

ДОДАТОК А

ЛІСТИНГ ПРОГРАМИ

Програмне налаштування конвеєрної лінії.

Приклад коду:

```

REGION обробка помилок та аварій

#Error :=
#Error_wd.%X0 OR // 01 помилка секції при прийомі
#Error_wd.%X1 OR // 02 помилка секції при видачі
#Error_wd.%X2 OR // 03 помилка частотного перетворювача
#Error_wd.%X3 OR // 04 зміна напрямку руху на зайнятій зоні
#Error_wd.%X4 OR // 05 немає зв'язку по PROFINET з частотним
перетворювачем
#Error_wd.%X5 OR // 06 РЕЗЕРВ
#Error_wd.%X6; // 07 помилка по сигналу "ПОЖЕЖА"

#Error_wd.%X7 := #Error;

//скидуємо біти помилок та поточний стан конвеєра
IF #Reset THEN
    #Error_wd := 0;
    #Error_id := 0;
END_IF;

// 01 помилка секції при прийомі
IF #State_zone = 255 AND #State_last = 2 THEN
    #Error_wd.%X0 := TRUE;
END_IF;

// 02 помилка секції при видачі
IF #State_zone = 255 AND #State_last = 4 THEN
    #Error_wd.%X1 := TRUE;
END_IF;

// 03 помилка частотного перетворювача
IF #Fault THEN
    #Error_wd.%X2 := TRUE;
END_IF;

// 04 зміна напрямку руху на зайнятій зоні
IF #Busy AND #Direction <> (#iRevers XOR #iLR) THEN
    #Error_wd.%X3 := TRUE;
END_IF;

// 05 немає зв'язку по PROFINET з частотним перетворювачем
IF "AlwaysFALSE" THEN
    #Error_wd.%X4 := TRUE;
END_IF;

// 06 РЕЗЕРВ
IF "AlwaysFALSE" THEN
    #Error_wd.%X5 := TRUE;
END_IF;

// 07 помилка по сигналу "ПОЖЕЖА"

```

```

THEN
    IF #iFire AND (#State_zone = 0 OR #State_zone = 1 OR #State_zone = 5)
        #Error_wd.%X6 := TRUE;
    END_IF;

    //фіксація першого значення кода помилки
    IF #Error_id = 0 THEN
        #t_error_id := #Error_wd AND B#16#7F;
        WHILE (#t_error_id > 0) DO
            #Error_id := #Error_id + 1;
            #t_error_id := SHR(IN := #t_error_id, N := 1);
        END_WHILE;
    END_IF;

END_REGION

REGION формуємо основні сигнали керування

    //встановлюємо напрямок руху
    #Direction := #iRevers XOR #iLR;

    //призначаємо датчик зони та датчик початку зони
    IF #Direction THEN
        #Sensor_zone := #Sensor_1;
        #Sensor_start := #Sensor_2;
        #State_back := #iState_2;
        #State_next := #iState_1;
    ELSE
        #Sensor_zone := #Sensor_2;
        #Sensor_start := #Sensor_1;
        #State_back := #iState_1;
        #State_next := #iState_2;
    END_IF;

    //вмикаємо таймер для контролю руху конвеєра
    #Time_move(IN := #Motion,
        PT := #iTime_move);

    //вмикаємо таймер часу роботи секції після видачі палети
    #Time_after(IN := #State_zone = 3,
        PT := #iTime_after);

    //вмикаємо таймер для автоматичного скидання помилки
    #Time_reset(IN := #Error AND NOT #t_reset AND #iAuto_reset AND NOT
#iFire,
        PT := #iTime_reset,
        Q => #t_reset);

    //формуємо імпульс скидання помилки вручну
    #i_reset := #iReset AND NOT #f_reset AND NOT #iAuto_reset AND NOT
#iFire;
    #f_reset := #iReset;

    //формуємо сигнал скидання
    #Reset := #t_reset OR #i_reset;

    //переводимо логіку в помилку
    IF #Error THEN
        #State_zone := 255;
    END_IF;

```

```

    #Busy := (#Sensor_start OR #State_zone <> 1) AND #iEnable;

END_REGION

REGION керування частотним перетворювачем секції конвеєра

    //розраховуємо задану швидкість (перерахунок відсотків)
    #Setpoint := REAL_TO_INT(#MAX_SPEED * MIN(IN1 := 100, IN2 := #iSpeed)
/ 100);

    //керування частотним перетворювачем
    #FC_Nord(HW_Identifier := #iHW_id,
        Enable := #Motion,
        Disable_Voltage := NOT #iEnable,
        Quick_Stop := #Error,
        Fault_ACK := #Reset,
        Direction := #Direction,
        Setpoint_1 := #Setpoint,
        Fault => #Fault,
        Operation_Enabled => #Operation,
        Actual_Value_2 => #Sensors);

    //читаємо з частотного перетворювача датчики каретки
    #Sensor_1 := NOT #Sensors.%X0;
    #Sensor_2 := NOT #Sensors.%X1;
    #SensorFloor := #Sensors.%X2;
    #SensorGap := #Sensors.%X3;

END_REGION

REGION логіка роботи зони

    //запам'ятовуємо останній стан зони перед зміною
    #State_last := #State_zone;

    //логіка роботи зони
    CASE #State_zone OF

        0:
            IF #Sensor_zone THEN
                #State_zone := 5;
            ELSIF #Time_move.Q THEN
                #State_zone := 1;
            END_IF;

        1:
            IF #Sensor_zone THEN
                #State_zone := 5;
            ELSIF #State_back = 5 THEN
                #State_zone := 2;
            END_IF;

        2:
            IF #Time_move.Q THEN
                #State_zone := 255;
            ELSIF #Sensor_zone THEN
                #State_zone := 5;
            END_IF;

        3:
            IF #Sensor_zone THEN
                #State_zone := 5;

```

```

        ELSIF #Time_after.Q THEN
            #State_zone := 1;
        END_IF;

4:
    IF #Time_move.Q THEN
        #State_zone := 255;
    ELSIF NOT #Sensor_zone THEN
        #State_zone := 3;
    END_IF;

5:
    IF NOT #Sensor_zone THEN
        #State_zone := 1;
    ELSIF #State_next = 2 THEN
        #State_zone := 4;
    END_IF;

254:
    IF NOT #Operation AND NOT #Motion THEN
        #State_zone := 0;
    END_IF;

255:
    IF #Reset THEN
        #State_zone := 254;
    END_IF;

ELSE
    #State_zone := 0;

END_CASE;

#Motion := (#State_zone = 0 OR #State_zone = 2 OR #State_zone = 3 OR
#State_zone = 4) AND #iEnable;

IF NOT #iEnable THEN
    #State_zone := 0;
END_IF;

END_REGION

REGION встановлюємо значення виходів

#oBusy := #Busy;
#oState := #State_zone;
#oMotion := #Motion;
#oError := #Error;
#oError_id := #Error_id;

END_REGION

```

ДОДАТОК Б
ДЕМОНСТРАЦІЙНИЙ МАТЕРІАЛ

