CEC Continuing Education Center

DesignNews

Embedded Studio Primer

DAY 2: Embedded Studio and the STM32

Sponsored by

Digi-Key ELECTRONICS

informa markets

Webinar Logistics

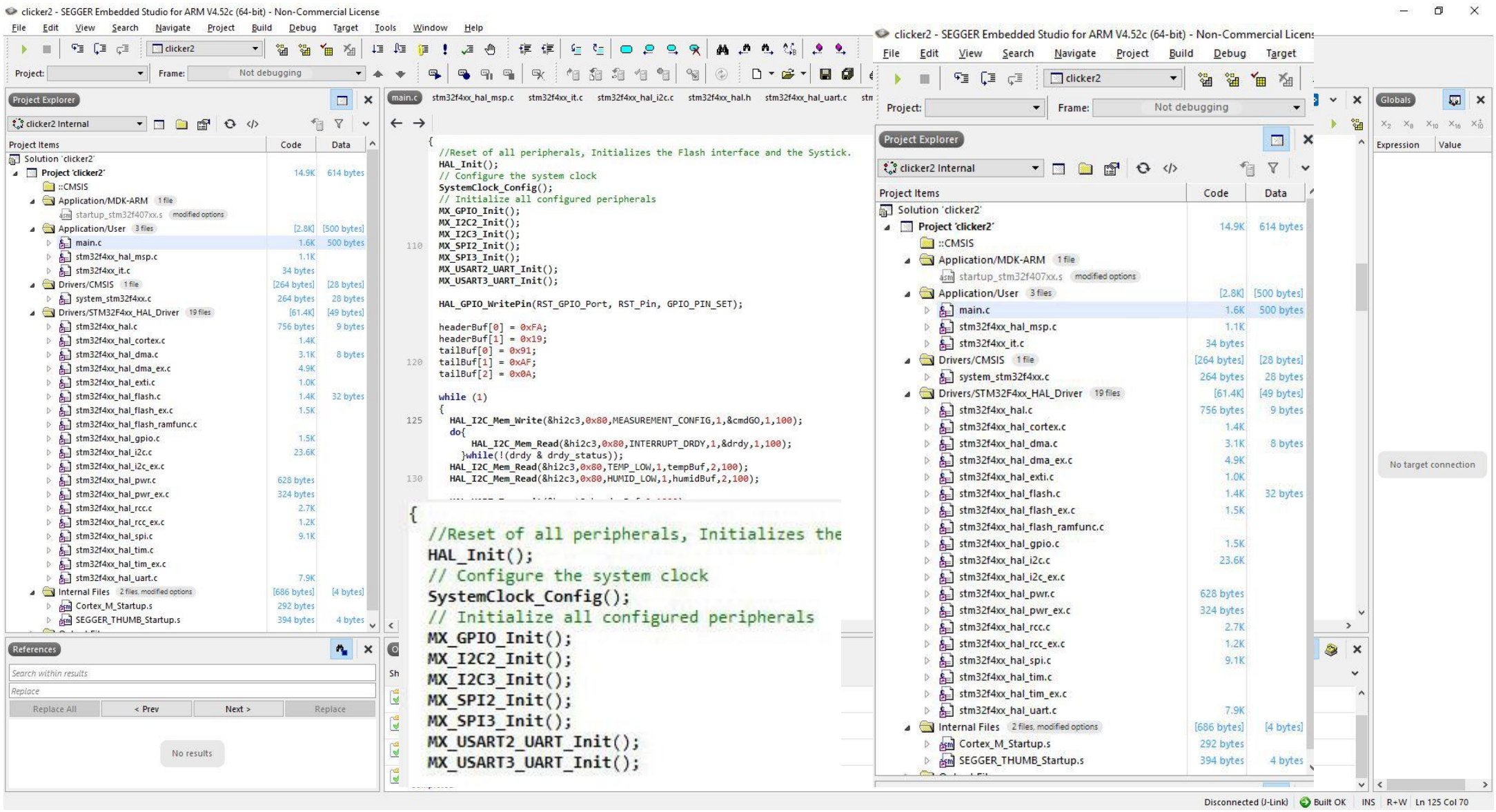
- Turn on your system sound to hear the streaming presentation.
- If you have technical problems, click “Help” or submit a question asking for assistance.
- Participate in ‘Group Chat’ by maximizing the chat widget in your dock.
- Submit questions for the lecturer using the Q&A widget. They will follow-up after the lecture portion concludes.

Fred Eady

Visit ‘Lecturer Profile’ in your console for more details.

