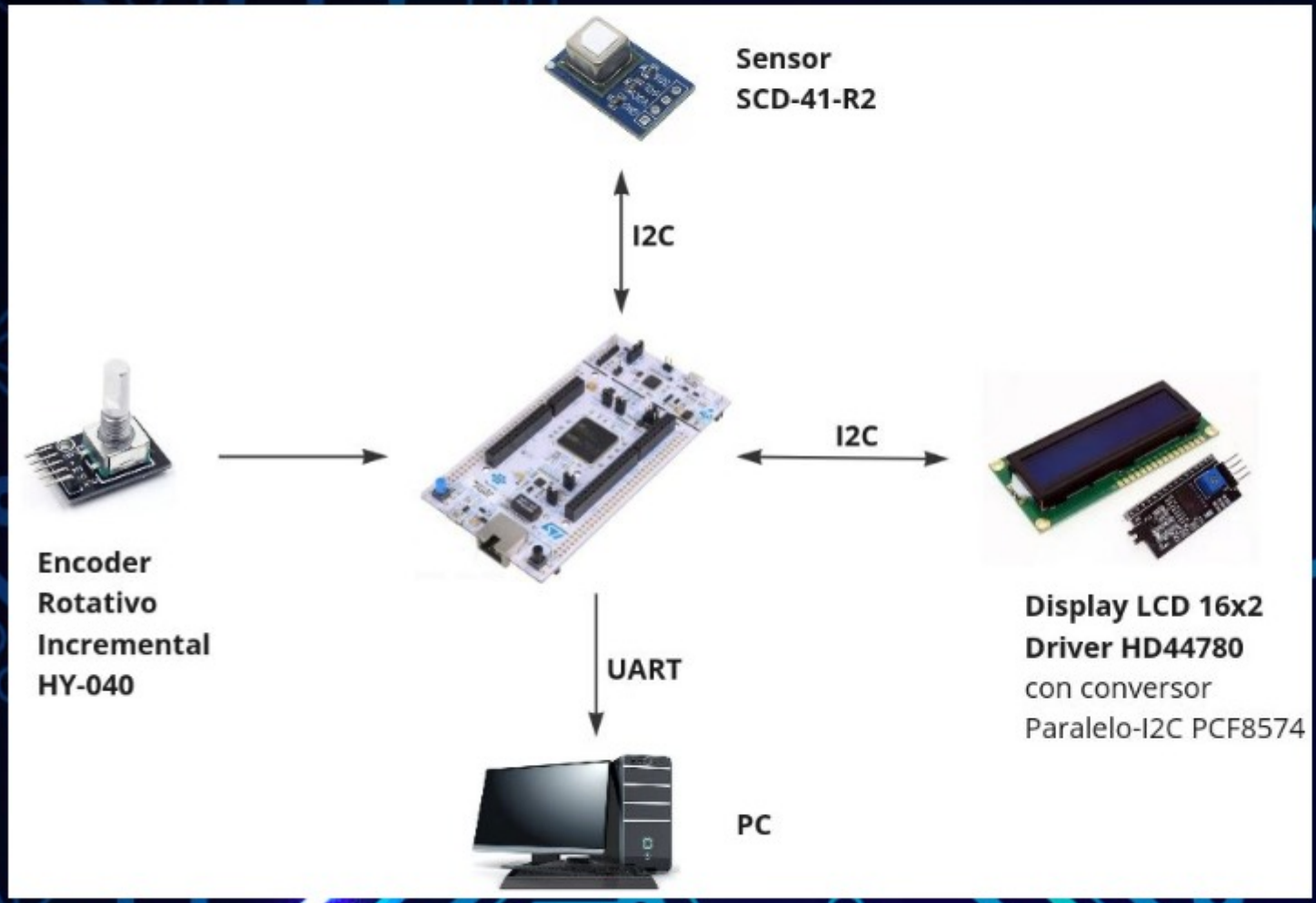
Programación de Microcontroladores

Docente - Patricio Bos - PdM - CESE

Alumno - Podoroska Iván

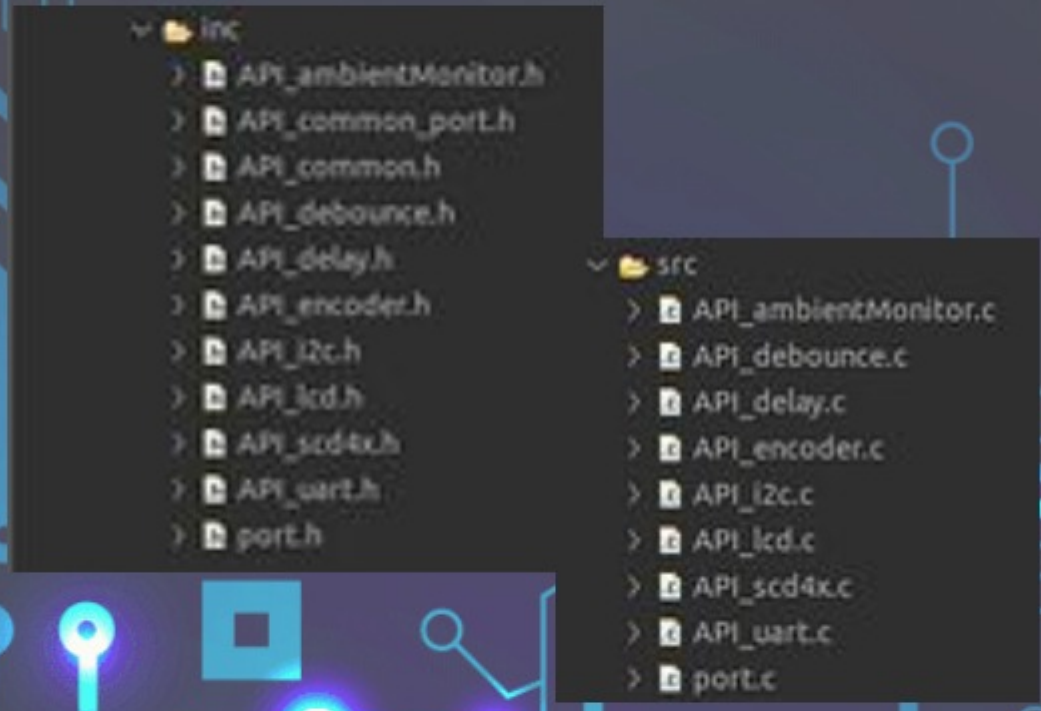
GITHUB - https://github.com/ivanpodo/PdM_workspace/tree/main/tp-final

TEMA TP



Podoroska Iván - PdM - CESE
Docente - Patricio Bos

Modularización



Podoroska Iván - PdM - CESE
Docente - Patricio Bos

Detalle GPIOs

```
GPIO_InitStruct.Pin = (GPIO_PIN_10 | GPIO_PIN_11 | GPIO_PIN_12);
GPIO_InitStruct.Mode = GPIO_MODE_INPUT;
GPIO_InitStruct.Pull = GPIO_PULLUP;
GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_LOW;

HAL_GPIO_Init(GPIOC, &GPIO_InitStruct);
```

