

**ĐẠI HỌC QUỐC GIA TP.HỒ CHÍ MINH**  
**TRƯỜNG ĐẠI HỌC BÁCH KHOA**  
**KHOA ĐIỆN – ĐIỆN TỬ**  
**BỘ MÔN TỰ ĐỘNG**

-----o0o-----



**ĐỒ ÁN MÔN HỌC**

**TÌM HIỂU PHẦN MỀM SIMIT – TIA PORTAL V16**  
**– PLCSIM ADVANCED – C#**

**GVHD: Trương Đình Châu**

**SVTH: Nguyễn Hoàng Dương**

**MSSV: 1810883**

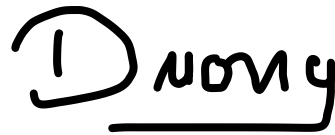
**TP. HỒ CHÍ MINH, THÁNG 07 NĂM 2021**

## *LỜI CẢM ƠN*

*Em xin chân thành cảm ơn Thầy Châu đã giúp đỡ, quan tâm, hướng dẫn chúng em hoàn thành tốt bài báo cáo này trong thời gian qua.*

*Tp. Hồ Chí Minh, ngày 12 tháng 07 năm 2021 .*

**Sinh viên**

A handwritten signature in black ink, appearing to read 'Dương' with a stylized 'D' and 'y'.

**Nguyễn Hoàng Dương**

## TÓM TẮT ĐỒ ÁN

Đồ án này trình bày về quá trình tìm hiểu, thiết kế và kết nối các thành phần trong lĩnh vực Digitalization và Automation.

## MỤC LỤC

1. GIỚI THIỆU .....	1
1.1 Tổng quan.....	1
1.2 Nhiệm vụ đề tài .....	1
2. THIẾT KẾ VÀ THỰC HIỆN PHẦN MỀM .....	1
2.1 SIMIT V10 .....	2
2.1.1 Các Component được sử dụng trong đề tài.....	2
2.1.2 Tạo Marco và Scale các giá trị trao đổi trong project .....	5
2.1.3 Thiết kế hệ thống cho project .....	6
2.2 TIA Portal V16.....	7
2.2.1 Chương trình điều khiển .....	8
2.2.2 Bảng I/O liên kết với SIMIT của chương trình .....	14
2.3 C# Window Form (Visual Studio 2019) .....	14
3. KẾT QUẢ THỰC HIỆN.....	22
3.1 Quá trình mô phỏng.....	22
3.2 Link video kết quả chạy mô phỏng.....	24
4. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN .....	25
4.1 Kết luận .....	25
4.2 Hướng phát triển .....	25
5. TÀI LIỆU THAM KHẢO .....	25
6. PHỤ LỤC .....	25

## 1. GIỚI THIỆU

### 1.1 Tổng quan

Tìm hiểu về Digital Twin sử dụng phần mềm Simit. Mô hình được điều khiển bởi PLC và lập trình trên TIA Portal V16 và giám sát bởi hệ thống SCADA sử dụng .NET Framework.

### 1.2 Nhiệm vụ đề tài

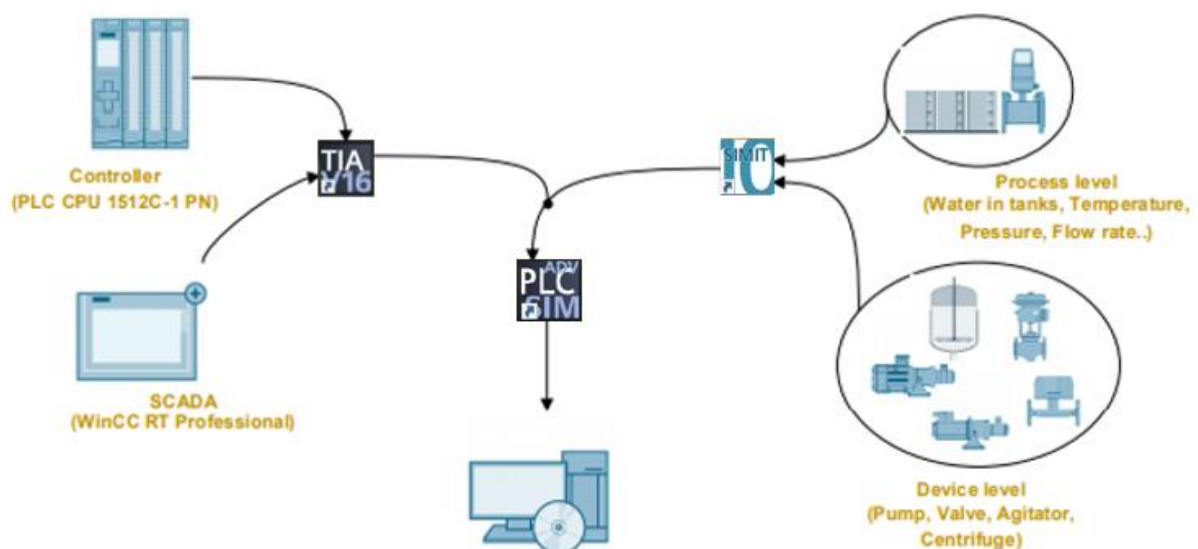
Nội dung 1: Tìm hiểu nguyên lý, cách thức hoạt động của các khối chức năng và công dụng các công cụ trên phần mềm Simit.

Nội dung 2: Lập trình các tín hiệu điều khiển bằng TIA Portal V16.

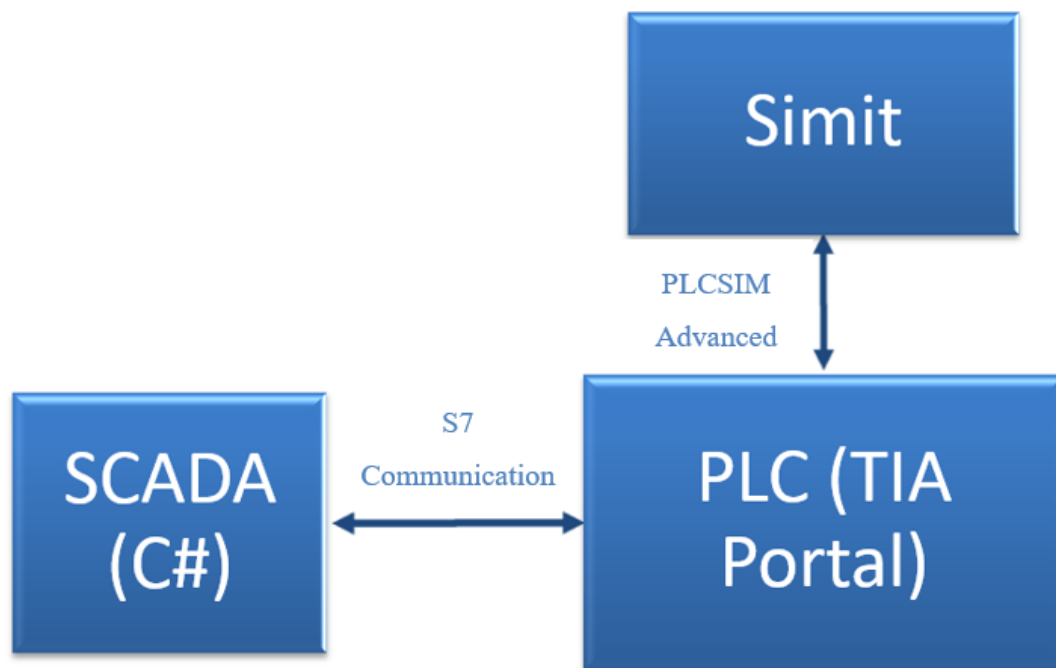
Nội dung 3: Kết nối hệ thống ảo Simit với PLC thông qua PLCSim Advanced.

Nội dung 4: Thiết kế giao diện SCADA bằng C#. Kết nối với PLC thông qua thư viện S7.

## 2. THIẾT KẾ VÀ THỰC HIỆN PHẦN MỀM



Hình 2.1 Tổng quan đề tài

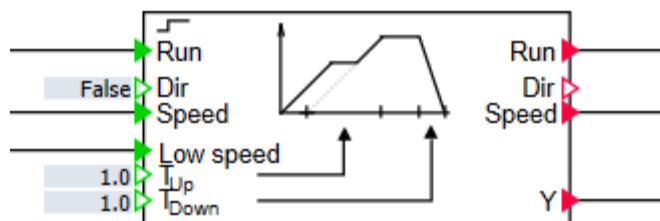


Hình 2.2 Sơ đồ thiết kế

## 2.1 SIMIT V10

### 2.1.1 Các Component được sử dụng trong đề tài

#### ➤ Drive P1



Hình 2.3 Drive P1

#### Ngõ vào:

**Run:** ngõ vào Binary, kích hoạt Drive, Drive ON khi Run là 1, OFF khi Run là 0.

**Dir:** Ngõ vào Binary, chọn chiều tăng giảm cho ngõ ra Y.

**Speed:** Ngõ vào Binary, chọn chế độ hoạt động cho drive, khi Speed là 1 thì ngõ ra Y có giá trị tối đa là 100, ngược lại ngõ ra Y có giá trị tối đa bằng Low speed.

