



УНИВЕРЗИТЕТ
У НОВОМ САДУ



ФАКУЛТЕТ
ТЕХНИЧКИХ НАУКА

Трг Доситеја Обрадовића 6, 21000 Нови Сад, Југославија
Деканат: 021 350-413; 021 450-810; Централа: 021 350-122
Рачуноводство: 021 58-220; Студентска служба: 021 350-763
Телефакс: 021 58-133; e-mail: ftndeap@uns.ns.ac.yu



DISTRIBUIRANI UPRAVLJAČKI SISTEMI

Zajedničke karakteristike nadzorno-upravljačkih (SCADA) sistema

-SKRIPTA-

Novi sad
2005

1. PODSISTEMI NADZORNO-UPRAVLJAČKOG PROGRAMSKOG SISTEMA	3
2. INTELLUTION IFIX	6
2.1. Dodavanje veličina	6
2.2. Formiranje i animacija slike	7
2.3. DataBinding	8
2.4. Alarmi i trendovi	9
2.5. Regulacija temperature	10
3. OMRON CX-SUPERVISOR	16
3.1. TEHNIČKI OPIS SISTEMA	16
3.2. OMRON SOFTVER	17
3.3. VIZUELIZACIJA	18
3.4. PROGRAMIRANJE	21
4. INTEGRACIJA POSTOJEĆIH SISTEMA	26
4.1. Korišćenje postojećih kontrola	26
4.2. Visual Basic OPC DA klijent	29

1. Podsistemi nadzorno-upravljačkog programskog sistema

Razvoj nadzorno-upravljačkih sistema predstavlja složen problem zbog velikog broja različitih merno-akvizicionih i upravljačkih uređaja i povezivanja sa njima. Ovo se pre svega odnosi na softverske *driver*-e preko kojih se vrši programsko povezivanje uređaja sa ostalim delovima sistema. Do sada su razni proizvođači razvijali svoje *driver*-e po sopstvenim specifikacijama, što je dovelo do nekompatibilnosti uređaja proizvedenih od strane različitih proizvođača.

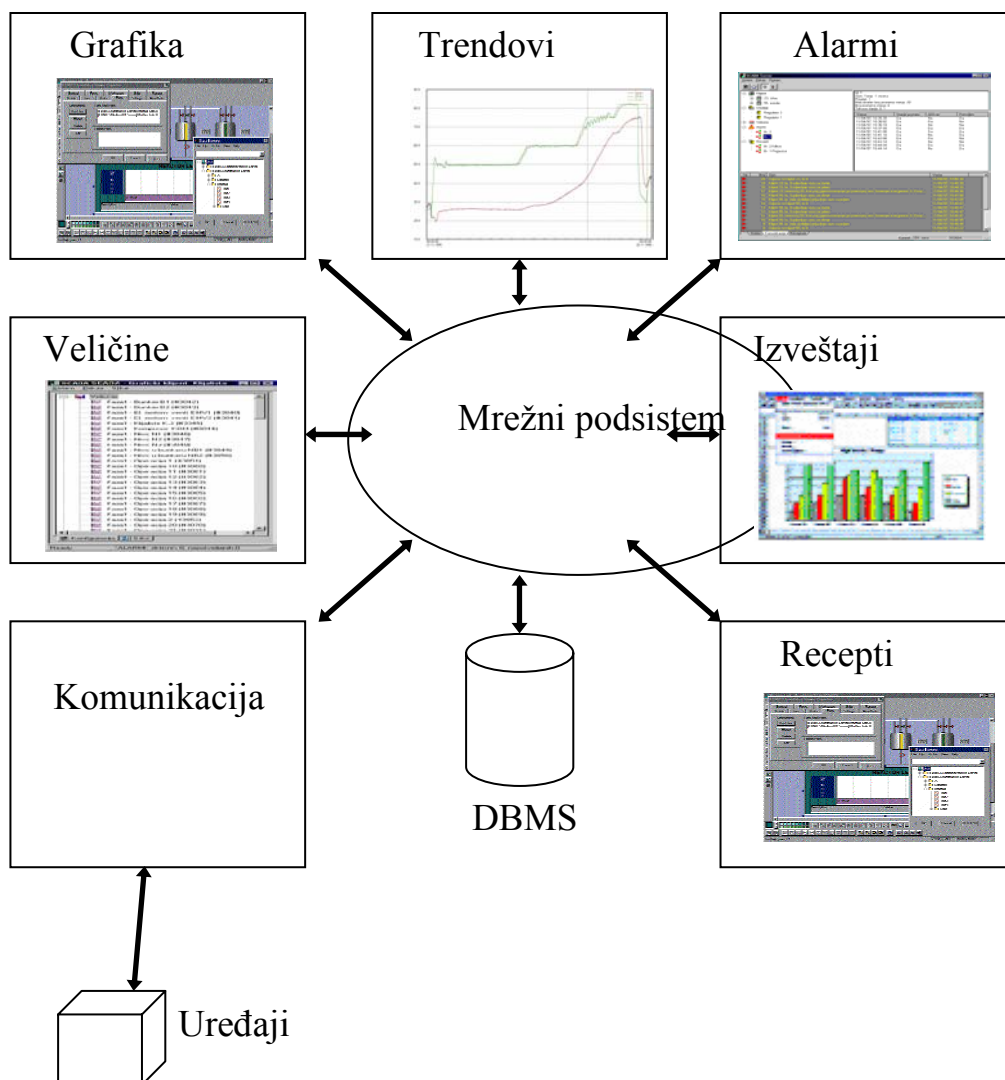
Pri realizaciji nadzorno-upravljačkih sistema primenom SCADA softvera pretpostavlja se da postoji postrojenje sa pratećom mernom opremom i izvršnim organima, da je data tehnološka šema i opis postrojenja, kao i elektro projekat na nivou postrojenja. Tada se SCADA softver projektuje tako da omogući jednostavno specifikiranje svih elemenata sistema, kao i jednostavno projektovanje operatorskog interfejsa i dispečerskih stanica. Pri tome se mora specificirati način komunikacije, čvorovi u mreži, vreme skaniranja pojedinih stanica ili pojedinih signala u stanici, kao i skup (bazu) podataka koji se prate i obrađuju. Širokom primenom OPC DA specifikacije olakšan je pristup veličinama iz aplikacija koje razvijaju različiti proizvođači.

Iako SCADA softvere razvijaju različiti proizvođači iz analize dostupnih sistema mogu se uočiti sličnosti u njihovoj arhitekturi. Takođe se može primetiti postojanje sličnih podsistema kao što su:

1. **Podsistem za definisanje veličina** u kome se definišu veličine i njihove osobine kao što su gornja i donja granica vrednosti veličine, jedinica mere, vreme očitavanja, inženjerska jedinica itd. Ulazne veličine predstavljaju vrednosti izmerenih fizičkih veličina iz procesa, a izlazne veličine vrednosti koje se šalju ka upravljačkim uređajima. Često se mogu definisati i memorijske veličine (koje služe za proračune) i sistemske veličine koje su specifične za upotrebljeni program.
2. **Podsistem za alarme** koji služi za definisanje i prikaz alarmnih stanja u sistemu. Alarmna stanja mogu predstavljati nedozvoljenu ili kritičnu vrednost veličine kao i nedozvoljenu akciju ili komandu operatera. Svaki alarm ima svoje osobine kao što su nivo ozbiljnosti alarma, mesto nastanka, kategorija, poruka koja se vezuje za alarm i slično. Podsistem za alarme omogućuje promenu stanja alarma putem operacije potvrde i brisanja.
3. **Podsistem za prikaz trendova** u kome se prikazuju poslednje promene vrednosti veličina (trendovi u realnom vremenu) i istorijat promene vrednosti veličina u toku dužeg vremenskog perioda (histogrami). Dobro osmišljeni podsistemi za prikaz trendova omogućuju i uporedni prikaz više veličina kao i arhiviranje dijagrama.
4. **Podsistem za recepture** omogućuje zadavanje više vrednosti veličina kao željene promene vrednosti veličina u vremenu.
5. U **podsistemu za izveštaje** se formiraju izveštaji o promenama vrednosti veličina, alarmima, akcijama operatera i ostalim aspektima rada postrojenja.

6. **Grafički podsistem** prikazuje stanje postrojenja u obliku koji je najpregledniji za čoveka (operatera) kako bi on mogao pravovremeno odreagovati na promenu stanja sistema. Osnovna ideja je da se letimičnim pogledom na ekran uoče nepravilnosti u radu postrojenja, da bi se brzo reagovalo i sprečilo neželjeno ponašanje. Vrednosti veličina se najčešće prikazuju u obliku brojeva ili “dinamičkih slika”, čime se olakšava uočavanje promena na slici. Pored prikaza stanja sistema grafički podsistem treba da omogući i izvršavanje neke akcije od strane operatera. Na primer klikom miša na neki objekat može se pokrenuti izvršavanje nekog ranije definisanog makroa ili skripta. U većini dostupnih sistema omogućeno je pisanje makroa u VBA (*Visual Basic for Application*) programskom jeziku koji se odlikuje jednostavnim sintaksom.
7. **Komunikacioni podsistem** omogućuje povezivanje SCADA sistema sa fizičkim uređajima koji vrše neposredan nadzor i upravljanje (PLC). Najčešće se ovo povezivanje vrši preko drajvera koji su razvijeni od strane proizvođača merne i upravljačke opreme. U SCADA softverskim sistemima nezavisnih proizvođača postoji velika paleta drajvera za opremu različitih proizvođača, dok je kod proizvođača SCADA softvera i upravljačke opreme akcenat stavlja na drajvere za sopstvene uređaje.
8. **Podsistem za pristup bazama podataka (DBMS)** omogućuje trajno čuvanje i pregled podataka u relacionim bazama podataka. Ranija rešenja su beležila podatke u datoteke u nestandardnom obliku. Novija rešenja koriste neki od standardnih načina arhiviranja podataka koji omogućuju korisniku lak pristup podacima kao i pristup podacima iz drugih softverskih sistema. Na *Microsoft Windows* operativnim sistemima često se koristi ODBC (*Open Database Connectivity*) i nešto savremenija ADO (*ActiveX Data Object*) tehnologija. Upotreba ovih tehnologija omogućuje lakšu pretragu podataka kao i formiranje izveštaja pomoću SQL (*Structured Query Language*) jezika.
9. **Mrežni podsistem** omogućuje povezivanje ostalih podsistema SCADA programa. Brzina rada celog sistema u mnogome zavisi od programskog rešenja ovog sistema. Obično se rešenja oslanjaju na operativni sistem i koriste poznate protokole (transportnog nivoa). Jedna grupa rešenja koristi DDE (*Dynamic Data Exchage*) tehnologiju koja je jednostavna za korišćenje a pokazuje dobre performanse u sistemima u kojima se prenosi mala količina podataka. U složenijim sistemima često je koriste sistemi distribuiranih objekata koji se zasnivaju na DCOM i CORBA tehnologijama.

Na slici 1.1 prikazani su podsistemi nadzorno-upravljačkog programskog sistema i njihovi međusobni odnosi.

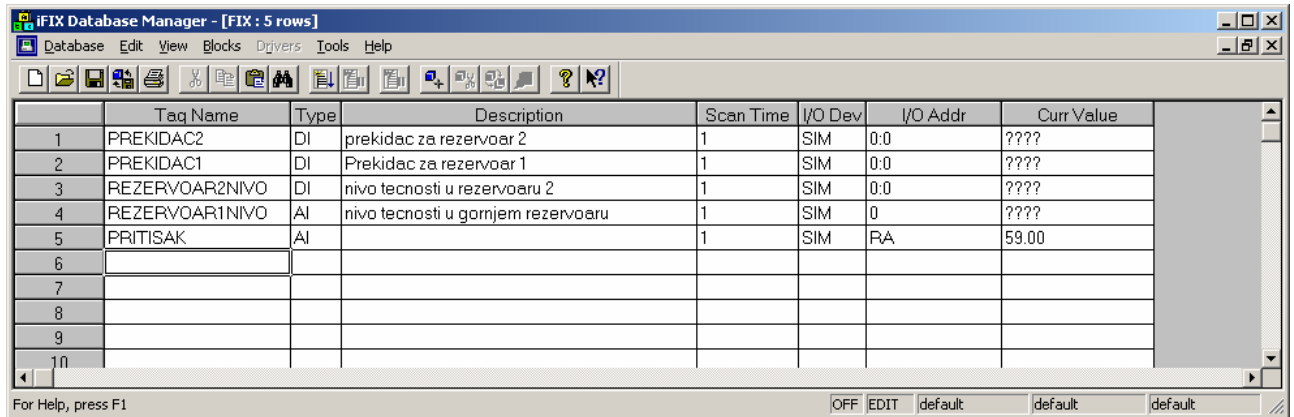


Slika 1.1. Podsystemi nadzorno-upravljačkog programskog sistema

2. Intellution IFix

2.1. Dodavanje veličina

Dodavanje veličina u Intellution IFix-u se vrši preko Database Manager-a (slika 2.1). Pri dodavanju veličine prvo se odabira njen tip. Tip može biti: analogni alarm (AA), analogni ulaz (AI), analogni izlaz (AO), digitalni alarm (DA), digitalni ulaz (DI)

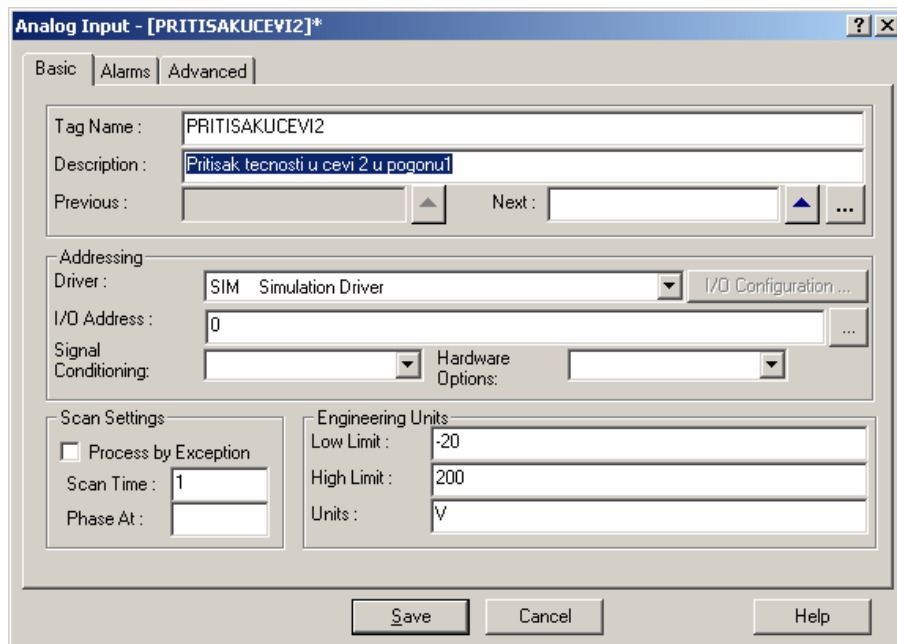


The screenshot shows the 'IFIX Database Manager' window with a menu bar (Database, Edit, View, Blocks, Drivers, Tools, Help) and a toolbar. Below is a table with 8 columns: Tag Name, Type, Description, Scan Time, I/O Dev, I/O Addr, and Curr Value. The table contains 5 rows of data, with the last row highlighted. The status bar at the bottom indicates 'For Help, press F1' and has buttons for 'OFF', 'EDIT', and 'default'.