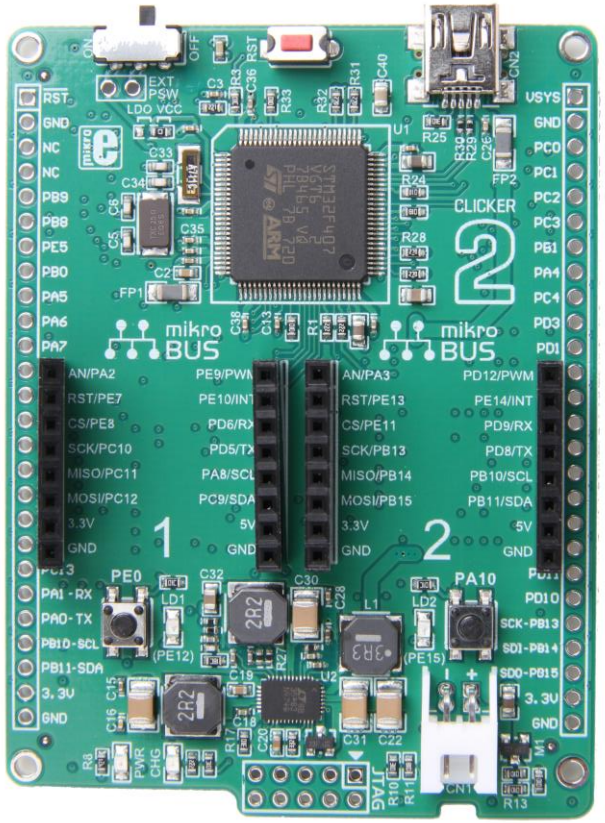
Embedded Studio Primer

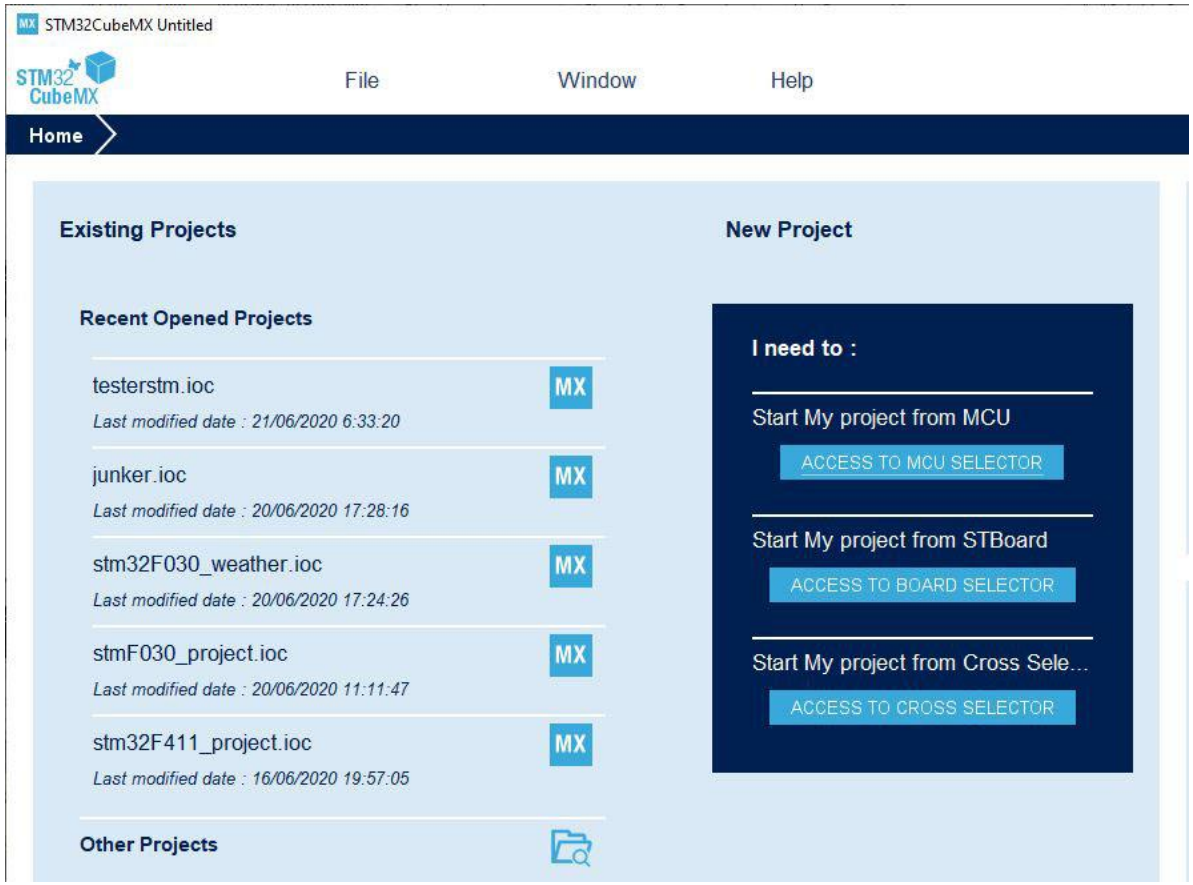
Embedded Studio and the STM32

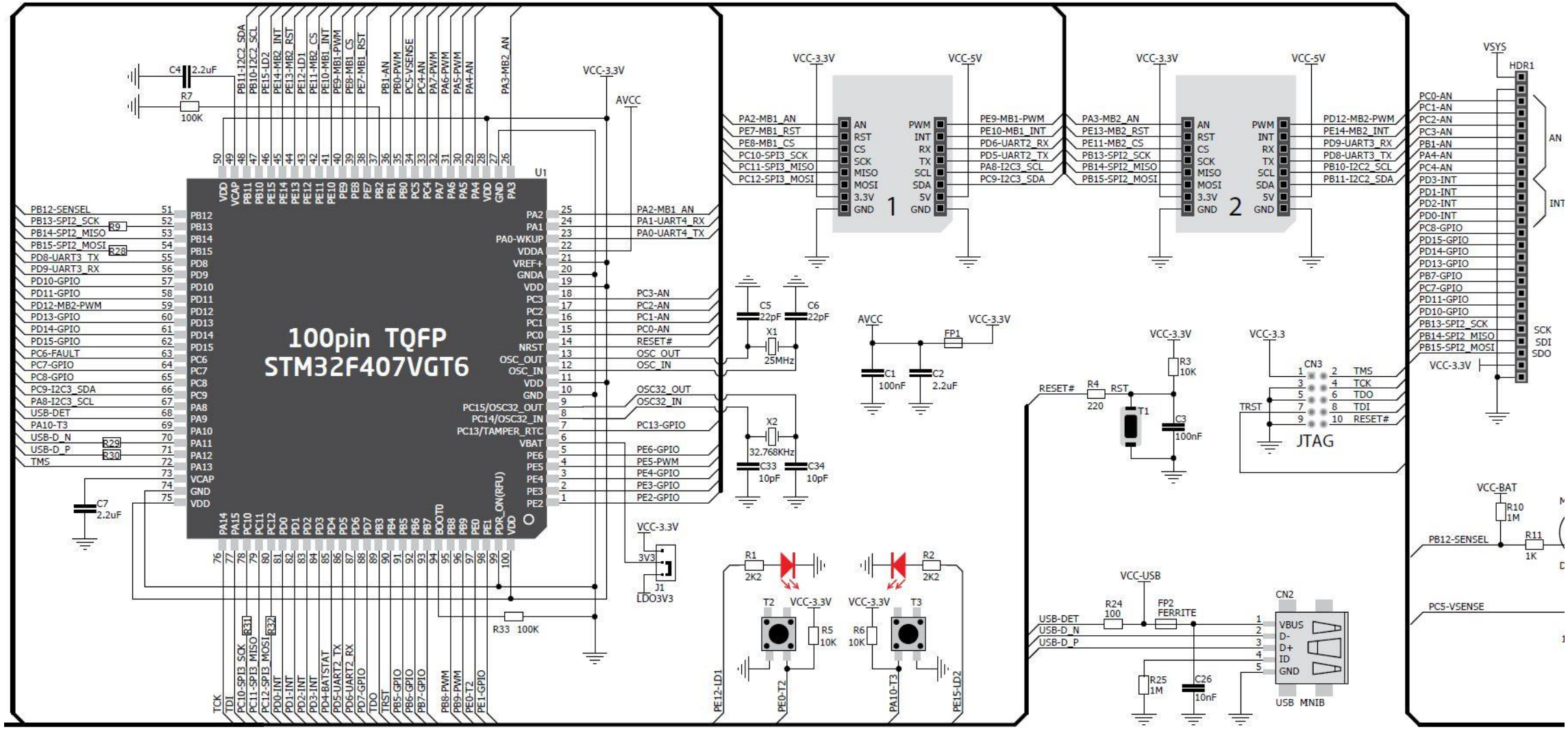


AGENDA

- Importing a Keil/STM32CubeMX Project
- Coding the Imported Keil Project
- Weather Over Wi-Fi

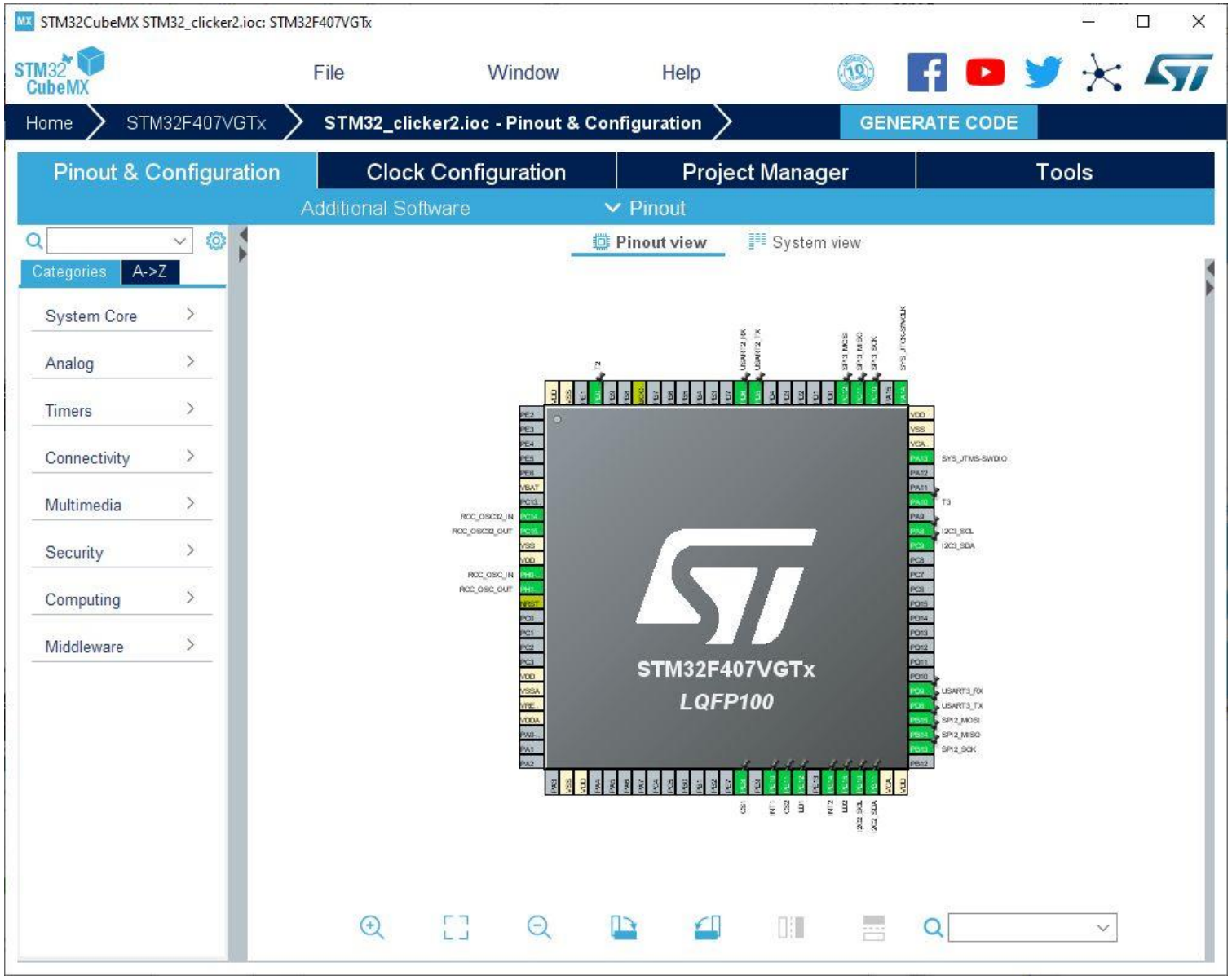


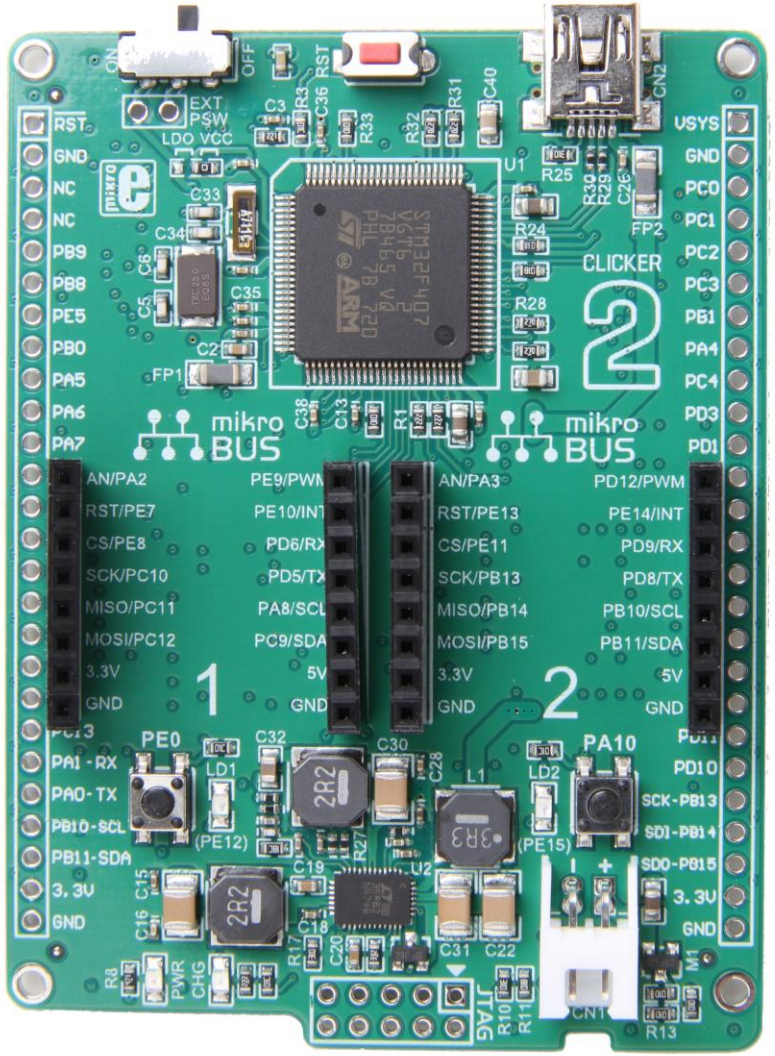



100pin TQFP
STM32F407VGT6
U1
VCC-3.3V
VCC-5V
AVCC
VSYS
HDR1
JTAG
CN3
CN2
USB MNIB
VCC-BAT
VCC-USB
C4 2.2uF
R7 100K
C5 22pF
C6 22pF
X1 25MHz
X2 32.768KHz
C33 10pF
C34 10pF
C1 100nF
C2 2.2uF
C7 2.2uF
FP1
R1 2K2
R2 2K2
R3 10K
R4 220
R5 10K
R6 10K
R10 1M
R11 1K
R24 100
R25 1M
C26 10nF
C3 100nF
FP2 FERRITE
T1
T2
T3
J1
LDO3V3
R33 100K
RESET#
TRST
TMS
TCK
TDI
TDO
SCK
SDI
SDO
PB12-SENSEL
PC5-VSENSE
USB-DET
USB-D_N
USB-D_P