**Low speed:** Đặt giá trị cho ngõ ra Y khi Speed là 0.

**Tup:** Thời gian ngõ ra Y đạt giá trị tối đa.

**Tdown:** Thời gian ngõ ra Y về 0.

**Ngõ ra:**

**Run:** Ngõ ra binary, bằng 1 khi ngõ vào Run bằng 1.

**Dir:** Ngõ ra binary, thể hiện chiều hiện tại của drive.

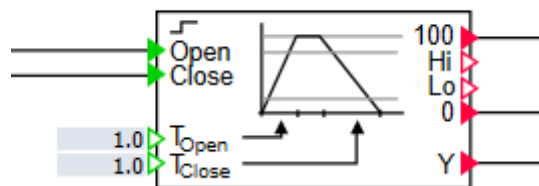
**Speed:** Ngõ ra binary, thể hiện chế độ hoạt động hiện tại của drive.

**Y:** Giá trị ngõ ra của drive có tầm từ -100 tới 100.

**Chức năng:**

Điều khiển tốc độ bơm của pump theo thời gian thực. Báo hiệu hoạt động của pump cho người dùng bằng các tín hiệu binary, người dùng có thể sử dụng các tín hiệu này để gửi về SCADA cho mục đích giám sát.

#### ➤ Drive V2:



Hình 2.4 Drive V2

**Ngõ vào:**

**Open:** Ngõ vào binary, kích hoạt drive chạy, drive ON khi Open là 1. Khi Open và Close cùng là 1 hoặc 0 thì drive giữ nguyên trạng thái hoạt động.

**Close:** Ngõ vào binary, kích hoạt drive dừng, drive OFF khi Close là 1.

**Tup:** Thời gian ngõ ra Y đạt giá trị tối đa.44

**Tdown:** Thời gian ngõ ra Y về 0.

**Ngõ ra:**

**100:** Ngõ ra binary, bằng 1 khi Y đạt giá trị tối đa.

**0:** Ngõ ra binary, bằng 1 khi Y về 0.

**Hi:** Ngõ ra binary, bằng 1 khi drive chạy.

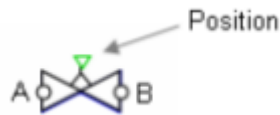
**Lo:** Ngõ ra binary, bằng 1 khi drive dừng.

**Y:** Giá trị ngõ ra của drive có tầm từ 0 tới 100.

**Chức năng:**

Điều khiển độ mở của valve theo thời gian thực. Báo hiệu hoạt động của valve cho người dùng bằng các tín hiệu binary, người dùng có thể sử dụng các tín hiệu này để gửi về SCADA cho mục đích giám sát.

➤ **Valve:**



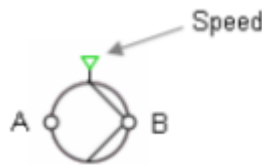
Hình 2.5 Valve

Valve đóng mở được điều khiển bằng ngõ vào Positon (0-100%) lấy từ ngõ ra của Drive V2.

Áp suất rơi trên Valve bằng áp suất điểm B trừ áp suất điểm A.

Chiều của dòng chảy phụ thuộc vào lưu lượng  $> 0$  hay  $< 0$ .

➤ **Pump:**



Hình 2.6 Pump

Pump được điều khiển tốc độ bằng chân Speed (0-100%) lấy từ ngõ ra của Drive P1.

Pump cũng có thể hiển thị chiều dòng chảy.



Hình 2.1.5 Chiều của Pump

➤ **PTnode:**

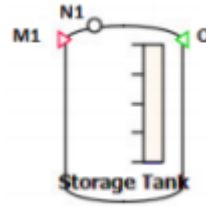


Hình 2.7 PTnode

PTnode mô phỏng các đầu giới hạn của flow network, mô phỏng môi trường nơi cuối cùng các đường ống tiếp xúc, ở đây PTnode cho phép ta cấu hình nhiệt độ và áp suất. Ta có thể thay đổi áp suất và nhiệt độ cho PTnode kể cả khi đang mô phỏng.

➤ **StorageTankLiquid:**





Hình 2.8 StorageTank

StorageTankLiquid mô phỏng bể chứa chất lỏng kín với môi trường ngoài và cho phép người dùng cấu hình môi trường chất lỏng chứa trong bể. Các hợp chất hay chất lỏng trong bể đều có thể được cấu hình bởi người dùng bằng cách điều chỉnh các thông số của nó.

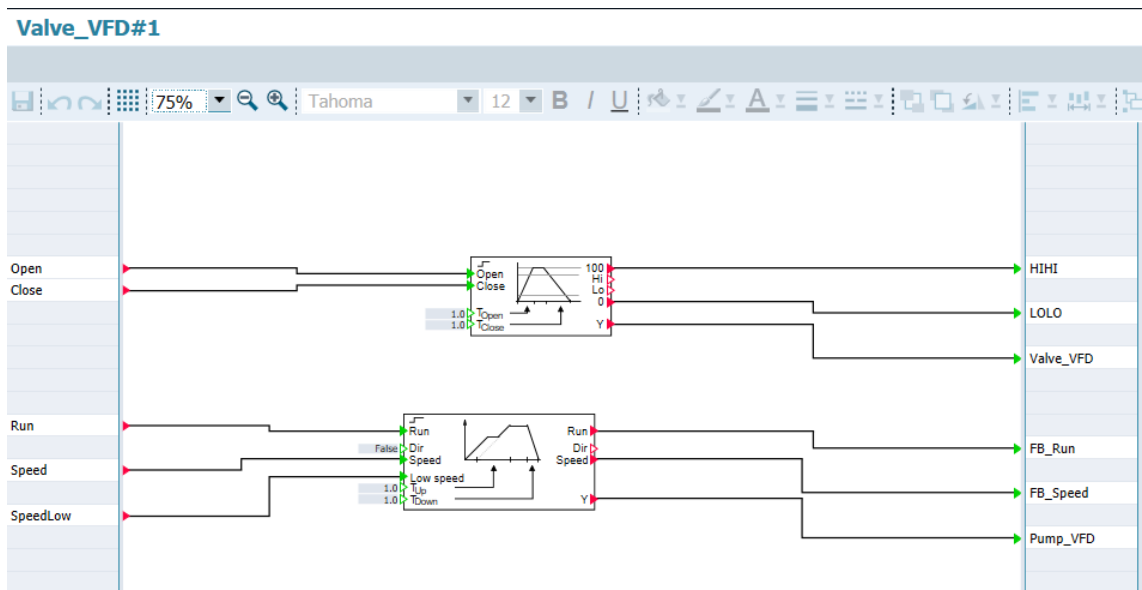
Các điểm nối N là nơi chất lỏng chảy vào và ra khỏi bể.

Điểm nối Q là nơi trao đổi nhiệt độ với các thiết bị heating, sử dụng cho việc đun nóng hay làm lạnh chất lỏng trong bể.

Điểm nối M để kết nối với các thiết bị đo như LevelIndicator.

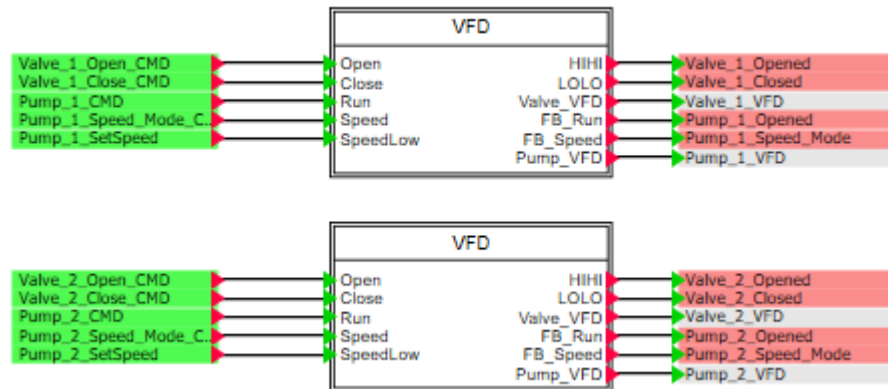
### 2.1.2 Tạo Marco và Scale các giá trị trao đổi trong project

Ta sử dụng 2 drive trên tất cả các pump và valve trong project nên sẽ có sự lặp lại trong lập trình, vì vậy tạo macro component sẽ giúp ta tiết kiệm thời gian lập trình.



Hình 2.9 Marco VFD

Trong chart System ta sẽ kéo tất cả các macro component cho các pump và valve trong project.



