CEC Continuing Education Center

DesignNews

C++ Primer for Embedded Systems

Day 4:

Embedded C++: Interfacing with the STM32 HAL

Sponsored by

DigiKey

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 ‘Attendee Chat’ by maximizing the chat widget in your dock.

Fred Eady

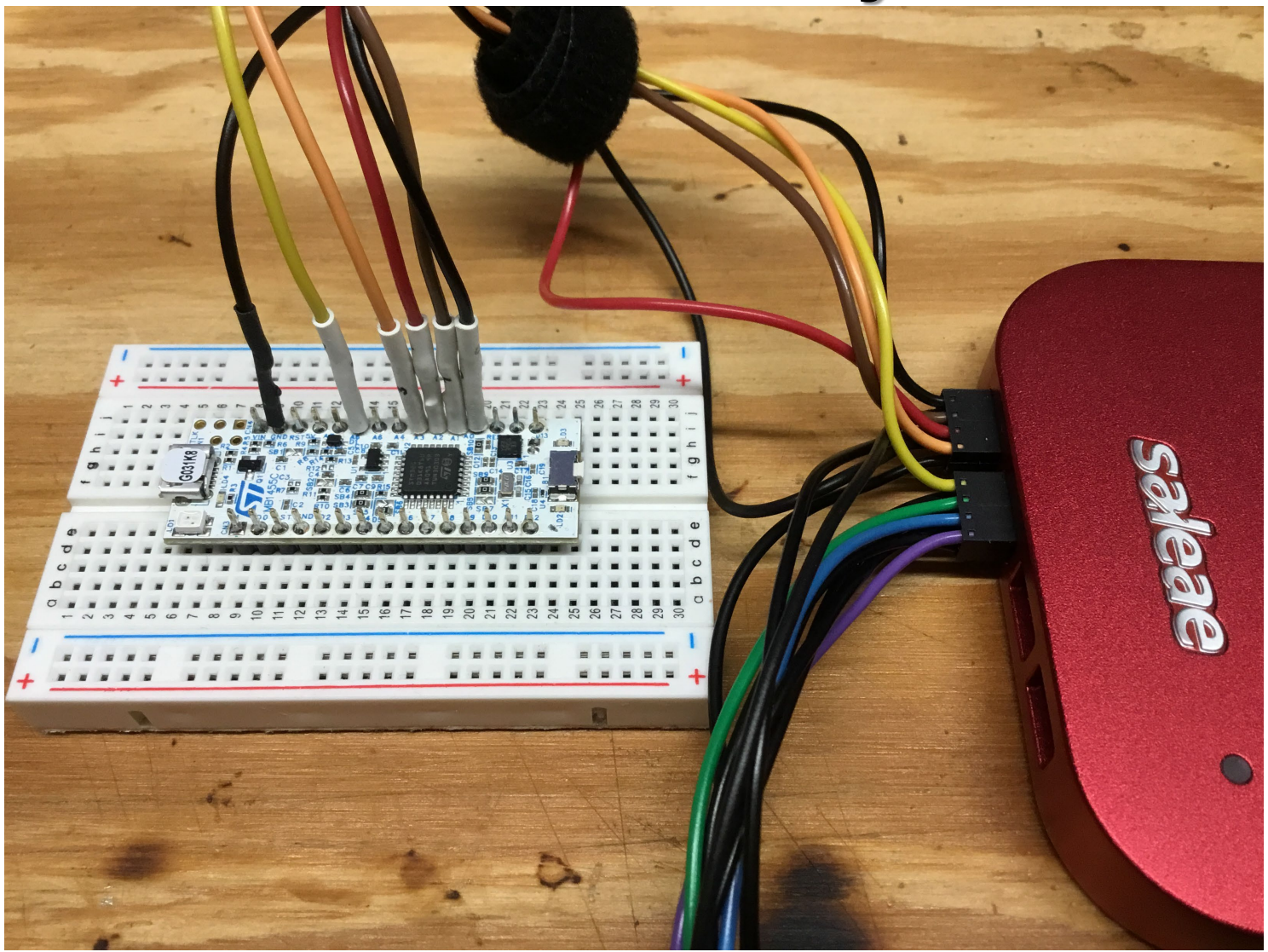
Visit ‘Lecturer Profile’ in your console for more details.