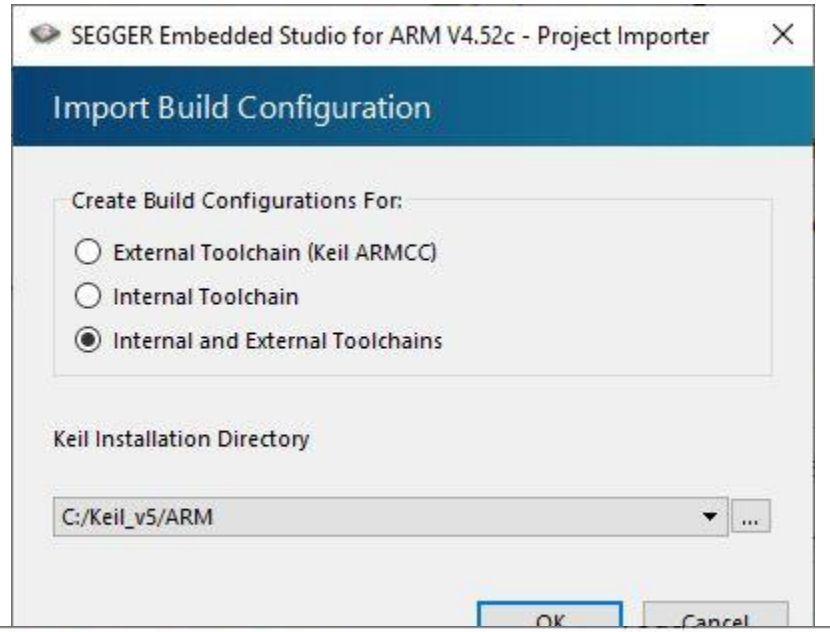
Embedded Studio Primer

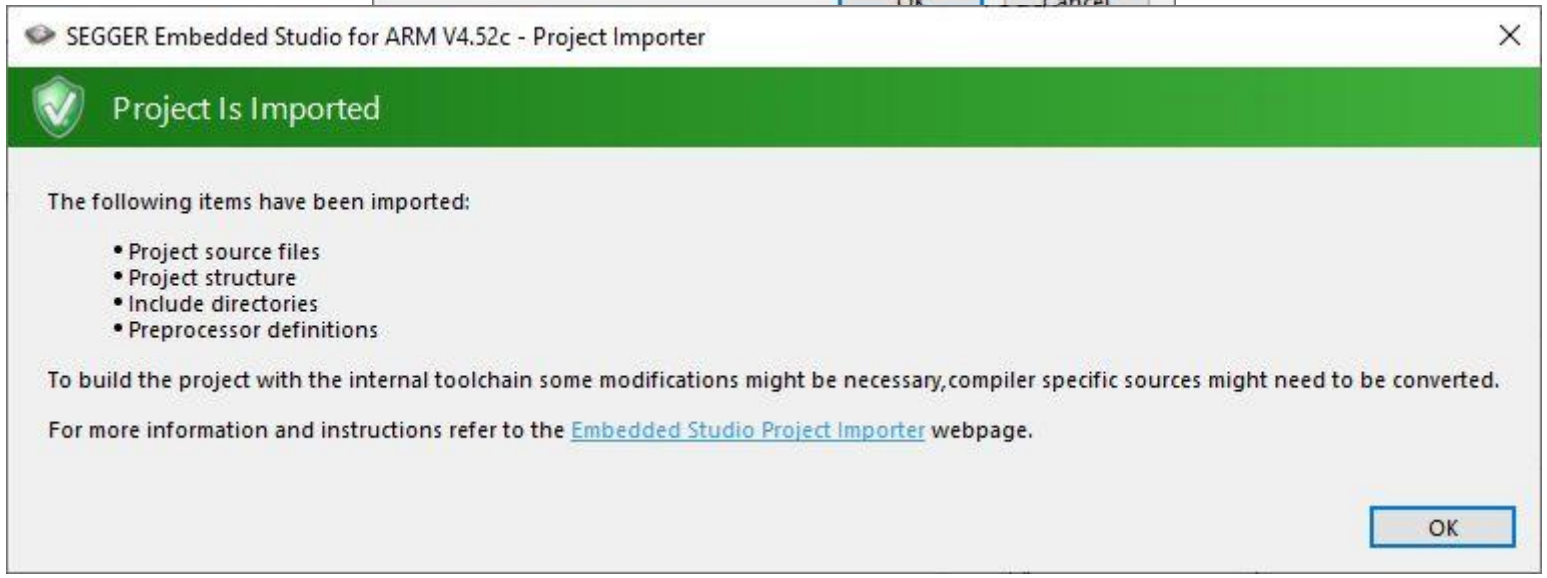
Importing a Keil/STM32CubeMX Project





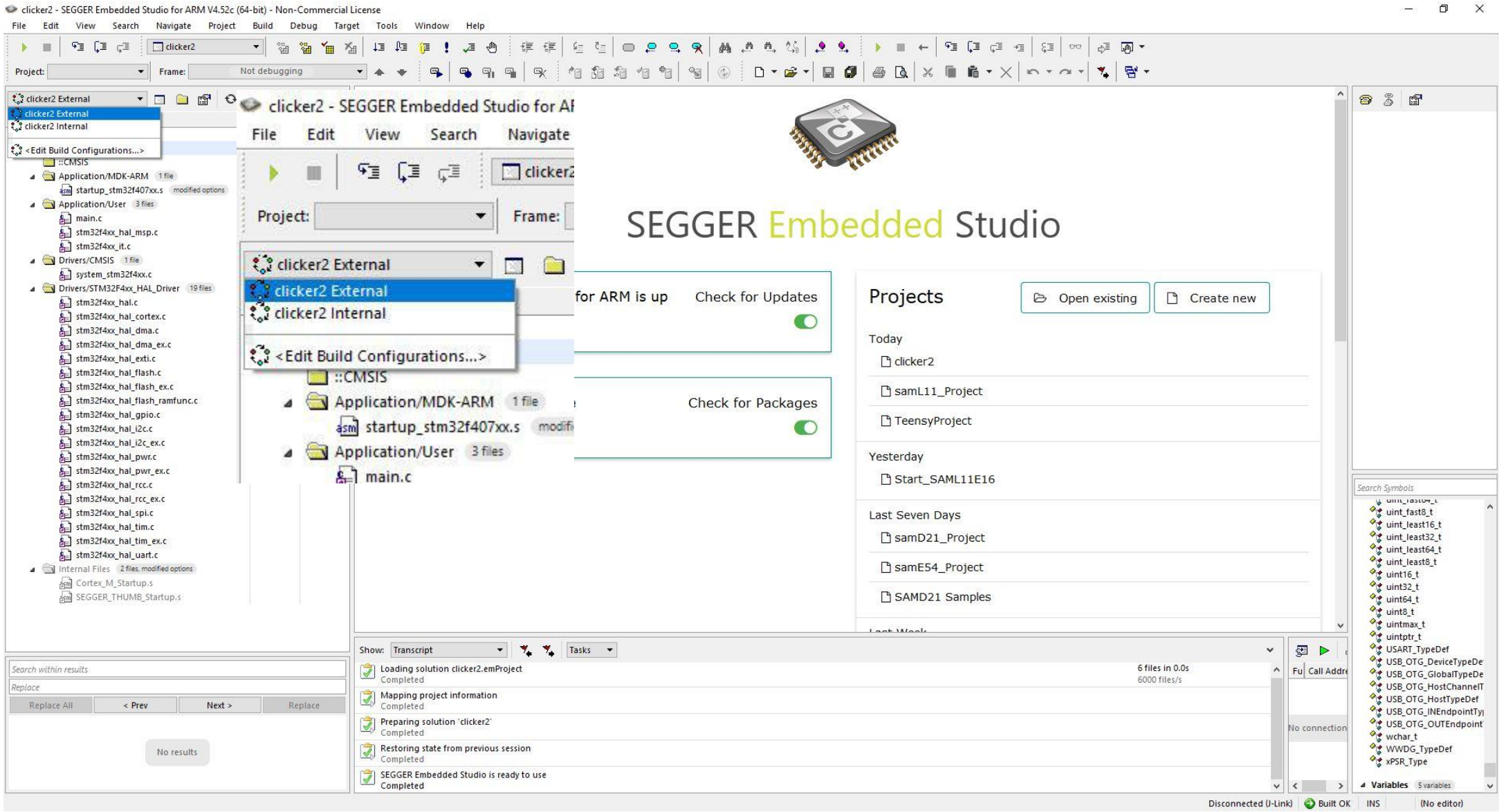
Importing a Keil/STM32CubeMX Project

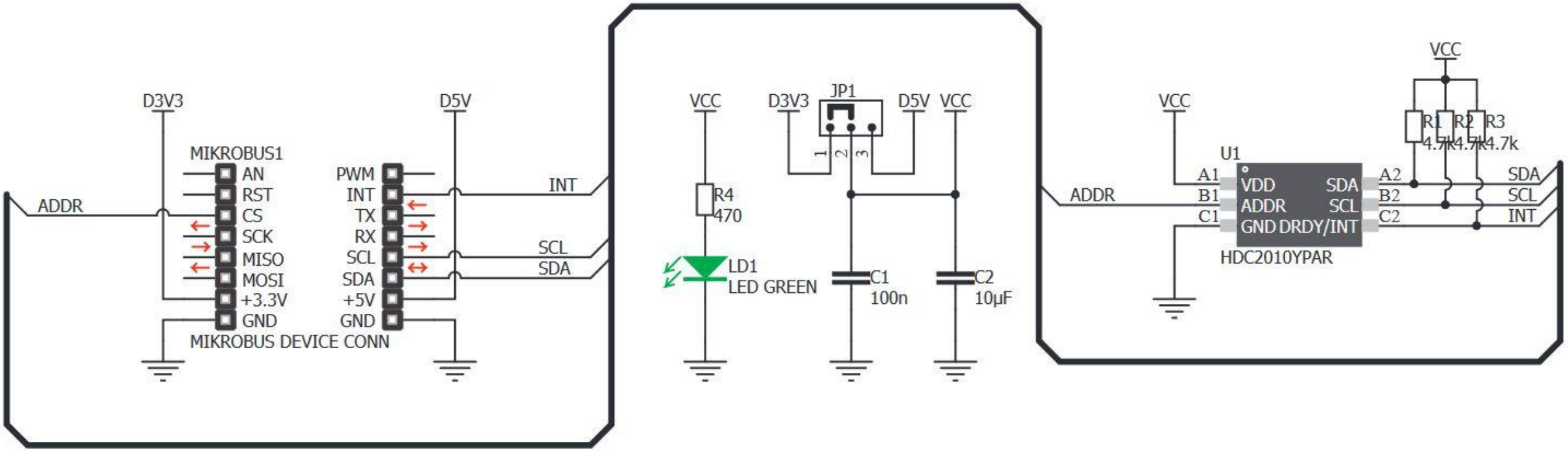




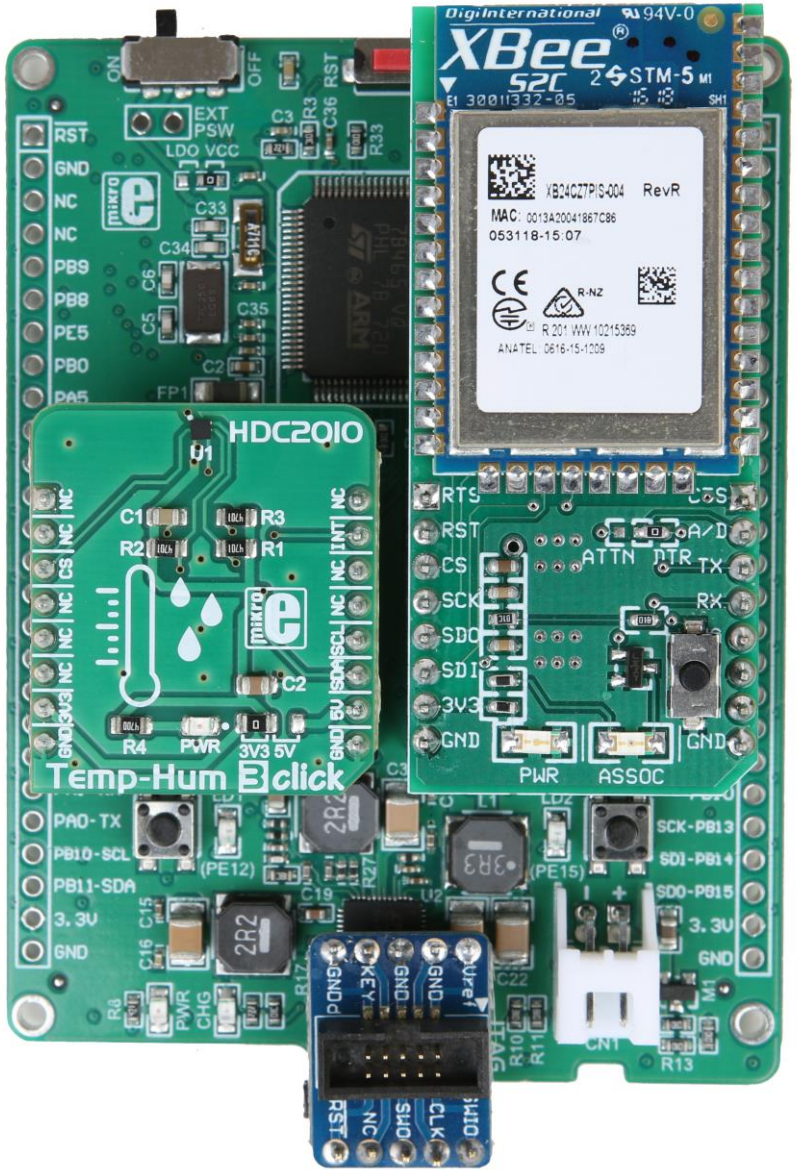
Embedded Studio Primer

Importing a Keil/STM32CubeMX Project




D3V3
D5V
MIKROBUS1
AN
RST
CS
SCK
MISO
MOSI
+3.3V
GND
PWM
INT
TX
RX
SCL
SDA
+5V
GND
MIKROBUS DEVICE CONN
ADDR
INT
SCL
SDA
VCC
R4
470
LD1
LED GREEN
D3V3
JP1
D5V
VCC
C1
100n
C2
10µF
VCC
U1
A1
B1
C1
VDD
ADDR
GND
SDA
SCL
DRDY/INT
A2
B2
C2
HDC2010YPAR
VCC
R1
R2
R3
4.7k
4.7k
4.7k
SDA
SCL
INT