	Tag Name	Type	Description	Scan Time	I/O Dev	I/O Addr	Curr Value
1	PREKIDAC2	DI	prekidac za rezervoar 2	1	SIM	0:0	????
2	PREKIDAC1	DI	Prekidac za rezervoar 1	1	SIM	0:0	????
3	REZERVOAR2NIVO	DI	nivo tecnosti u rezervoaru 2	1	SIM	0:0	????
4	REZERVOAR1NIVO	AI	nivo tecnosti u gornjem rezervoaru	1	SIM	0	????
5	PRITISAK	AI		1	SIM	RA	59.00
6							
7							
8							
9							

Slika 2.1 IFix Database Manager

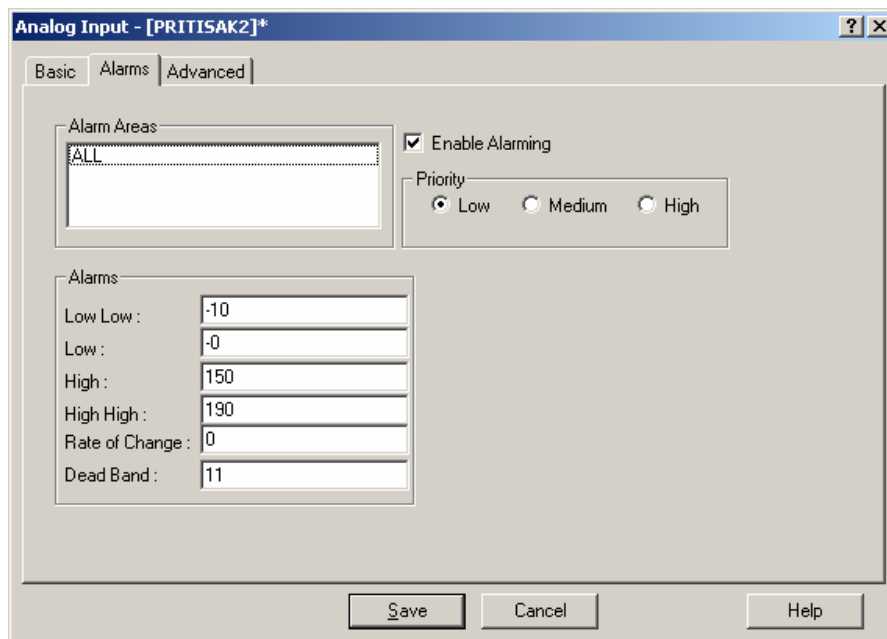
U osnovne karakteristike veličine spadaju njeno Ime (Tag name) i opis (Description). Osobina Scan Time predstavlja vreme između dva očitavanja veličine. Za svaku veličine može se uneti donja granica (Low Limit) i gornja granica (High Limit).



The screenshot shows the 'Analog Input - [PRITISAKUCEVI2]*' dialog box with three tabs: Basic, Alarms, and Advanced. The 'Basic' tab is active. It contains fields for Tag Name (PRITISAKUCEVI2), Description (Pritisak tecnosti u cevi 2 u pogonu1), Previous, and Next. The Addressing section includes Driver (SIM Simulation Driver), I/O Address (0), Signal Conditioning, and Hardware Options. The Scan Settings section has checkboxes for Process by Exception, Scan Time (1), and Phase At. The Engineering Units section has fields for Low Limit (-20), High Limit (200), and Units (V). Buttons for Save, Cancel, and Help are at the bottom.

Slika 2.2. Osnovne osobine veličina

Alarmne vrednosti za veličinu se postavljaju na kartici Alarms. Prioritet alarma može biti nizak (Low), srednji (Medium), ili visok (High). Vrednost koja predstavlja kritičnu donju vrednost postavlja se u polje Low Low, donja vrednost veličine postavlja se u polje Low, gornja vrednost u polje High, a kritična gornja vrednost u polje High High. Ako se vrednost veličine promeni za vrednost veću od one definisane u polju Rate of Change, sistem će generisati alarm. Veličina Dead Band definiše koliko treba da se promeni vrednost veličine u alarmnom stanju da bi alarm bio ponovljen.



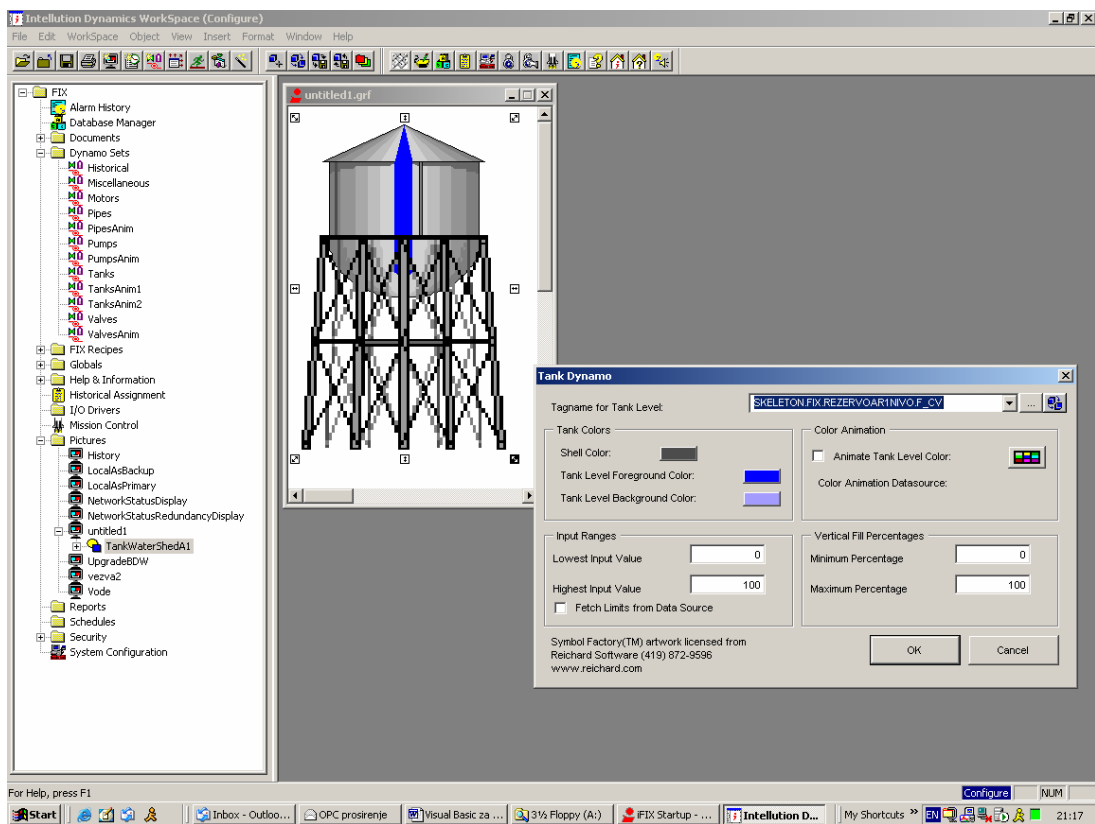
Slika 2.3. Definisanje alarmnih vrednosti

2.2. **Formiranje i animacija slike**

Osnova korisničkog interfejsa u IFix sistemu predstavlja slika (picture). Na sliku se mogu dodati neka od osnovnih kontrola kao što su linija, poligon ili graf itd.

IFix sadrži skupove grafičkih kontrola koje je moguće animirati na osnovu vrednosti neke od veličina. Ti skupovi se zovu Dynamo Set-ovi. Postojeći Dynamo Set ovi sadrže animacije pumpi, rezervoara, cevi i motora, a moguće je i praviti nove.

Po dodavanju neke od animiranih kontrola na sliku pojavljuje se dijalog u kome je potrebno definisati veličinu na osnovu koje će biti vršena animacija.

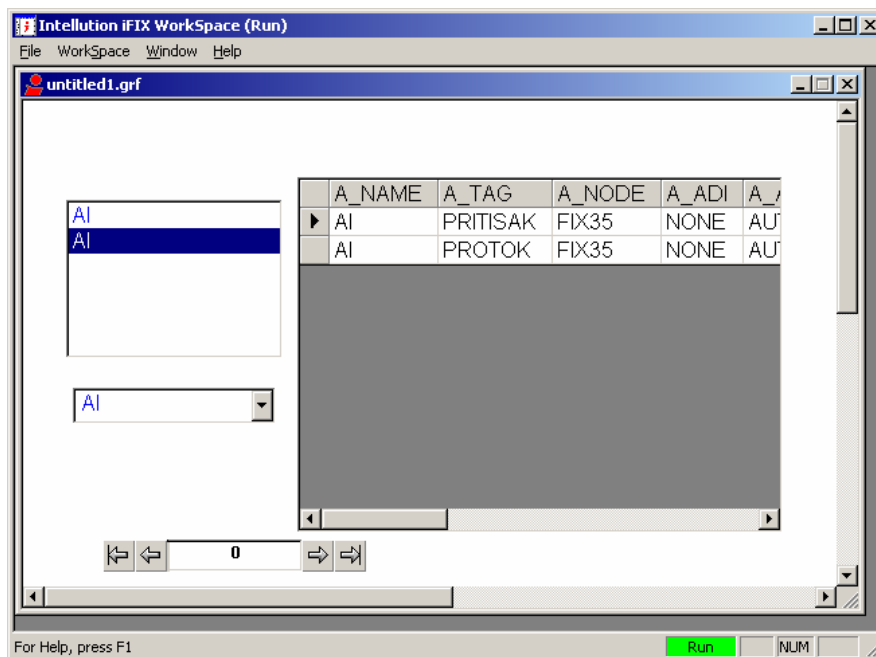


Slika 2.4 Animiranje slika

2.3. DataBinding

Intellution je proširio funkcionalnost ovih kontrola i formirao VisiconX Data Control, VisiconX Grid Control, VisiconX List Box i VisiconX Combo Box.

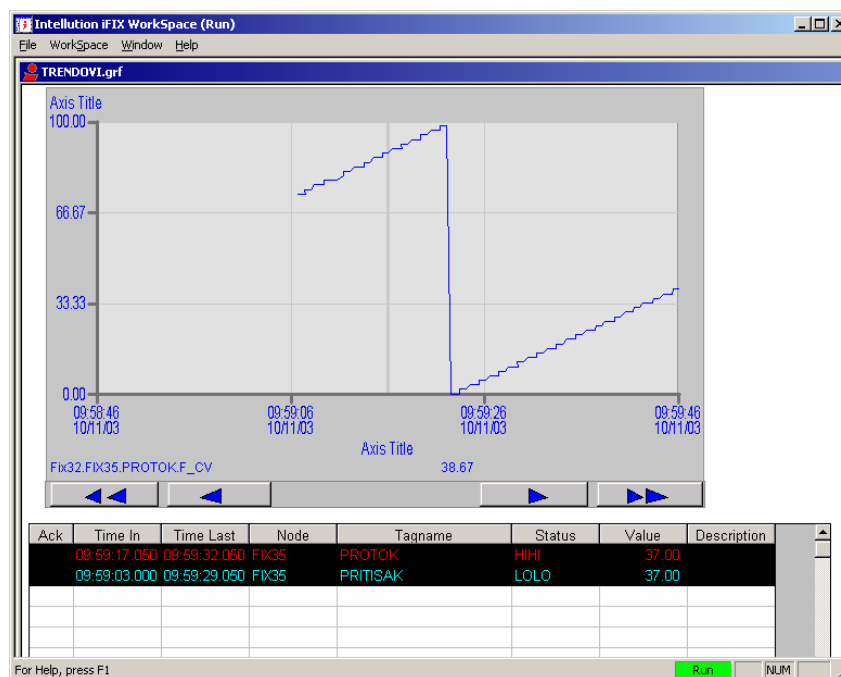
VisiconX Data control služi za povezivanje sa bazom, dok ostale kontrole prikazuju vrednosti iz Data kontrole.



Slika 2.5 VisicoX kontrole

2.4. Alarmi i trendovi

Kontrola za prikaz alarma je Alarm Summary. Za prikaz trendova koristi se kontrola iz Historical Dynamo Set-a



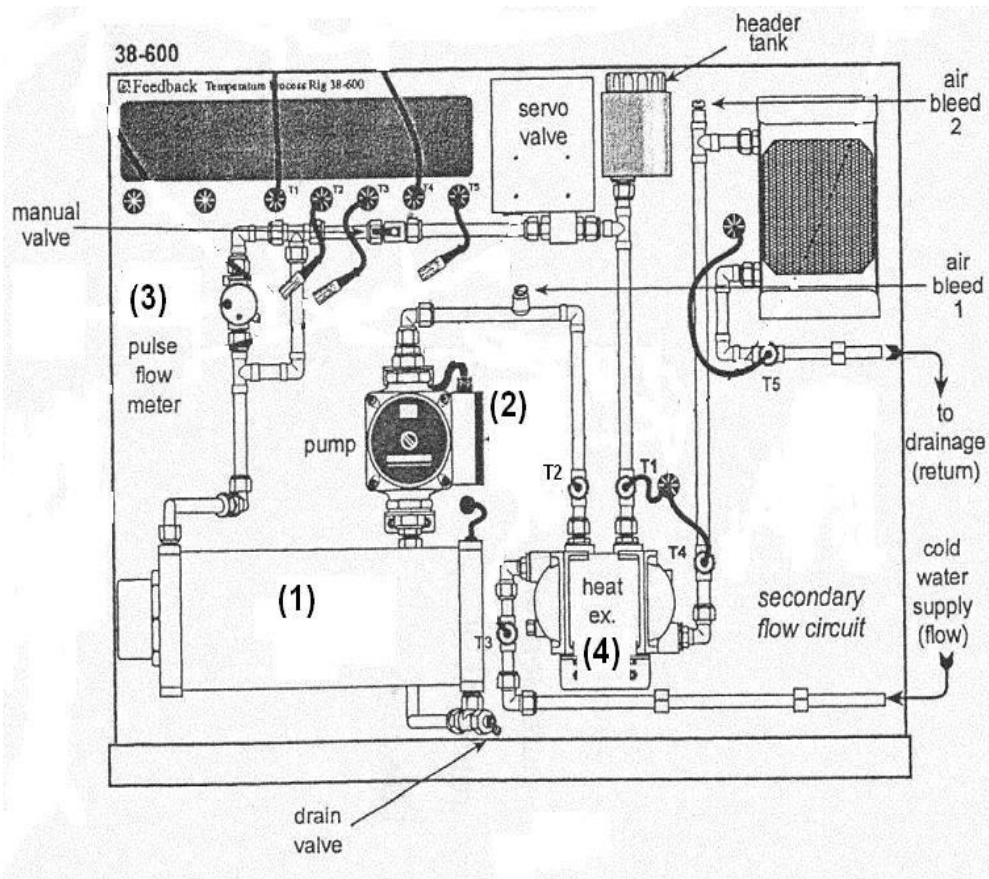
Slika 2.6. Kontrole za trendove i alarme

2.5. Regulacija temperature

Zadatak Konfigurisati IFix SCADA sistem za nadzor i upravljanje laboratorijsko-industrijskom tablom za regulaciju temperature (Feedback 38-600)

Opis sistema (FEEDBACK 38-600)

Sistem se sastoji od dva vodena toka čiji se sadržaji međusobno ne mešaju. Primarni krug je zatvoren i kroz njega protiče topla voda koja služi za zagrevanje vode koja protiče kroz sekundarni krug. Izgled sistema prikazan je na slici 2.7.