Hình 2.10 Marco

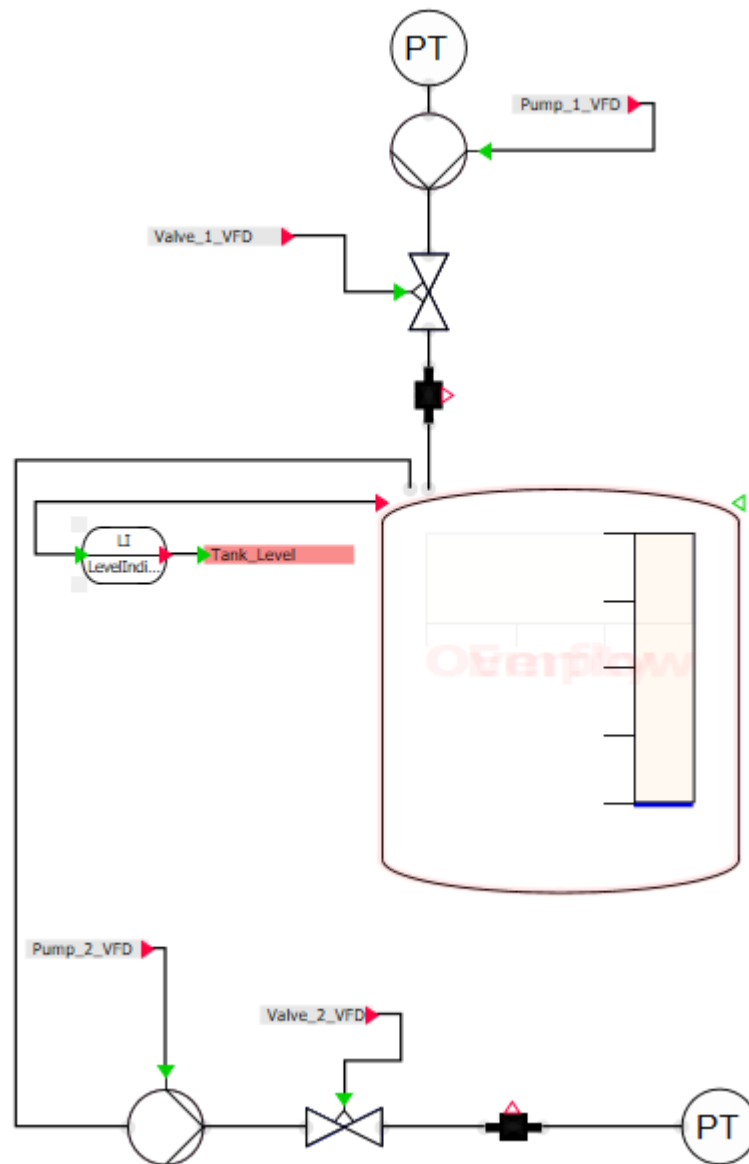
Ta scale các giá trị như mực nước, độ mở valve, tốc độ bơm và tốc độ đặt trực tiếp trong bảng tín hiệu coupling.

▼ Inputs										
Default	Symbol name	Address	Data type	System	Device	Module	Comment	Scaling	Lower	Upper
0.0	Tank_Level	IW0	WORD	0	0	1		Unipolar	0	10
<input type="checkbox"/>		I2.0	BOOL	0	0	1				
<input type="checkbox"/>		I2.1	BOOL	0	0	1				
<input type="checkbox"/>		I2.2	BOOL	0	0	1				
<input type="checkbox"/>		I2.3	BOOL	0	0	1				
<input type="checkbox"/>		I2.4	BOOL	0	0	1				
<input type="checkbox"/>		I2.5	BOOL	0	0	1				
<input type="checkbox"/>		I2.6	BOOL	0	0	1				
<input type="checkbox"/>		I2.7	BOOL	0	0	1				
<input type="checkbox"/>		I3.0	BOOL	0	0	1				
<input type="checkbox"/>		I3.1	BOOL	0	0	1				
▼ Outputs										
Symbol name	Address	Data type	System	Device	Module	Comment	Scaling	Lower	Upper	
Pump_1_SetSpeed	QW0	WORD	0	0	1		Unipolar	0	100	
Pump_2_SetSpeed	QW2	WORD	0	0	1		Unipolar	0	100	
Valve_1_Open_CMD	Q4.0	BOOL	0	0	1					
Valve_1_Close_CMD	Q4.1	BOOL	0	0	1					
Pump_1_CMD	Q4.2	BOOL	0	0	1					
Pump_2_CMD	Q4.3	BOOL	0	0	1					
Valve_2_Open_CMD	Q4.4	BOOL	0	0	1					
Valve_2_Close_CMD	Q4.5	BOOL	0	0	1					
Pump_1_Speed_Mode_C...	Q4.6	BOOL	0	0	1					
Pump_2 Speed Mode C...	Q4.7	BOOL	0	0	1					

Hình 2.11 Scale tín hiệu

### 2.1.3 Thiết kế hệ thống cho project

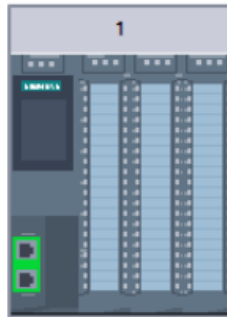
Chart System cho hệ thống:



Hình 2.12 Chart hệ thống

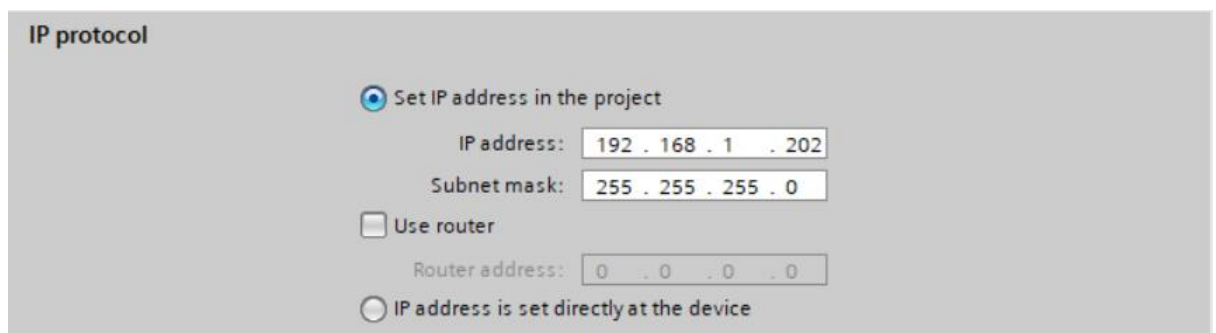
## 2.2 TIA Portal V16

Ta sử dụng phần mềm Tia Portal v16 sp 3 để lập trình chương trình điều khiển và các chế độ hoạt động cho các valve, pump trong Simit. Tia Portal sẽ gửi tín hiệu điều khiển xuống Simit và nhận tín hiệu đáp ứng về để chương trình điều khiển xử lý và SCADA có thể giám sát sự vận hành của hệ thống. Ở đây, ta mô phỏng PLC CPU 1512C-1 PN làm controller cho hệ thống.



Hình 3.1 CPU 1512C-1 PN

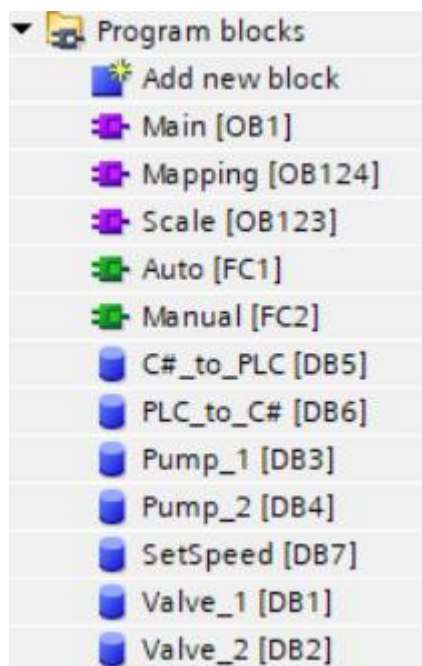
Đặt địa chỉ cho PLC:



Hình 3.2 Set IP cho PLC

### 2.2.1 Chương trình điều khiển

- **Các khối hàm trong chương trình**



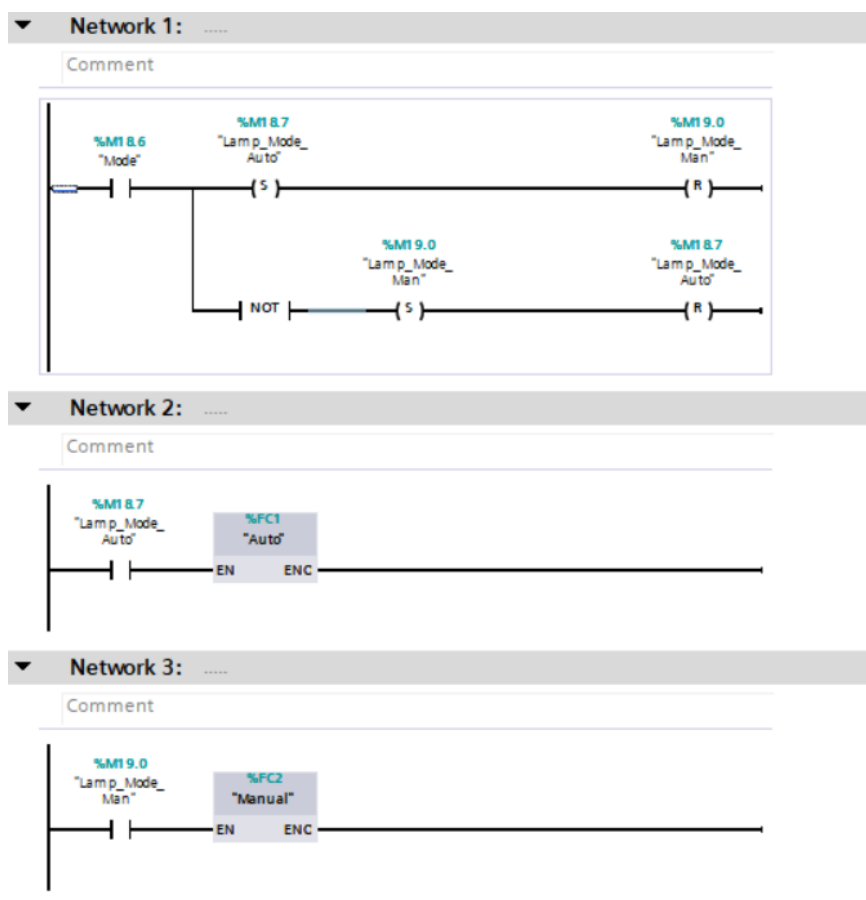
Hình 3.3 Danh sách các block sử dụng trong chương trình