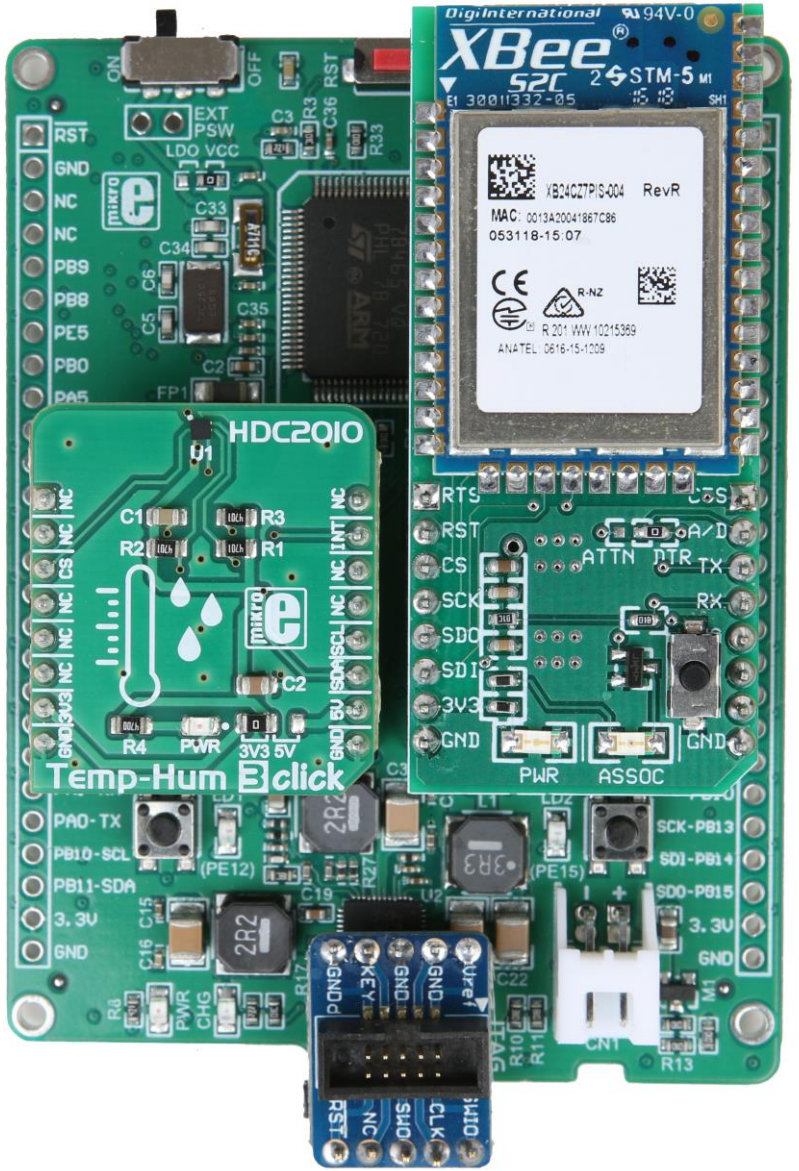
```
//*****************************************************************
//* HDC2010 REGISTERS
//*****************************************************************
#define TEMP_LOW 0x00
#define TEMP_HIGH 0x01
#define HUMID_LOW 0x02
#define HUMID_HIGH 0x03
#define INTERRUPT_DRDY 0x04
#define TEMP_MAX 0x05
#define HUMID_MAX 0x06
#define INTERRUPT_CONFIG 0x07
#define TEMP_OFFSET_ADJUST 0x08
#define HUM_OFFSET_ADJUST 0x09
#define TEMP_THR_L 0x0A
#define TEMP_THR_H 0x0B
#define HUMID_THR_L 0x0C
#define HUMID_THR_H 0x0D
#define CONFIG 0x0E
#define MEASUREMENT_CONFIG 0x0F
#define MID_L 0xFC
#define MID_H 0xFD
#define DEVICE_ID_L 0xFE
#define DEVICE_ID_H 0xFF
```