CEC Continuing Education Center

AGENDA

DigiKey

- **Create an STMCubeIDE C++ GPIO Project**



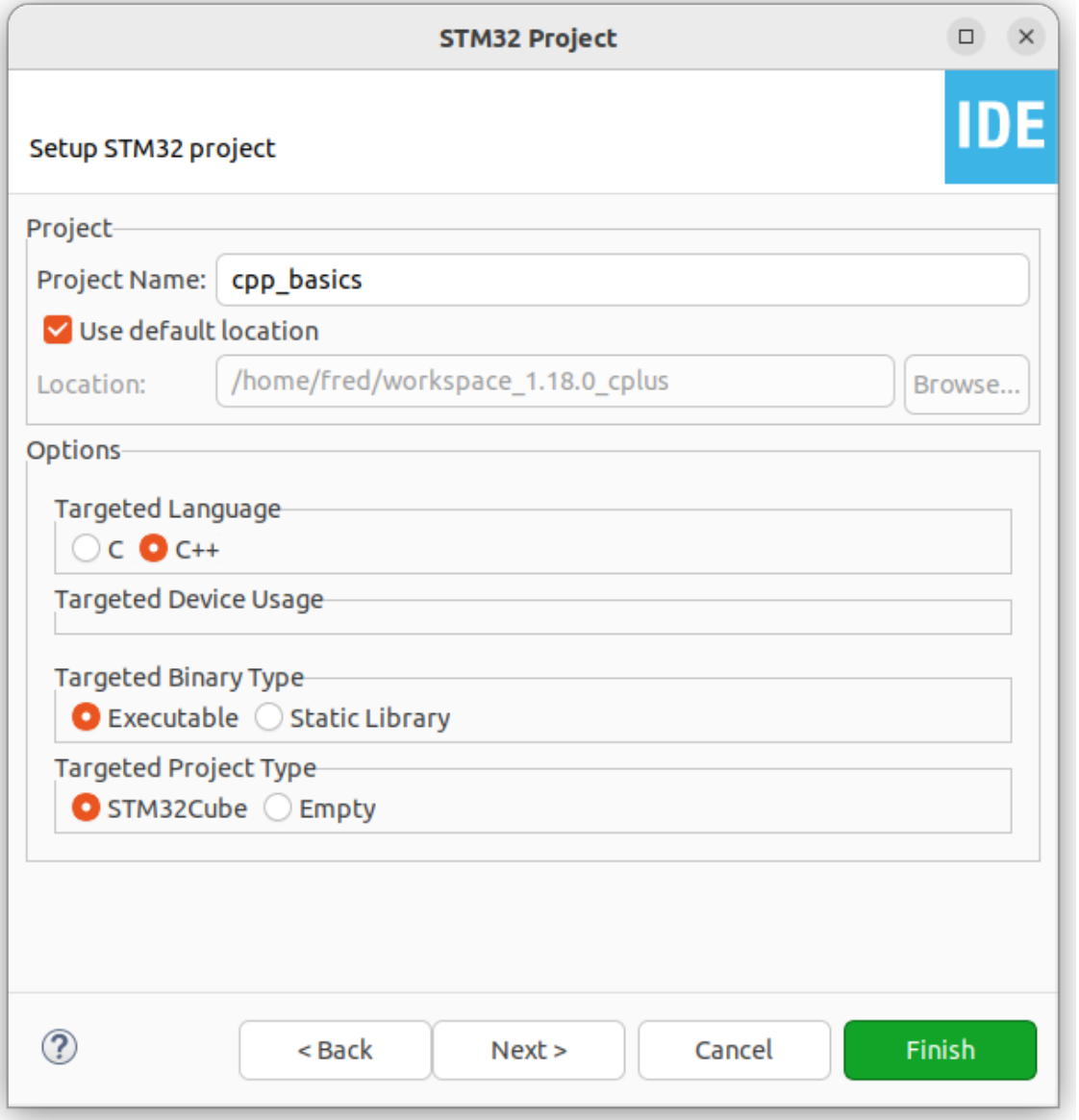
STM32Gpio Class Components

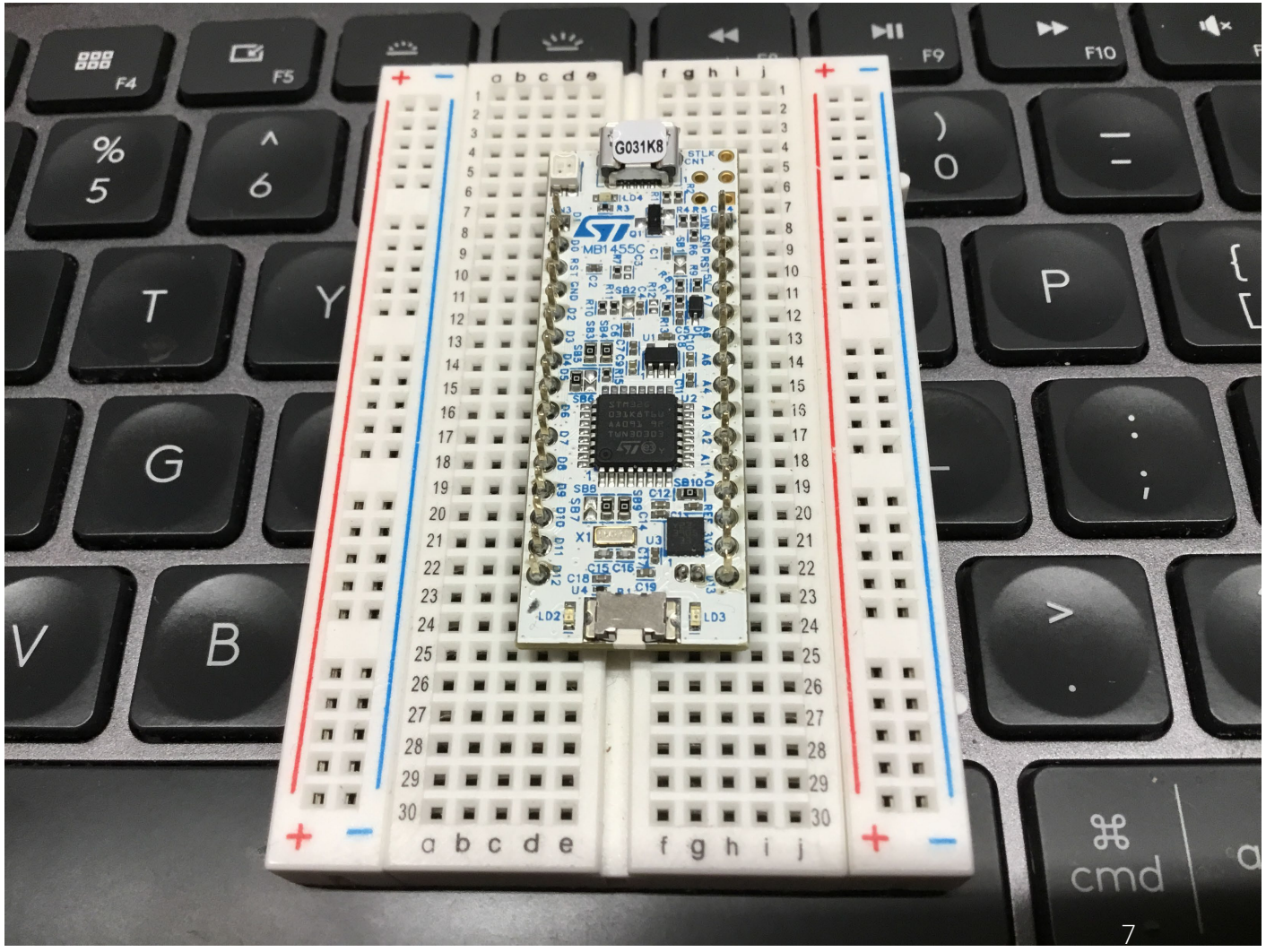
```
14  class STM32Gpio                                 // Class Keyword - Class Name
15  {
16  public:                                         // Access specifier - Members (attributes and methods) accessible outside of the class
17      STM32Gpio(GPIO_TypeDef* port,uint16_t pin); // Constructor
18
19      void SetPin();                              // Methods (Functions)
20      void ClrPin();
21      void TogPin();
22      bool ReadPin();
23
24  private:                                        // Access specifier - Members not accessible outside of the class
25      GPIO_TypeDef* stm32port;                    // Attributes
26      uint16_t stm32pin;
27  };
```