Slika 2.7. Tabla za upravljanje temperaturom

U primarnom krugu se nalazi rezervoar sa grejačem (1), ON/OFF tipa, koji se puni vodom pre puštanja sistema u rad. Zagrejana voda se iz rezervoara potiskuje u sistem pomoću centrifugalne pumpe 2 (*pump*). Merač protoka 3 (*pulse flow meter*) daje signal srazmeran veličini protoka u primarnom krugu. Preko transmitera, signal iz merača se vodi na displej, koji nije prikazan na slici.

Ručni ventili (*manual valve*) u oba kruga služe za ručnu regulaciju protoka. Na rezervoaru postoji ručni ventil za ispuštanje vode po potrebi iz primarnog kruga (*drain valve*).

Sekundarni krug (*secondary flow circuit*) se snabdeva vodom iz nezavisnog izvora (*cold water supply*). U izmenjivaču toplote 4 (*heat exc.*) se vrši zagrevanje vode sekundarnog kruga na taj način što voda primarnog kruga, koja može da dostigne i temperaturu od 70 st. C predaje toplotu vodi sekundarnog kruga.

Temperatura se meri na 5 pozicija, kao što se i vidi sa slike (T1,T2,T3,T4,T5). Od najvećeg značaja je temperatura T4 jer je to u stvari veličina koju regulišemo u ovom sistemu. Regulišemo je na taj način što ili menjamo otvorenost servo ventila (*servo valve*) ili broj obrtaja centrifugalne pumpe. Voda u sekundarnom krugu prolazi i kroz hladnjak, tako da je temperatura T5 niža od T4.

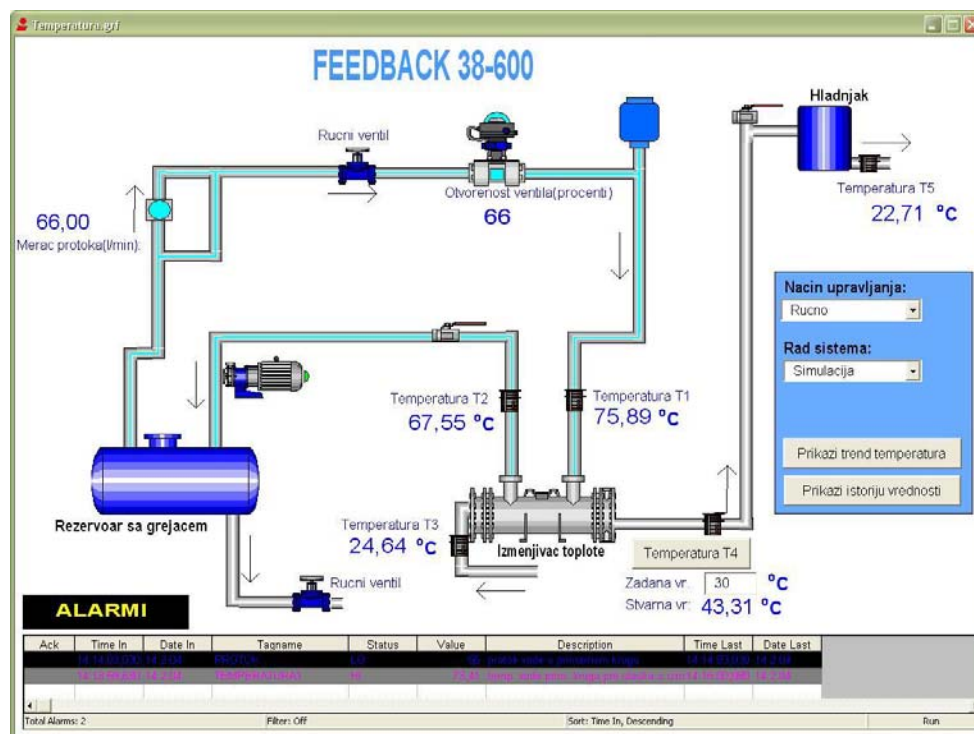
Na sistemu postoje blende za ispuštanje nepoželjnog vazduha iz cevi (*air bleed* 1,2). Nakon prolaska kroz hladnjak, voda se usmerava u neki drugi sistem ili ide ponovo na zagrevanje u sekundarni krug(*to drainage, return*).

Konfigurisanje SCADA-e

SCADA je softver za kontrolu i nadzor sistema. Nadzor se vrši na taj način što se podaci iz sistema koji se nadzire, dobijaju od senzora (temperature, pritiska...) i prikazuju na ekranu računara.

U slučaju FEEDBACK 38-600 ti podaci se prikupljaju pomoću akvizicione A/D (analogno/digitalne) kartice koja se nalazi u PC-u na kojem se pokreće program za nadzor sistema. Korišten je „Intellution iFix 3.5" softverski paket za konfigurisanje SCADA-e za nadzor FEEDBACK 38-600.

Korisnički interfejs se konfigurira na taj način što se crta slika koja bi trebalo da predstavlja realni sistem. Izgled glavnog ekrana prikazan je na slici 2.8.



Slika 2.8. Tabla za nadzor i upravljanje temperaturom

Nakon završetka crtanja slike, pristupa se formiranju baze podataka koja bi trebalo da sadrži sve promenljive sistema. „Intellution iFix" omogućava u sebi ima integrisanu bazu podataka koja se popunjava korišćenjem vrlo jednostavnog editora. Potprogram za manipulaciju podacima iz baze naziva se Database Manager.

Promenljiva u sistemu može biti digitalnog ili analognog tipa, ulazna ili izlazna. Ovi atributi promenljivih, kao i vreme skeniranja na A/D kartici, vrednosti alarmnih nivoa za svaku od promenljivih, definišu se u Database Manager-u. Promenljive u ovom sistemu izlistane su u sledećoj tabeli.

Tabela 1. Spisak veličina

	Tag Name	Type	Description	Scan Time	I/O Dev	I/O Addr	Curr Value
1	PROTOK	AI	protok vode u primarnom krugu	1	SIM	0	42
2	TEMPERATURA1	AI	temp. vode prim. kruga pre ulaska u izm.	1	SIM	0	72,18
3	TEMPERATURA2	AI	temp. vode prim. kruga posle izmenjivac.	1	SIM	0	63,79
4	TEMPERATURA3	AI	temp. vode sek. kruga pre ulaska u izm.	1	SIM	0	21,98
5	TEMPERATURA4	AI	temp. vode sek. kruga posle izmenjivaca	1	SIM	0	42,82
6	TEMPERATURA5	AI	temp. vode sek. kruga posle hladnjaka	1	SIM	0	22,52
7	SERVO_VENTIL	AO	otvorenost servo ventila	—	SIM	0	42,00
8	PUMPA_RADI	DO	Uključuje/isključuje pumpu	—	SIM	0:0	CLOSE

Svaka promenljiva u bazi ima svoje ime (Tag Name), tip, opis (Description), vreme skeniranja, naziv akvizicione A/D kartice (u ovom slučaju je SIM, tj. simulacioni drajver je korišten, jer u toku izrade SCADA-e nije još bilo uslova za testiranje na stvarnom sistemu), adresu memorijske lokacije na kartici (I/O Addr), kao i trenutnu vrednost (Curr Value).

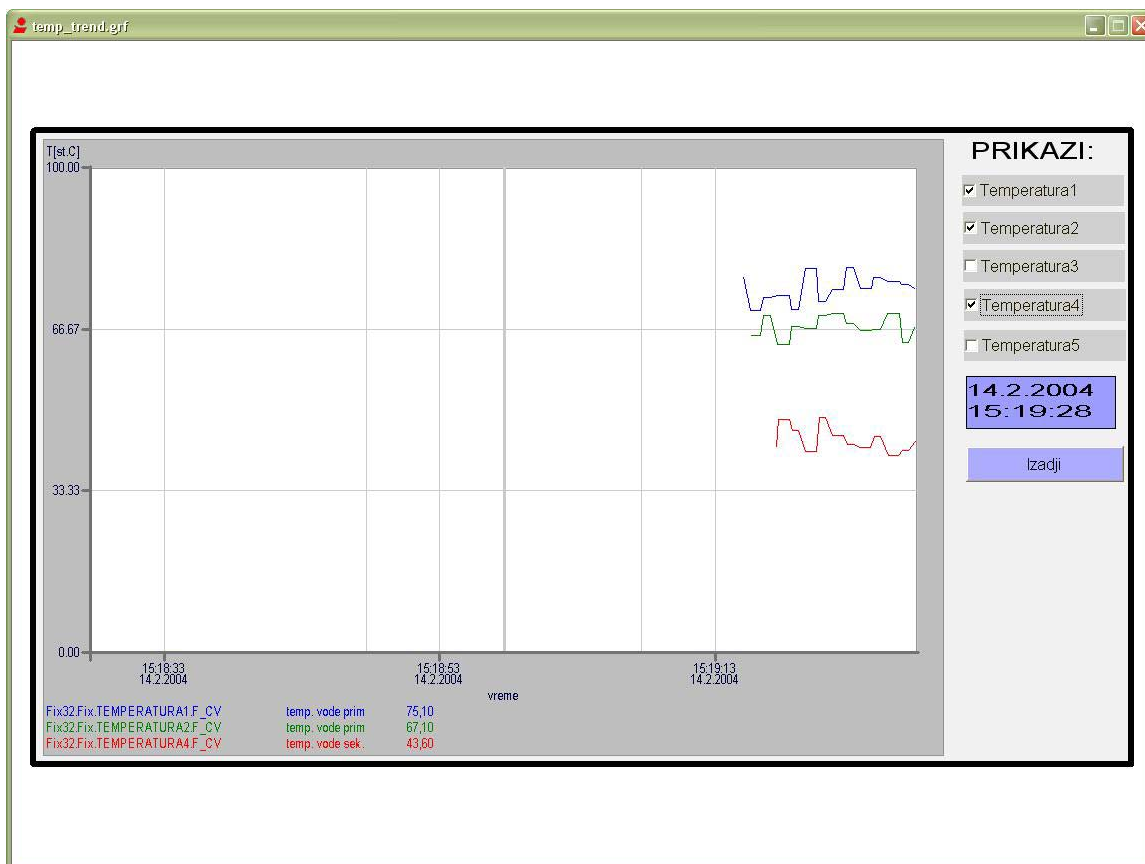
Svaki element slike se animira, tj. menja mu se boja ili oblik u toku rada SCADA-e. To se postiže povezivanjem svakog od tih elemenata slike sa nekom promenljivom iz baze, pa kada se menja vrednost promenljive animira se odgovarajući objekat na slici.

Dok SCADA radi, u pozadini se izvršava kod napisan u Visual Basic-u. Deo tog koda je izgenerisan automatski prilikom crtanja slike. Za svaki od elemenata na slici vezan je određen broj metoda i atributa. Na primer, jedna od metoda, tj. funkcija, jeste dvoklik mišem na neki od objekata na slici. U Visual Basic-u se piše kod koji se izvršava kada se na taj objekat sa slike klikne dva puta mišem. Pored ove postoji još mnogo drugih, kao što su: pritisak tastera, otpuštanje tastera, klik levim tasterom miša,...

```
Private Sub CBUkljuciPumpu_Click()
    Fix32.Fix.PUMPA_RADI.F_CV = True
    KontrolaPumpe.Visible = False
End Sub
```

Atributi objekata jesu njegova boja, oblik, vidljivost, dimenzije,... koji se takođe menjaju pisanjem određenog koda u Basic-u.

Postoji mogućnost ubacivanja OLE objekata na sliku i njihovog animiranja. Na primer, u ovom projektu je korišten *Chart* za prikaz vrednosti temperatura u toku realnog vremena. To je prikazano na slici 2.9, i do ove slike se u SCADA-i dolazi klikom na taster „Prikaži trend temperatura”.



Slika 2.9. Istorijat promene vrednosti veličina

„Intellution iFix" u sebi ima implementiran tzv. History Collect, potprogram koji služi za prikupljanje podataka iz sistema, tačnije vrednosti promenljivih. Te vrednosti se smeštaju u datoteke koje se nalaze u okviru Intellution-ove istorijske baze (History Database) podataka. Manipulacija tim podacima vrši se u okviru potprograma Historical Assignment.

Istorijske vrednosti promenljivih mogu se prikazivati korišćenjem OLE objekata (Visicon.vxData i Visicon.vxGrid). Tehnologija za pristup podacima u bazi je ADO (ActiveX Data Object). Visicon.vxData je OLE kontrola koja služi za dohvatanja podataka iz baze, dok se kontrola Visicon.vxGrid koristi za prikazivanje istih. Na slici 2.10. je prikazan način pristupa ovim podacima.

