/\* USER CODE BEGIN Header \*/

/\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* @file : main.c

\* @brief : Main program body

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* @attention

\*

\* <h2><center>&copy; Copyright (c) 2021 STMicroelectronics.

\* All rights reserved.</center></h2>

\*

\* This software component is licensed by ST under BSD 3-Clause license,

\* the "License"; You may not use this file except in compliance with the

\* License. You may obtain a copy of the License at:

\* opensource.org/licenses/BSD-3-Clause

\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*/

/\* USER CODE END Header \*/

/\* Includes ------------------------------------------------------------------\*/

**#include** "main.h"

**#include** <string.h>

**#define** A 10

**#define** B 11

**#define** C 12

**#define** D 13

**#define** E 14

**#define** F 15

**#define** G 16

**#define** H 17

**#define** I 18

**#define** J 19

**#define** K 20

**#define** L 21

**#define** M 22

**#define** N 23

**#define** O 24

**#define** P 25

**#define** Q 26

**#define** R 27

**#define** S 28

**#define** T 29

**#define** U 30

**#define** V 31

**#define** W 32

**#define** X 33

**#define** Y 34

**#define** Z 35

**int** ALL\_LED;

**int** ALL\_LED1;

**int** ALL\_LED2;

**int** ALL\_LED3;

**int** ALL\_LED4;

**int** ALL\_LED5 = 0xFF;

**int** a=255;

**int** b=80;

**int** c=0;

**char** bang[255] = {P,I,F}; //NHAP CHUOI (0=O)

/\* Private includes ----------------------------------------------------------\*/

/\* USER CODE BEGIN Includes \*/

/\* USER CODE END Includes \*/

/\* Private typedef -----------------------------------------------------------\*/

/\* USER CODE BEGIN PTD \*/

/\* USER CODE END PTD \*/

/\* Private define ------------------------------------------------------------\*/

/\* USER CODE BEGIN PD \*/

/\* USER CODE END PD \*/

/\* Private macro -------------------------------------------------------------\*/

/\* USER CODE BEGIN PM \*/

/\* USER CODE END PM \*/

/\* Private variables ---------------------------------------------------------\*/

/\* USER CODE BEGIN PV \*/

/\* USER CODE END PV \*/

/\* Private function prototypes -----------------------------------------------\*/

**void** **SystemClock\_Config**(**void**);

**static** **void** **MX\_GPIO\_Init**(**void**);

/\* USER CODE BEGIN PFP \*/

/\* USER CODE END PFP \*/

/\* Private user code ---------------------------------------------------------\*/

/\* USER CODE BEGIN 0 \*/

/\* USER CODE END 0 \*/

**int** x;

**int** BCC[180]={0xFF,0xFF,0xFF,0xFF,0xFF,

0x22,0x42,0xFE,0x02,0x02, //1

0x46,0x8A,0x92,0x92,0x62, //2

0x84,0x82,0x92,0xB2,0xCC, //3

0x18,0x28,0x48,0xFE,0x08, //4

0xF4,0x92,0x92,0x92,0x8C, //5

0x3c,0x52,0x92,0x92,0x0C, //6

0x80,0x8E,0x90,0xA0,0xC0, //7

0x6C,0x92,0x92,0x92,0x6C, //8

0x60,0x92,0x92,0x94,0x78, //9

0x7E,0x90,0x90,0x90,0x7E, //A

0xFE,0x92,0x92,0x92,0x6C, //B

0x7C,0x82,0x82,0x82,0x44, //C

0xFE,0x82,0x82,0x82,0x7C, //D

0xFE,0x92,0x92,0x92,0x92, //E

0xFE,0x90,0x90,0x90,0x80, //F

0x7C,0x82,0x82,0x92,0x9E, //G

0xFE,0x10,0x10,0x10,0xFE, //H

0x00,0x00,0xFE,0x00,0x00, //I

0x84,0x82,0xFC,0x80,0x80, //J

0xFE,0x10,0x28,0x44,0x82, //K

0xFE,0x02,0x02,0x02,0x02, //L

0xFE,0x40,0x30,0x40,0xFE, //M

0xFE,0x20,0x10,0x08,0xFE, //N

0x7C,0x82,0x82,0x82,0x7C, //O

0xFE,0x90,0x90,0x90,0x60, //P

0x7C,0x82,0x8A,0x84,0x7A, //Q

0xFE,0x90,0x98,0x94,0x62, //R

0x64,0x92,0x92,0x92,0x4C, //S

0x80,0x80,0xFE,0x80,0x80, //T

0xFC,0x02,0x02,0x02,0xFC, //U

0xF8,0x04,0x02,0x04,0xF8, //V

0xFC,0x02,0x1C,0x02,0xFC, //W

0xC6,0x28,0x10,0x28,0xC6, //X

0xE0,0x10,0x1E,0x10,0xE0, //Y

0x8C,0x8A,0x92,0xA2,0xC2}; //Z

/\*\*

\* @brief The application entry point.