- Cấu trúc chương trình**

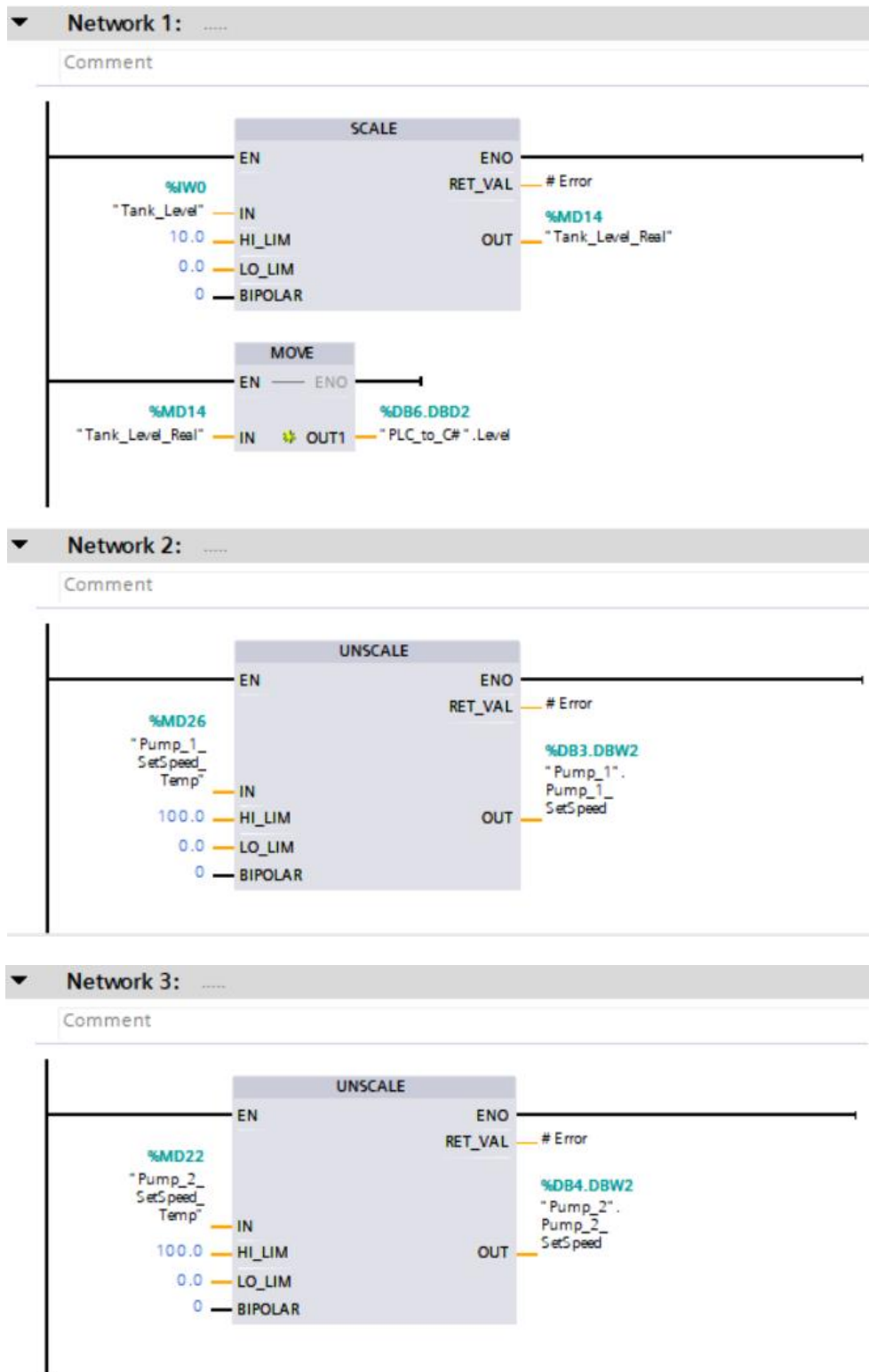
Tên Block	Chức năng
Main (OB1)	Gọi các Function Block điều khiển chương trình
Mapping (OB124)	Đồng bộ các tín hiệu truyền, nhận giữa SCADA, Simit và chương trình điều khiển
Scale (OB123)	Scale các giá trị mực nước và tốc độ bơm từ Simit gửi về Tia Portal và ngược lại (giá trị raw)
Auto ( FC1)	Chương trình điều khiển ở chế độ tự động
Manual (FC2)	Chương trình điều khiển ở chế độ manual
Các khối DB	Chứa các các tín hiệu truyền, nhận giữa SCADA, Simit và chương trình điều khiển

- Bảng biến của các khối DB và các khối chức năng**

- Main (OB1):**



- Scale (OB123)



- Mapping (OB124):

IF...	CASE... OF...	FOR... TO DO...	WHILE... DO...	(*...*)	REGION
1	"Pump_1_CMD" := "Pump_1".Pump_1_CMD;				
2	"Pump_2_CMD" := "Pump_2".Pump_2_CMD;				
3					
4	"Valve_1_Open_CMD" := "Valve_1".Valve_1_Open_CMD;				
5	"Valve_2_Open_CMD" := "Valve_2".Valve_2_Open_CMD;				
6					
7	"Valve_1_Close_CMD" := "Valve_1".Valve_1_Close_CMD;				
8	"Valve_2_Close_CMD" := "Valve_2".Valve_2_Close_CMD;				
9					
10	"Pump_1".Pump_1_Opened := "Pump_1_Opened";				
11	"Pump_2".Pump_2_Opened := "Pump_2_Opened";				
12					
13	"Valve_1".Valve_1_Opened := "Valve_1_Opened";				
14	"Valve_1".Valve_1_Closed := "Valve_1_Closed";				
15	"Valve_2".Valve_2_Opened := "Valve_2_Opened";				
16	"Valve_2".Valve_2_Closed := "Valve_2_Closed";				
17					
18	"Pump_1".Pump_1_Speed_Mode := "Pump_1_Speed_Mode";				
19	"Pump_2".Pump_2_Speed_Mode := "Pump_2_Speed_Mode";				
20					
21	"Pump_1_Speed_Mode_CMD" := "Pump_1".Pump_1_Speed_Mode_CMD;				
22	"Pump_2_Speed_Mode_CMD" := "Pump_2".Pump_2_Speed_Mode_CMD;				
23					
24	"Pump_1_SetSpeed" := "Pump_1".Pump_1_SetSpeed;				
25	"Pump_2_SetSpeed" := "Pump_2".Pump_2_SetSpeed;				
26					
27	"Pump_1_SetSpeed_Temp" := "SetSpeed".Pump_1_SetSpeed_Real;				
28	"Pump_2_SetSpeed_Temp" := "SetSpeed".Pump_2_SetSpeed_Real;				
29					
30	"Open_Pump_1" := "C#_to_PLC".Open_Pump_1;				
31	"Open_Pump_2" := "C#_to_PLC".Open_Pump_2;				
32	"Open_Valve_1" := "C#_to_PLC".Open_Valve_1;				
33	"Open_Valve_2" := "C#_to_PLC".Open_Valve_2;				
34	"Close_Valve_1" := "C#_to_PLC".Close_Valve_1;				
35	"Close_Valve_2" := "C#_to_PLC".Close_Valve_2;				
36					
37	"PLC_to_C#".Lamp_Mode_Auto := "Lamp_Mode_Auto";				
38	"PLC_to_C#".Lamp_Mode_Man := "Lamp_Mode_Man";				
39					
40	"Mode" := "C#_to_PLC".Mode;				

- **Auto (FC1):**

```

1  "Pump_1".Pump_1_CMD := TRUE;
2  "Pump_2".Pump_2_CMD := TRUE;
3  "Valve_1".Valve_1_Open_CMD := TRUE;
4  "Valve_2".Valve_2_Open_CMD := TRUE;
5
6  IF "Tank_Level_Real" < 2 THEN
7      // Statement section IF
8      "Valve_2".Valve_2_Open_CMD := FALSE;
9      "Valve_2".Valve_2_Close_CMD := TRUE;
10     ;
11 END_IF;
12 IF "Tank_Level_Real" > 8 THEN
13     // Statement section IF
14     "Valve_1".Valve_1_Open_CMD := FALSE;
15     "Valve_1".Valve_1_Close_CMD := TRUE;
16     ;
17 END_IF;
18 IF "Tank_Level_Real" > 2 AND "Tank_Level_Real" < 8 THEN
19     // Statement section IF
20     "Valve_1".Valve_1_Open_CMD := TRUE;
21     "Valve_2".Valve_2_Open_CMD := TRUE;
22     "Valve_1".Valve_1_Close_CMD := FALSE;
23     "Valve_2".Valve_2_Close_CMD := FALSE;
24 END_IF;
25
26
27

```

**Giải thích code:** Ở chế độ Auto thì ta đặt giới hạn trên và giới hạn dưới cho mực nước trong Tank.

- Nếu mực nước lớn hơn 8m thì Van 1 sẽ đóng.
- Nếu mực nước xuống dưới 2m thì Van 2 sẽ đóng.
- Nếu mực nước ở khoảng giữa thì 2 Van đều được mở.