ISTORIJA VREDNOSTI VELICINA									
	NODE	TAG	FIELD	VALUE	DATETIME	MODE	STATUS	ALARM	INTE
	FIX	TEMPERATURA2	F_CV	62,26126	14.2.2004 18:58:27	SAMPLE	Data is valid	No Alarm - OK	00:0
	FIX	TEMPERATURA2	F_CV	68,52757	14.2.2004 18:59:27	SAMPLE	Data is valid	No Alarm - OK	00:0
	FIX	TEMPERATURA2	F_CV	69,34233	14.2.2004 19:00:27	SAMPLE	Data is valid	No Alarm - OK	00:0
	FIX	TEMPERATURA2	F_CV	67,93916	14.2.2004 19:01:27	SAMPLE	Data is valid	No Alarm - OK	00:0
	FIX	TEMPERATURA2	F_CV	68,26527	14.2.2004 19:02:27	SAMPLE	Data is valid	No Alarm - OK	00:0
	FIX	TEMPERATURA2	F_CV	64,44439	14.2.2004 19:03:27	SAMPLE	Data is valid	No Alarm - OK	00:0
	FIX	TEMPERATURA2	F_CV	64,88808	14.2.2004 19:04:27	SAMPLE	Data is valid	No Alarm - OK	00:0
	FIX	TEMPERATURA2	F_CV	65,34309	14.2.2004 19:05:27	SAMPLE	Data is valid	No Alarm - OK	00:0
	FIX	TEMPERATURA2	F_CV	67,13695	14.2.2004 19:06:27	SAMPLE	Data is valid	No Alarm - OK	00:0
	FIX	TEMPERATURA2	F_CV	66,84131	14.2.2004 19:07:27	SAMPLE	Data is valid	No Alarm - OK	00:0
	FIX	TEMPERATURA2	F_CV	68,61571	14.2.2004 19:08:27	SAMPLE	Data is valid	No Alarm - OK	00:0
	FIX	TEMPERATURA2	F_CV	69,05135	14.2.2004 19:09:27	SAMPLE	Data is valid	No Alarm - OK	00:0
	FIX	TEMPERATURA2	F_CV	64,51479	14.2.2004 19:10:27	SAMPLE	Data is valid	No Alarm - OK	00:0
	FIX	TEMPERATURA2	F_CV	60,99176	14.2.2004 19:11:27	SAMPLE	Data is valid	No Alarm - OK	00:0
	FIX	TEMPERATURA2	F_CV	66,43087	14.2.2004 19:12:27	SAMPLE	Data is valid	No Alarm - OK	00:0
	FIX	TEMPERATURA2	F_CV	65,43102	14.2.2004 19:13:27	SAMPLE	Data is valid	No Alarm - OK	00:0
	FIX	TEMPERATURA2	F_CV	67,36687	14.2.2004 19:14:27	SAMPLE	Data is valid	No Alarm - OK	00:0
	FIX	TEMPERATURA2	F_CV	64,6692	14.2.2004 19:15:27	SAMPLE	Data is valid	No Alarm - OK	00:0
	FIX	TEMPERATURA2	F_CV	66,73199	14.2.2004 19:16:27	SAMPLE	Data is valid	No Alarm - OK	00:0
	FIX	TEMPERATURA2	F_CV	63,53084	14.2.2004 19:17:27	SAMPLE	Data is valid	No Alarm - OK	00:0

Slika 2.10. Tabelarni prikaz promene vrednosti veličina

Alarm History je kontrola koja služi za prikazivanje alarma u sistemu. Svaka promenljiva čija vrednost pređe alarmni nivo prikazuje se pomoću ove kontrole na ekranu i obaveštava operatera da se desio alarmni događaj. Označavanjem tog alarma, operater daje do znanja SCADA-i da je registrovao alarmni događaj koji tada nestaje sa ekrana. Ovu kontrolu vidimo na dnu slike 2.8.

SCADA za FEEDBACK 38-600 je konfigurisana tako da može da se izabere ručni ili automatski način upravljanja. Kod ručnog je dozvoljeno podešavanje otvorenosti servoventila i uključivanje/isključivanje pumpe. U automatskom režimu, zadaje se željena vrednost temperatura T4. Na osnovu algoritma upravljanja napisanog u Basic-u koji se u pozadini izvršava, vrši se podešavanje otvorenosti servoventila kao i kontrola rada pumpe.

Postoje dva režima rada sistema: „simulacija" i „realni sistem". Kada je odabrana simulacija, vrednosti temperatura se dobijaju iz generatora slučajnih vrednosti iz Visual Basic-a. Ako je odabran rad u realnom sistemu, tada bi trebalo da se vrednosti promenljivih dobijaju sa akvizicione A/D kartice i da se prikazuju na ekranu. Jedna od glavnih komponenata SCADA-e je *Timer*. Njegov atribut je vremenski interval definisan u mikrosekundama, koji kada tajmer izbroji se resetuje na nulu. Nakon isteka tog vremenskog intervala, ako je u pitanju simulacija, generisaće se neke slučajne vrednosti temperatura i protoka, a ako je izabran rad sa realnim sistemom, vrši će se očitavanje vrednosti promenljivih iz akvizicione A/D kartice.

```
Private Sub CFixPicture_Initialize()
    Fix32.Fix.PUMPA_RADIF_CV = True
    Fix32.Fix.SERVO_VENTILF_CV = 50
    Fix32.Fix.TEMPERATURA1F_CV = 40
    Fix32.Fix.TEMPERATURA2F_CV = 35
```

```

Fix32.Fix.TEMPERATURA3.F_CV = 30
Fix32.Fix.TEMPERATURA4.F_CV = 35
Fix32.Fix.TEMPERATURA5.F_CV = 25
ComboBoxUpravljanje.AddItem "Rucno"
ComboBoxUpravljanje.AddItem "Automatski"
ComboBoxRezim.AddItem "Simulacija"
ComboBoxRezim.AddItem "Realni sistem"
TextBox1.Value = Fix32.Fix.SERVO_VENTIL.F_CV
ComboBoxUpravljanje.Value = "Rucno"
ComboBoxRezim.Value = "Simulacija"
ZadVrT4.Value = 30
zadanoT4 = ZadVrT4.Value
End Sub

Private Sub tajmer_OnTimeOut(ByVal lTimerId As Long)
zadanoT4 = ZadVrT4.Value
If ComboBoxUpravljanje.Value = "Rucno" Then
If Fix32.Fix.PUMPA_RAD1.F_CV = False Or Fix32.Fix.SERVO_VENTIL.F_CV = 0 Then
Fix32.Fix.PROTOK.F_CV = 0
Fix32.Fix.TEMPERATURA1.F_CV = Fix32.Fix.TEMPERATURA1.F_CV - 0.1
Fix32.Fix.TEMPERATURA2.F_CV = Fix32.Fix.TEMPERATURA2.F_CV - 0.1
Fix32.Fix.TEMPERATURA3.F_CV = Fix32.Fix.TEMPERATURA3.F_CV - 0.1
Fix32.Fix.TEMPERATURA4.F_CV = Fix32.Fix.TEMPERATURA4.F_CV - 0.1
Fix32.Fix.TEMPERATURA5.F_CV = Fix32.Fix.TEMPERATURA5.F_CV - 0.1
Else
Fix32.Fix.TEMPERATURA1.F_CV = Rnd * 10 + 70
Fix32.Fix.TEMPERATURA2.F_CV = Rnd * 10 + 60
Fix32.Fix.TEMPERATURA3.F_CV = Rnd * 5 + 20
Fix32.Fix.TEMPERATURA4.F_CV = Rnd * 10 + 40
Fix32.Fix.TEMPERATURA5.F_CV = Rnd * 5 + 20
Fix32.Fix.PROTOK.F_CV = Fix32.Fix.SERVO_VENTIL.F_CV
End If
End If
If ComboBoxUpravljanje.Value = "Automatski" Then
Fix32.Fix.PUMPA_RAD1.F_CV = True
If Fix32.Fix.TEMPERATURA4.F_CV > zadanoT4 Then
If Fix32.Fix.SERVO_VENTIL.F_CV > 0 Then
Fix32.Fix.SERVO_VENTIL.F_CV = Fix32.Fix.SERVO_VENTIL.F_CV - 1
End If
Fix32.Fix.TEMPERATURA4.F_CV = Fix32.Fix.TEMPERATURA4.F_CV - 0.2
End If
If zadanoT4 > Fix32.Fix.TEMPERATURA4.F_CV Then
If Fix32.Fix.SERVO_VENTIL.F_CV < 100 Then
Fix32.Fix.SERVO_VENTIL.F_CV = Fix32.Fix.SERVO_VENTIL.F_CV + 1
End If
Fix32.Fix.TEMPERATURA4.F_CV = Fix32.Fix.TEMPERATURA4.F_CV + 0.2
End If
Fix32.Fix.TEMPERATURA1.F_CV = Rnd * 10 + 70
Fix32.Fix.TEMPERATURA2.F_CV = Rnd * 10 + 60
Fix32.Fix.TEMPERATURA3.F_CV = Rnd * 5 + 20
Fix32.Fix.TEMPERATURA5.F_CV = Rnd * 5 + 20
Fix32.Fix.PROTOK.F_CV = Fix32.Fix.SERVO_VENTIL.F_CV
End If
End Sub

```

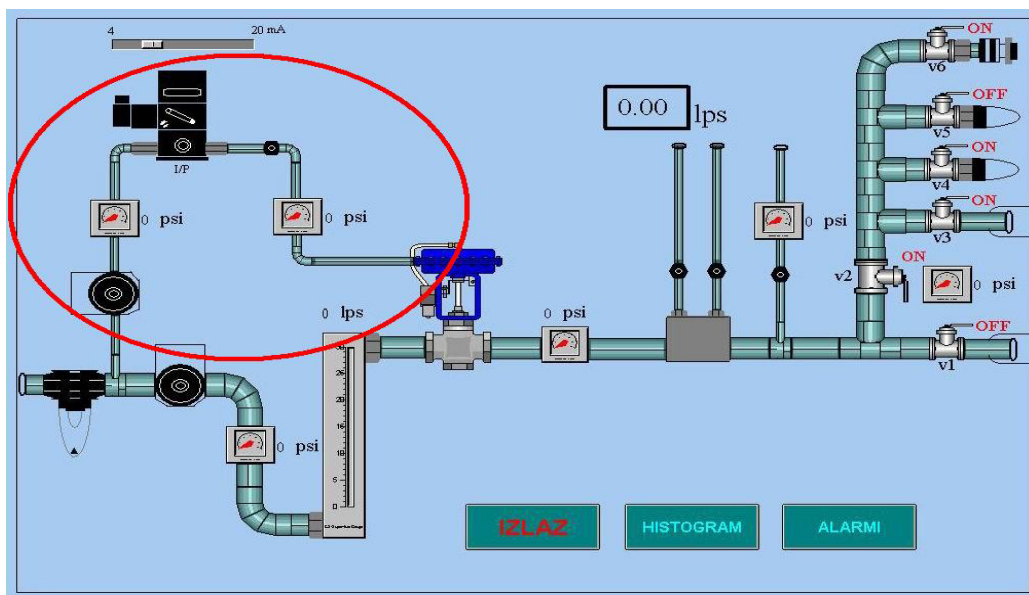

3. OMRON CX-SuperVisor

3.1. TEHNIČKI OPIS SISTEMA

Sistem za regulaciju pritiska se sastoji od kompresora, demonstracione table, rezervoara i PLC-a. Kompresor je priključen na ulaznu granu table i on na svom izlazu daje pritisak koji nije stabilan.

Tabla ima dve grane, glavnu ili regulisanu i regulacionu. Zbog nestabilnosti pritiska iz kompresora na obe grane se nalaze uređaji koji održavaju stabilan pritisak u granama, u glavnoj je 10 psi (pound square inch), a u regulacionoj je 25 psi.

U regulacionoj grani se iza stabilizatora pritiska nalazi jedan merač pritiska, zatim pretvarač struje u pritisak i na kraju grane još jedan merač pritiska. Prvi merač pritiska meri pritisak na ulazu regulacione grane, I / P pretvarač pretvara struju koja se dobija iz PLC-a u pritisak i sa tim pritiskom se upravlja pneumatskim ventilom. Merač pritiska na kraju regulacione grane meri koliki pritisak upravlja pneumatskim ventilom.



Slika 3.1. – Regulaciona grana

U glavnoj grani se iza stabilizatora pritiska nalazi merač pritiska i merač protoka, koji mere ove veličine ispred pneumatskog ventila. Pneumatski ventil kojim se upravlja regulacionom granom se nalazi ispred merača protoka koji radi na principu diferencijalnog pritiska. Ispred i iza merača protoka se nalaze merači pritiska. Na kraju glavne grane se nalaze potrošači koji su povezani sa tablom direktno i preko rezervoara, koji služi da stabilizuje pritisak iz table.

Svi merači pritiska su gravirani do 30 psi, jer se na tabli ne može ostvariti veća vrednost pritiska.

3.2. OMRON SOFTVER

Glavne karakteristike OMRON CX-Supervisor SCADA-e su:

- animacija, jednostavni crteži kao i složeniji OLE objekti
- mnogo ActiveX kontrola
- biblioteka sa preko 3000 grafičkih simbola
- recepture za prenos procesnih podataka ka/sa PLC-a
- alarmne funkcije koje saopštavaju o neželjenim vrednostima I/O promenljivih
- sigurnosne mere koje omogućavaju ograničen pristup pojedincu ili grupi
- izbor tri programska jezika: CX-Supervisor script, VBA i JAVA
- povezivanje sa bazom podataka
- podržavanje Microsoft COM / DCOM, DDE, OPC, OLE, ActiveX i ADO / OLEDB

CX Automation Suite u sebi integriše i sisteme za kontrolu procesa, kao i sisteme za prenos poslovnih informacija pomoću jedne iste mrežne arhitekture. Za korisnika to znači da se podaci o parametrima procesa i podaci o poslovnim efektima mogu zajedno prikazati u formi tabela koje se dalje mogu uređivati nekim od poznatih spreadsheet procesora (Excel, Delphi i sl.).

Pomoću notebook PC računara moguće je priključenje bilo gde u mreži, prikupljanje željenih informacija ili vizualizovanje procesa i promena parametara. Za ovakve poslove razvijen je modul CX Server Lite, koji intenzivno koristi interfejsse ActiveX i COM.

Programski paket CX Automation Suite je sastavljen od sledećih modula:

CX Supervisor: ovaj softverski modul se koristi za razvoj interfejsa SCADA sistema, koji omogućuje vizuelizaciju, arhiviranje svih parametara procesa radi kasnije analize, alarmiranje i podešavanje parametara sistema.

CX Programmer: ovaj softverski modul se koristi za programiranje PLC-ova i to počev od onih najmanjih, tzv. mikro PLC-ova, do najnovijih kontrolera visokih performansi.

CX Simulator: u fazi razvoja i ispitivanja programa ponekad je zamorno da se često vrši download izvršnog koda kako bi se proverila funkcionalnost ili izvršila sitna "doterivanja". CX Simulator omogućuje računaru da emulira CPU jedinicu PLC-a.