STM32Gpio Class Components

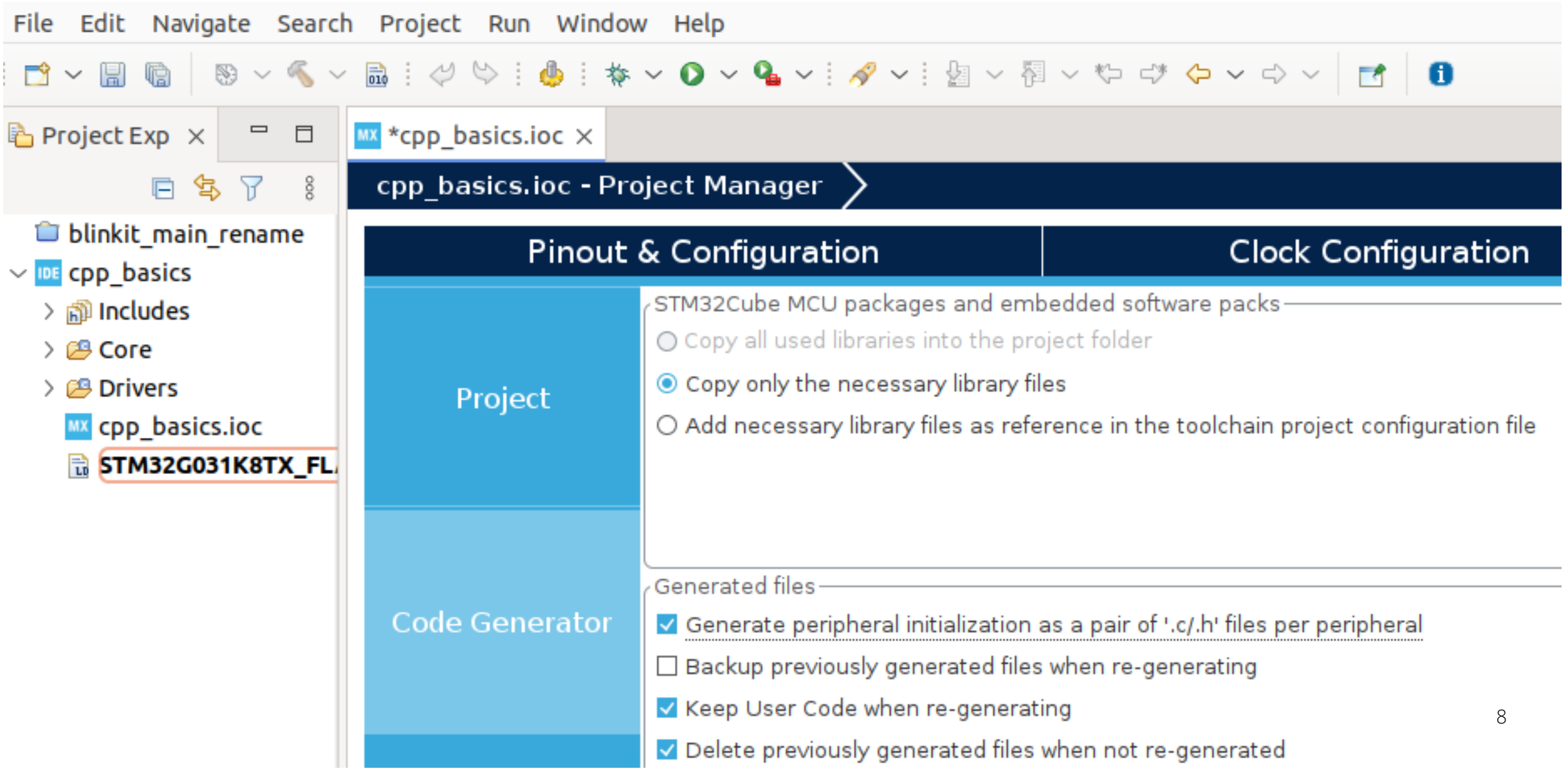
```
 9 #include "STM32Gpio.hpp"
10
11 STM32Gpio::STM32Gpio(GPIO_TypeDef* port,uint16_t pin)   // Constructor parameter list
12     :                                                  // Parameter list/Member Initializer Delimiter
13     stm32port(port),                                   // Member Initializer list
14     stm32pin(pin)
15 {}
16
17 // The :: in each function declaration is known as the scope resolution operator
18
19 void STM32Gpio::SetPin()
20 {
21     return HAL_GPIO_WritePin(stm32port,stm32pin,GPIO_PIN_SET);
22 }
23
24 void STM32Gpio::ClrPin()
25 {
26     return HAL_GPIO_WritePin(stm32port,stm32pin,GPIO_PIN_RESET);
27 }
28
29 void STM32Gpio::TogPin()
30 {
31     return HAL_GPIO_TogglePin(stm32port,stm32pin);
32 }
33
34 bool STM32Gpio::ReadPin()
35 {
36     return HAL_GPIO_ReadPin(stm32port,stm32pin);
37 }
```

