

CONTROL OF AUTOMATION SYSTEMS THROUGH PORTABLE DEVICES

Lt.col.lect.univ.drd.ing. SORA Daniel

Departamentul Regional de Studii pentru Managementul Resurselor de Apărare

Abstract:

Many people will be able to leave their laptop at the office and handle essentially all of their mobile computing and communications tasks with a pocket-sized device. Now, the smartphone might not be one's first choice for spreadsheets and documents, but let's face it, everybody has one (in business, anyway, all over the world), and those that have one will buy a new one sometime in the next two years as cellular contracts expire and products and wireless technologies continue their rapid evolution. Smartphones are as powerful as PCs from just a few years ago, with significantly better software, user interfaces, and flexibility. The level of capability in contemporary smartphones is remarkable and continues to grow.¹⁸² PDA devices and other mobile handheld devices make it easier than ever to develop remote applications that transmit and receive information from a remote site back to a host computer. The LabVIEW PDA Module extends the LabVIEW graphical development environment to PDA devices so we can create PDA applications that run on Pocket PC and Windows Mobile devices. We can create portable solutions for a wide spectrum of applications, such as field test systems, remote control and monitoring systems, and portable data acquisition systems.

Noțiuni introductive

Dezvoltarea rapidă a comunicațiilor fără fir din ultimii ani a permis utilizatorilor să se conecteze la internet aproape de oriunde, fie prin intermediul punctelor de acces (gratuite sau nu), fie prin infrastructura companiilor de telefonie mobilă care oferă viteze de conectare tot mai mari, la prețuri accesibile. Integrarea

¹⁸² Craig Mathias – “*Can the smartphone replace the laptop?*”, MOBILE TECHNOLOGIES AND TRENDS, 08.10.2009

conceptelor de PDA (inițial creat pentru a fi un fel de agendă electronică) și de telefon inteligent (smartphone) a creat posibilitatea de a avea la îndemână un mic computer care poate să ofere acum mult mai multe funcții decât ne puteam aștepta în urmă cu câțiva ani. Acestea au devenit la fel de puternice ca și PC-urile de la începutul acestui deceniu, având însă progrese semnificative în ceea ce privește flexibilitatea, interfața cu utilizatorul sau programele disponibile, și capacitățile acestora continuă să crească.

Popularitatea în creștere a dispozitivelor mobile, dar și instrumentele de dezvoltare puse la dispoziție de producătorii de sisteme de operare mobile sau de producătorii de medii de dezvoltare profesionale (cum este National Instruments) au permis crearea de programe pentru PDA-uri și telefoane inteligente capabile să transmită și să recepționeze informații la / de la un server de date, rezultând soluții portabile pentru o gamă largă de aplicații cum sunt sistemele de testare portabile, cele de monitorizare și / sau control, sau sistemele portabile de achiziții de date.

Modulul PDA (sau Mobile cum se numește în cea mai recentă versiune) din pachetul LabVIEW produs de National Instruments extinde mediul grafic de dezvoltare LabVIEW pentru a permite crearea de programe pentru dispozitive portabile care rulează sistemul de operare Windows Mobile.

LabVIEW utilizează modelul Open System Interconnection (OSI), astfel că implementarea rețelelor de comunicații fără fir este la fel de simplă ca și în cazul rețelelor clasice cu fir. În cadrul modelului OSI, rețelele de comunicații fără fir (cunoscute și sub codificarea IEEE 802.11) reprezintă doar unul din cele șapte niveluri existente. Suportul pentru comunicațiile Bluetooth oferit începând cu versiunea LabVIEW 7.1 extinde opțiunile disponibile pentru crearea de aplicații fără fir pentru PDA.

Metode de comunicații în rețea în LabVIEW

Instrumentele virtuale (VI) – elementele de bază ale mediului de programare grafică LabVIEW – pot comunica în rețea cu alte procese, inclusiv cele care rulează în alte aplicații sau pe alte calculatoare, pentru a îndeplini următoarele tipuri de sarcini:

- Schimburi de date în timp real cu alte instrumente virtuale din rețea utilizând mecanismul variabilelor distribuite (shared variables);
- Publicarea pe internet a imaginilor panourilor de control și a documentației instrumentelor virtuale;
- Transmiterea de date ale instrumentelor virtuale prin e-mail sau SMS;
- Construirea de instrumente virtuale care comunică cu alte aplicații sau instrumente virtuale prin protocoale de nivel scăzut, cum sunt TCP sau UDP.

Fiecare facilitate de comunicare în rețea din LabVIEW corespunde unor nevoi ale diferitelor aplicații, astfel:

- Shared Variable – pentru utilizarea în comun a datelor în timp real cu alte instrumente virtuale de pe același calculator, de pe un alt calculator din rețea, sau cu un server;

- DataSocket cu protocolul PSP sau DSTP – pentru utilizarea în comun a datelor în timp real cu alte instrumente virtuale de pe același calculator, de pe un alt calculator din rețea, sau cu un server (necesită programare suplimentară față de mecanismul shared variable);
- DataSocket cu protocolul OPC – pentru citirea datelor dintr-o rețea industrială cu utilizarea DataSocket ca și client OPC;
- Funcțiile TCP – pentru comunicarea cu un instrument care utilizează un protocol bazat pe TCP;
- Funcțiile UDP – pentru comunicarea cu un pachet software care utilizează un protocol bazat pe UDP;
- Funcțiile IrDA – stabilesc o conexiune fără fir în infraroșu cu un alt calculator;
- Funcțiile Bluetooth – stabilesc o conexiune fără fir cu un dispozitiv Bluetooth.