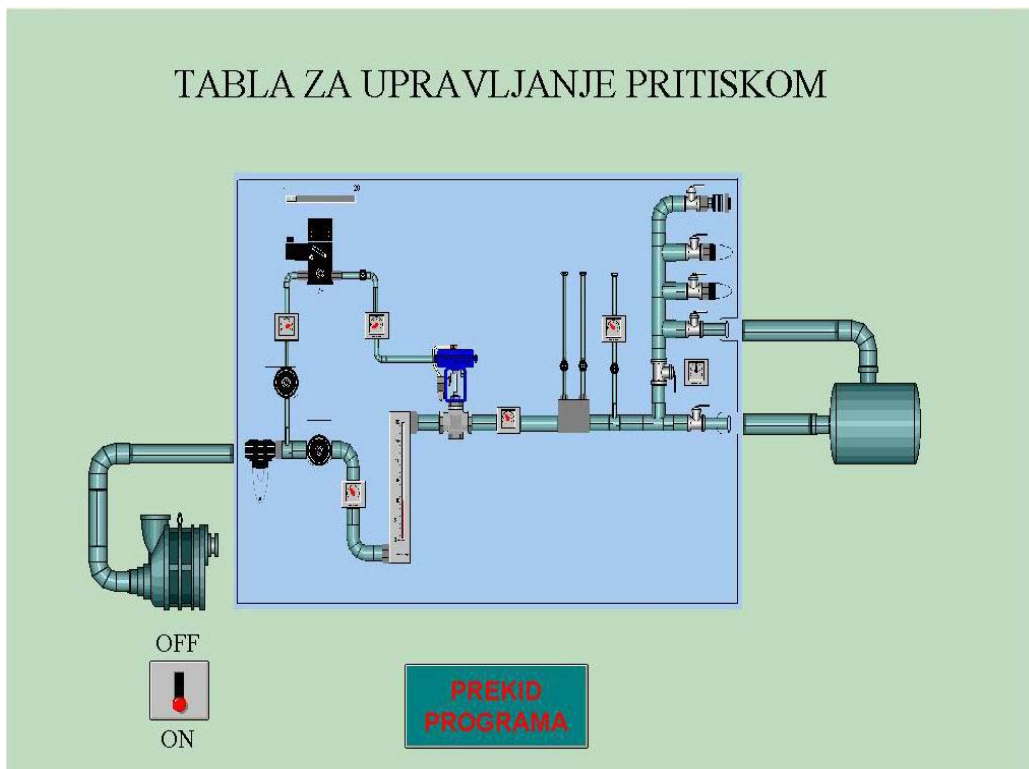
CX Protocol: namenjen je podršci OMRON-ovim Protocol Macro Support modulima (PMSU). PMSU predstavljaju interfejs ka serijskim periferijama i podržavaju oba smera komunikacije.

CX Server Lite: obezbeđuje fleksibilnost CX Automation Suite paketa, omogućavajući da standardni Office paketi (Excel, Word, Visual Basic, Delphi) pristupaju uređajima putem ActiveX standarada. CX Server Lite sadrži skup grafičkih komponenti koje olakšavaju razvoj aplikacija: drag&drop tehnikom kreiraju se veze između objekata.

CX Server OPC: softverski modul predstavlja skup OLE/COM protokola namenjenih podešavanju relacija između upravljačkih programa PLC-a i Office aplikacija. Kreiranjem svojevrstnih tagova u upravljačkom softveru PLC-a pomoću ovog softverskog modula, moguće je uspostaviti vezu između određenih parametara procesa i baze podataka. CX Server OPC integriše PLC uređaje u otvorenu arhitekturu savremenih mreža, kroz koje se ravnopravno prenose različiti tipovi podataka: SCADA i VBA aplikacije, Excel tabele, itd. CX Server OPC podržava razvoj malih OPC klijent aplikacija u Excel-u ili Visual Basic-u uz pomoć ugrađenih grafičkih objekata.

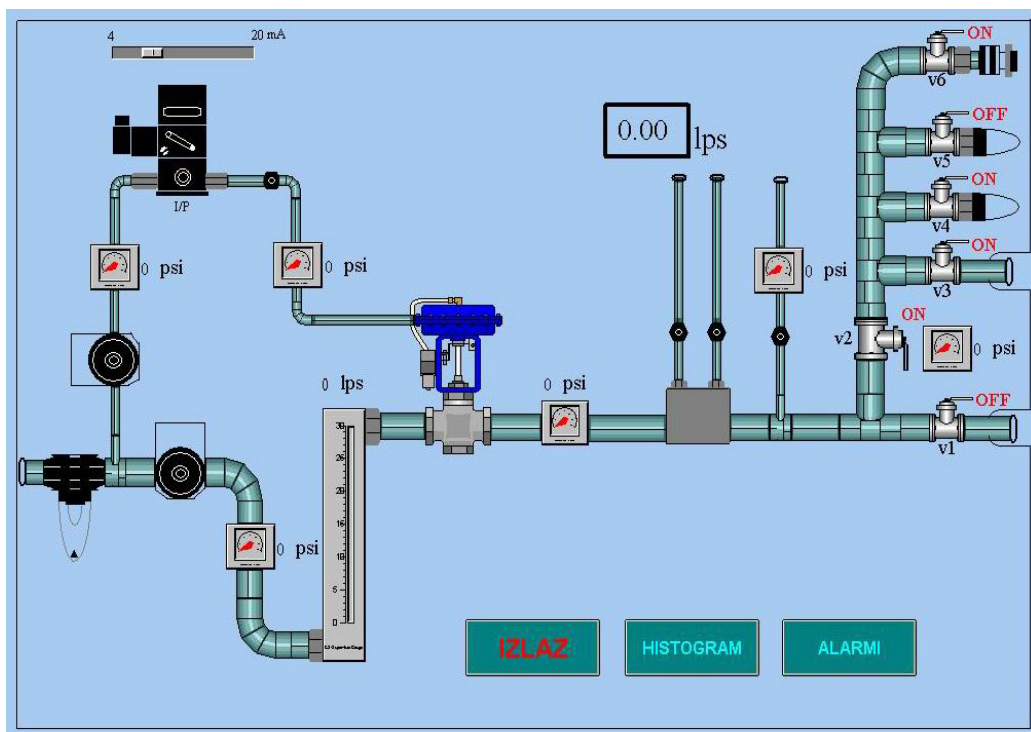
3.3. VIZUELIZACIJA

Ceo sistem za upravljanje pritiskom se sastoji od četiri slike. Kad se pokrene aplikacija dobija se slika celog sistema (slika 3.1), druga je slika table (slika 3.2), a treća i četvrta su slike manjeg formata na kojima su histogram i alarmna stanja.



Slika 3.2. Prikaz sistema za regulaciju pritiska

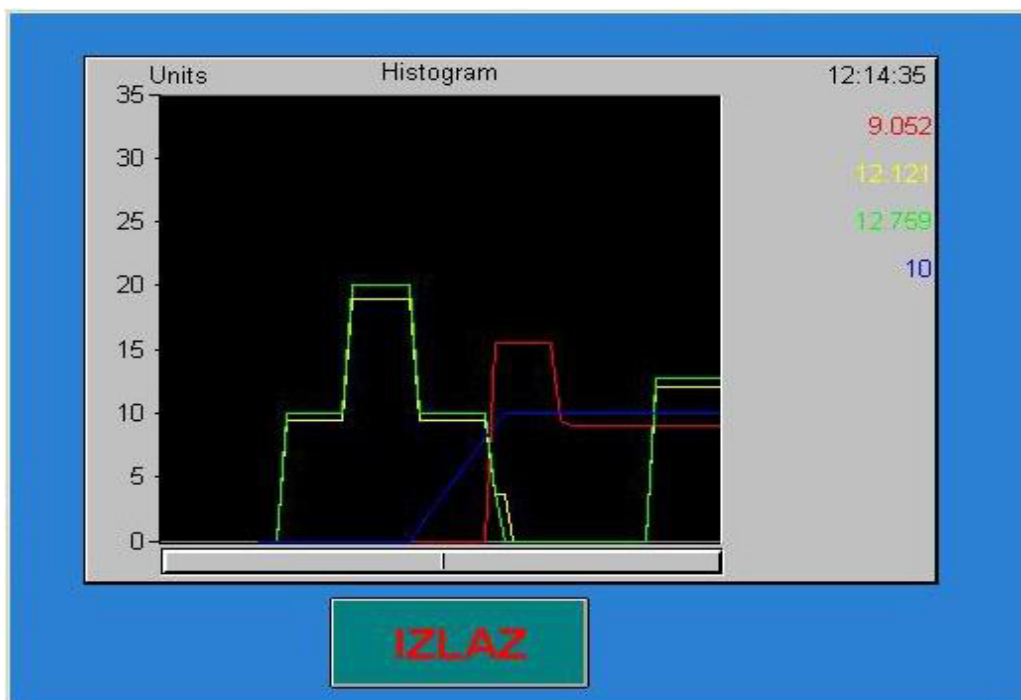
Slika 3.2. predstavlja celokupan sistem za regulaciju pritiska. Prekidač služi za uključivanje i isključivanje kompresora. Pritiskom na sliku table dobija detaljan prikaz table, dok se pritiskom na taster "PREKID PROGRAMA" završava izvršavanje aplikacije.



Slika 3.3. – Detaljna slika table

Detaljan prikaz table je na slici 3.3. Klizač u gornjem levom uglu table služi za upravljanje strujom koja se šalje na I / P pretvarač.

Pritiskom na taster "HISTOGRAM" se dobija grafik na kome se prate vrednosti struje – crvena boja, pritiska vazduha u rezervoaru – plava boja i protoka ispred i iza pneumatskog ventila – zelena i žuta boja.



Slika 3.4. – Forma sa histogramom

Pritiskom na taster "ALARM!" ulazi se u prozor gde su zapisana alarmna stanja. Crvenim slovima su označeni alarmi koje korisnik nije potvrdio, plavim oni koji su potvrđeni a zelenim oni koji su otklonjeni.

Date	Time	Message	Status
12/14/200	12:09:30	NIJE OTVOREN NIJEDAN VENTIL ZA IZLAZ, f	Alarm
12/14/200	12:08:41	Alarm alarm3 Cleared	Cleared

Below the table is a green button with the text 'IZLAZ' in red.

Slika 3.5. – Forma sa alarmima

Ventili na desnoj strani table su ON/OFF tipa i služe za uključivanje i isključivanje potrošača, i određivanje da li će potrošači biti priključeni direktno ili preko rezervoara.

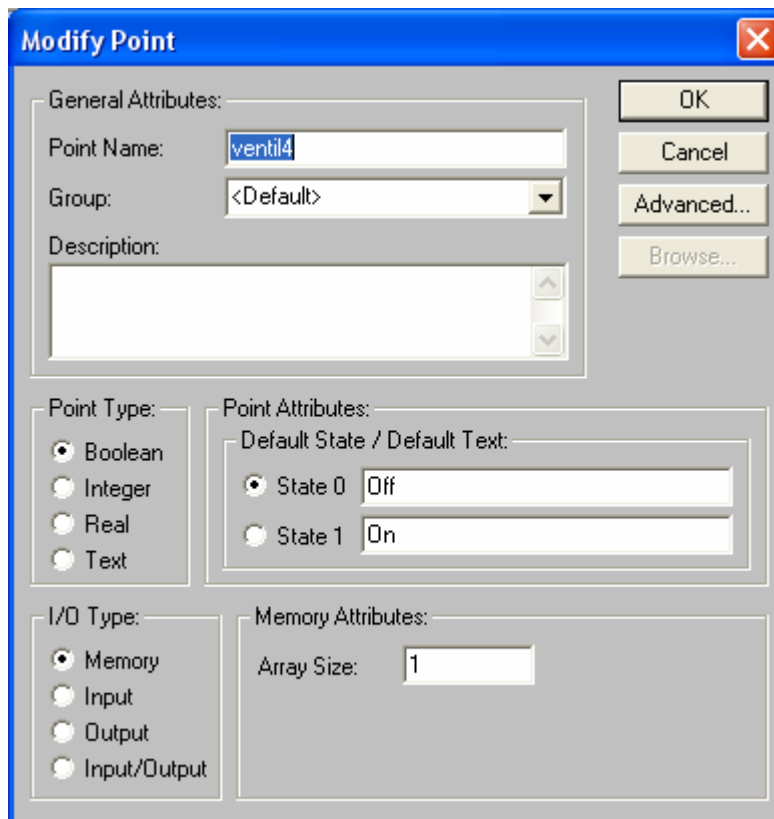
Pritiskom na taster "IZLAZ" se vraća u prozor s početka, slika 3.1.

3.4. PROGRAMIRANJE

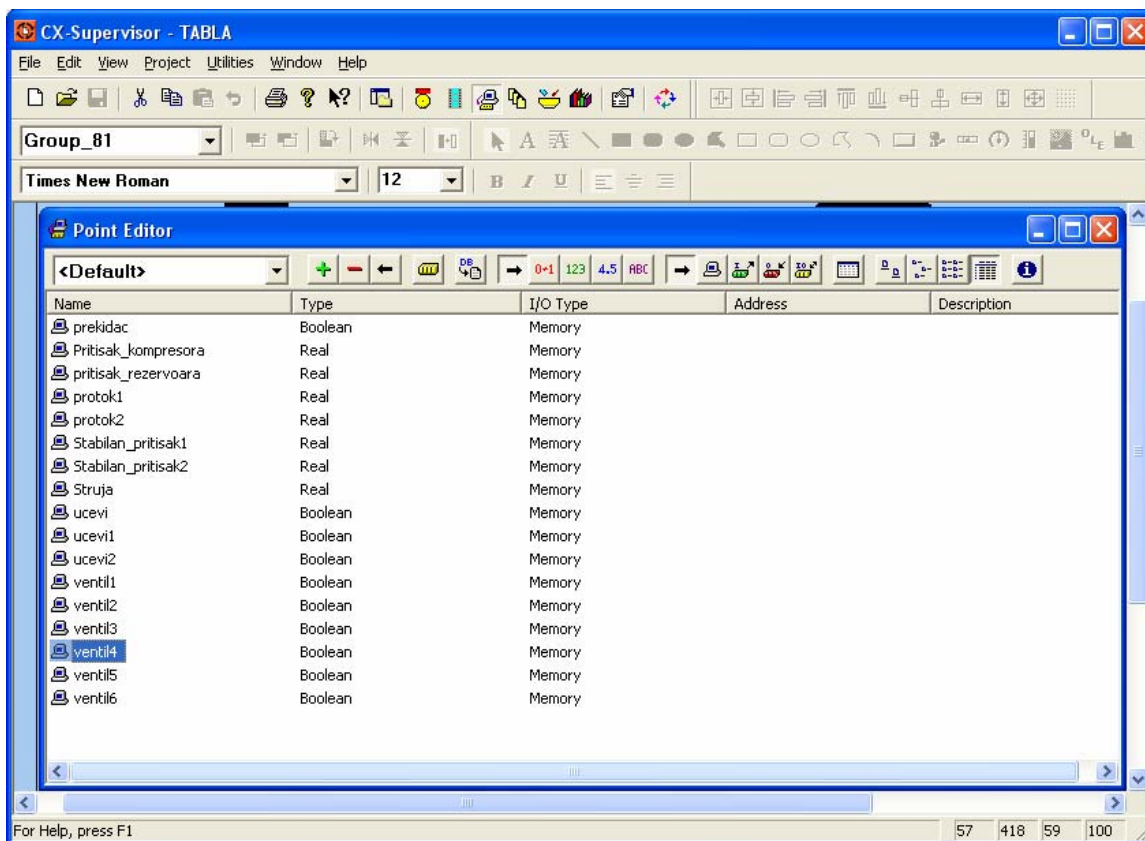
Definisanje promenljivih

Otvaranjem prozora *Point Editor* i klikom na zeleni plus dodaje se promenljiva, daje joj se ime, definiše tip, početna vrednost i eventualno se daje njen opis. Promenljive mogu biti ulazne, izlazne, ulazno/izlazne i memorijskog tipa. Sve promenljive koje se koriste su memorijskog tipa zbog ograničenja demo verzije programa kao i nemogućnosti definisanja PLC-a koji se zahteva pri upotrebi drugih vrsta promenljivih.