Create a C++ Project Named cpp_basics

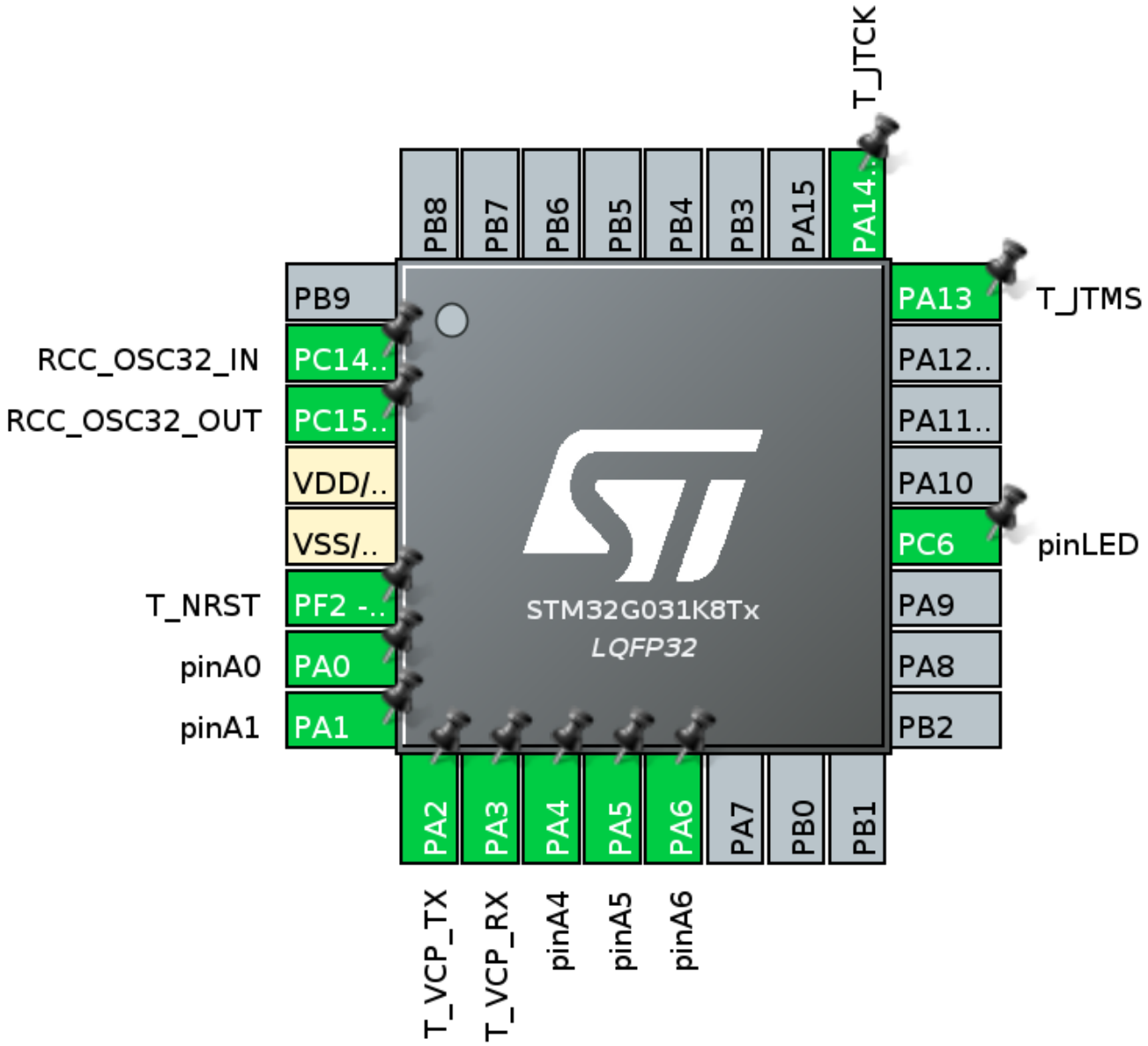




Configure the C++ Project for .c and .h File Generation

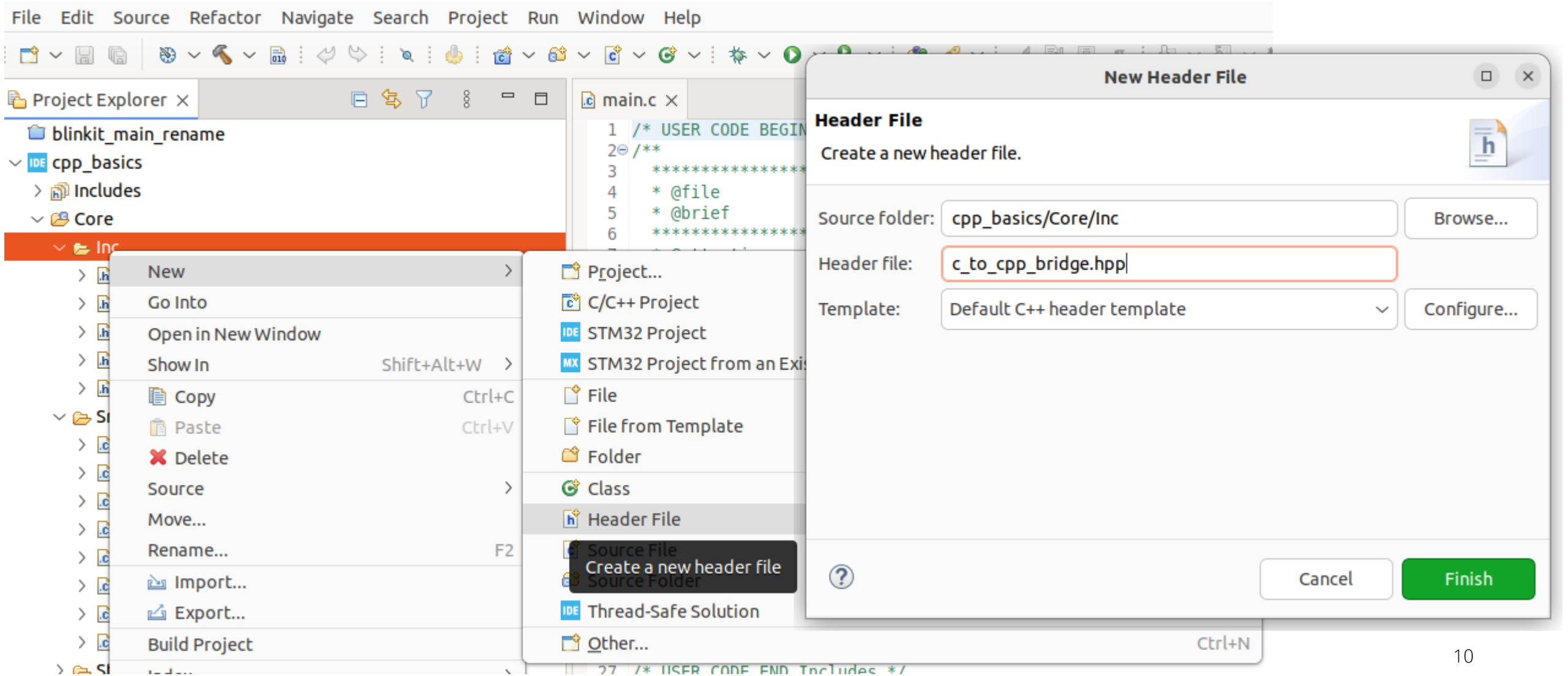


Configure the NUCLEO-G031K8 Hardware

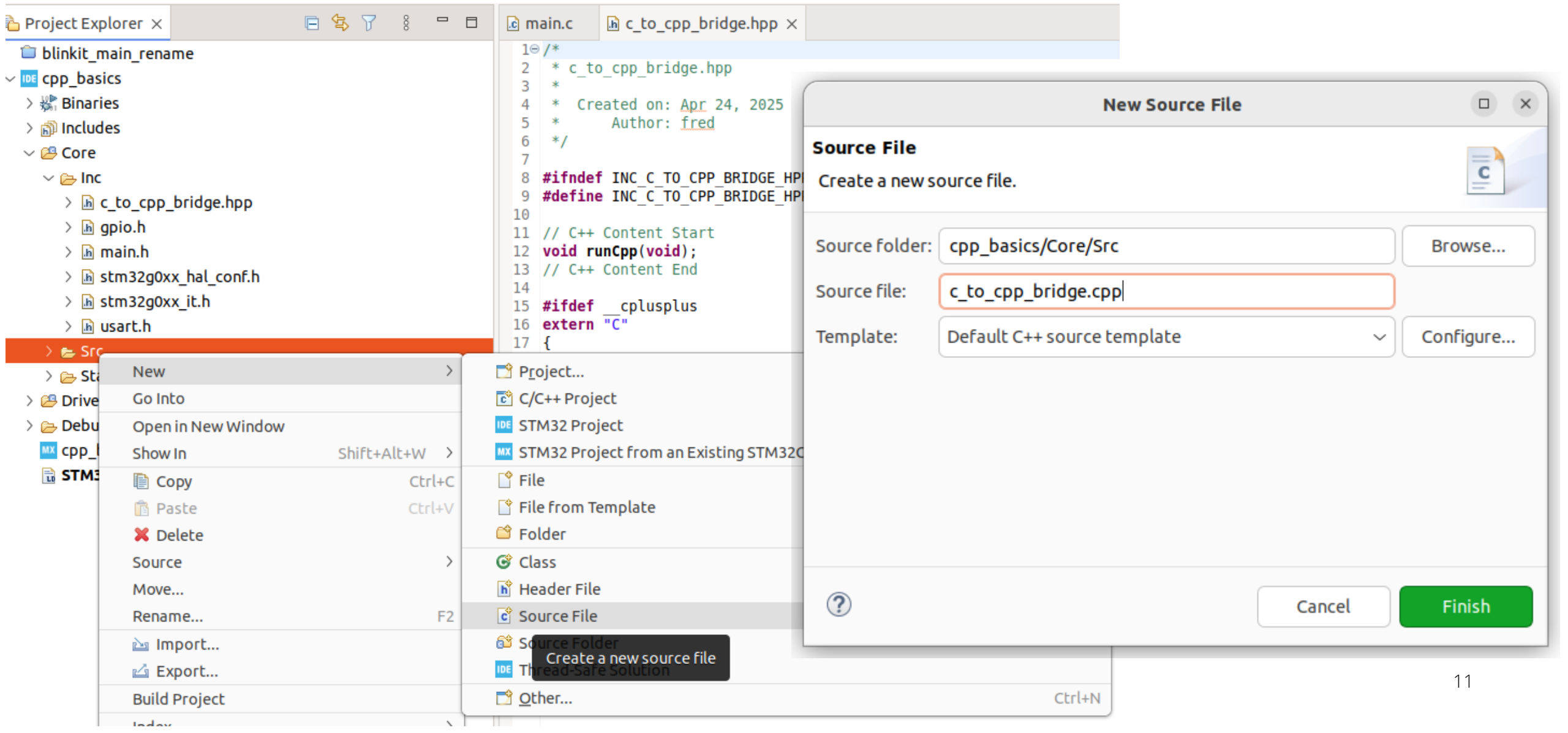



```
59 /* Private defines -----------------
60 #define T_NRST_Pin GPIO_PIN_2
61 #define T_NRST_GPIO_Port GPIOF
62 #define pinA0_Pin GPIO_PIN_0
63 #define pinA0_GPIO_Port GPIOA
64 #define pinA1_Pin GPIO_PIN_1
65 #define pinA1_GPIO_Port GPIOA
66 #define T_VCP_TX_Pin GPIO_PIN_2
67 #define T_VCP_TX_GPIO_Port GPIOA
68 #define T_VCP_RX_Pin GPIO_PIN_3
69 #define T_VCP_RX_GPIO_Port GPIOA
70 #define pinA4_Pin GPIO_PIN_4
71 #define pinA4_GPIO_Port GPIOA
72 #define pinA5_Pin GPIO_PIN_5
73 #define pinA5_GPIO_Port GPIOA
74 #define pinA6_Pin GPIO_PIN_6
75 #define pinA6_GPIO_Port GPIOA
76 #define pinLED_Pin GPIO_PIN_6
77 #define pinLED_GPIO_Port GPIOC
78 #define T_JTMS_Pin GPIO_PIN_13
79 #define T_JTMS_GPIO_Port GPIOA
80 #define T_JTCK_Pin GPIO_PIN_14
81 #define T_JTCK_GPIO_Port GPIOA
```