- **Manual (FC2):**

```

1  "Pump_1".Pump_1_CMD := "Open_Pump_1";
2  "Pump_2".Pump_2_CMD := "Open_Pump_2";
3
4  "Valve_1".Valve_1_Open_CMD := "Open_Valve_1";
5  "Valve_2".Valve_2_Open_CMD := "Open_Valve_2";
6
7  "Valve_1".Valve_1_Close_CMD := "Close_Valve_1";
8  "Valve_2".Valve_2_Close_CMD := "Close_Valve_2";

```

- **Valve\_1 (DB1):**

Tên	Chức năng
Valve_1_Open_CMD	Điều khiển mở Valve 1
Valve_1_Opened	Nhận tín hiệu valve đã mở từ Simit
Valve_1_Close_CMD	Điều khiển đóng Valve 1
Valve_1_Closed	Nhận tín hiệu valve đã đóng từ Simit



- **Valve\_2 (DB2):**

Tên	Chức năng
Valve_2_Open_CMD	Điều khiển mở Valve 2
Valve_2_Opened	Nhận tín hiệu valve đã mở từ Simit
Valve_2_Close_CMD	Điều khiển đóng Valve 1
Valve_2_Closed	Nhận tín hiệu valve đã đóng từ Simit

- **Pump\_1 (DB3)**

Tên	Chức năng
Pump_1_CMD	Điều khiển mở Pump 1
Pump_1_Opened	Nhận tín hiệu Pump đã mở từ Simit
Pump_1_Speed_Mode	Điều khiển chế độ cho Pump 1
Pump_1_Speed_Mode_CMD	Nhận tín hiệu Pump 1 hoạt động ở chế độ nào từ Simit
Pump_1_SetSpeed	Đặt tốc độ bơm cho Pump

- **Pump\_2 (DB4)**

Tên	Chức năng
Pump_2_CMD	Điều khiển mở Pump 2
Pump_2_Opened	Nhận tín hiệu Pump đã mở từ Simit
Pump_2_Speed_Mode	Điều khiển chế độ cho Pump 2
Pump_2_Speed_Mode_CMD	Nhận tín hiệu Pump 1 hoạt động ở chế độ nào từ Simit
Pump_2_SetSpeed	Đặt tốc độ bơm cho Pump

- **C#\_to\_PLC (DB5)**

Tên	Chức năng
Mode	Chọn chế độ từ SCADA
Open_Valve_1	Mở Valve 1 từ SCADA
Close_Valve_1	Đóng Valve 1 từ SCADA
Open_Valve_2	Mở Valve 2 từ SCADA
Close_Valve_2	Đóng Valve 2 từ SCADA
Open_Pump_1	Mở Pump 1 từ SCADA
Open_Pump_2	Mở Pump 2 từ SCADA

- **PLC\_to\_C# (DB6)**

Tên	Chức năng
Lamp_Mode_Auto	Hiển thị đèn báo Auto
Lamp_Mode_Man	Hiển thị đèn báo Manual
Level	Hiển thị mực nước tank

- **SetSpeed (DB7)**

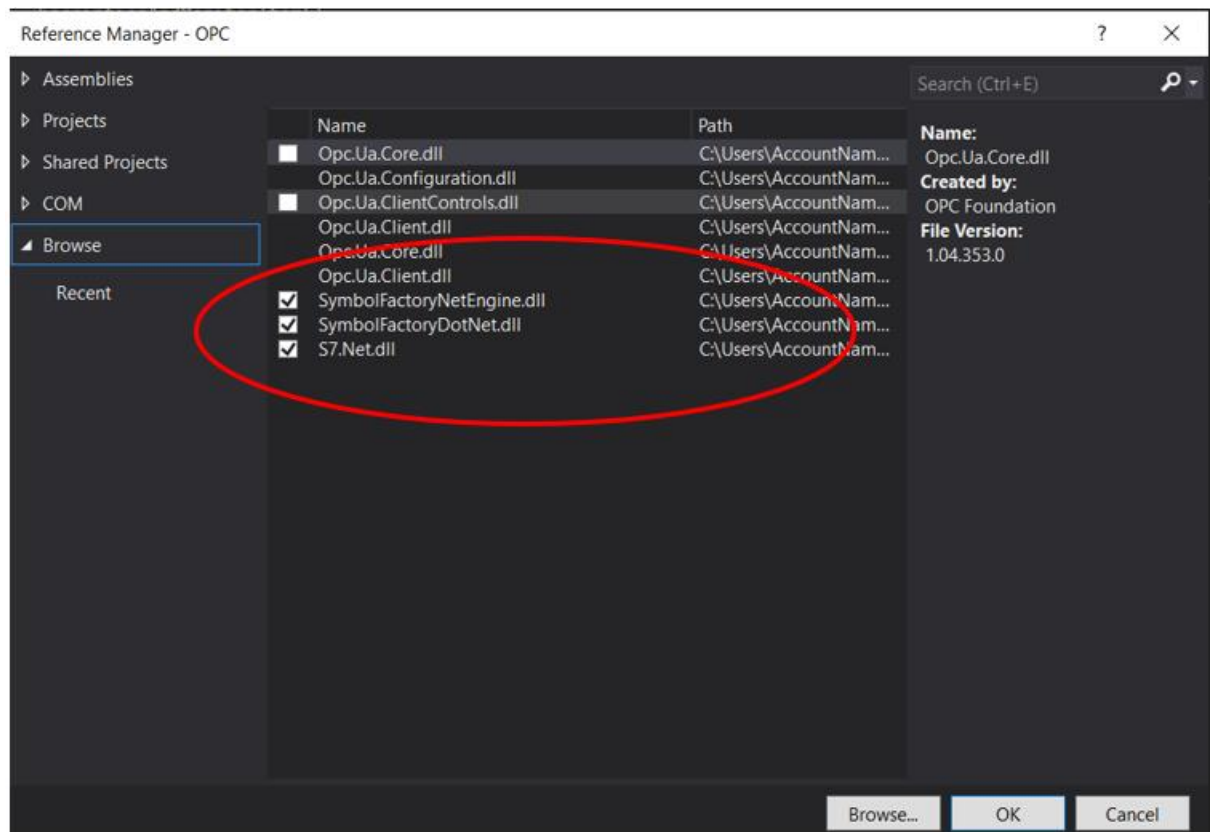
Tên	Chức năng
Pump_1_SetSpeed_Real	Đặt tốc độ bơm cho Pump 1
Pump_2_SetSpeed_Real	Đặt tốc độ bơm cho Pump 2