\* @retval int

\*/

**int** **main**(**void**)

{

/\* USER CODE BEGIN 1 \*/

/\* USER CODE END 1 \*/

/\* MCU Configuration--------------------------------------------------------\*/

/\* Reset of all peripherals, Initializes the Flash interface and the Systick. \*/

HAL\_Init();

/\* USER CODE BEGIN Init \*/

/\* USER CODE END Init \*/

/\* Configure the system clock \*/

SystemClock\_Config();

/\* USER CODE BEGIN SysInit \*/

/\* USER CODE END SysInit \*/

/\* Initialize all configured peripherals \*/

MX\_GPIO\_Init();

/\* USER CODE BEGIN 2 \*/

/\* USER CODE END 2 \*/

/\* Infinite loop \*/

/\* USER CODE BEGIN WHILE \*/

**while** (1)

{

/\* USER CODE END WHILE \*/

/\* USER CODE BEGIN 3 \*/

}

/\* USER CODE END 3 \*/

}

/\*\*

\* @brief System Clock Configuration

\* @retval None

\*/

**void** **SystemClock\_Config**(**void**)

{

RCC\_OscInitTypeDef RCC\_OscInitStruct = {0};

RCC\_ClkInitTypeDef RCC\_ClkInitStruct = {0};

/\*\* Initializes the RCC Oscillators according to the specified parameters

\* in the RCC\_OscInitTypeDef structure.

\*/

RCC\_OscInitStruct.OscillatorType = RCC\_OSCILLATORTYPE\_HSI;

RCC\_OscInitStruct.HSIState = RCC\_HSI\_ON;

RCC\_OscInitStruct.HSICalibrationValue = RCC\_HSICALIBRATION\_DEFAULT;

RCC\_OscInitStruct.PLL.PLLState = RCC\_PLL\_NONE;

**if** (HAL\_RCC\_OscConfig(&RCC\_OscInitStruct) != *HAL\_OK*)

{

Error\_Handler();

}

/\*\* Initializes the CPU, AHB and APB buses clocks

\*/

RCC\_ClkInitStruct.ClockType = RCC\_CLOCKTYPE\_HCLK|RCC\_CLOCKTYPE\_SYSCLK

|RCC\_CLOCKTYPE\_PCLK1|RCC\_CLOCKTYPE\_PCLK2;

RCC\_ClkInitStruct.SYSCLKSource = RCC\_SYSCLKSOURCE\_HSI;

RCC\_ClkInitStruct.AHBCLKDivider = RCC\_SYSCLK\_DIV1;

RCC\_ClkInitStruct.APB1CLKDivider = RCC\_HCLK\_DIV1;

RCC\_ClkInitStruct.APB2CLKDivider = RCC\_HCLK\_DIV1;

**if** (HAL\_RCC\_ClockConfig(&RCC\_ClkInitStruct, FLASH\_LATENCY\_0) != *HAL\_OK*)

{

Error\_Handler();

}

}

/\*\*

\* @brief GPIO Initialization Function

\* @param None

\* @retval None

\*/

**static** **void** **MX\_GPIO\_Init**(**void**)

{

GPIO\_InitTypeDef GPIO\_InitStruct = {0};

/\* GPIO Ports Clock Enable \*/

\_\_HAL\_RCC\_GPIOC\_CLK\_ENABLE();

\_\_HAL\_RCC\_GPIOA\_CLK\_ENABLE();

\_\_HAL\_RCC\_GPIOB\_CLK\_ENABLE();

/\*Configure GPIO pin Output Level \*/

HAL\_GPIO\_WritePin(GPIOC, GPIO\_PIN\_13, *GPIO\_PIN\_RESET*);

/\*Configure GPIO pin Output Level \*/

HAL\_GPIO\_WritePin(GPIOA, GPIO\_PIN\_0|GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3

|GPIO\_PIN\_4|GPIO\_PIN\_5|GPIO\_PIN\_6|GPIO\_PIN\_7, *GPIO\_PIN\_RESET*);

/\*Configure GPIO pin : PC13 \*/

GPIO\_InitStruct.Pin = GPIO\_PIN\_13;

GPIO\_InitStruct.Mode = GPIO\_MODE\_OUTPUT\_PP;

GPIO\_InitStruct.Pull = GPIO\_NOPULL;

GPIO\_InitStruct.Speed = GPIO\_SPEED\_FREQ\_LOW;

HAL\_GPIO\_Init(GPIOC, &GPIO\_InitStruct);

/\*Configure GPIO pins : PA0 PA1 PA2 PA3

PA4 PA5 PA6 PA7 \*/

GPIO\_InitStruct.Pin = GPIO\_PIN\_0|GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3

|GPIO\_PIN\_4|GPIO\_PIN\_5|GPIO\_PIN\_6|GPIO\_PIN\_7;

GPIO\_InitStruct.Mode = GPIO\_MODE\_OUTPUT\_PP;

GPIO\_InitStruct.Pull = GPIO\_NOPULL;