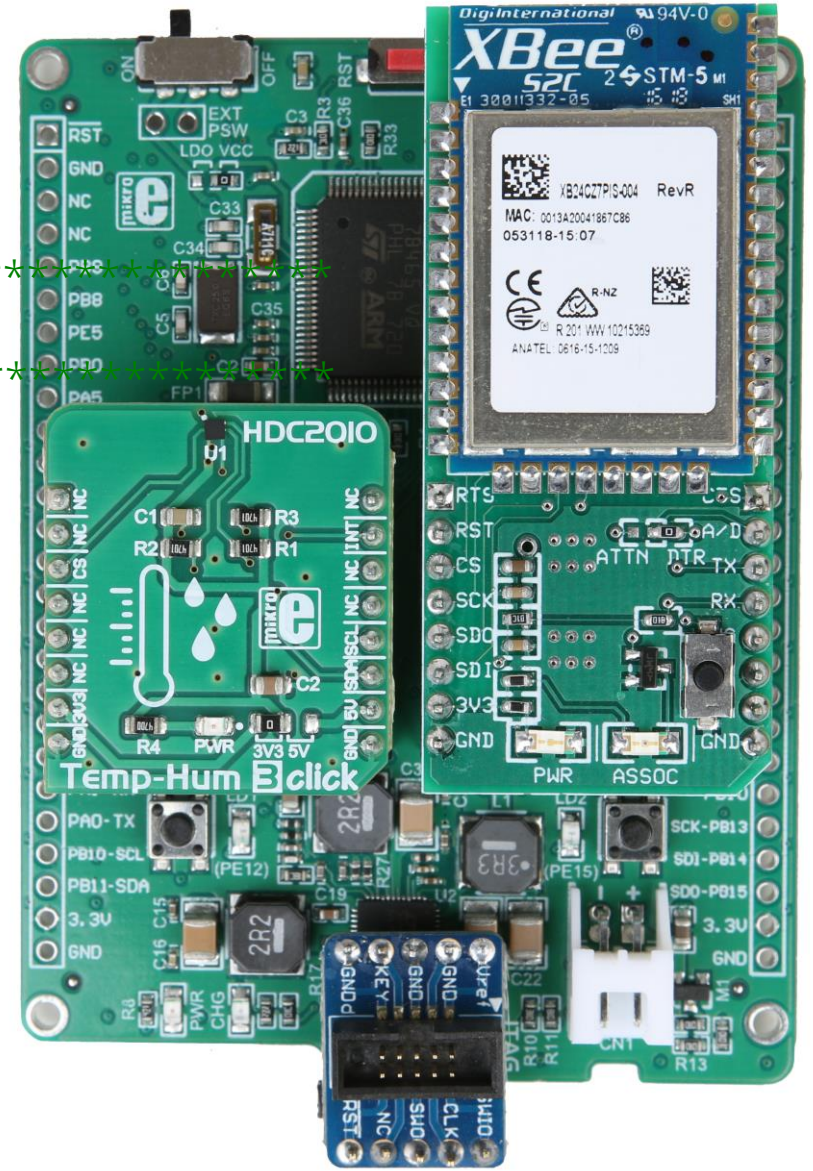
Embedded Studio Primer

Coding the Imported Keil Project


```
static void MX_USART3_UART_Init(void)
{
  huart3.Instance = USART3;
  huart3.Init.BaudRate = 9600;
  huart3.Init.WordLength = UART_WORDLENGTH_8B;
  huart3.Init.StopBits = UART_STOPBITS_1;
  huart3.Init.Parity = UART_PARITY_NONE;
  huart3.Init.Mode = UART_MODE_TX_RX;
  huart3.Init.HwFlowCtl = UART_HWCONTROL_NONE;
  huart3.Init.OverSampling = UART_OVERSAMPLING_16;
  if (HAL_UART_Init(&huart3) != HAL_OK)
  {
    Error_Handler();
  }
}
```



```
//**********************************************************************
//* PRINTF KEIL
//**********************************************************************
int fputc(int ch, FILE *f)
{
      HAL_UART_Transmit(&huart3, (uint8_t *)&ch, 1, 0xFFFF);
      return ch;
}
```



```
//*****************************************************
//* PRINTF STUFF
//*****************************************************
typedef struct __printf_tag
{
  size_t charcount;
  size_t maxchars;
  char *string;
  int (*output_fn)(int, struct __printf_tag *ctx);
} __printf_t;

void uart3_putc(char ch)
{
  HAL_UART_Transmit(&huart3, (uint8_t *)&ch, 1, 0xFFFF);
}

int uart3_printf(const char *fmt, ...)
{
  int n;
  va_list ap;
  __printf_t iod;
  va_start(ap, fmt);
  iod.string = 0;
  iod.maxchars = INT_MAX;
  iod.output_fn = uart3_putc;
  n = __vfprintf(&iod, fmt, ap);
  va_end(ap);
  return n;
}
```

SEGGER Embedded Studio for ARM printf-style output

SEGGER Embedded Studio for ARM provides a solution for just this case by using some internal functions and data types in the SEGGER Embedded Studio for ARM library. These functions and types are define in the header file <__vfprintf.h>.

The first thing to introduce is the **__printf_t** type which captures the current state and parameters of the format conversion:

```
typedef struct __printf_tag
{
  size_t charcount;
  size_t maxchars;
  char *string;
  int (*output_fn)(int, struct __printf_tag *ctx);
} __printf_t;
```

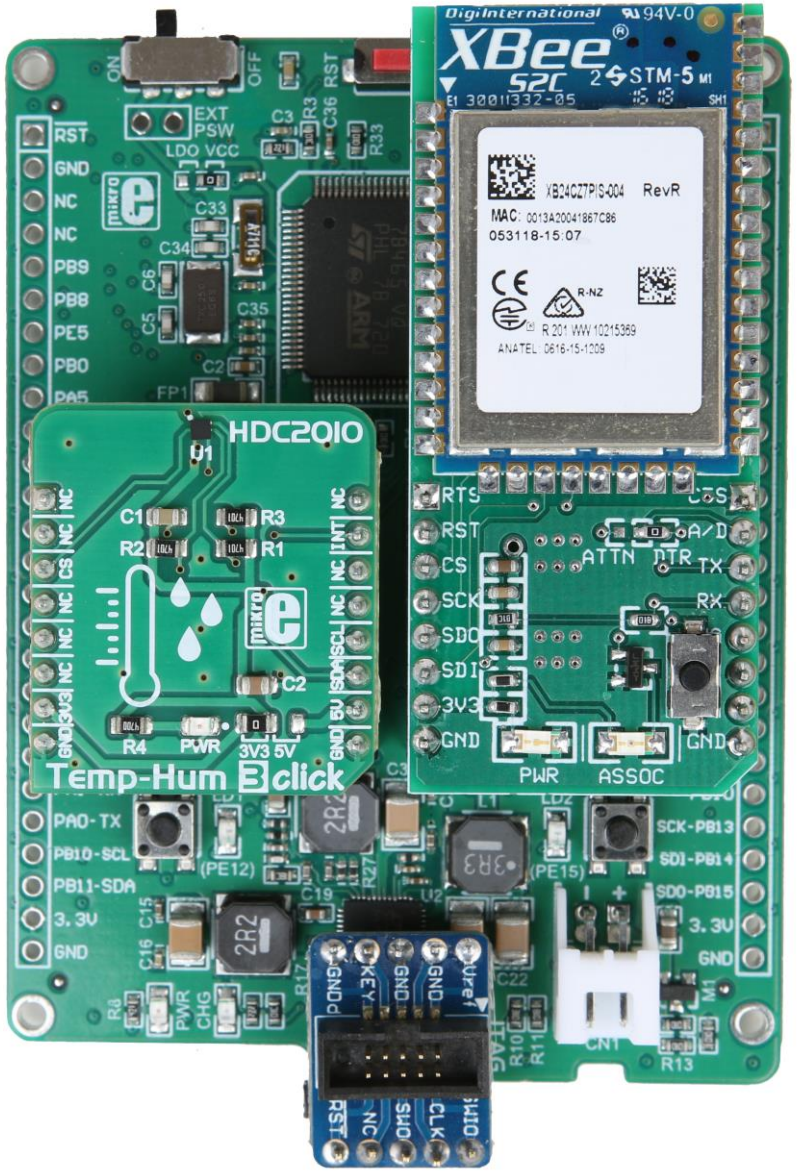
This type is used by the library functions to direct what the formatting routines do with each character they need to output. If `string` is non-zero, the character is appended is appended to the string pointed to by **string**; if `output_fn` is non-zero, the character is output through the function **output_fn** with the context passed as the second parameter.

The member **charcount** counts the number of characters currently output, and **maxchars** defines the maximum number of characters output by the formatting routine **__vfprintf**.

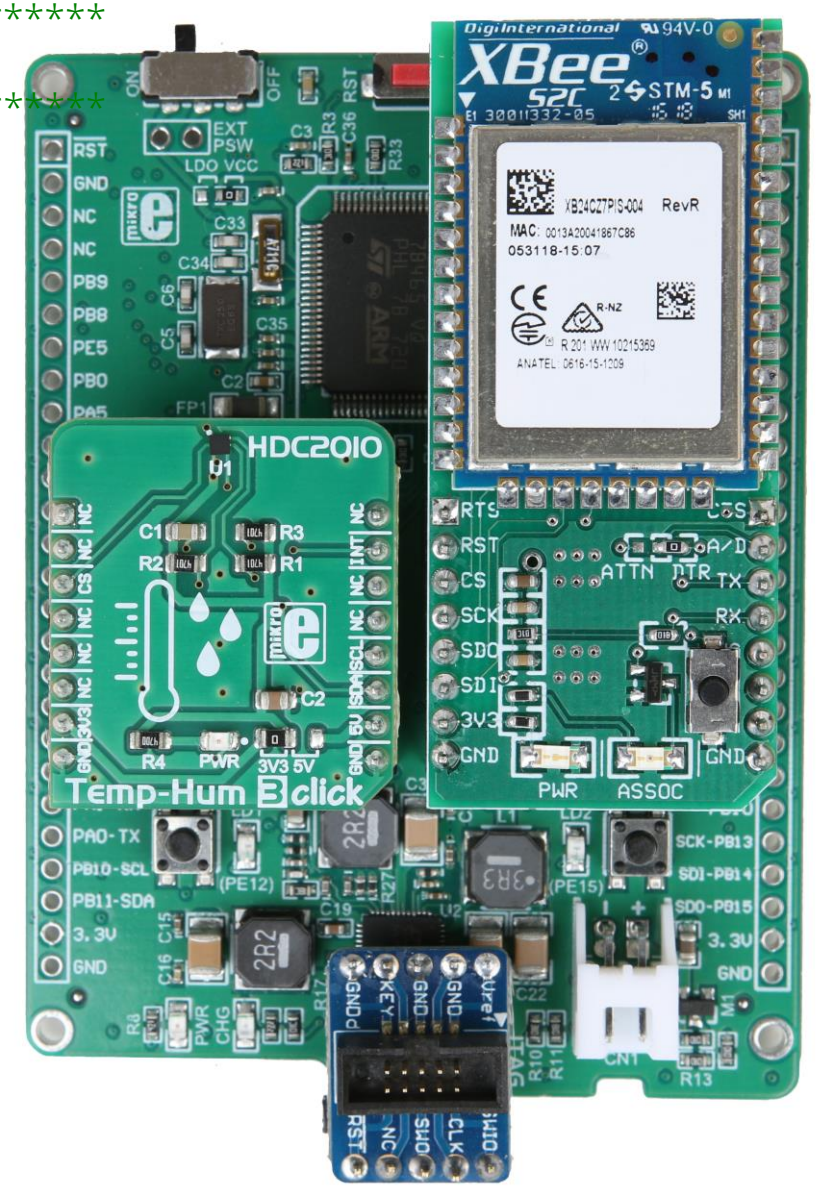
We can use this type and function to rewrite **uart0_printf**:

```
int uart0_printf(const char *fmt, ...)
{
  int n;
  va_list ap;
  __printf_t iod;
  va_start(ap, fmt);
  iod.string = 0;
  iod.maxchars = INT_MAX;
  iod.output_fn = uart0_putc;
  n = __vfprintf(\&iod, fmt, ap);
  va_end(ap);
  return n;
}
```