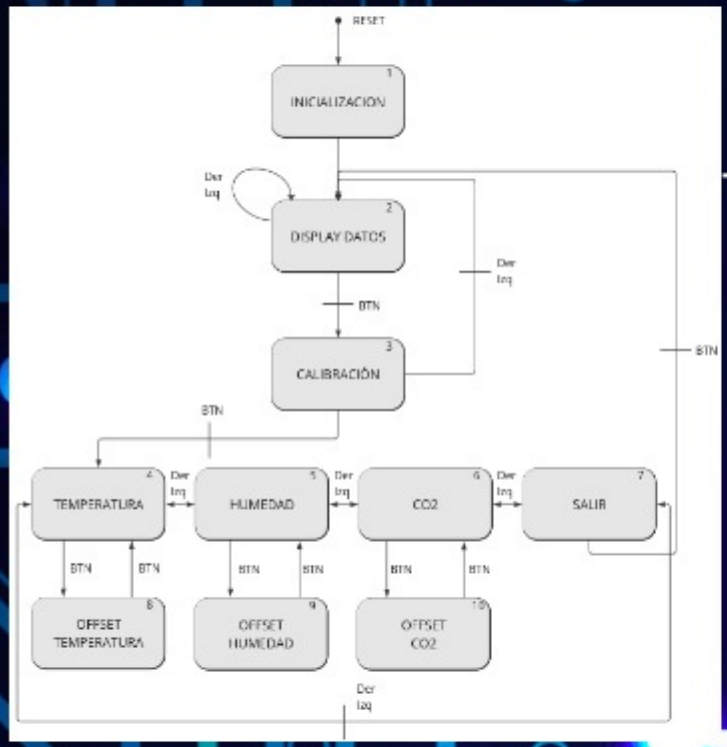
```
GPIO_InitStruct.Pin       = GPIO_PIN_8|GPIO_PIN_9;
GPIO_InitStruct.Mode      = GPIO_MODE_AF_OD;
GPIO_InitStruct.Pull      = GPIO_NOPULL;
GPIO_InitStruct.Speed     = GPIO_SPEED_FREQ_LOW;
GPIO_InitStruct.Alternate = GPIO_AF4_I2C1;

HAL_GPIO_Init(GPIOB, &GPIO_InitStruct);
```

```
huart_.Instance          = UART_INSTANCE;
huart_.Init.BaudRate     = UART_BAUDRATE;
huart_.Init.Parity       = UART_PARITY_NONE;
huart_.Init.StopBits     = UART_STOPBITS_1;
huart_.Init.WordLength   = UART_WORDLENGTH_8B;
huart_.Init.Mode         = UART_MODE_TX_RX;
huart_.Init.HwFlowCtl    = UART_HWCONTROL_NONE;
huart_.Init.OverSampling = UART_OVERSAMPLING_16;
```

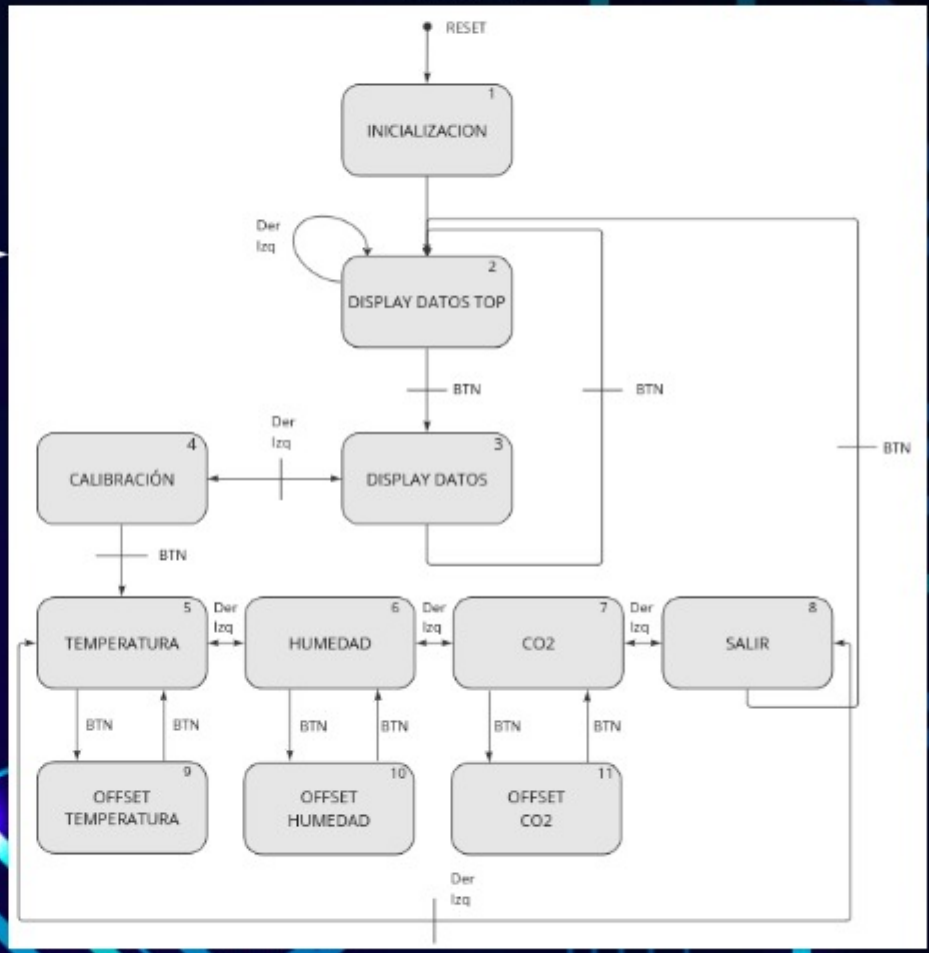
```
hi2c1.Instance             = I2C1;
hi2c1.Init.ClockSpeed      = I2C_CLOCK_RATE;
hi2c1.Init.DutyCycle       = I2C_DUTYCYCLE_2;
hi2c1.Init.OwnAddress1     = 0;
hi2c1.Init.AddressingMode  = I2C_ADDRESSINGMODE_7BIT;
hi2c1.Init.DualAddressMode = I2C_DUALADDRESS_DISABLE;
hi2c1.Init.OwnAddress2     = 0;
hi2c1.Init.GeneralCallMode = I2C_GENERALCALL_DISABLE;
hi2c1.Init.NoStretchMode   = I2C_NOSTRETCH_DISABLE;
```