Create C++ Header File - c_to_cpp_bridge.hpp



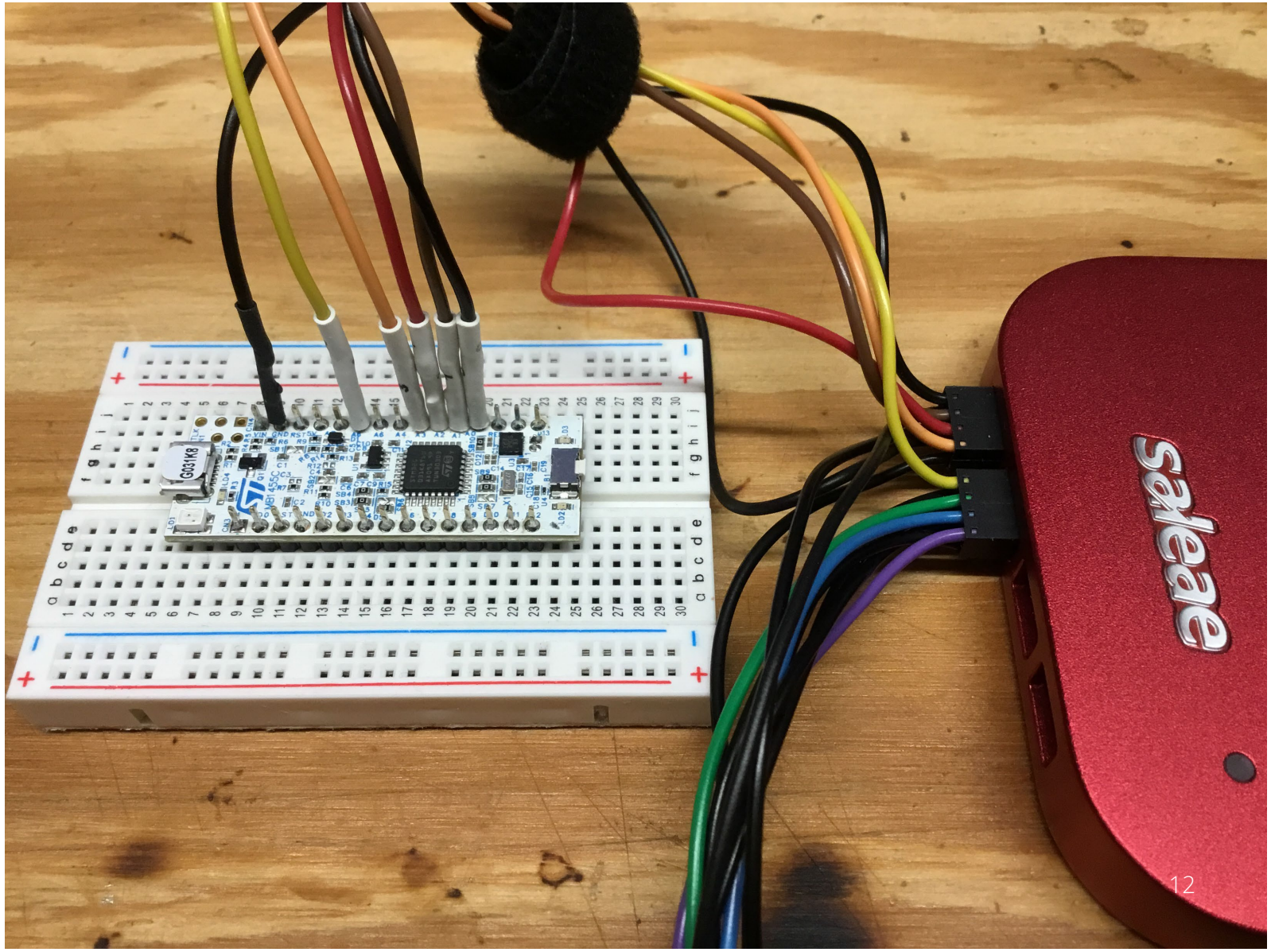
Create C++ Source File - c_to_cpp_bridge .cpp



Populate C++ Header File - c_to_cpp_bridge .hpp

c_to_cpp_bridge.hpp

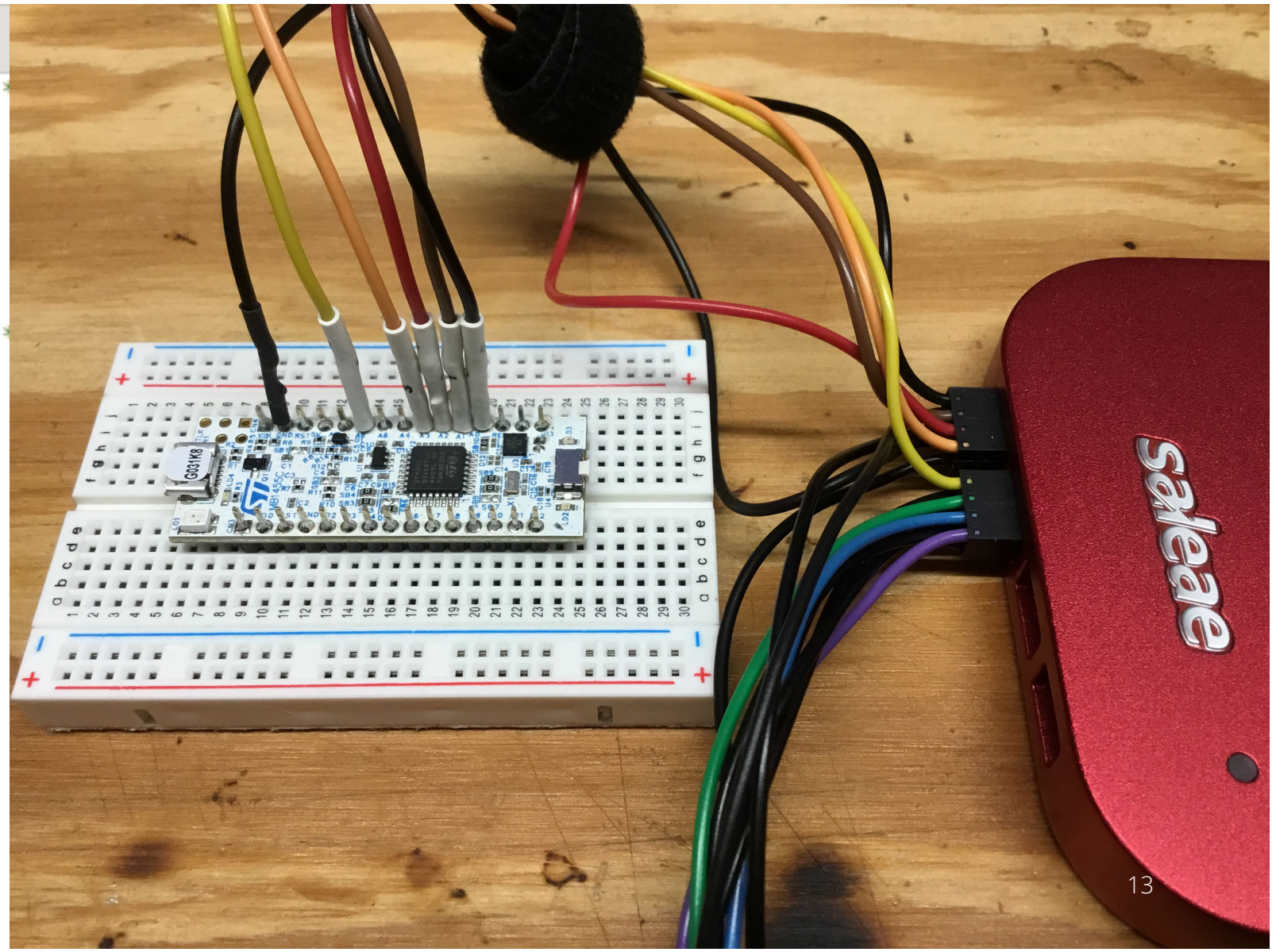
```
1  //**********************************
2  //* C++ Primer for Embedded Systems
3  //* c_to_cpp_bridge.hpp
4  //* Last Update: Apr 26, 2025
5  //* Author: Fred Eady
6  //* Notes:
7  //**********************************
8
9  #ifndef INC_C_TO_CPP_BRIDGE_HPP_
10 #define INC_C_TO_CPP_BRIDGE_HPP_
11
12 // C++ Content Start
13 void runCpp(void);
14 // C++ Content End
15
16 // C Content Start
17 #ifdef __cplusplus
18 extern "C"
19 {
20 #endif
21
22 void call_runCpp(void);
23
24 #ifdef __cplusplus
25 }
26 #endif
27 // C Content End
28
29
30
31 #endif /* INC_C_TO_CPP_BRIDGE_HPP_ */
```



