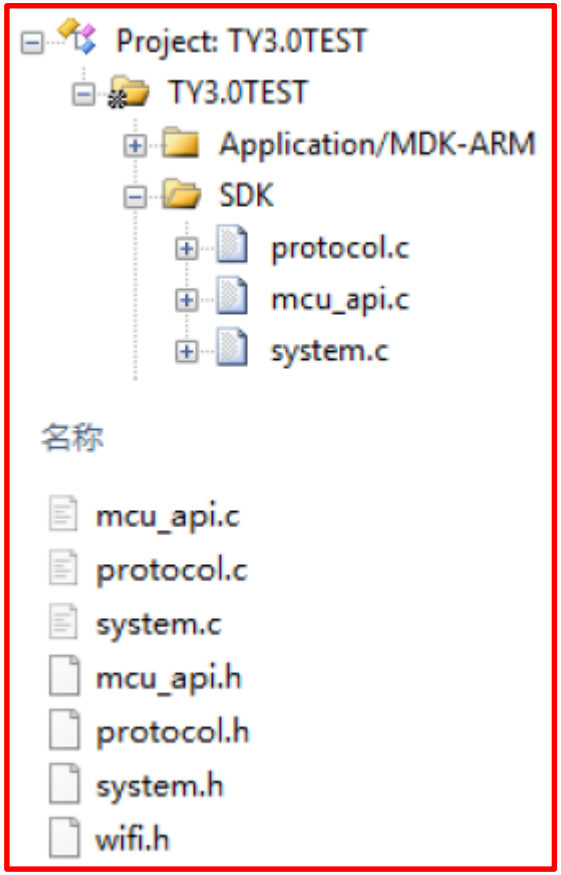
- UART Tx

```
85 /*****************************************************************************
86                          2:串口单字节发送函数
87 请将MCU串口发送函数填入该函数内,并将接收到的数据作为参数传入串口发送函数
88 *****************************************************************************/
89
90 /**
91  * @brief  串口发送数据
92  * @param[in] {value} 串口要发送的1字节数据
93  * @return Null
94  */
95 void uart_transmit_output(unsigned char value)
96 {
97     #error "请将MCU串口发送函数填入该函数,并删除该行"
98
99 /*
100     //Example:
101     extern void Uart_PutChar(unsigned char value);
102     Uart_PutChar(value);                              //串口发送函数
103 */
104 }
```

```
103 void uart_transmit_output(unsigned char value)
104 {
105   //#error "Please fill in the MCU serial port send function and delete the line"
106     Uart_PutChar(value);
107 }
```

CONT.

- SDK




```c
#include "wifi.h"
...
void main(void)
{
    wifi_protocol_init();
    ...
    while(1)
    {
        wifi_uart_service();
        ...
    }
    ...
}
```

CONT.

- **DPID (Data Pointer ID) upload**

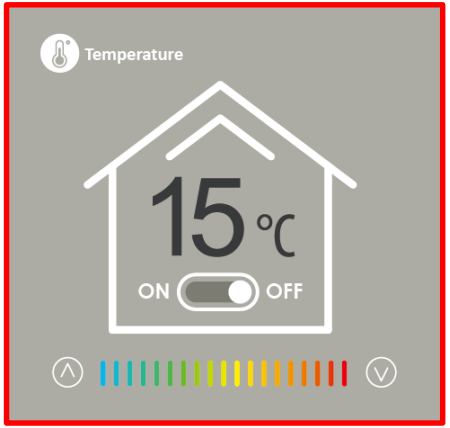
```
1  mcu_dp_bool_update(DPID_SWITCH,1); //BOOL 型数据上报
2  mcu_dp_value_update(DPID_TEMPER_SET,25); //VALUE 型数据上报
3  mcu_dp_string_update(DPID_DAY,"1234",4); //STRING 型数据上报
```

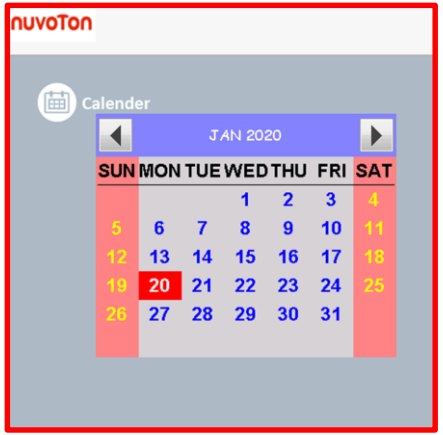
- **DPID download**

```
11  static unsigned char dp_download_switch_handle(const unsigned char v
12  alue[], unsigned short length)
13  {
14    //示例:当前 DP 类型为 BOOL
15    unsigned char ret;
16    //0:关状态/1:开状态
17    unsigned char switch1;
18
19    switch1 = mcu_get_dp_download_bool(value,length);
```

Integrate Thermostat HMI

- **MENU1 - Fan, Cool, Heat & Dehumidify**
  - On / Off and level control
- **MENU2**
  - Current temp.
  - Temp. control
  - Backlight control
  - Modbus control
- **MENU3**
  - Calendar date-scheduler