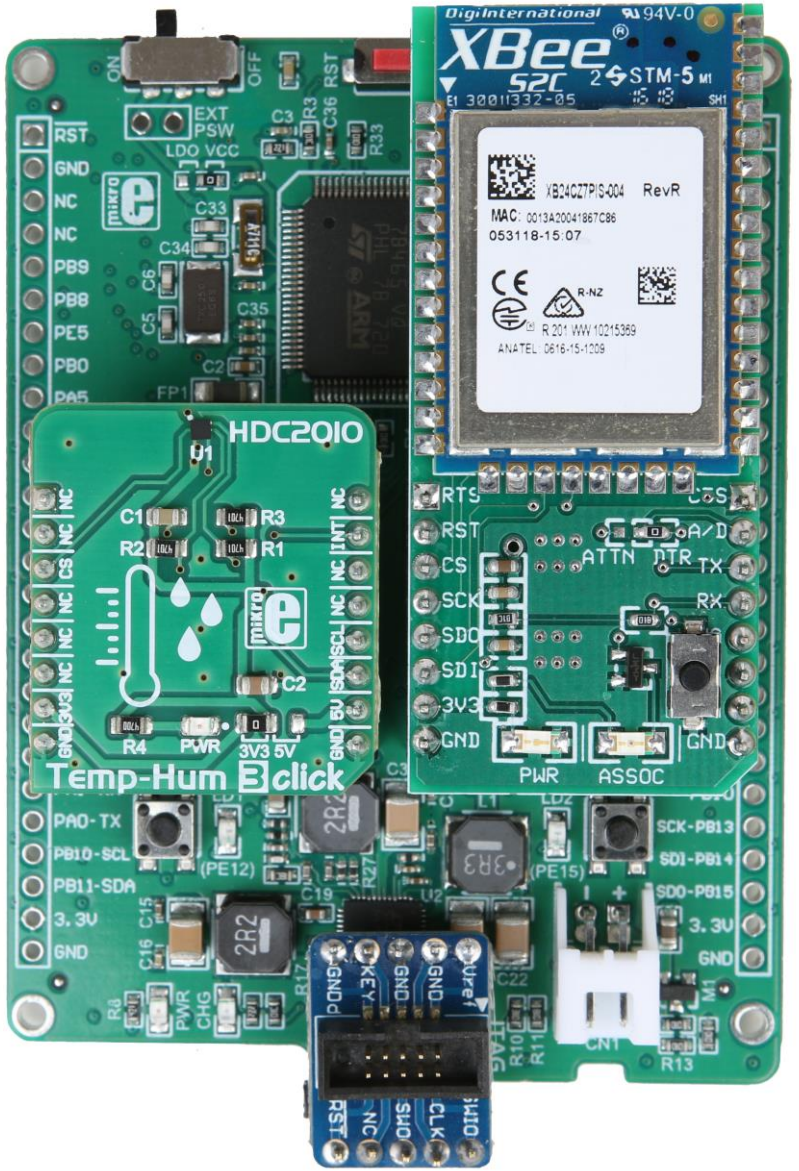
```
static void MX_I2C2_Init(void)
{
  hi2c2.Instance = I2C2;
  hi2c2.Init.ClockSpeed = 100000;
  hi2c2.Init.DutyCycle = I2C_DUTYCYCLE_2;
  hi2c2.Init.OwnAddress1 = 0;
  hi2c2.Init.AddressingMode = I2C_ADDRESSINGMODE_7BIT;
  hi2c2.Init.DualAddressMode = I2C_DUALADDRESS_DISABLE;
  hi2c2.Init.OwnAddress2 = 0;
  hi2c2.Init.GeneralCallMode = I2C_GENERALCALL_DISABLE;
  hi2c2.Init.NoStretchMode = I2C_NOSTRETCH_DISABLE;
  if (HAL_I2C_Init(&hi2c2) != HAL_OK)
  {
    Error_Handler();
  }
}
```



```
//*****************************************************************************
//* MAIN
//*****************************************************************************
int main(void)
{
  //Reset of all peripherals, Initializes the Flash interface and the Systick.
  HAL_Init();
  // Configure the system clock
  SystemClock_Config();
  // Initialize all configured peripherals
  MX_GPIO_Init();
  MX_I2C2_Init();
  MX_I2C3_Init();
  MX_SPI2_Init();
  MX_SPI3_Init();
  MX_USART2_UART_Init();
  MX_USART3_UART_Init();
  //Drive XBee RESET pin HI
  HAL_GPIO_WritePin(RST_GPIO_Port, RST_Pin, GPIO_PIN_SET);
  //packet header and footer
  headerBuf[0] = 0xFA;
  headerBuf[1] = 0x19;
  tailBuf[0] = 0x91;
  tailBuf[1] = 0xAF;
  tailBuf[2] = 0x0A;
```



```
while (1)
{
  HAL_I2C_Mem_Write(&hi2c3,0x80,MEASUREMENT_CONFIG,1,&cmdGO,1,100);
  do{
     HAL_I2C_Mem_Read(&hi2c3,0x80,INTERRUPT_DRDY,1,&drdy,1,100);
   }while(!(drdy & drdy_status));
  HAL_I2C_Mem_Read(&hi2c3,0x80,TEMP_LOW,1,tempBuf,2,100);
  HAL_I2C_Mem_Read(&hi2c3,0x80,HUMID_LOW,1,humidBuf,2,100);

  HAL_UART_Transmit(&huart3,headerBuf,2,1000);
  HAL_UART_Transmit(&huart3,tempBuf,2,1000);
  HAL_UART_Transmit(&huart3,humidBuf,2,1000);
  HAL_UART_Transmit(&huart3,tailBuf,3,1000);

  tempVal = make16(tempBuf[1],tempBuf[0]);
  humidVal = make16(humidBuf[1],humidBuf[0]);
  tempC = (((float)tempVal/65536)*165)-40;
  tempF = ((float)tempC * 1.8)+32;
  humidity = ((float)humidVal/65536)*100;

  uart3_printf("\r\nTemperature C = %3.2f\r\n",tempC);
  HAL_Delay(10);
  uart3_printf("Temperature F = %3.2f\r\n",tempF);
  HAL_Delay(10);
  uart3_printf("Humidity = %3.2f%%\r\n\r\n",humidity);
  HAL_Delay(1000);
 }
}
```

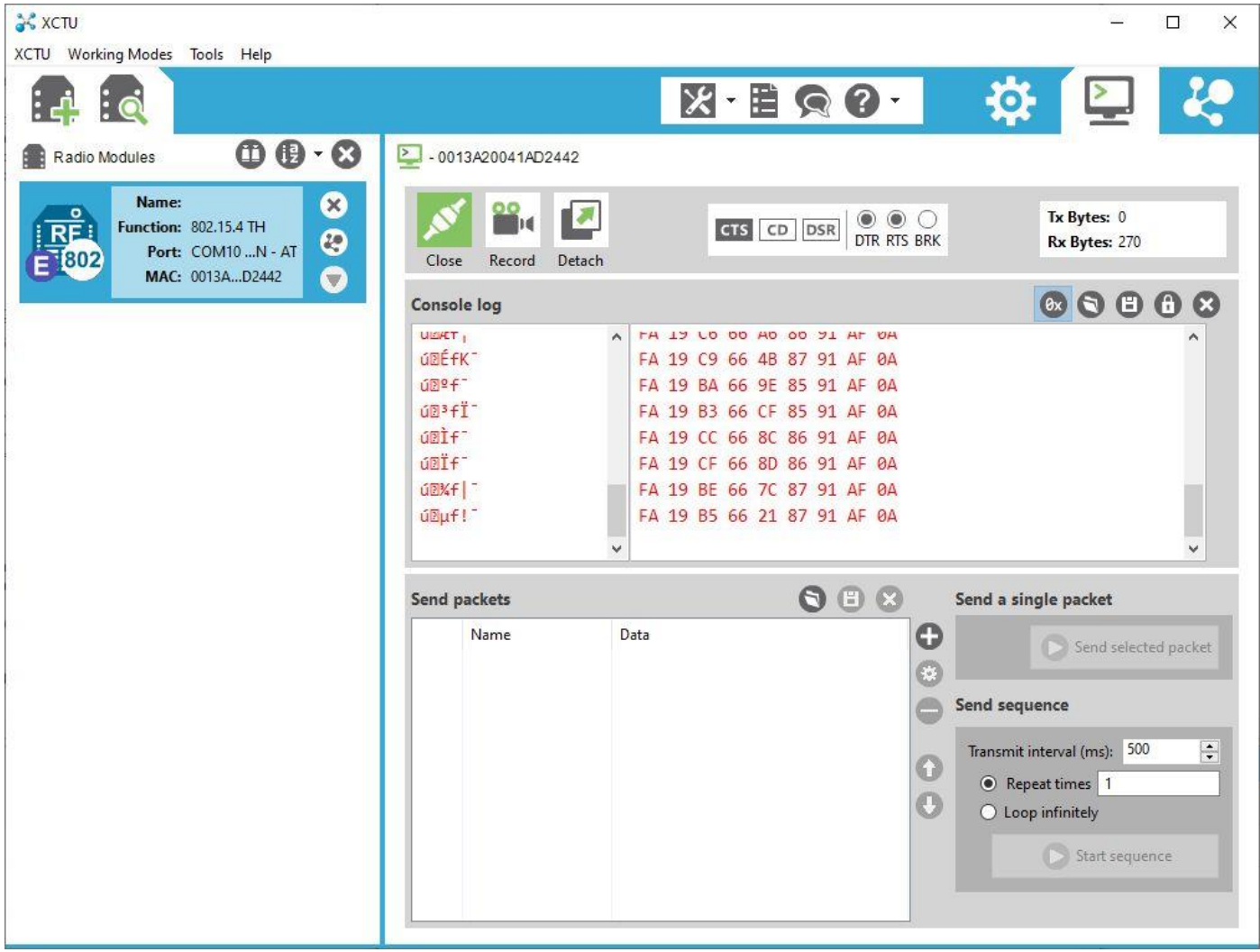



LAN
USB
j-link Pro
SEGGER
www.segger.com
Target
XBee
HDC2010
Temp-Hum Click
PWR
ASSOC