Populate C++ Source File - c_to_cpp_bridge .cpp

c_to_cpp_bridge.cpp

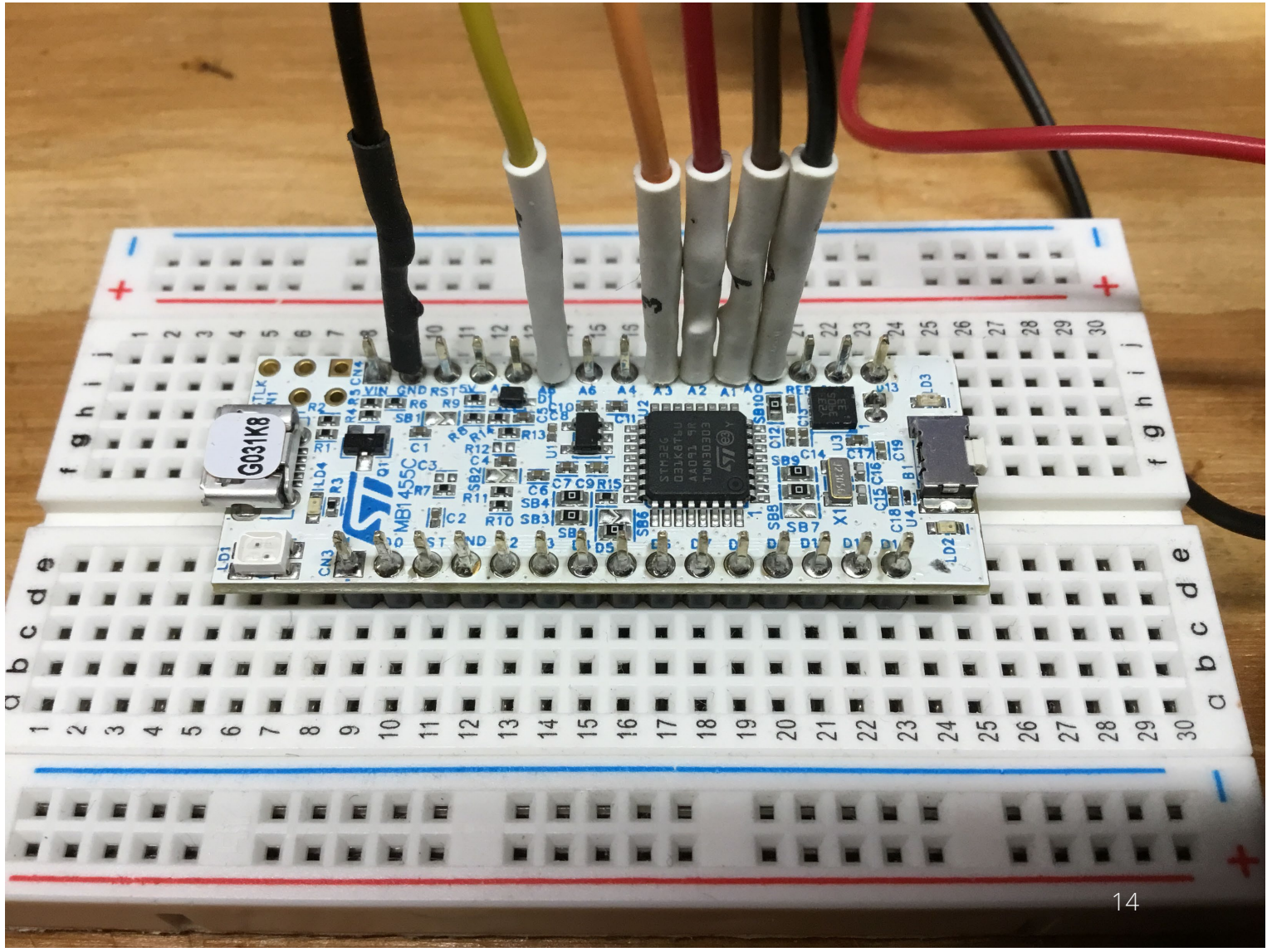
```
1  //*****************************************
2  //* C++ Primer for Embedded Systems
3  //* c_to_cpp_bridge.cpp
4  //* Last Update: Apr 26, 2025
5  //* Author: Fred Eady
6  //* Notes:
7  //* Include .hpp file for all C++ classes
8  //*****************************************
9
10 #include "main.h"
11
12 // Run C++ application
13 void runCpp()
14 {
15     // C++ App Code Goes Here
16 }
17
18 // Function calls from main.c
19 extern "C"
20 {
21     void call_runCpp()
22     {
23         runCpp();
24     }
25 }
```