GPIO\_InitStruct.Speed = GPIO\_SPEED\_FREQ\_LOW;

HAL\_GPIO\_Init(GPIOA, &GPIO\_InitStruct);

/\*Configure GPIO pin : PB9 \*/

GPIO\_InitStruct.Pin = GPIO\_PIN\_9;

GPIO\_InitStruct.Mode = GPIO\_MODE\_IT\_FALLING;

GPIO\_InitStruct.Pull = GPIO\_PULLDOWN;

HAL\_GPIO\_Init(GPIOB, &GPIO\_InitStruct);

/\* EXTI interrupt init\*/

HAL\_NVIC\_SetPriority(*EXTI9\_5\_IRQn*, 0, 0);

HAL\_NVIC\_EnableIRQ(*EXTI9\_5\_IRQn*);

}

**void** **HAL\_GPIO\_EXTI\_Callback**(uint16\_t GPIO\_PIN){

**if**(GPIO\_PIN==GPIO\_PIN\_9){

HAL\_GPIO\_WritePin(GPIOA, ALL\_LED5, *GPIO\_PIN\_RESET*);

**for**(**int** m=0;m<a;m++){ //cai dat goc hien thi

**for**(**int** n=0;n<b;n++){

**asm**("NOP");

}

}

**for**(**int** i = 0;i<**strlen**(bang);i++){

x=bang[i]\*5;

ALL\_LED = BCC[x];

ALL\_LED1 = BCC[x+1];

ALL\_LED2 = BCC[x+2];

ALL\_LED3 = BCC[x+3];

ALL\_LED4 = BCC[x+4];

HAL\_GPIO\_WritePin(GPIOA, ALL\_LED, *GPIO\_PIN\_SET*);

**for**(**int** i=0;i<200;i++)

**asm** ("NOP");

HAL\_GPIO\_WritePin(GPIOA, ALL\_LED, *GPIO\_PIN\_RESET*);

**for**(**int** i=0;i<100;i++)

**for**(**int** j=0;j<2;j++)

**asm** ("NOP");

HAL\_GPIO\_WritePin(GPIOA, ALL\_LED1, *GPIO\_PIN\_SET*);

**for**(**int** i=0;i<200;i++)

**asm** ("NOP");

HAL\_GPIO\_WritePin(GPIOA, ALL\_LED1, *GPIO\_PIN\_RESET*);

**for**(**int** i=0;i<100;i++)

**for**(**int** j=0;j<2;j++)

**asm** ("NOP");

HAL\_GPIO\_WritePin(GPIOA, ALL\_LED2, *GPIO\_PIN\_SET*);

**for**(**int** i=0;i<200;i++)

**asm** ("NOP");

HAL\_GPIO\_WritePin(GPIOA, ALL\_LED2, *GPIO\_PIN\_RESET*);

**for**(**int** i=0;i<100;i++)

**for**(**int** j=0;j<2;j++)

**asm** ("NOP");

HAL\_GPIO\_WritePin(GPIOA, ALL\_LED3, *GPIO\_PIN\_SET*);

**for**(**int** i=0;i<200;i++)

**asm** ("NOP");

HAL\_GPIO\_WritePin(GPIOA, ALL\_LED3, *GPIO\_PIN\_RESET*);

**for**(**int** i=0;i<100;i++)

**for**(**int** j=0;j<2;j++)

**asm** ("NOP");

HAL\_GPIO\_WritePin(GPIOA, ALL\_LED4, *GPIO\_PIN\_SET*);

**for**(**int** i=0;i<200;i++)

**asm** ("NOP");

HAL\_GPIO\_WritePin(GPIOA, ALL\_LED4, *GPIO\_PIN\_RESET*);

**for**(**int** i=0;i<200;i++)

**for**(**int** j=0;j<4;j++)

**asm** ("NOP");

}

}

}

/\* USER CODE BEGIN 4 \*/

/\* USER CODE END 4 \*/

/\*\*

\* @brief This function is executed in case of error occurrence.

\* @retval None

\*/

**void** **Error\_Handler**(**void**)

{

/\* USER CODE BEGIN Error\_Handler\_Debug \*/

/\* User can add his own implementation to report the HAL error return state \*/

/\* USER CODE END Error\_Handler\_Debug \*/

}

**#ifdef** USE\_FULL\_ASSERT

/\*\*

\* @brief Reports the name of the source file and the source line number

\* where the assert\_param error has occurred.

\* @param file: pointer to the source file name

\* @param line: assert\_param error line source number

\* @retval None

\*/

**void** assert\_failed(uint8\_t \*file, uint32\_t line)

{

/\* USER CODE BEGIN 6 \*/

/\* User can add his own implementation to report the file name and line number,

tex: printf("Wrong parameters value: file %s on line %d\r\n", file, line) \*/

/\* USER CODE END 6 \*/

}

**#endif** /\* USE\_FULL\_ASSERT \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* (C) COPYRIGHT STMicroelectronics \*\*\*\*\*END OF FILE\*\*\*\*/