Embedded Studio Primer

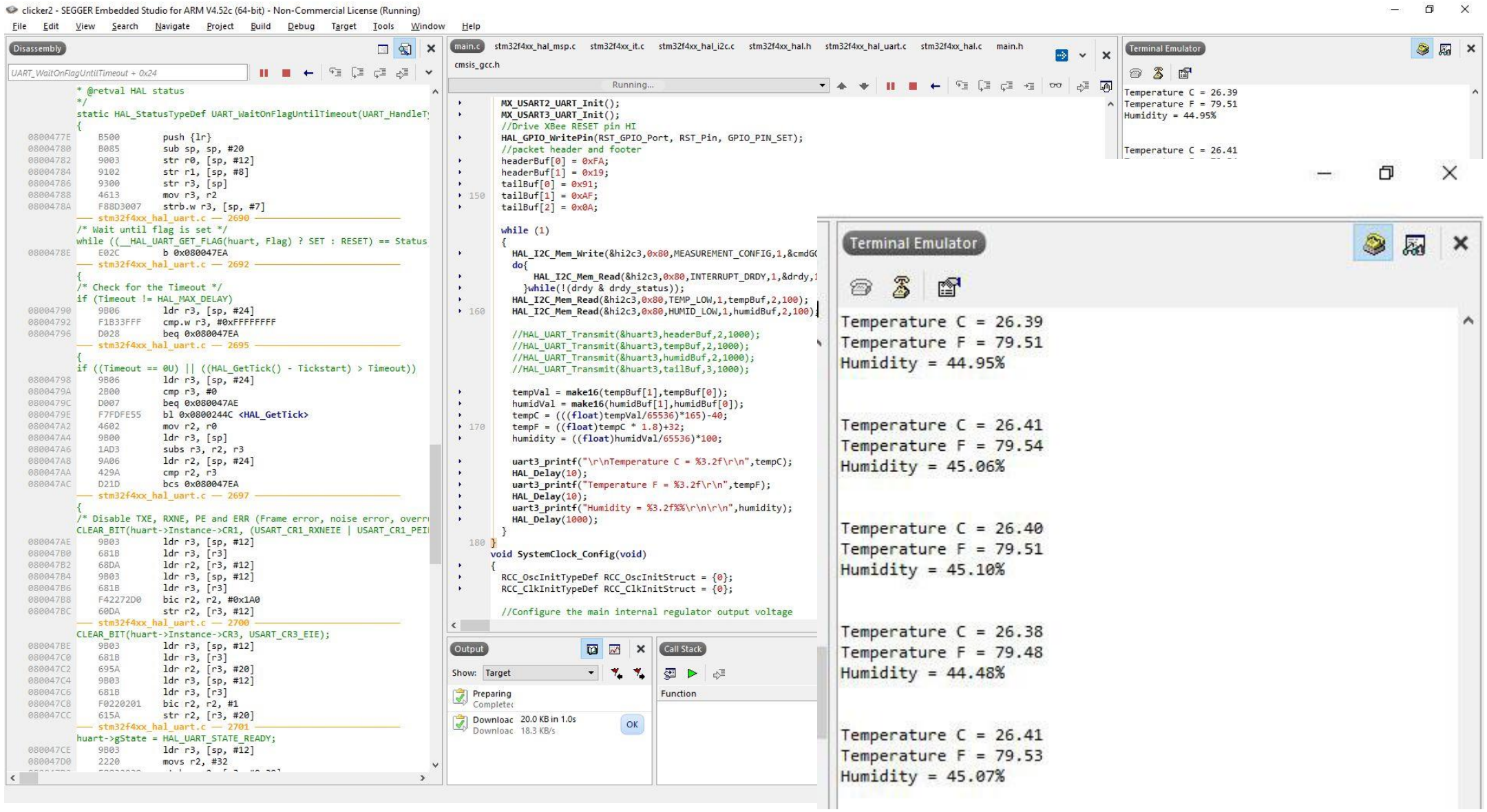
Coding the Imported Keil Project


```
while (1)
 {
  HAL_I2C_Mem_Write(&hi2c3,0x80,MEASUREMENT_CONFIG,1,&cmdGO,1,100);
  do{
     HAL_I2C_Mem_Read(&hi2c3,0x80,INTERRUPT_DRDY,1,&drdy,1,100);
   }while(!(drdy & drdy_status));
  HAL_I2C_Mem_Read(&hi2c3,0x80,TEMP_LOW,1,tempBuf,2,100);
  HAL_I2C_Mem_Read(&hi2c3,0x80,HUMID_LOW,1,humidBuf,2,100);

  HAL_UART_Transmit(&huart3,headerBuf,2,1000);
  HAL_UART_Transmit(&huart3,tempBuf,2,1000);
  HAL_UART_Transmit(&huart3,humidBuf,2,1000);
  HAL_UART_Transmit(&huart3,tailBuf,3,1000);

  tempVal = make16(tempBuf[1],tempBuf[0]);
  humidVal = make16(humidBuf[1],humidBuf[0]);
  tempC = (((float)tempVal/65536)*165)-40;
  tempF = ((float)tempC * 1.8)+32;
  humidity = ((float)humidVal/65536)*100;

  //uart3_printf("\r\nTemperature C = %3.2f\r\n",tempC);
  //HAL_Delay(10);
  //uart3_printf("Temperature F = %3.2f\r\n",tempF);
  //HAL_Delay(10);
  //uart3_printf("Humidity = %3.2f%%\r\n\r\n",humidity);
  HAL_Delay(1000);
 }
```