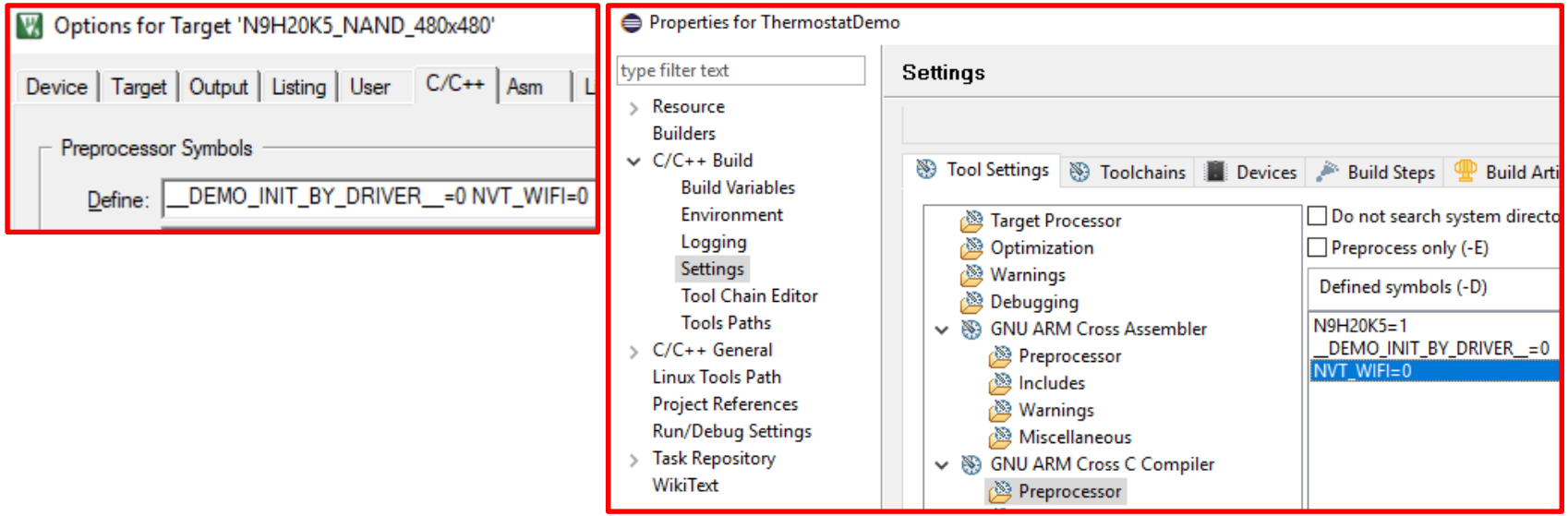




N9H20 Thermostat

CONT.

- Change UART1 from print debug message to connect with Tuya IoT module device
- Baud-rate is 9600 bps
- Default Thermostat HMI is disabled Tuya IoT feature “NVT_WIFI=0”



CONT.

- If “NVT_WIFI=1”
  - Please make sure your have Tuya IoT module device (WBR3 / WR3)
  - Build TUYA_IoT library
  - Copy TUYA_IoT.lib or libTUYA_IoT.a to \N9H_emWin_Template\Thermostat_N9H20_NonOS\Tuya_IoT
  - Re-build Thermostat

CONT.

- **Build TUYA_IoT library**
  - In Thermostat template v2.1, there is a new added folder called “Tuya_IoT” (\N9H_emWin_Template\Thermostat_N9H20_NonOS\Tuya_IoT)
  - Into “Tuya_IoT” then copy the sub-folder called “TUYA_IoT” to (\N9H20_emWin_NonOS-master\BSP\Driver\Source)
  - Fill correct PID to “protocol.h” then re-build “TUYA_IoT”

```
46 #define PRODUCT_KEY "please fill your PID"    //开发平台创建产品后生成的16位字符产品唯一标识
```

  - Your PID should be in downloaded SDK “protocol.h”

Online Resource

Nuvoton Website www.nuvoton.com

Forum

- http://forum.nuvoton.com
- http://www.nuvoton-MCU.com
- http://bbs.21ic.com/iclist-187-1.html

Media

- https://www.facebook.com/NuvotonNuMicro/
- ID: nuvoton_mcu
- @NuvotonMCU

BSP

- https:// github.com/OpenNuvoton
- https://gitlab.com/OpenNuvoton
- https://gitee.com/OpenNuvoton

eShop

- https://direct.nuvoton.com
- http://nuvoton.tmall.com/
- www.techdesign.com/market/nuvoton

Joy of innovation
nuvoTon

Q & A

www.nuvoton.com

CONT.

- **How to register Tuya IoT account?**
  - https://auth.tuya.com/?from=https%3A%2F%2Fiot.tuya.com%2F

CONT.

- How to purchase Tuya IoT device (WBR3 / WR3)?

CONT.

- **How to pair Tuya IoT device?**
  - Download Android APP “TuyaSmart” and follow the pairing flow

CONT.

- **May I use Tuya IoT cloud service directly without another charge fee?**
  - Yes.

CONT.

- **Block diagram between Tuya IoT module and N9H20**

tuya

1 概述

涂鸦 Wi-Fi 串口通用协议为涂鸦定制的 Wi-Fi 模组串口通用协议，主要用于涂鸦 Wi-Fi 模组与其它 MCU 串口直连做串口通信，其架构框图如下图所示。