Point Editor ima mogućnost sortiranja promenljivih po tipu, a daje i informacije o svim korišćenim veličinama.



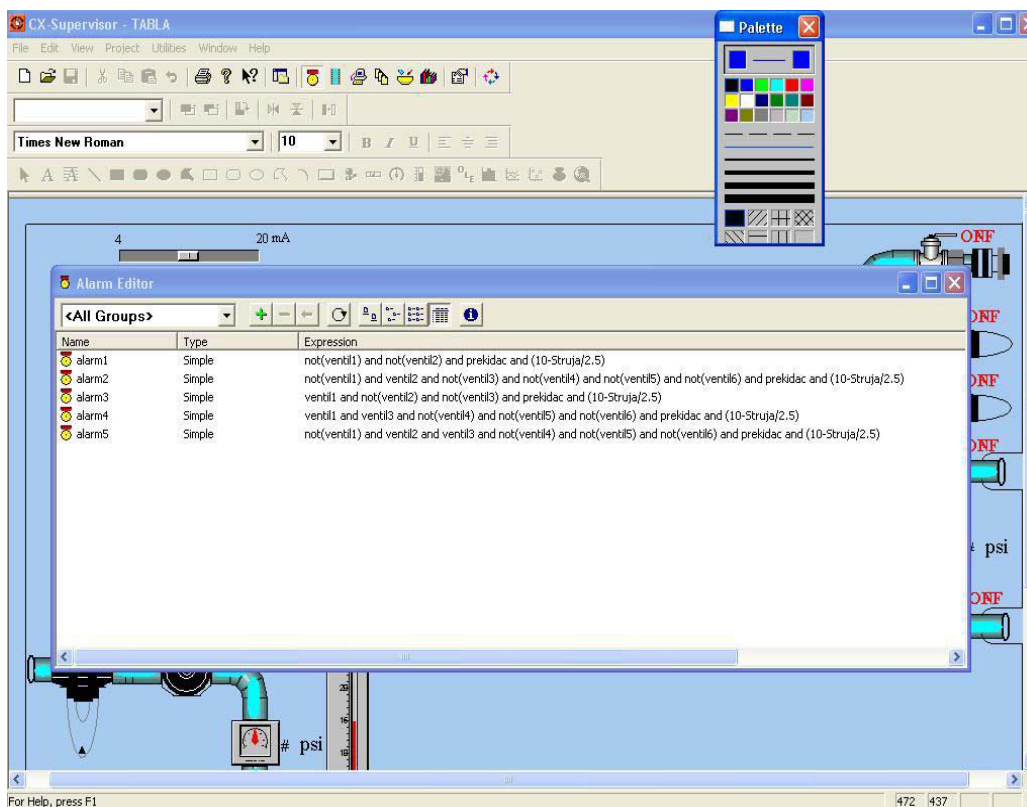
Slika 3.6. Point Editor – jedna promenljiva



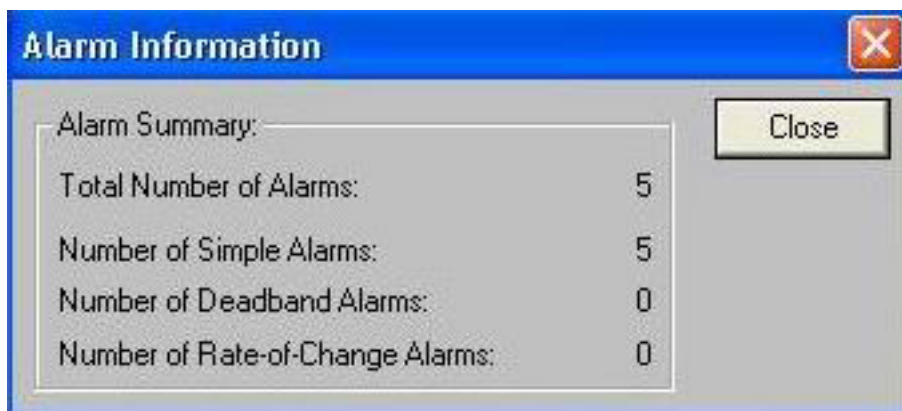
Slika 3.7. - Prikaz promenljivih potrebnih za eksperiment

Alarmi

Otvaranjem prozora *Alarm Editor* i klikom na zeleni plus dodaje se alarm, daje mu se ime, definiše tip, vrednost nakon koje se aktivira alarm i opis alarmnog stanja. Postoje tri vrste alarma koji se mogu definisati: simple, deadband i rate-of-change. Izbor se vrši u zavisnosti od tipa promenljive koja se posmatra (npr. Simple alarm je korišćen kod prijavljivanja požara, provala i nestanka napajanja, a Deadband kod alarma za nivo i pritisak). Alarmne vrednosti su usko povezane sa veličinama iz *Point Editor*a.



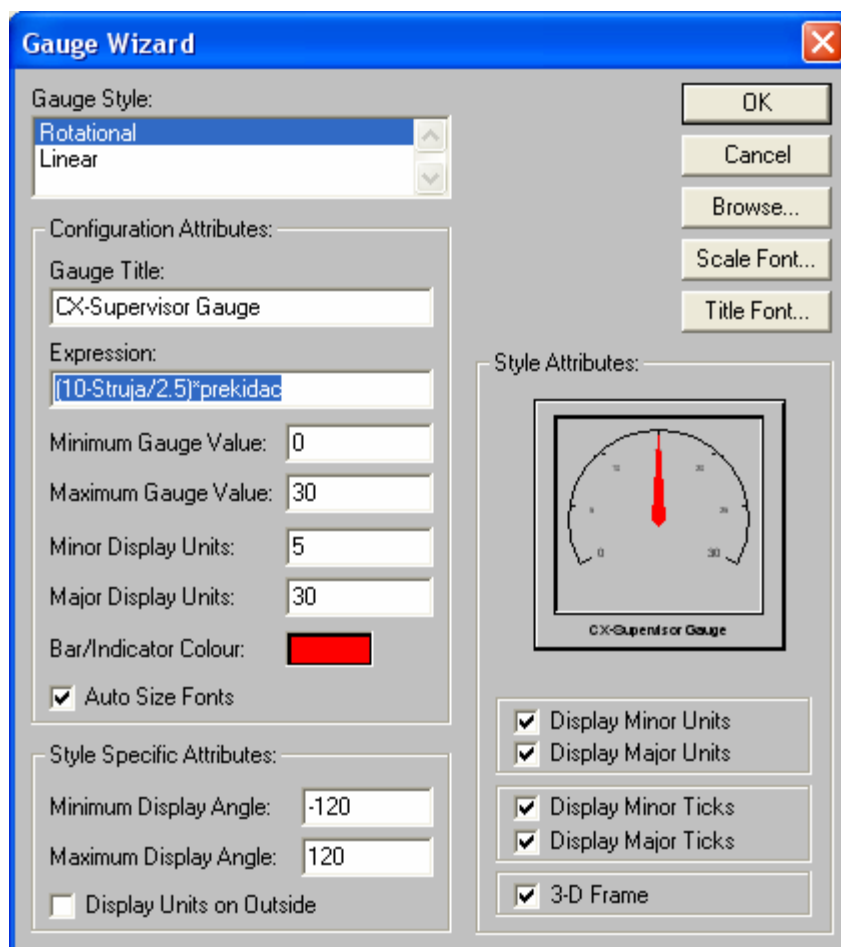
Slika 2.8. Alarm Editor



Slika 2.9. - Sumarni prikaz alarma

Animacija

Neke kontrole se mogu direktno animirati pomoću vrednosti neke veličine ili izraza. Te kontrole se nalaze na paleti *Graphic Object Bar*. Na slici 3.10 prikazan je način povezivanja prikaza instrumenta sa kazaljkom na vrednost izraza.



Slika 3.10. Animacija kontrole pomocu vrednosti izraza

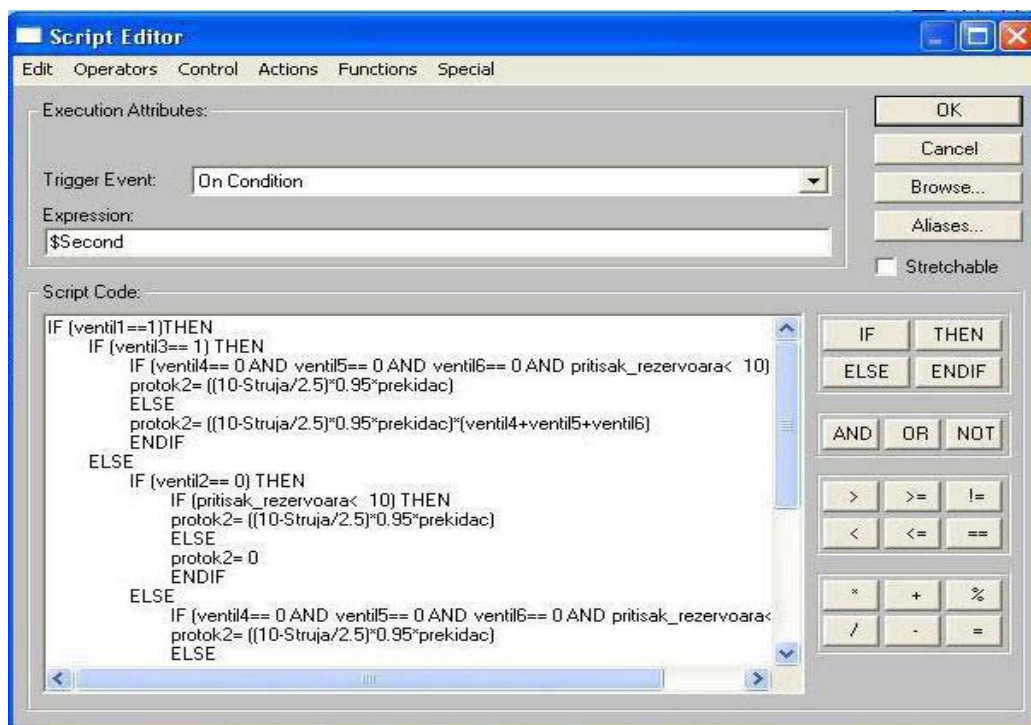
Programski kod

Svaki objekat se posebno programira i između njih postoji međusobna zavisnost (npr. protok1 i protok2, objekti za vizuelizaciju pritiska u cevima...). Programski kod se piše za funkcionalnost, vizuelizaciju ili animiranje objekata.

Animiranje objekata je znatno olakšano postojanjem *Animation editora*, u kojem se na jednostavan način može animirati svaki objekat (npr. punjenje i pražnjenje cevi, promena nivoa u rezervoarima, promena boja, prikaz analognih i digitalnih vrednosti promenljivih...).

Pokretanjem *Script Editora*, koji je u *Animation Editor-u*, pojavljuje se prozor kao na slici 3.11. i u njemu se unosi kod i slučaj, događaj kad on treba da se izvrši (to može biti svake sekunde, samo pritiskom na odgovarajući taster ili neki drugi od ponuđenih uslova).

Isti postupak se primenjuje za sve ostale elemente.



Slika 3.11. – Kod pisan u Skript Editoru

4. Integracija postojećih sistema

U praksi nije redak slučaj da su pojedina postrojenja u okviru istog preduzeća geografski udaljena. Tako se može dogoditi da se nadzorno-upravljački sistemi za upravljanje pojedinim postrojenjima razvijaju pomoću komponenti različitih proizvođača. Povezivanje SCADA sistema takvih postrojenja do sada najčešće nije bilo moguće ili je bilo neophodno proširiti komunikacione podsisteme sa novim protokolima za komunikaciju što predstavlja dugačak i komplikovan proces.

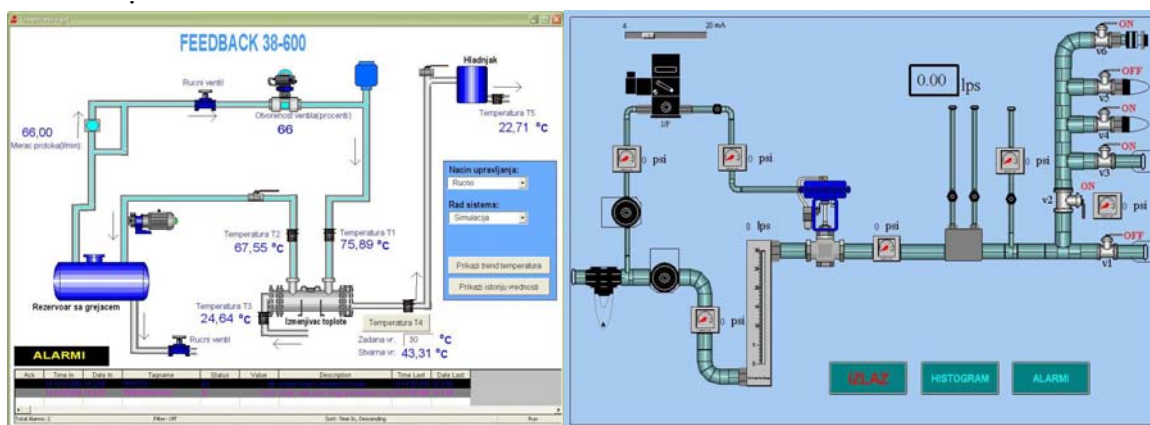
Povezivanje više podsistema SCADA sistema biće detaljnije objašnjeno u ovom poglavlju. Ono se može izvršiti na neki od sledećih načina:

1. Povezivanje pomoću postojećih OPC komponenti (*CX OPC Communications Control*)
2. Pisanjem Visual Basic programa koji preko univerzalnih interfejsa (*OPC DA*) pristupaju serverima

4.1. Korišćenje postojećih kontrola

Kako je specifikacija za OPC server veličina široko prihvaćena od strane proizvođača SCADA softvera (Omron, National Instruments, Festo...) povezivanje SCADA sistema različitih proizvođača je prilično olakšano. Da bi povezivanje bilo uspešno potrebno je da SCADA sistemi podržavaju OPC DA 2.0 specifikaciju.

Kao primer ovakvog povezivanja formiraćemo sistem koji omogućuje nadzor nad laboratorijsko-industrijskim tablama formiranim u poglavljima 2 i 3. Na slici 4.1 prikazana je tabla za regulaciju temperature realizovana u IFix 3.5 razvojnom alatu. Drugi deo sistema služi za nadzor i upravljanje tablom za protok i razvijen je u CX-Supervisor 1.1 okruženju proizvođača Omron Corporation.

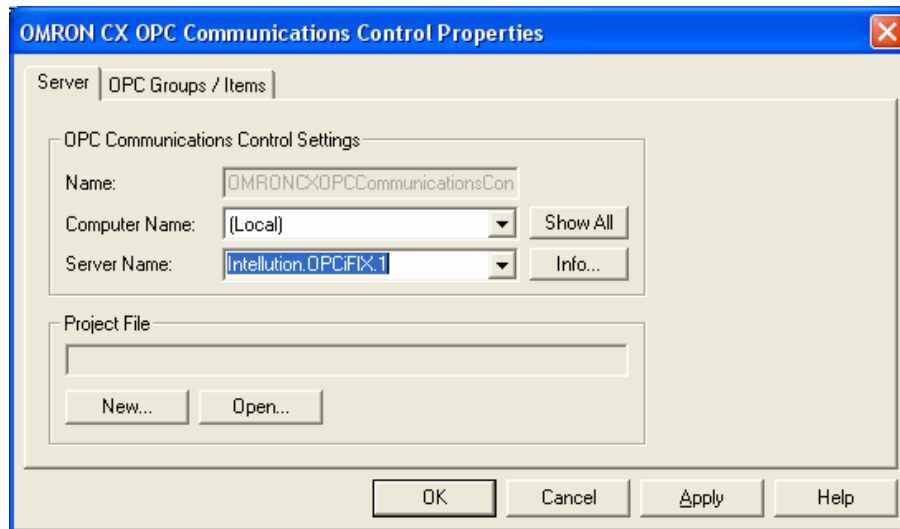


Slika 4.1. SCADA sistemi korišćeni za povezivanje (IFix i Omron)

Povezivanje ova dva sistema omogućeno je na taj način što Omron SCADA poseduje komponentu Omron CX OPC Communications Control koja može da

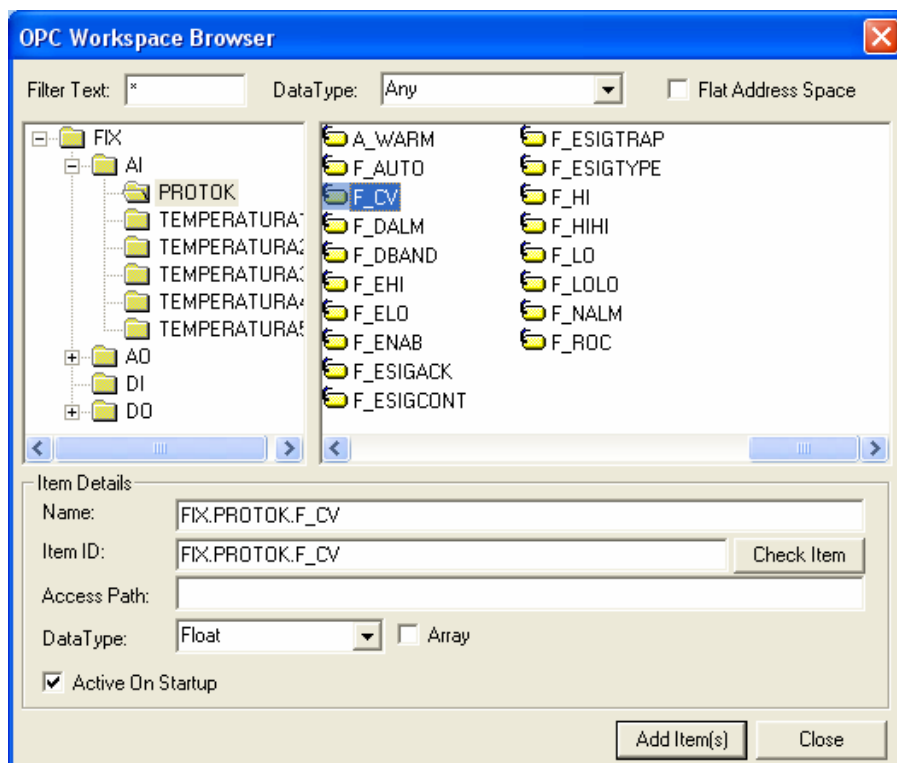
komunicira sa OPC DA 2.0 serverom. Takođe je bilo neophodno da Intellution IFix SCADA implementira interfejs OPC DA 2.0 specifikacije. Problem pri ovoj integraciji predstavlja činjenica da Omronova kontrola za povezivanje radi samo sa serverima koji se nalaze u okviru lokalne mreže sa računarima na kojima je Microsoft Windows operativni sistem.

Omron CX OPC Communications Control predstavlja univerzalni klijent OPC DA 2.0 servera. Ova kontrola se dodaje tako što se na formu prvo doda OLE kontrola a zatim se izabere “*OMRON OPC Communication Control*”. Prvo je potrebno definisati server sa kojim se želi komunicirati, što je prikazano na slici 4.2.

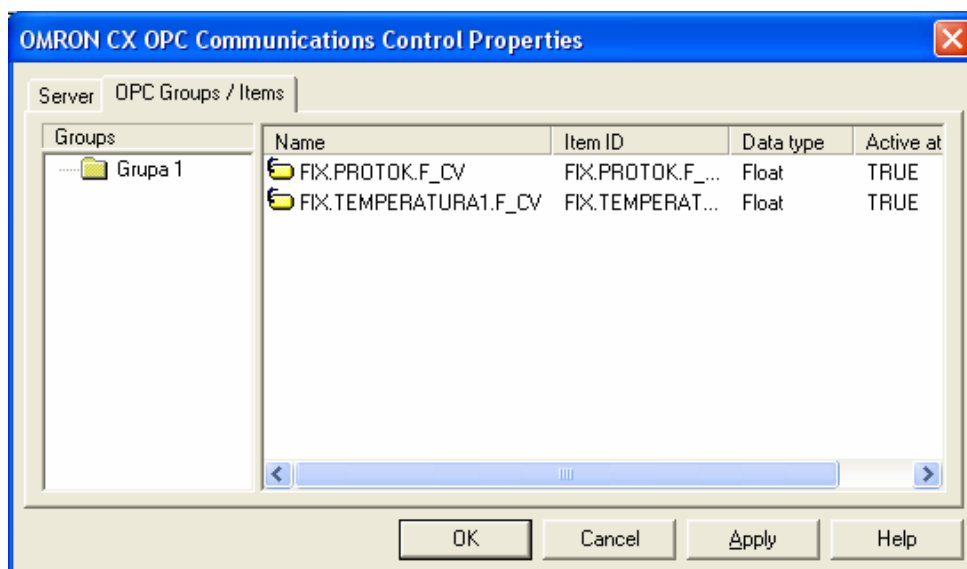


Slika 4.2. Definisanje OPC servera u OMRON OPC kontroli

Zatim je potrebno dodati grupu veličina, i veličine iz Intellution.OPCiFIX.1 servera koje je potrebno koristiti u OMRON okruženju. Moguće je dodati samo veličine prethodno definisane u IFix okruženju. Na slici 4.3 prikazan je postupak dodavanja veličina u grupu, dok je na slici 4.4. prikazana grupa veličina i veličine dodate u nju.

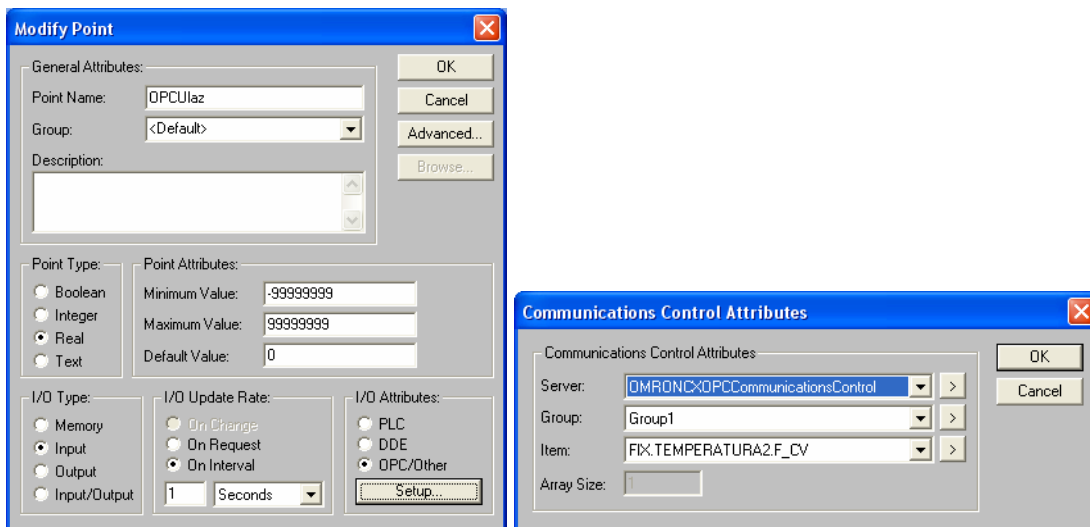


Slika 4.3. Dodavanje veličina u OMRON OPC kontroli



Slika 4.4. Prikaz grupe veličina i njenih veličina

Nakon ovoga je moguće dodati veličinu u OMRON koja dobija vrednost iz OPC servera na način prikazan na slici 4.5.



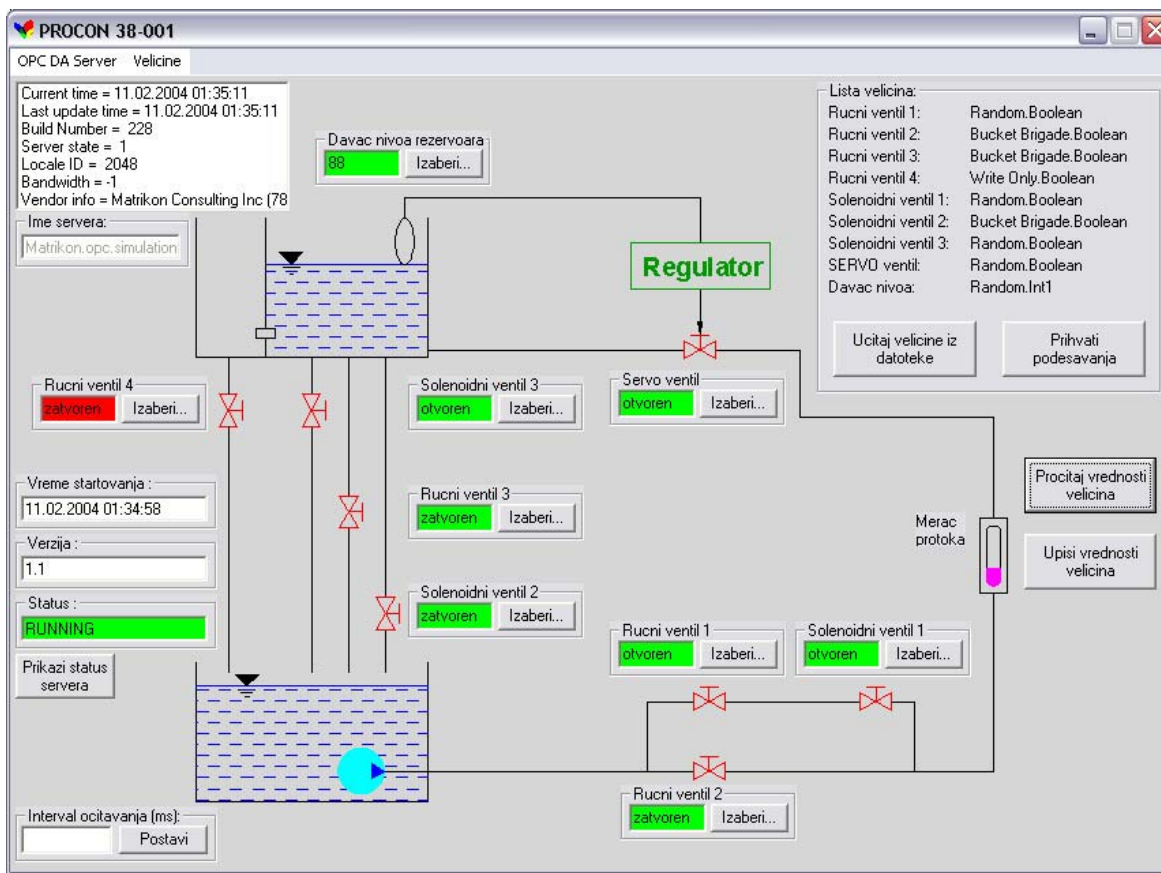
Slika 4.5 Dodavanje veličine dobijene iz IFIX OPC DA servera

4.2. Visual Basic OPC DA klijent

OPC (OLE For Process Control) fondacija je neprofitabilna organizacija koja za cilj ima definisanje specifikacija za interfejsse koji će biti korišćeni u industrijskim komunikacijama. Najširu primenu doživeo je OPC server za pristup trenutnim vrednostima veličina, Data Access (DA) server. Detaljan opis ove i ostalih OPC specifikacija biće priložen uz treći deo scripti iz ovog ispita.

Ovde će biti prikazan samo osnovni klijent za pristup serveru. Osnovne funkcije koje koristi ovakav klijent omogućuju povezivanje sa serverom, kreiranje grupe veličina, dodavanje veličina u grupu i čitanje i izmenu vrednosti veličina pomoću sinhronih metoda.

Takođe će biti predstavljen uprošćeni prikaz laboratorijsko-industrijske table za regulaciju nivom razvijen kao OPC DA klijent u Visual Basic-u. Forma koja se pojavljuje pokretanjem klijentske aplikacije prikazana je na slici 4.6.



Error! Slika 4.6. Izgled forme za nadzor

Na formi za nadzor procesa se dobijaju samo podaci o veličinama koje je korisnik izabrao iz servera veličina. Ukoliko je pravilno konfigurisana, forma za nadzor procesa korisniku omogućava i da izvrši očitavanje, kao i upis, veličina u bilo kom trenutku. Pored toga, korisnik u *StatusBar* kontroli stalno može da prati status servera, vreme startovanja i verzija servera.

Povezivanje je moguće izvesti i iz forme za nadzor i iz forme za konfigurisanje i to tako da se u meniju "OPC DA server" izabere opcija "Connect". Nakon unošenja imena servera veličina i izbora opcije "Connect" menija "OPC DA server", korisnik kreira objekat *OPCServer* i povezuje se sa serverom veličina.

Ispod opcije "Connect" menija "OPC DA server", nalazi se i opcija "Disconnect" koja omogućava prekid veze sa **OPC** serverom. Za povezivanje i prekid veze se serverom veličina korišćena je metoda *Connect*, odnosno *Disconnect*, objekta *OPCServer*.

OPC Data Access Automation klijent za nadzor i upravljanje laboratorijsko-industrijskom tablom PROCON 38-001 prati promene i vrednosti devet veličina iz procesa prikazanog na osnovnoj procesnoj tabli. Korisnik može u svakom trenutku da prekine očitavanje neke veličine i da je zameni sa drugom veličinom iz **OPC** servera. To se izvodi pritiskom na taster "Izmeni...". Nakon toga će se pojaviti forma sa slike 5.3. koja omogućava jednostavnu zamenu procesne veličine.