Create the STM32Gpio Class

STM32Gpio.hpp

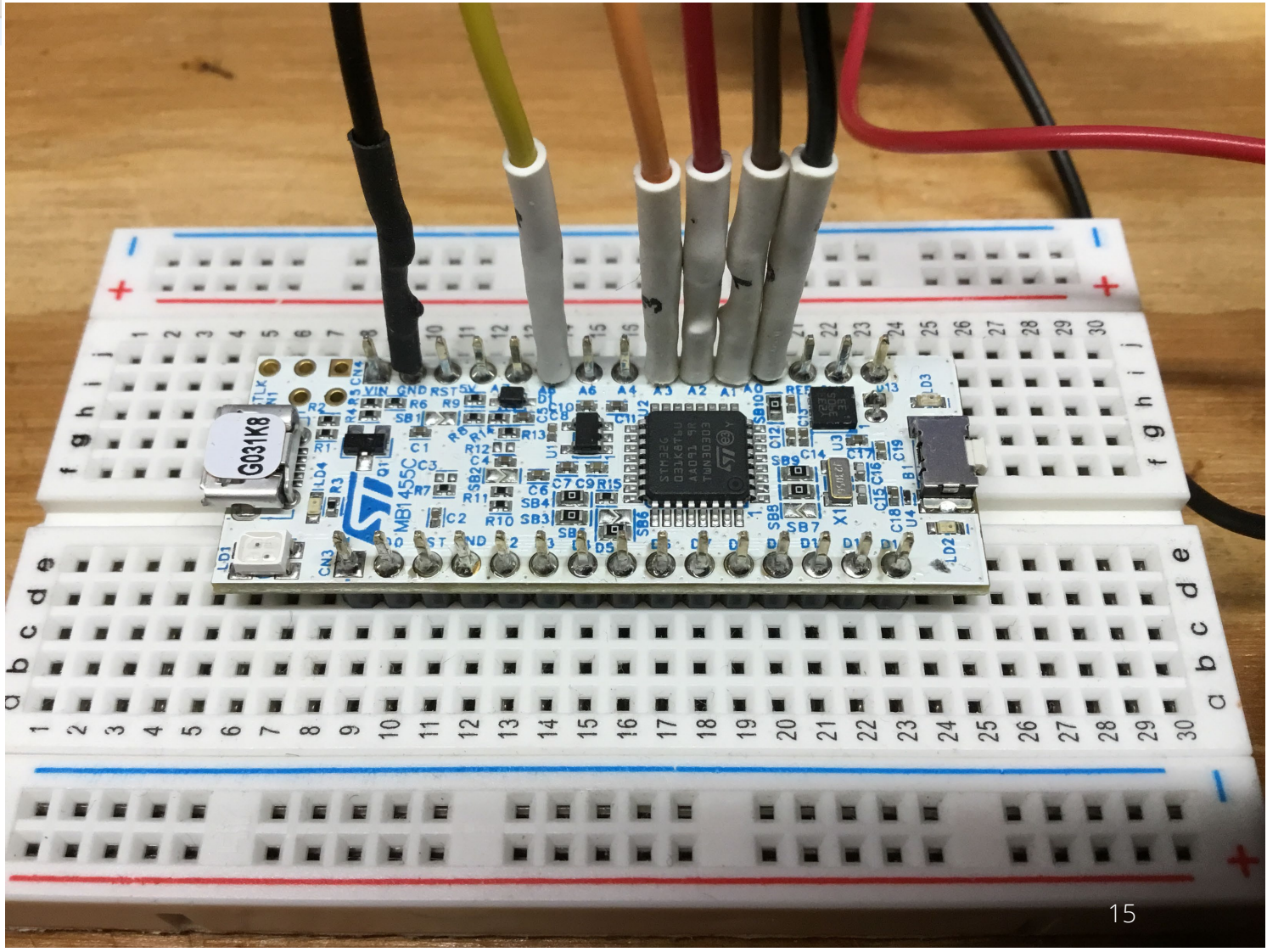
```cpp
//*************************************************
//* C++ Primer for Embedded Systems
//* c_to_cpp_bridge.hpp
//* Last Update: Apr 27, 2025
//* Author: Fred Eady
//* Notes:
//*************************************************

#ifndef INC_STM32GPIO_HPP_
#define INC_STM32GPIO_HPP_

#include "stm32g0xx_hal.h"

class STM32Gpio
{
public:
    STM32Gpio(GPIO_TypeDef* port,uint16_t pin);

    void SetPin();
    void ClrPin();
    void TogPin();
    bool ReadPin();

private:
    GPIO_TypeDef* stm32port;
    uint16_t stm32pin;
};
```



Create the STM32Gpio Class

STM32Gpio.cpp

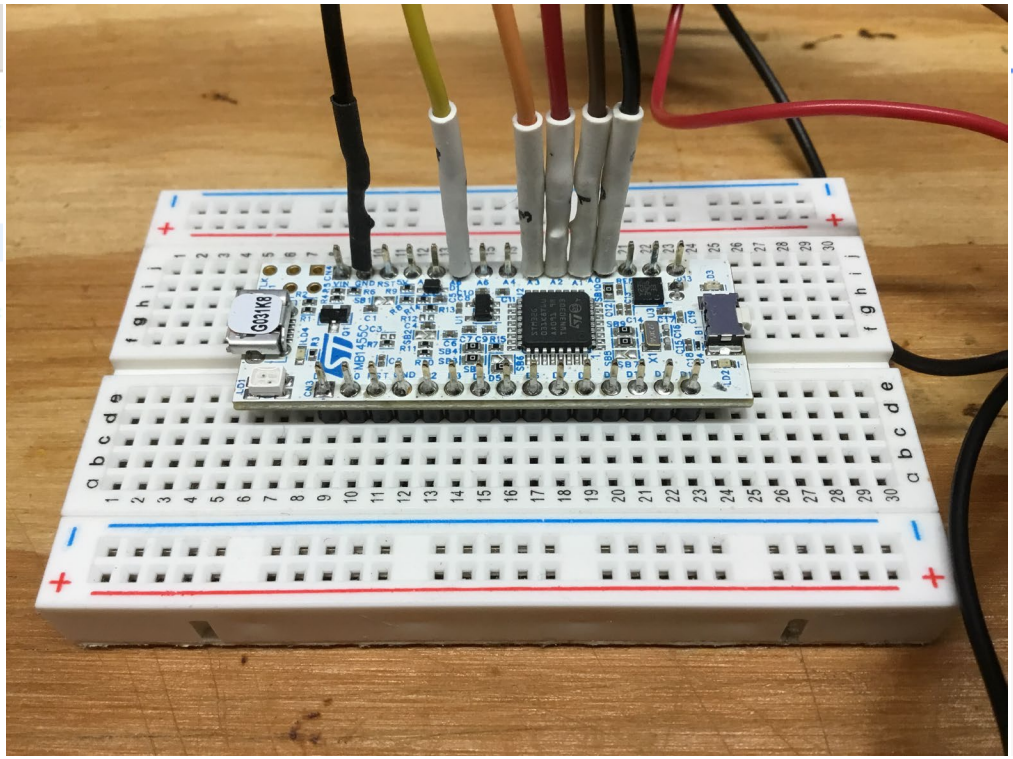
```cpp
//*****************************************************************
//* C++ Primer for Embedded Systems
//* STM32Gpio.cpp
//* Last Update: Apr 27, 2025
//* Author: Fred Eady
//* Notes:
//*****************************************************************

#include "STM32Gpio.hpp"

STM32Gpio::STM32Gpio(GPIO_TypeDef* port,uint16_t pin)
    :
    stm32port(port),
    stm32pin(pin)
{}

void STM32Gpio::SetPin()
{
    return HAL_GPIO_WritePin(stm32port,stm32pin,GPIO_PIN_SET);
}

void STM32Gpio::ClrPin()
{
    return HAL_GPIO_WritePin(stm32port,stm32pin,GPIO_PIN_RESET);
}

void STM32Gpio::TogPin()
{
    return HAL_GPIO_TogglePin(stm32port,stm32pin);
}

bool STM32Gpio::ReadPin()
{
    return HAL_GPIO_ReadPin(stm32port,stm32pin);
}
```



Modify main.c

main.c

```
/* USER CODE BEGIN Header */
//**********************************
//* C++ Primer for Embedded Systems
//* Author: Fred Eady
//* Last Update: 04-28-2025
//* Notes:
//**********************************
/* USER CODE END Header */
/* Includes ------------------------
#include "main.h"
#include "usart.h"
#include "gpio.h"

/* Private includes ----------------
/* USER CODE BEGIN Includes */
#include "c_to_cpp_bridge.hpp"
/* USER CODE END Includes */