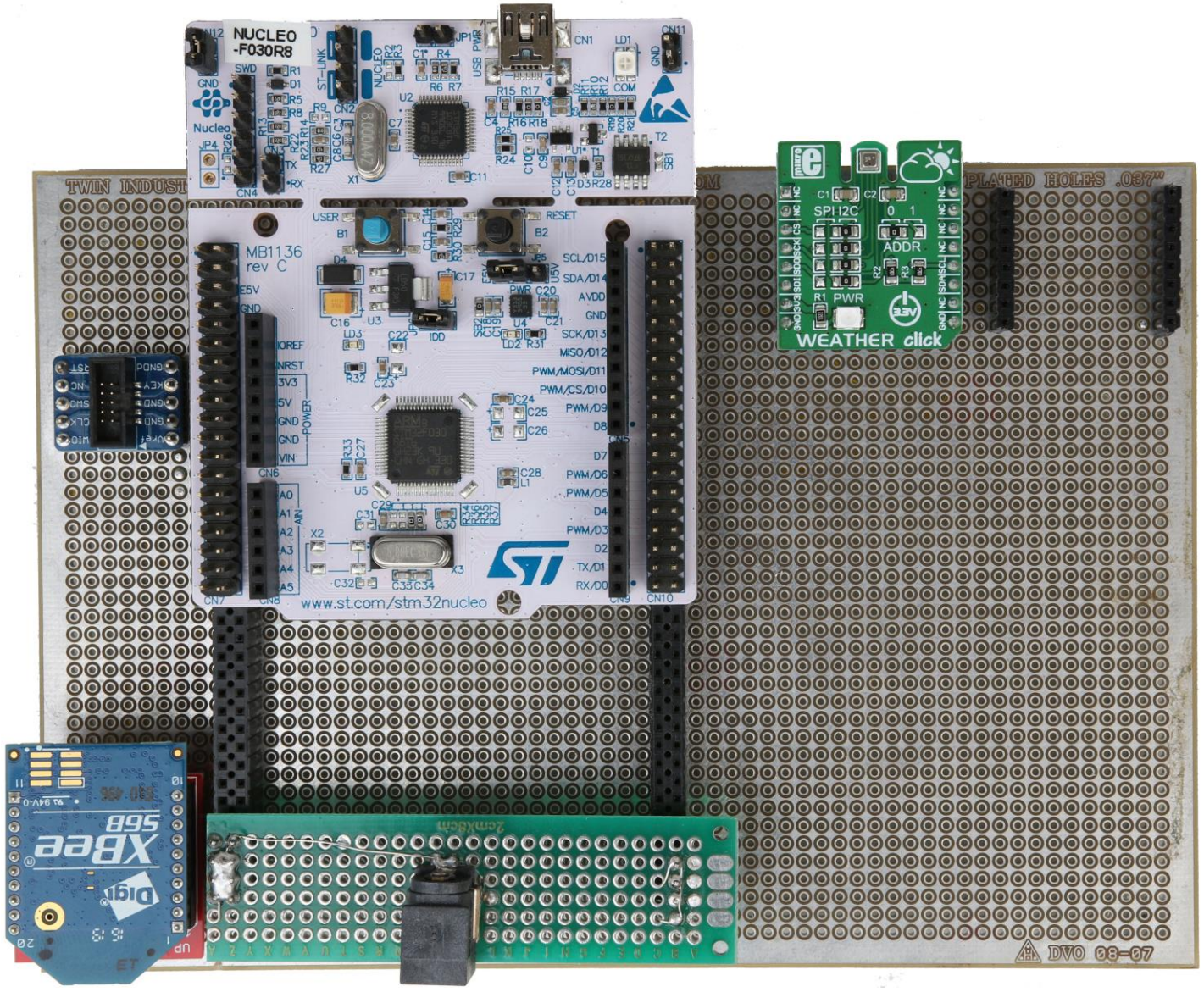
Embedded Studio Primer

Coding the Imported Keil Project



Embedded Studio Primer

Weather Over Wi-Fi

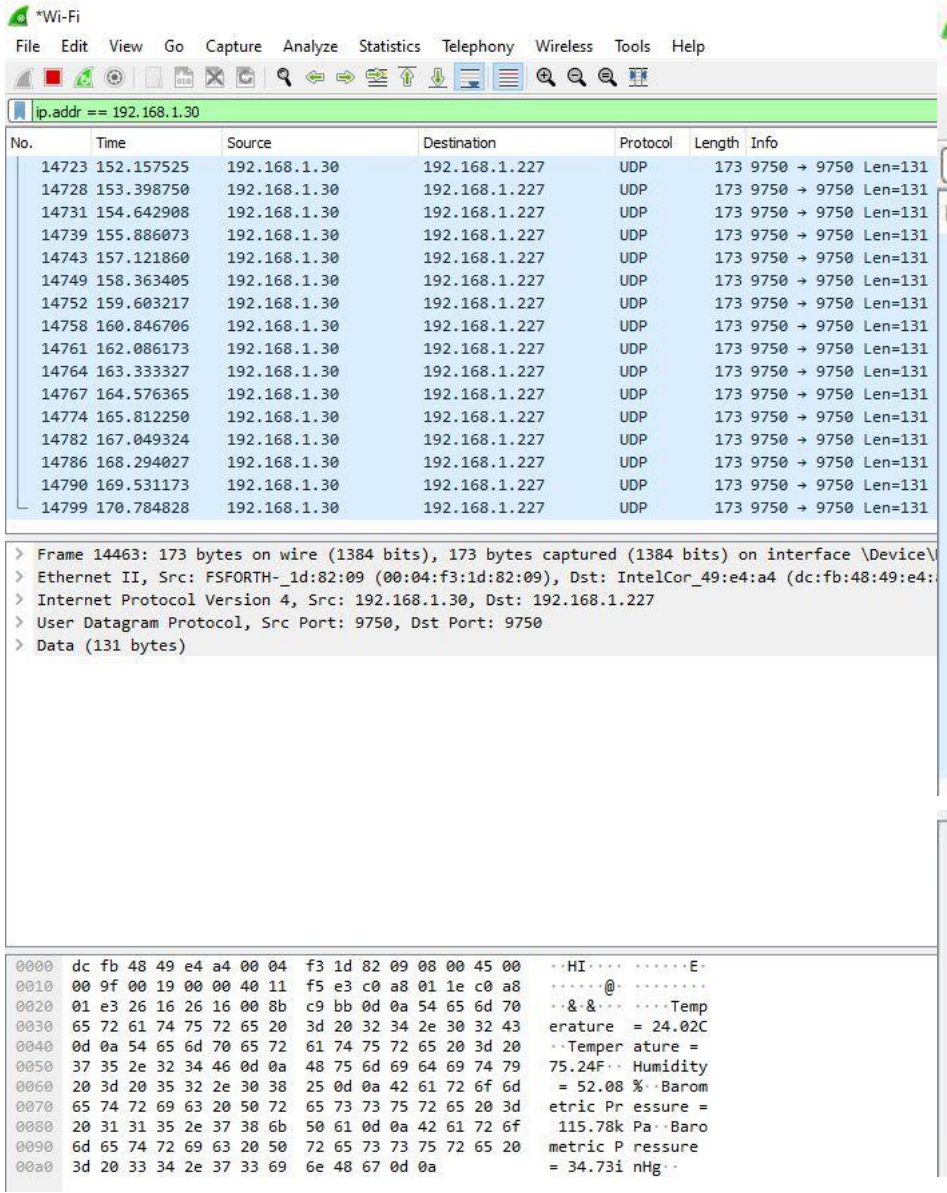


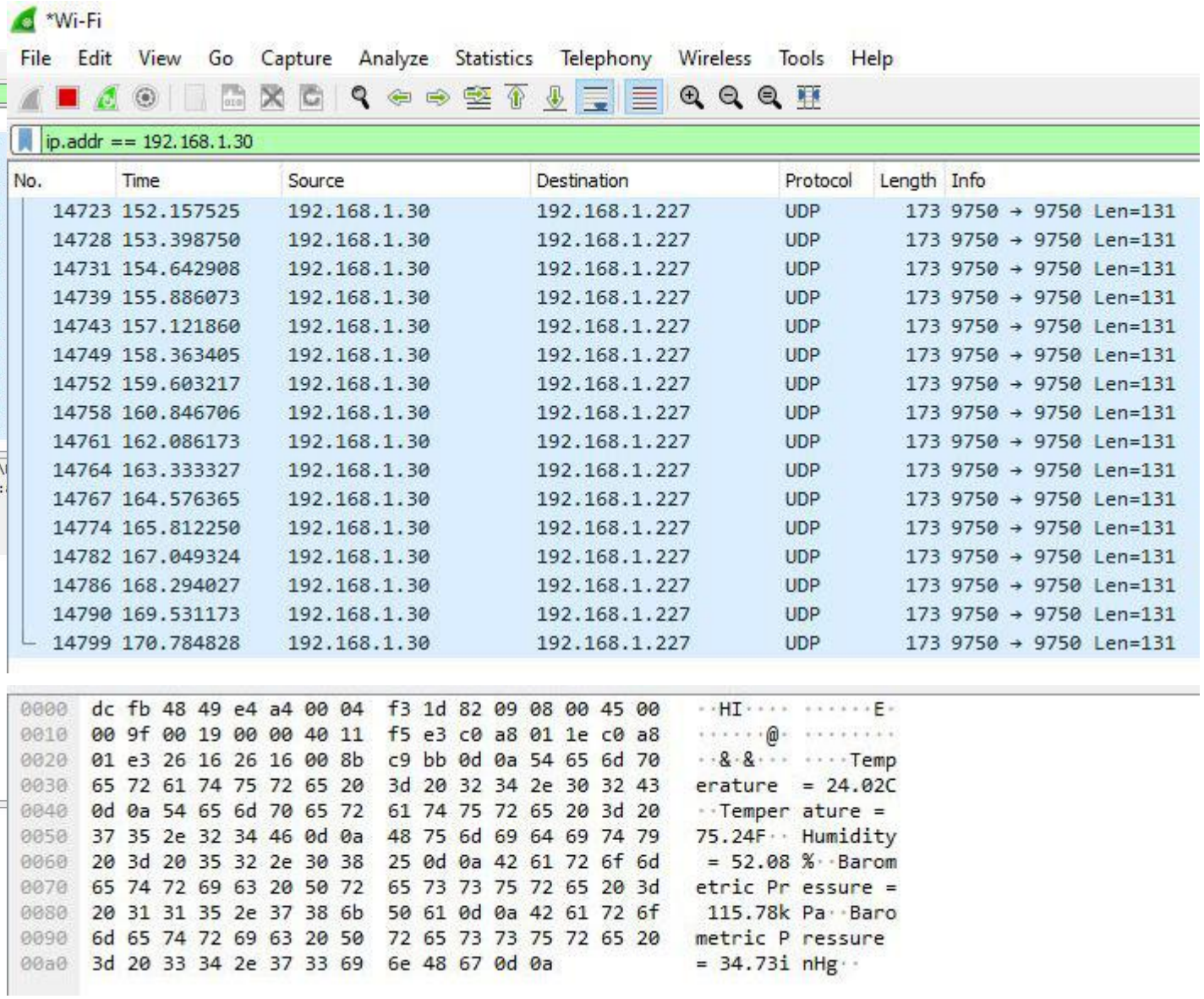
```
SysTick_Config(SystemCoreClock/1000);
 Weather_initializeClick();
 while (1)
 {
   Weather_readSensors();
   tc = BME280_getTemperature();
   tf = (tc * 1.8) + 32;
   th = BME280_getHumidity();
   tk = BME280_getPressure();
   tm = tk * 0.30;
   uart2_printf("\r\nTemperature = %3.2fC\r\n",tc);
   uart2_printf("Temperature = %3.2fF\r\n",tf);
   uart2_printf("Humidity = %3.2f%%\r\n",th);
   uart2_printf("Barometric Pressure = %3.2fkPa\r\n",tk);
   uart2_printf("Barometric Pressure = %3.2finHg\r\n",tm);
   HAL_Delay(100);
   HAL_GPIO_WritePin(LEDGRN_GPIO_Port, LEDGRN_Pin, GPIO_PIN_RESET);
   HAL_Delay(500);
   HAL_GPIO_WritePin(LEDGRN_GPIO_Port, LEDGRN_Pin, GPIO_PIN_SET);
   HAL_Delay(500);
 }
```

```
//*****************************************************************
//* PRINTF STUFF
//*****************************************************************
typedef struct __printf_tag
{
  size_t charcount;
  size_t maxchars;
  char *string;
  int (*output_fn)(int, struct __printf_tag *ctx);
} __printf_t;

void uart2_putc(char ch)
{
  HAL_UART_Transmit(&huart2, (uint8_t *)&ch, 1, 0xFFFF);
}

int uart2_printf(const char *fmt, ...)
{
  int n;
  va_list ap;
  __printf_t iod;
  va_start(ap, fmt);
  iod.string = 0;
  iod.maxchars = INT_MAX;
  iod.output_fn = uart2_putc;
  n = __vfprintf(&iod, fmt, ap);
  va_end(ap);
  return n;
}
```

Embedded Studio Primer

Weather Over Wi-Fi





Thank you for attending

Please consider the resources below:

- https://www.mikroe.com
- https://www.st.com/en/development-tools/stm32cubemx.html
- https://www.digi.com

CEC Continuing Education Center

DesignNews

Thank You

Sponsored by

Digi-Key ELECTRONICS

informa markets