**Assignment – 4**

**while** (1)

{

/\* USER CODE END WHILE \*/

/\* USER CODE BEGIN 3 \*/

/\*

if (HAL\_GPIO\_ReadPin(sw2\_GPIO\_Port, GPIO\_PIN\_2) == 0)

{

c++;

printf("Reading pin status\n");

}

\*/

// printf("count=%d\n",c);

**switch**(c)

{

**case** 1:

{

**printf**("In case 1\n");

HAL\_GPIO\_WritePin(LED5\_GPIO\_Port, GPIO\_PIN\_5,1);

HAL\_Delay(1000);

HAL\_GPIO\_WritePin(LED5\_GPIO\_Port, GPIO\_PIN\_5,0);

HAL\_Delay(1000);

**break**;

}

**case** 2:

{

**printf**("In case 2\n");

HAL\_GPIO\_WritePin(LED14\_GPIO\_Port, GPIO\_PIN\_14,1);

HAL\_GPIO\_WritePin(LED5\_GPIO\_Port, GPIO\_PIN\_5,1);

HAL\_Delay(1000);

HAL\_GPIO\_WritePin(LED14\_GPIO\_Port, GPIO\_PIN\_14,0);

HAL\_GPIO\_WritePin(LED5\_GPIO\_Port, GPIO\_PIN\_5,0);

HAL\_Delay(1000);

**break**;

}

**case** 3:

{

**printf**("In case 3\n");

HAL\_GPIO\_WritePin(LED14\_GPIO\_Port, GPIO\_PIN\_14,1);

HAL\_GPIO\_WritePin(LED5\_GPIO\_Port, GPIO\_PIN\_5,1);

HAL\_Delay(500);

**break**;

}

**case** 4:

{

**printf**("In case 4\n");

HAL\_GPIO\_WritePin(LED14\_GPIO\_Port, GPIO\_PIN\_14,0);

HAL\_GPIO\_WritePin(LED5\_GPIO\_Port, GPIO\_PIN\_5,0);

HAL\_Delay(500);

**break**;

}

**case** 5:

{

c=1;

**break**;

}

**default** :

{

HAL\_GPIO\_WritePin(LED14\_GPIO\_Port, GPIO\_PIN\_14,0);

HAL\_GPIO\_WritePin(LED5\_GPIO\_Port, GPIO\_PIN\_5,0);

}

}

}

/\* USER CODE END 3 \*/

}

/\* USER CODE BEGIN 4 \*/

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

{

**if**(GPIO\_Pin == sw1\_Pin)

{

//HAL\_GPIO\_TogglePin(LED14\_GPIO\_Port, LED14\_Pin);

//printf("Button pressed event!!\n");

c++;

**printf**("count : %d",c);

}

**else** {

\_\_NOP();

}

}

/\* USER CODE END 4 \*/

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated