

```

/*-----*/
Use :Example program showing how to use a quadrature encoder component.
/*-----*/
void FCM_Main()
{

```

```

// Name: Call Component Macro, Type: Call Component Macro: LCDI2C1::Start()
FCD_0be11_LCDI2C1_Start();

```

```

// Name: Call Component Macro, Type: Call Component Macro: QuadEncoder1::Enable()
FCD_017b1_QuadEncoder1_Enable();

```

```

// Name: Interrupt IOC, Type: Interrupt: Enable IOC0
sei();
PCMSK0=0xF;
PCICR |= (1 << PCIE0);

```

```

// Name: Loop, Type: Loop: While 1
while (1)
{

```

```

// Name: Input, Type: Input: B6 -> SwA
FCV_SWA = GET_PORT_PIN(B,6);

```

```

// Name: Decision, Type: Decision: SegnalaErrori > 0?
if (FCV_SEGNALAERRORI > 0)
{

```

```

// Name: Call Component Macro, Type: Call Component Macro: LCDI2C1::Cursor(10, 0)
FCD_0be11_LCDI2C1_Cursor(10, 0);

```

```

// Name: Call Component Macro, Type: Call Component Macro: LCDI2C1::PrintString("PERDITA DI PASSI")
FCD_0be11_LCDI2C1_PrintString("PERDITA DI PASSI", 17);

```

```

} else {

```

```

// Name: Decision, Type: Decision: SWA?
if (FCV_SWA)
{

```

```

// Name: Call Component Macro - Pulsante premuto, Type: Call Component Macro: QuadEncoder1::ResetCounter()
FCD_017b1_QuadEncoder1_ResetCounter();

```

```

// Name: , Type: Calculation:
// Count = 0
FCV_COUNT = 0;

```

```

} else {

```

```

// Name: Decision, Type: Decision: Count = 360?
if (FCV_COUNT == 360)
{

```

```

// Name: Call Component Macro, Type: Call Component Macro: QuadEncoder1::ResetCounter()
FCD_017b1_QuadEncoder1_ResetCounter();

```

```

// Name: , Type: Calculation:
// Count = 0
FCV_COUNT = 0;

```

```

} else {

```

```

// Name: Call Component Macro, Type: Call Component Macro: LCDI2C1::Clear()
FCD_0be11_LCDI2C1_Clear();

```

```

// Name: Call Component Macro, Type: Call Component Macro: Count=QuadEncoder1::ReadCounter()
FCV_COUNT = FCD_017b1_QuadEncoder1_ReadCounter();

```

```

// Name: Call Component Macro, Type: Call Component Macro: LCDI2C1::PrintNumber(Count)
FCD_0be11_LCDI2C1_PrintNumber(FCV_COUNT);

```

```

}

```

```

}

```

```

}

```

```

// Name: Decision, Type: Decision: Delay > 1000?
if (FCV_DELAY > 1000)
{

```

```

} else {

```

```

// Name: , Type: Calculation:
// Delay = Delay + 1
FCV_DELAY = FCV_DELAY + 1;

```

```

}

```

```

}

```

```

}

```

```
/*-----*\
Use :sottoprogramma chiamato dall'Interrupt
\*-----*/
void FCM_Call_IRQ()
{
```

```
    // Name: Call Component Macro, Type: Call Component Macro: changed=QuadEncoder1::CheckForChanges()
    FCV_CHANGED = FCD_017b1_QuadEncoder1__CheckForChanges();
```

```
    // Name: verifica perdita di passi, Type: Decision: changed = 255?
    if (FCV_CHANGED == 255)
    {
```

```
        // Name: Calculation, Type: Calculation:
        // SegnalaErrori = 1
        FCV_SEGNALAERRORI = 1;
```

```
    // } else {
```

```
    }
```

```
}
```