```



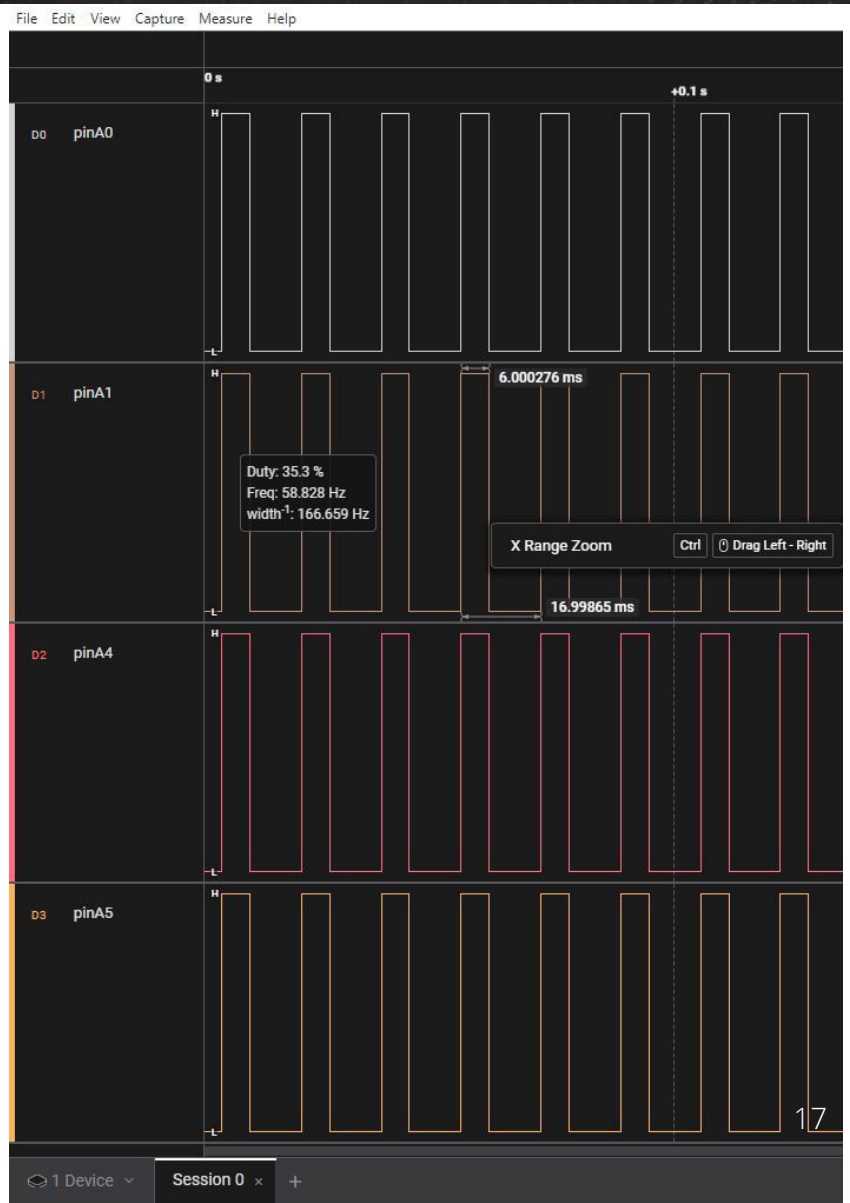
main.c

```
/* USER CODE END SysInit */

  /* Initialize all configured perip
  MX_GPIO_Init();
  MX_USART2_UART_Init();
  /* USER CODE BEGIN 2 */
  call_runCpp();
  /* USER CODE END 2 */

  /* Infinite loop */
  /* USER CODE BEGIN WHILE */
  while (1)
  {
      //HAL_GPIO_TogglePin(pinLED_GP
      //HAL_Delay(100);
    /* USER CODE END WHILE */

    /* USER CODE BEGIN 3 */
```

Run It

c_to_cpp_bridge.cpp

```cpp
//*****************************************************************************
//* C++ Primer for Embedded Systems
//* c_to_cpp_bridge.cpp
//* Last Update: Apr 26, 2025
//* Author: Fred Eady
//* Notes:
//* Include .hpp file for all C++ classes used in the c_to_cpp_bridge.cpp file
//*****************************************************************************

#include "main.h"
#include "STM32Gpio.hpp"

// Run C++ application
void runCpp()
{
    STM32Gpio iopins(GPIOA,pinA0_Pin|pinA1_Pin|pinA4_Pin|pinA5_Pin);
    STM32Gpio led(pinLED_GPIO_Port,pinLED_Pin);
    while(1)
    {
        iopins.SetPin();
        HAL_Delay(5);
        iopins.ClrPin();
        HAL_Delay(10);
    }
}

// Function calls from main.c
extern "C"
{
    void call_runCpp()
    {
        runCpp();
    }
}
```



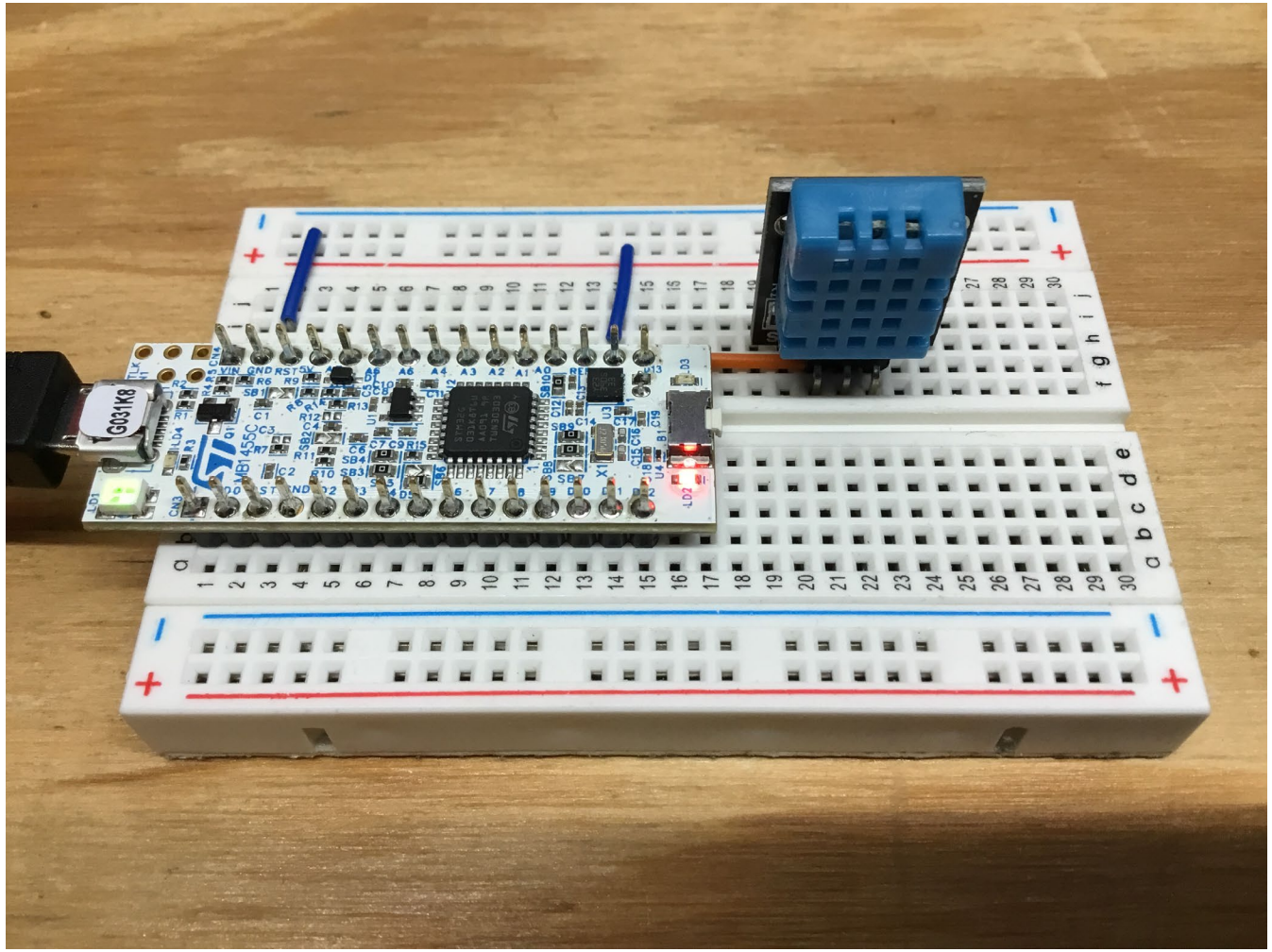
Next Time…

Thank you for attending!!!

Please consider the resources below:

- **Today's Download Package**

MORE TO COME..



CEC Continuing Education Center

DesignNews

Thank You

Sponsored by

DigiKey

informa markets