## 2.2.2 Bảng I/O liên kết với SIMIT của chương trình

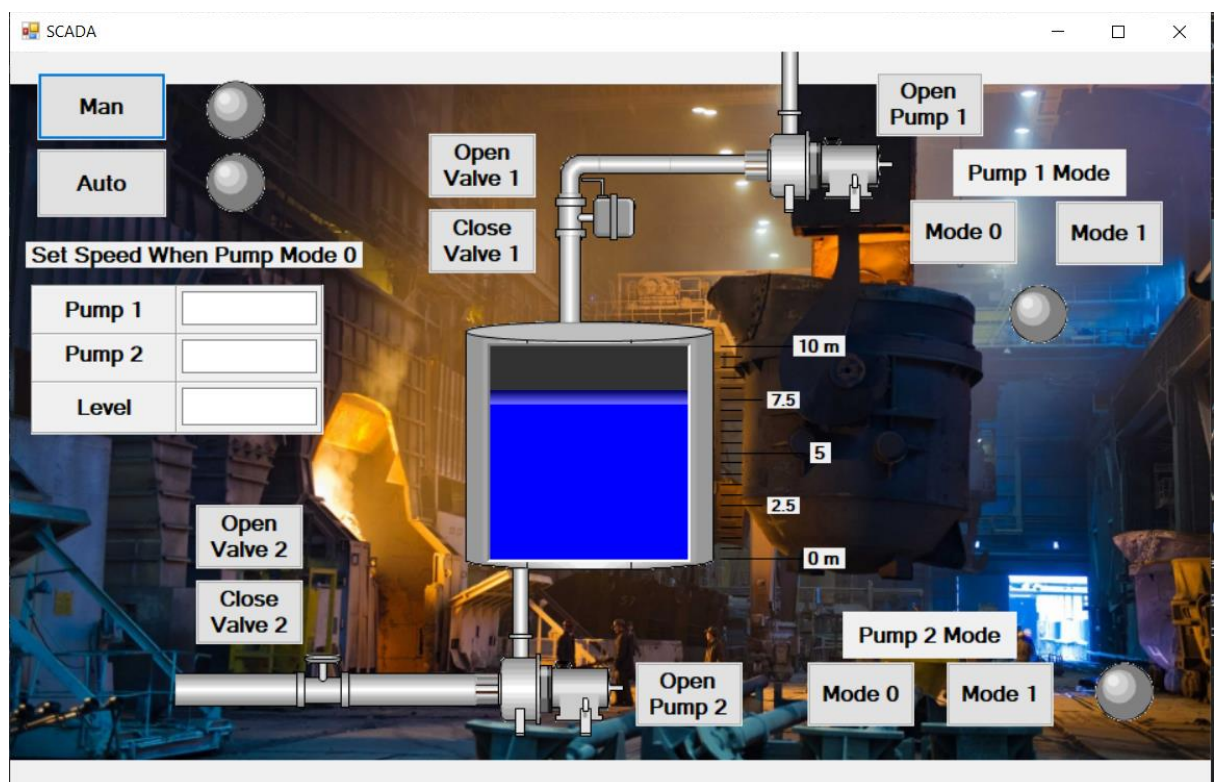
Name	Type	Address	Comment
Valve_1_Open_CMD	Bool	%Q4.0	
Valve_1_Close_CMD	Bool	%Q4.1	
Pump_1_CMD	Bool	%Q4.2	
Pump_2_CMD	Bool	%Q4.3	
Valve_2_Open_CMD	Bool	%Q4.4	
Valve_2_Close_CMD	Bool	%Q4.5	
Pump_1_Speed_Mode_CMD	Bool	%Q4.6	
Pump_2_Speed_Mode_CMD	Bool	%Q4.7	
Valve_1_Opened	Bool	%I10.0	
Valve_1_Closed	Bool	%I10.1	
Pump_1_Opened	Bool	%I10.3	
Pump_1_Speed_Mode	Bool	%I10.4	
Pump_2_Opened	Bool	%I10.5	
Pump_2_Speed_Mode	Bool	%I10.6	
Valve_2_Opened	Bool	%I11.1	
Valve_2_Closed	Bool	%I11.2	
Pump_1_SetSpeed	Int	%QW0	
Pump_2_SetSpeed	Int	%QW2	
Tank_Level	Int	%IW0	

## 2.3 C# Window Form (Visual Studio 2019)

- Tạo 1 project mới và thêm thư viện SymbolFactory để thiết kế giao diện SCADA cho đề tài. Đồng thời thêm thư viện S7.net để có thể giao tiếp với PLC thông qua chuẩn S7.



- Thiết kế giao diện SCADA trên Form Design



- Viết chương trình thực hiện
  - Khai báo 1 PLC mới và tạo các Class để đọc DB từ PLC

```

Plc _Plc = new Plc(CpuType.S71500, "192.168.1.202", 0, 0);
PLCRead _PLCRead = new PLCRead();
PLCReadValve1 _PLCReadValve1 = new PLCReadValve1();
PLCReadValve2 _PLCReadVavle2 = new PLCReadValve2();
PLCReadPump1 _PLCReadPump1 = new PLCReadPump1();
PLCReadPump2 _PLCReadPump2 = new PLCReadPump2();
PLCWrite _PLCWrite = new PLCWrite();
1 reference
private void Form1_Load_1(object sender, EventArgs e)
{
    if (_Plc.Open() == ErrorCode.NoError)
    {
        timer1.Interval = 1000;
        timer1.Enabled = true;
    }
}

```

- PLCRead: Đọc Data từ khối DB6

```

namespace Project_202
{
    2 references
    public class PLCRead
    {
        1 reference
        public bool auto_mode { get; set; }

        1 reference
        public bool man_mode { get; set; }






        2 references
        public double Level { get; set; }
    }
}

```

PLC_to_C#				
		Name	Data type	Offset
1	Static			
2		Lamp_Mode_Auto	Bool	0.0
3		Lamp_Mode_Man	Bool	0.1
4		Level	Real	2.0

- PLCReadValve1: Đọc Data của Valve 1

```
namespace Project_202
{
    2 references
    public class PLCReadValve1
    {
        0 references
        public bool Valve_1_Open_CMD { get; set; }
        1 reference
        public bool Valve_1_Opened { get; set; }
        0 references
        public bool Valve_1_Close_CMD { get; set; }
        0 references
        public bool Valve_1_Closed { get; set; }
        0 references
        public byte Valve_1 { get; set; }
    }
}
```

Valve_1				
		Name	Data type	Offset
1		Static		
2		Valve_1_Open_CMD	Bool	0.0
3		Valve_1_Opened	Bool	0.1
4		Valve_1_Close_CMD	Bool	0.2
5		Valve_1_Closed	Bool	0.3

- PLCReadValve2: Đọc Data của Valve 2

```
namespace Project_202
{
    2 references
    public class PLCReadValve2
    {
        0 references
        public bool Valve_2_Open_CMD { get; set; }
        1 reference
        public bool Valve_2_Opened { get; set; }
        0 references
        public bool Valve_2_Close_CMD { get; set; }
        0 references
        public bool Valve_2_Closed { get; set; }
        0 references
        public byte Valve_2 { get; set; }
    }
}
```

Valve_2				
	Name	Data type	Offset	
1	Static			
2	Valve_2_Open_CMD	Bool	0.0	
3	Valve_2_Opened	Bool	0.1	
4	Valve_2_Close_CMD	Bool	0.2	
5	Valve_2_Closed	Bool	0.3	

- PLCReadPump1: Đọc Data Pump 1

```
namespace Project_202
{
    2 references
    public class PLCReadPump1
    {
        0 references
        public bool Pump_1_CMD { get; set; }
        1 reference
        public bool Pump_1_Opened { get; set; }
        1 reference
        public bool Pump_1_Speed_Mode { get; set; }
        0 references
        public byte Pump_1 { get; set; }
    }
}
```

Pump_1				
	Name	Data type	Offset	
1	Static			
2	Pump_1_CMD	Bool	0.0	
3	Pump_1_Opened	Bool	0.1	
4	Pump_1_Speed_Mode	Bool	0.2	
5	Pump_1_Speed_Mod...	Bool	0.3	
6	Pump_1_SetSpeed	Word	2.0	

- PLCReadPump2: Đọc Data Pump 2

```
namespace Project_202
{
    2 references
    public class PLCReadPump2
    {
        0 references
        public bool Pump_2_CMD { get; set; }
        1 reference
        public bool Pump_2_Opened { get; set; }
        1 reference
        public bool Pump_2_Speed_Mode { get; set; }
        0 references
        public byte Pump_2 { get; set; }
    }
}
```

Pump_2			
	Name	Data type	Offset
1	Static		
2	Pump_2_CMD	Bool	0.0
3	Pump_2_Opened	Bool	0.1
4	Pump_2_Speed_Mode	Bool	0.2
5	Pump_2_Speed_Mod...	Bool	0.3
6	Pump_2_SetSpeed	Word	2.0

- PLCWrite: Write Data từ C# xuống PLC

```

namespace Project_202
{
    2 references
    public class PLCWrite
    {
        1 reference
        public double Pump_1_SetSpeed_Real { get; set; }
        1 reference
        public double Pump_2_SetSpeed_Real { get; set; }
    }
    4 references
    public static class PLCWriteStatic
    {
        2 references
        public static double Pump_1_SetSpeed_Real { get; set; }
        2 references
        public static double Pump_2_SetSpeed_Real { get; set; }
    }
}

```

SetSpeed			
	Name	Data type	Offset
1	Static		
2	Pump_1_SetSpeed_Real	Real	0.0
3	Pump_2_SetSpeed_Real	Real	4.0



- Ngoài ra ta có thể Write trực tiếp đến địa chỉ của Data Block trên PLC

```

1reference
private void buttonMan_Click(object sender, EventArgs e)
{
    _Plc.Write("DB5.DBX0.0", false);
}

1reference
private void buttonAuto_Click(object sender, EventArgs e)
{
    _Plc.Write("DB5.DBX0.0", true);
}

1reference
private void openValve1_Click(object sender, EventArgs e)
{
    _Plc.Write("DB5.DBX0.2", false);
    _Plc.Write("DB5.DBX0.1", true);
}

1reference
private void closeValve1_Click(object sender, EventArgs e)
{
    _Plc.Write("DB5.DBX0.1", false);
    _Plc.Write("DB5.DBX0.2", true);
}

1reference
private void openValve2_Click(object sender, EventArgs e)
{
    _Plc.Write("DB5.DBX0.4", false);
    _Plc.Write("DB5.DBX0.3", true);
}

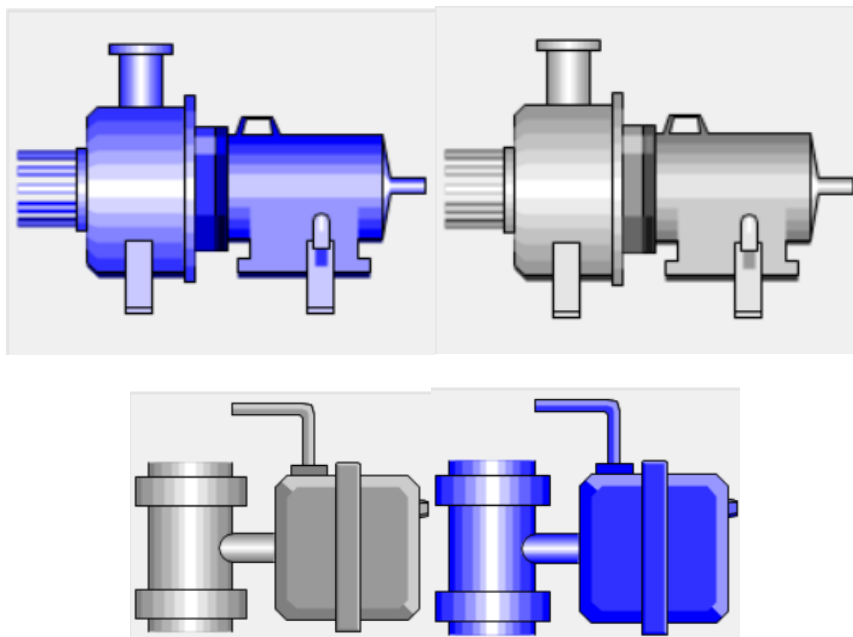
1reference
private void closeValve2_Click(object sender, EventArgs e)
{
    _Plc.Write("DB5.DBX0.3", false);
    _Plc.Write("DB5.DBX0.4", true);
}
    
```