MEF Principal

INICIAL



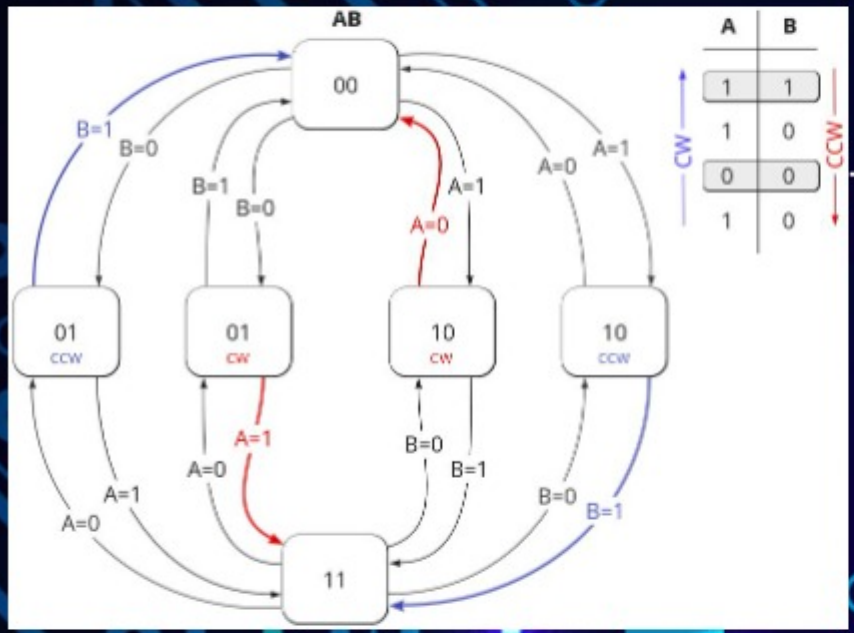
FINAL



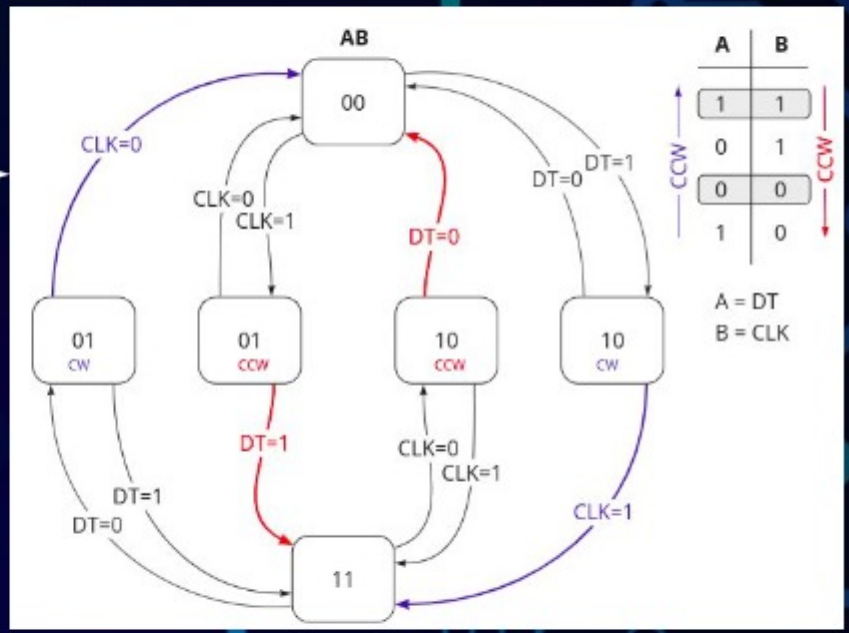
Podoroska Iván - PdM - CESE
Docente - Patricio Bos

MEF Encoder

INICIAL



FINAL



Podoroska Iván - PdM - CESE
Docente - Patricio Bos

Detalle

SETUP

```c
int main(void)
{
    HAL_Init();

    BSP_LED_Init(LED1);
    BSP_LED_Init(LED2);
    BSP_LED_Init(LED3);

    BSP_LED_Off(LED1);
    BSP_LED_Off(LED2);
    BSP_LED_Off(LED3);

    GPIO_Init();

    SystemClock_Config();

    if(uartInit() != true)
    {
        Error_Handler();
    }

    DBN_FSMinit(&swButton,  &Encoder_SW,  &swDelay);

    ENC_setGPIOs(&Encoder_DT, &Encoder_CLK);
    ENC_encoderInit();

    delayInit(&ledDelay, BLINK_TIME);

    if(AMB_MON_Init() != true)
    {
        Error_Handler();
    }
```

LOOP

```c
int main(void)
{
    HAL_Init();

    BSP_LED_Init(LED1);
    BSP_LED_Init(LED2);
    BSP_LED_Init(LED3);

    BSP_LED_Off(LED1);
    BSP_LED_Off(LED2);
    BSP_LED_Off(LED3);

    GPIO_Init();

    SystemClock_Config();

    if(uartInit() != true)
    {
        Error_Handler();
    }

    DBN_FSMinit(&swButton,  &Encoder_SW,  &swDelay);

    ENC_setGPIOs(&Encoder_DT, &Encoder_CLK);
    ENC_encoderInit();

    delayInit(&ledDelay, BLINK_TIME);

    if(AMB_MON_Init() != true)
    {
        Error_Handler();
    }
```

Podoroska Iván - PdM - CESE
Docente - Patricio Bos

DEMOSTRACIÓN

VIDEO DEMOSTRATIVO: https://drive.google.com/file/d/1c6Spt7LK5JwAIFcHeT-6YWO50yyMYHFK/view?usp=drive_link

DEMO UART: https://drive.google.com/file/d/1jQVafHF2vWL_g4ZWpDJu-ALKsj6rms3a/view?usp=drive_link

Podoroska Iván - PdM - CESE
Docente - Patricio Bos