Očitavanje i upisivanje vrednosti veličina se vrši izborom opcije "Očitaj vrednosti velicina", odnosno opcije "Upisi vrednosti velicina", koje se nalaze u meniju "Velicine". Meni "Velicine" imaju i forma za nadzor procesa i forma za konfigurisanje. Forma za nadzor procesa ima i mogućnost da se pritiskom na odgovarajući taster koji se nalazi na formi, vrši čitanje i upisivanje vrednosti veličina.

Očitavanje vrednosti veličina će u *TextBox* kontrolama ispisivati vrednosti (*Values*) veličina iz liste veličina, a boja pozadine *TextBox* kontrole će dati kvalitet (*Quality*) očitane veličine. Boje su dodeljene tako da se kvalitetu *OPCQualityGood* pridružuje zelena boja, kvalitetu *OPCQualityBad* crvena boja, dok je kvalitetu *OPCQualityUncertain* pridružena žuta boja. U isto vreme će *ToolTip*-ovima *TextBox* kontrola biti pridruženo vreme očitavanja vrednosti veličina. Svi navedeni podaci biće prikazani i na formi za nadzor procesa i na formi za konfigurisanje.

Objekat OPCServer

Automation objekat *OPCServer* se povezuje sa **OPC** serverom korišćenjem metode *Connect* **OPC** servera. Tom prilikom se prosleđuje parametar *ime servera* i, opciono, parametar *Node*. Parametar *Node* predstavlja udaljeni računar sa kojim klijent može da komunicira koristeći **DCOM**. Parametar *ime servera* je korisnik naveo koristeći *TextBox* kontrolu forme za konfigurisanje. Deo programskog koda koji sledi opisuje upotrebu metode *Connect*.

```
Dim SERVER As OPCServer
Private Sub cmdSTART_Click()
```

```
    On Error GoTo errHandle
    Set SERVER = New OPCServer
    SERVER.Connect (txtImeServera.Text)
    SERVER.OPCGroups.Add ("NovaGrupa")
errHandle:
    MsgBox "Server nije podesen!", vbExclamation, "PROCON: GRESKA!"
End Sub
```

U ovoj proceduri se, ako dođe do pojave greške, naredbom *On Error GoTo errHandle* aktivira rutina za rukovanje greškom. O metodama za rukovanje greškama će biti reči nešto kasnije u radu.

Nakon kreiranja objekta *OPCServer* i povezivanja sa **OPC** serverom, korisnik dobija mogućnost i da pristupi osobinama servera veličina i da koristi metode koje su u serveru implementirane. Pristup nekim osobinama je prikazan programskim kodom koji sledi.

Objekat OPCGroup

Svaki objekat *OPCServer* sadrži i objekat *OPCGroups*, tj. objekat kolekcije **OPC** grupa. Ovaj objekat u sebi čuva informacije o grupama koje se nalaze na **OPC** serveru. Za kreiranje grupe i njeno dodavanje u kolekciju *OPCGroups* koristi se metoda *Add*.

OPCGroup objekat je neophodan za bilo kakvu manipulaciju ili pristup podacima iz izvora podataka, kojima je, naravno, dozvoljen pristup korišćenjem **OPC Data Access Automation** klijenta. Sledeće dve linije koda objašnjavaju na koji način se nekoj promenljivoj koja je tipa *OPCGroup* dodeljuje postojeća grupa.

```
Dim gru As New OPCGroup
Set gru = SERVER.OPCGroups.GetOPCGroup("NovaGrupa")
```

Pretpostavljeno je da je ime objekta *OPCServer SERVER*, i da se klijentska aplikacija povezala sa serverom veličina. Ovako se promenljivoj *gru* dodeljuje grupa koja se zove "NovaGrupa". Grupa "NovaGrupa" je član kolekcije grupa objekta *OPCServer* koji se zove *SERVER*. Nakon ovakvog definisanja promenljive *gru* (ova promenljiva je tipa *OPCGroup*), moguće je pristupati podacima koji se nalaze u toj grupi i u izvoru podataka. Tako je, na primer, pomoću sledeće linije koda prebrojan broj veličina koje se nalaze u kolekciji *OPCItems* grupe *gru* i dodeljen promenljivoj *I*, tipa integer.

```
I = gru.OPCItems.Count
```

U nastavku će biti navedeno kako se metoda *GetOPCGroup*, kolekcije *OPCGroup*, koristi za očitavanje i upisivanje vrednosti veličina u server veličina.

Objekat OPCItems

Svaki objekat *OPCGroup* sadrži u sebi i objekat kolekcije veličina, *OPCItems*, koji čuva informacije o veličinama te grupe (broj veličina u grupi, njihova imena,...). U nastavku sledi deo koda koji prikazuje kako se u kolekciju *OPCItems* korišćenjem metode *AddItem* dodaju veličine u grupu *gru* objekta *OPCServer*. Objekat *OPCServer* se zove *SERVER*.

```
Private Sub cmdPostaviPROM_Click()
Dim gru As New OPCGroup
Dim Item As OPCItem
Set gru = SERVER.OPCGroups.GetOPCGroup("NovaGrupa")
With gru.OPCItems
Set Item = .AddItem(CStr(lblRV1.Caption), 1)
Set Item = .AddItem(CStr(lblRV2.Caption), 2)
Set Item = .AddItem(CStr(lblRV3.Caption), 3)
Set Item = .AddItem(CStr(lblRV4.Caption), 4)
Set Item = .AddItem(CStr(lblSV1.Caption), 5)
Set Item = .AddItem(CStr(lblSV2.Caption), 6)
Set Item = .AddItem(CStr(lblSV3.Caption), 7)
Set Item = .AddItem(CStr(lblSERVO.Caption), 8)
Set Item = .AddItem(CStr(lblDavac.Caption), 9)
End With
End Sub
```


Kao što se se može videti i u ovoj proceduri se prvo koristi metoda *GetOPCGroup* objekta *OPCGroups*, čime se određuje kojoj se grupi želi pristupiti i dodati veličina. Tek potom se vrši dodavanje veličine i to zadavanjem imena i, opciono, određivanjem njenog jedinstvenog identifikatora (*ClientHandle*). Tako je, na primer, u jednoj od linija koda za ime veličine izabran natpis (*Caption*) labela koja se zove *lblSERVO* i toj veličini je dodeljen jedinstveni identifikator 8.

Objekat *OPCItem*

Relevantne fizičke veličine, procesa koji se odvijaju u nekom postrojenju predstavljene su veličinama (*OPCItem*-ima). Pristupanjem bilo kojoj veličini iz procesa moguće je dobiti vrednosti i osobine te veličine (trenutna vrednost, kvalitet, vreme poslednjeg očitavanja, prava pristupa, itd.). U prethodnom tekstu je objašnjen način pristupa veličinama i njihovim osobinama preko grupa. To, međutim, nije jedini način pristupa veličinama. **OPC Data Access Automation** interfejs omogućava da se podaci o osobinama (*Properties*) veličina dobiju i "direktnim" pristupanjem određenoj veličini. "Direktno" pristupanje je u smislu da se veličina, kojoj se želi pristupiti i čije nas osobine interesuju, ne mora dodati u grupu. Ovaj pristup nam omogućavaju metode *OPCServera* koje će biti nabrojane i objašnjene u nastavku teksta.

Metoda **OPC** servera *QueryAvailableProperties* vraća listu ID kodova i opisa postojećih osobina (*Property*) za te ID kodove. Metoda **OPC** servera *GetItemProperties* vraća listu trenutnih vrednosti za prosleđene ID kodove osobina (*Property*). Poslednja je metoda **OPC** servera *LookupItemIDs* koja vraća listu identifikatora veličina (*ItemIDs*) za svaki prosleđeni ID kod osobine (*Property*).

Za svaku veličinu iz servera moguće je očitavanje i upisivanje njene vrednosti. To se radi korišćenjem metode *SyncRead* odnosno *SyncWrite* objekta *OPCGroup*. Ovim metodama je omogućeno čitanje, odnosno upisivanje, kako svih vrednosti veličina iz navedene grupe, tako i svake veličine pojedinačno.

Zahtevi koje je razvijana klijentska aplikacija morala da zadovolji, su i da postoji mogućnost čitanja i upisivanja, i da, u zavisnosti od pročitanih vrednosti veličina, kontrolama na formi promeni neke osobine. Tako je, na primer, zahtevano postavljanje boje pozadine *TextBox* kontrole u kojoj je ispisana vrednost veličine, u skladu sa kvalitetom (*Quality*) veličine. Za tu *TextBox* kontrolu, takođe je zahtevano i da njen "*ToolTip*" prikazuje vreme očitavanja vrednosti posmatrane veličine.

Sledeći kod prikazuje korišćenje metode *SyncRead*, a nakon toga i metode *SyncWrite*.

```
Public Sub cmdRead_Click()  
Dim Source As Integer  
Dim srvHandle() As Long  
Dim Values() As Variant  
Dim Err() As Long  
Dim gru As New OPCGroup  
Dim Prom(9) As OPCItem  
Dim i As Integer
```

```
Set gru = SERVER.OPCGroups.GetOPCGroup("NovaGrupa")
```

```

ReDim srvHandle(gru.OPCItems.Count)

For i = 1 To gru.OPCItems.Count
    Set Prom(i) = gru.OPCItems.Item(i)
    srvHandle(i) = Prom(i).ServerHandle
    ItemID = Prom(i).ItemID
Next i

Source = 1 ' 1 je OPC_DS_CACHE
On Error GoTo errHandle
gru.SyncRead Source, gru.OPCItems.Count, srvHandle, Values, Err

If VelicinaLOCK(1) = False Then
    Quality2Color Prom(1).Quality, txtValue1
    ValueNaSrpskom Values(1), txtValue1
    txtValue1.ToolTipText = ("Vreme: " & Prom(1).TimeStamp)
    Quality2Color Prom(1).Quality, frmUSER.txtValue1
    ValueNaSrpskom Values(1), frmUSER.txtValue1
    frmUSER.txtValue1.ToolTipText = ("Vreme: " & Prom(1).TimeStamp)
End If
.
.
If VelicinaLOCK(9) = False Then
    Quality2Color Prom(9).Quality, txtValue9
    txtValue9.Text = Values(9)
    txtValue9.ToolTipText = ("Vreme: " & Prom(9).TimeStamp)
    Quality2Color Prom(9).Quality, frmUSER.txtValue9
    frmUSER.txtValue9.Text = Values(9)
    frmUSER.txtValue9.ToolTipText = ("Vreme: " & Prom(9).TimeStamp)
End If

TmrSAT.Enabled = True
Set Item = Nothing
Set gru = Nothing
Exit Sub
errHandle:
MsgBox ("Proverite da li ste dobro ispisali imena velicina koje zelite da posmatrate!"), vbExclamation,
"PROCON 38-001: GRESKA!"
End Sub

```

Deo u proceduri *cmdRead_Click* koji opisuje dodelu vrednosti, kvaliteta i boje ("if petlja"), je izostavljen, jer se taj deo koda ponavlja za svih devet veličina koje posmatramo. Ovde je prikazan samo kod kojim se ove tri osobine, dodeljuju delovima *TextBox* kontrola koje se zovu *txtValue1* i *txtValue9*.

Procedura *cmdRead_Click* očitava vrednosti (*Values*) navedenih osam veličina, njihove kvalitete (*Quality*) i vremena kada su očitavanja izvršena (*TimeStamp*). Promenljiva kojoj je dodeljena vrednost 1 (*source = 1*) označava izvor podataka. U ovoj klijentskoj aplikaciji, izvor podataka kome se pristupa za dobijanje vrednosti veličina je

keš memorija (*OPC_DataSource_Cache*). Pored keš memorije, kao izvoru podataka za dobijanje vrednosti veličina iz servera veličina, moguće je i direktno pristupati uređajima koji očitavaju vrednosti tih veličina (*OPC_DataSource_Device*). Ukoliko je kao izvor podataka naveden uređaj, dolazi do direktnog pristupa uređaju. Uređaj potom očitava traženu veličinu, i na kraju se od njega preuzimaju podaci o vrednosti veličine. Još jedna razlika između ova dva načina pristupa podacima o veličinama se ogleda u tome što, kod očitavanja vrednosti veličine iz keš memorije, i grupa u kojoj se veličina nalazi i veličina moraju biti aktivna. Ukoliko to nije slučaj, osobina veličine kvalitet (*Quality*) će promeniti stanje u *OPC_QUALITY_OUT_OF_SERVICE*. Kada je izvor podataka uređaj, stanje grupe i veličine nije bitno za očitavanje trenutne vrednosti veličine.

I u ovoj proceduri je korišćena naredba *On Error GoTo errHandle* koja pri pojavi greške aktivira rutine za obradu greške. U nastavku sledi kod koji prikazuje kako je u ovom radu korišćena metode *SyncWrite*.

```
Private Sub cmdWrite_Click()  
Dim srvHandle() As Long  
Dim Values() As Variant  
Dim Err() As Long  
Dim gru As New OPCGroup  
Dim Item As OPCItem  
Dim I As Integer  
  
Set gru = SERVER.OPCGroups.GetOPCGroup("NovaGrupa")  
ReDim srvHandle(gru.OPCItems.Count)  
ReDim Values(gru.OPCItems.Count)  
  
For I = 1 To gru.OPCItems.Count  
Set Item = gru.OPCItems.Item(I)  
srvHandle(I) = Item.ServerHandle  
Next I  
Values(1) = txtValue1.Text  
VelicinaLOCK(1) = False  
Values(2) = txtValue2.Text  
VelicinaLOCK(2) = False  
Values(3) = txtValue3.Text  
VelicinaLOCK(3) = False  
Values(4) = txtValue4.Text  
VelicinaLOCK(4) = False  
Values(5) = txtValue5.Text  
VelicinaLOCK(5) = False  
Values(6) = txtValue6.Text  
VelicinaLOCK(6) = False  
Values(7) = txtValue7.Text  
VelicinaLOCK(7) = False  
Values(8) = txtValue8.Text  
VelicinaLOCK(8) = False  
Values(9) = txtValue9.Text  
VelicinaLOCK(9) = False
```

```
gru.SyncWrite gru.OPCItems.Count, srvHandle, Values, Err

Set gru = Nothing
Set Item = Nothing
End Sub
```

Niz od devet promenljivih koje se zovu *VelicinaLOCK (i)*, gde je *i* indeks koji označava o kojoj promenljivoj je reč *i* može biti od 1 do 9, je tipa *Boolean* i menja vrednost ukoliko korisnik postavi fokus na *TextBox* kontrolu kojoj je dodeljena promenljiva. Na ovaj način je korisniku data mogućnost da unese izmene u *TextBox* kontrolu i da pozivom metode *SyncWrite* izmeni vrednost veličine, bez obzira na to koliki je interval očitavanja vrednosti veličina. Tek nakon što unese sve izmene koje želi, korisnik poziva metodu za upisivanje unesenih vrednosti.