C#_to_PLC			
	Name	Data type	Offset
1	Static		
2	Mode	Bool	0.0
3	Open_Valve_1	Bool	0.1
4	Close_Valve_1	Bool	0.2
5	Open_Valve_2	Bool	0.3
6	Close_Valve_2	Bool	0.4
7	Open_Pump_1	Bool	0.5
8	Open_Pump_2	Bool	0.6

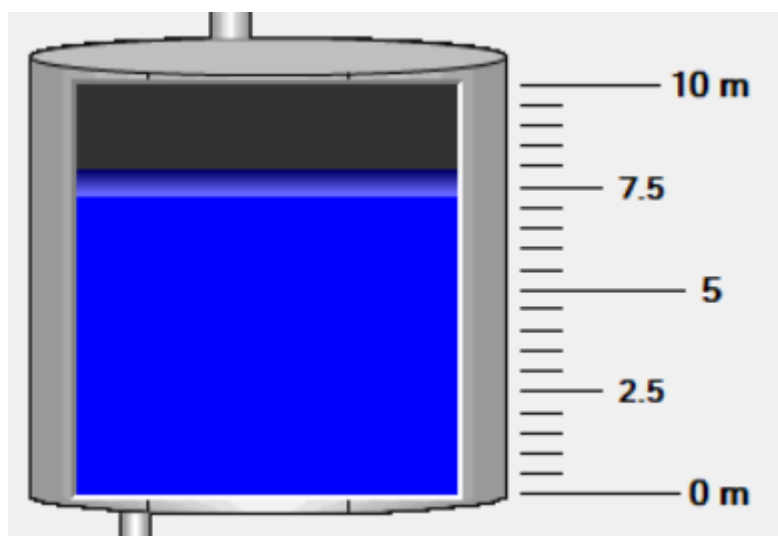


- Khi Pump hay Valve đang hoạt động thì sẽ sáng màu và ngược lại tối màu khi ở chế độ đóng.

```
8 references
private void ON_OFF(StandardControl symbol, bool Value)
{
    if (Value == true)
    {
        symbol.FillColorMode = SymbolFactoryNetEngine.FillColorModeOptions.Shaded;
    }
    else
    {
        symbol.FillColorMode = SymbolFactoryNetEngine.FillColorModeOptions.Original;
    }
}
```



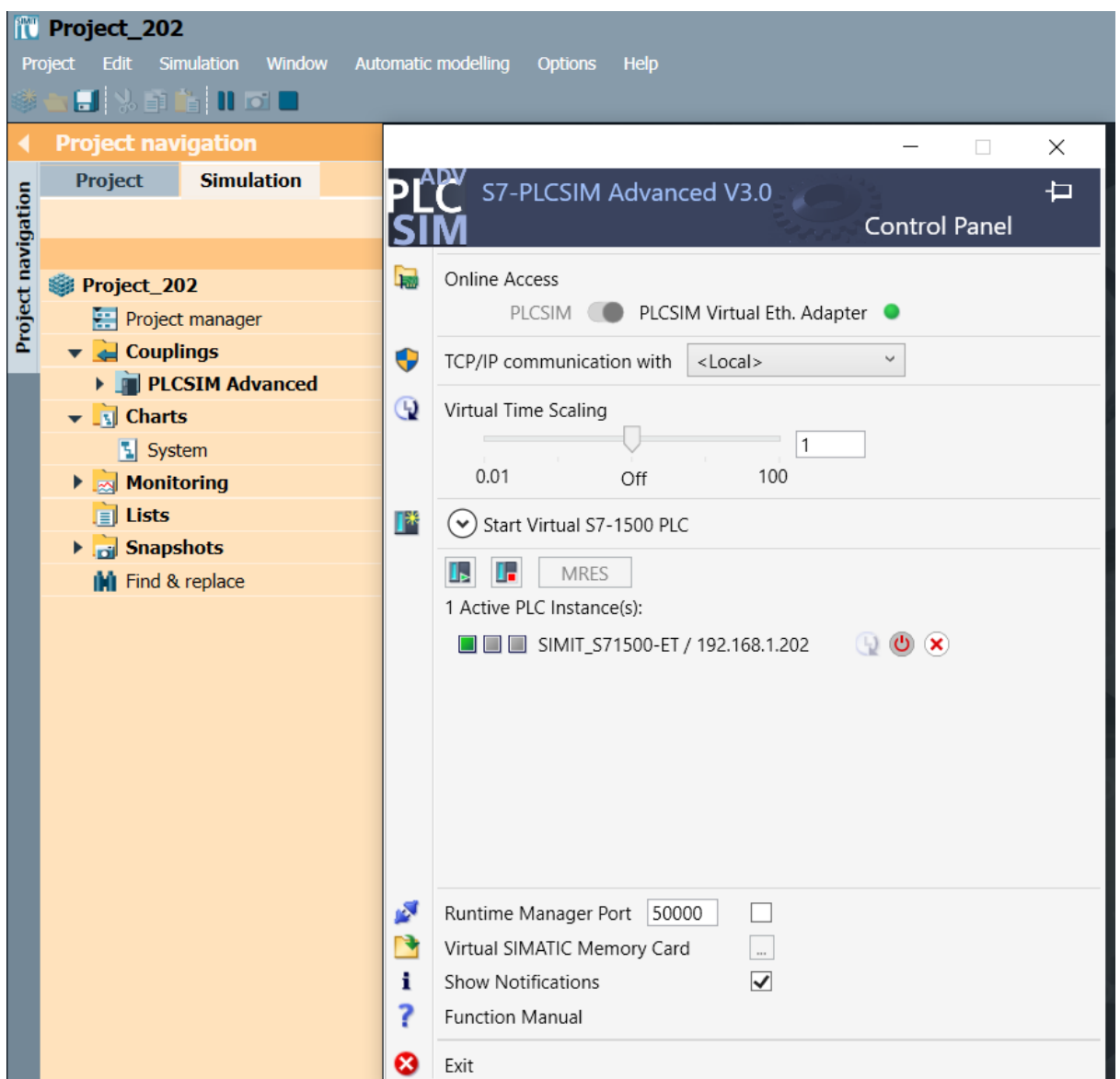
- Sử dụng CutawayControl để thể hiện mực nước trong Tank:

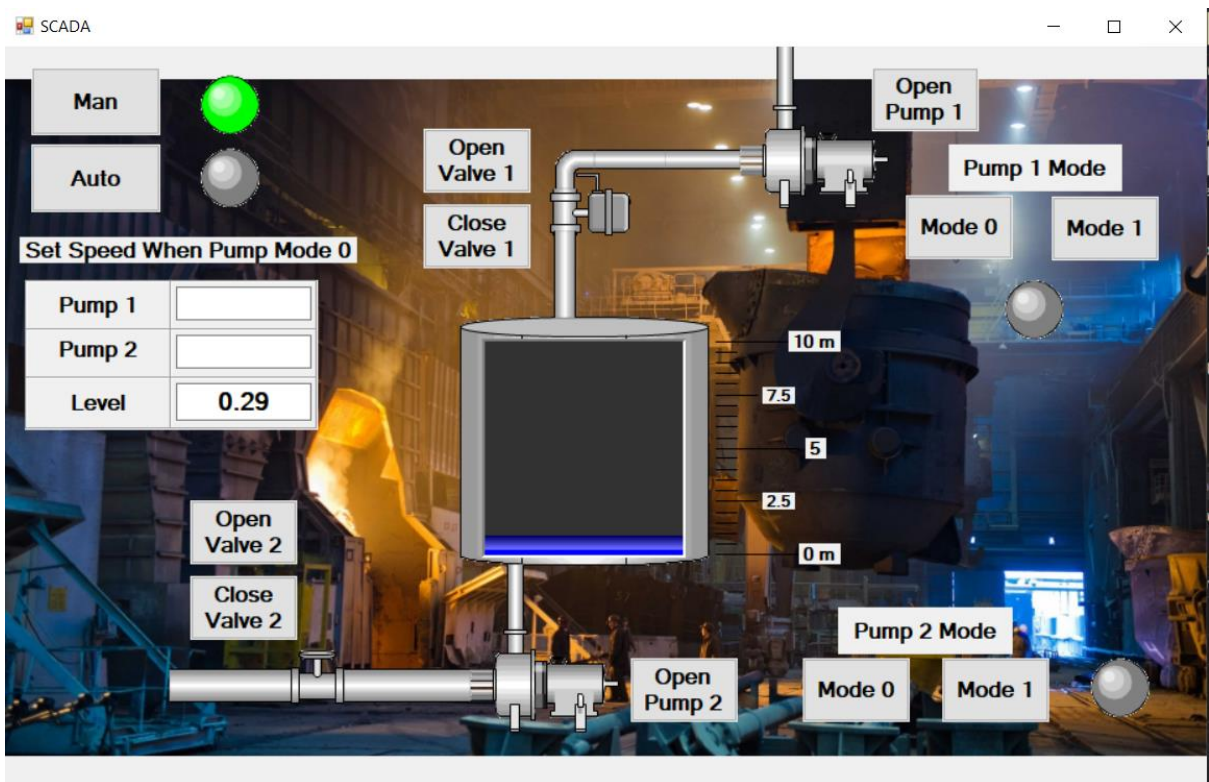
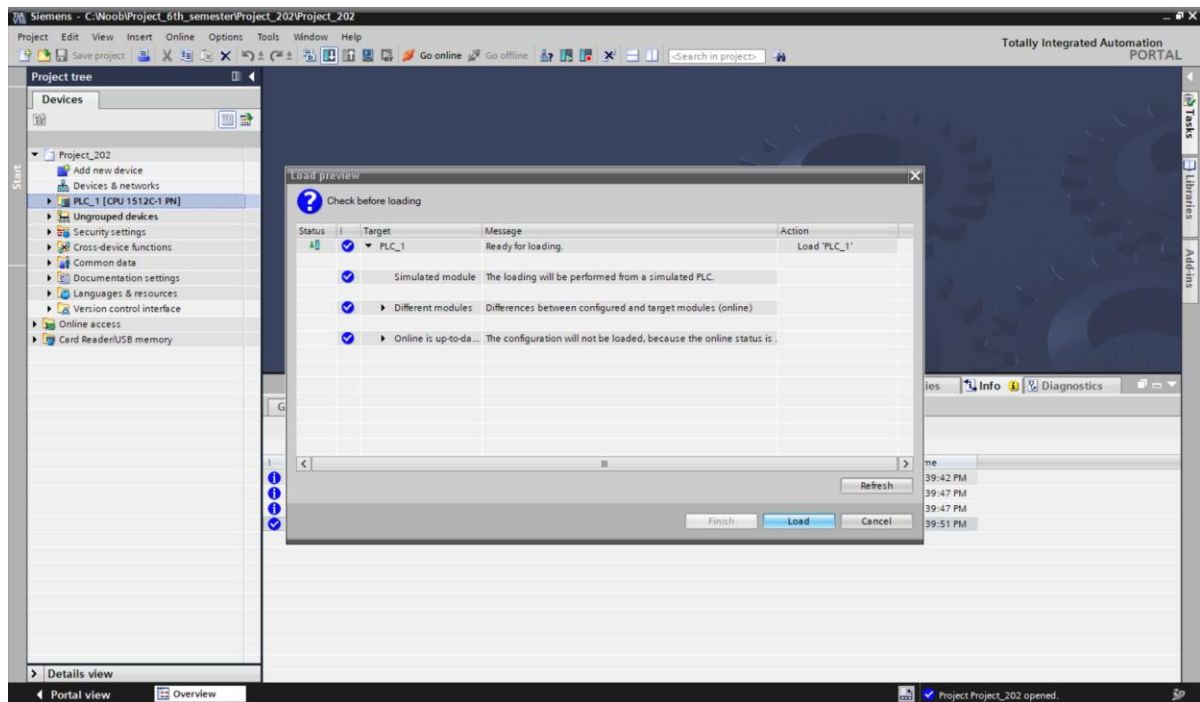


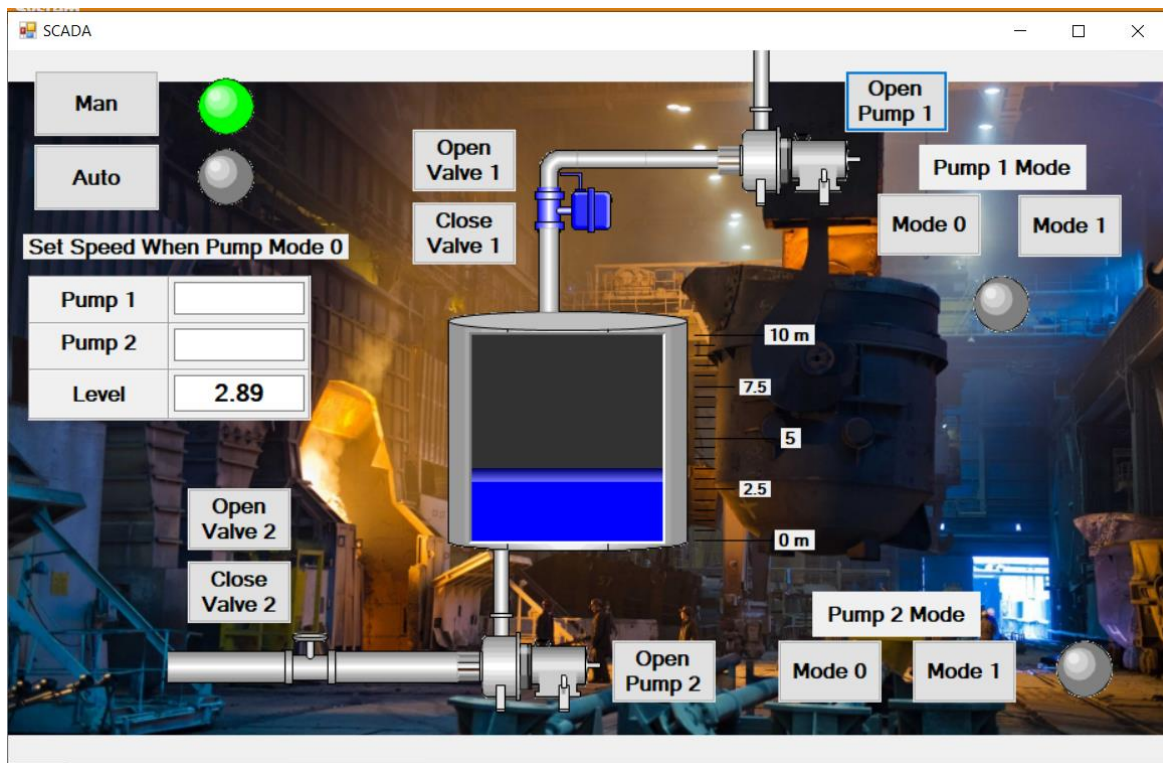
### 3. KẾT QUẢ THỰC HIỆN

#### 3.1 Quá trình mô phỏng

- Khởi chạy các file Simit, TIA Portal, Visual Studio và PLCSIM Advanced.
- Sau khi chạy mô phỏng của Simit thì Simit sẽ tự tạo 1 instance trên PLCSIM Advanced. Sau đó, ta chỉ cần download chương trình từ TIA Portal xuống instance vừa được tạo là có thể điều khiển hệ thống trên Simit thông qua TIA Portal.
- Một số hình ảnh khi chạy mô phỏng:







[https://youtu.be/RMO2t\\_QmEg8](https://youtu.be/RMO2t_QmEg8)

[https://youtu.be/RMO2t\\_QmEg8](https://youtu.be/RMO2t_QmEg8)

## 4. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### 4.1 Kết luận

Qua quá trình tìm hiểu ở trên, đã giúp chúng ta tiếp cận về công nghệ Digital Twin đang được sử dụng rộng rãi trong các nhà máy công nghiệp. Tuy đề tài vẫn còn nhiều thiếu sót nhưng phần nào đã giúp chúng ta hình dung ra được về Digitalization.

### 4.2 Hướng phát triển

Từ đó, chúng ta có thể phát triển thêm đề tài trên bằng cách chạy song song giữa mô hình thực và ảo để có thể phân tích các dữ liệu, sai số và từ đó đưa ra các giải pháp tối ưu nhất.

## 5. TÀI LIỆU THAM KHẢO

- ✓ Link down SymbolFactory Lib:  
<https://mega.nz/file/Iu4l0QCR#cmo1xwHwYmWbyaUvgMiVATF2E3OrYnHTIDKOH8dCjH0>
- ✓ Link tham khảo dự án của thầy Châu:  
[https://drive.google.com/drive/u/1/folders/1e2VOgBQ8YtCZSJ-F\\_YbrnUNM0IKGtm-z](https://drive.google.com/drive/u/1/folders/1e2VOgBQ8YtCZSJ-F_YbrnUNM0IKGtm-z)
- ✓ Link hướng dẫn kết nối PLC với C# bằng thư viện S7.net:  
<https://www.youtube.com/watch?v=Q9OfuvZ08L4&list=PL7i0pecxh8jnyt67MuVuisJpR8Xjz1OzX>

## 6. PHỤ LỤC