Pe lângă aceste facilități de comunicare în rețea, se pot pune la dispoziție și accesa date în LabVIEW prin următoarele metode:

- Funcțiile File I/O – pentru crearea de fișiere care conțin date care pot fi citite de alte aplicații sau instrumente virtuale;
- VI Server – pentru controlul altor instrumente virtuale sau aplicații LabVIEW de pe calculatorul local sau de pe alte calculatoare din rețea;
- Funcțiile ActiveX – pentru accesarea facilităților multora dintre aplicațiile Microsoft (de exemplu pentru inserarea reprezentărilor grafice ale formelor de undă în foile de calcul Excel).

Aplicații de rețea de tip Client – Server în LabVIEW

În LabVIEW se pot crea aplicații de tip client pentru a subscrie la date și a folosi facilități ale altor aplicații sau de tip server pentru a pune la dispoziție datele sau facilitățile LabVIEW altor aplicații.

Pentru a putea accesa date sau proprietăți și a invoca metode ale altei aplicații, trebuie stabilit protocolul de rețea corespunzător. De exemplu, protocolul HTTP este ideal pentru publicarea pe internet a panourilor de control ale instrumentelor virtuale, dar pentru crearea de instrumente virtuale care utilizează date create de alte instrumente virtuale trebuie utilizat protocolul TCP/IP. De asemenea se pot utiliza tehnologiile ActiveX în LabVIEW pentru a crea aplicații server sau client ActiveX.

Protocolul TCP în rețelele fără fir, ca și în cele cu fir, asigură transmisiile în rețea și livrarea datelor în ordinea corectă, fără erori, pierderi sau duplicări. Conexiunea TCP retransmite automat datagrama care conține datele împreună cu un antet care indică adresele sursă și destinație, până când primește confirmarea transmisiei.

În LabVIEW, stabilirea unei comunicații TCP este similară cu stabilirea unui proces de citire/scriere de fișiere sau a unei comunicații de intrare/ieșire cu un instrument, presupunând: deschiderea conexiunii, citirea/scrierea datelor, închiderea conexiunii. Se poate folosi un PDA ca și gazdă a unei aplicații de tip client sau server.

În figura 1 este prezentat un exemplu tipic de aplicație client care inițiază o conexiune cu un server la distanță în rețea.

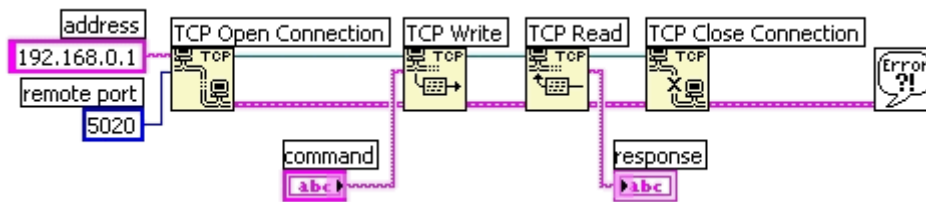


Figura 1. Exemplu de comunicație de tip Client TCP

Funcția **TCP Open Connection** deschide o conexiune de rețea TCP la adresa 192.168.0.1 folosind portul 5020, funcția **TCP Write** scrie datele prin conexiunea deschisă de funcția **TCP Open Connection**, funcția **TCP Read** citește un număr de octeți reprezentând răspunsul primit prin conexiunea de rețea, iar funcția **TCP Close Connection** închide conexiunea TCP.

În figura 2 este prezentat un exemplu tipic de aplicație server care așteaptă conexiuni de la distanță și le răspunde în mod corespunzător.

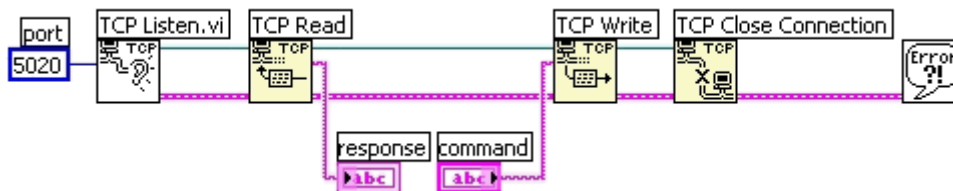


Figura 2. Exemplu de comunicație de tip Server TCP

Instrumentul virtual **TCP Listen VI** creează un proces care așteaptă o conexiune de rețea TCP la portul 5020, funcția **TCP Read** citește un număr de octeți de la conexiunea TCP, funcția **TCP Write** scrie datele prin conexiunea de rețea, iar funcția **TCP Close Connection** închide conexiunea TCP.

Utilizarea în comun a datelor în timp real prin mecanismul Shared Variables

Variabilele distribuite (Shared variables) sunt elemente software pre-configurate care permit transmiterea datelor între instrumentele virtuale, reprezentând valori sau puncte I/O. Acestea pot fi create numai pe calculatoare care rulează versiuni ale sistemelor de operare Windows sau pe sisteme în timp real cu utilizarea modulului LabVIEW Real Time. Pentru citirea sau scrierea variabilelor de pe alte platforme se utilizează funcțiile DataSocket.

Opțiunile de configurare ale variabilelor distribuite sunt disponibile prin intermediul casetei de dialog Shared Variable Properties. Aici, în cadrul meniului Variable Type se poate opta între valorile Network-Published în cazul în care se dorește ca variabilele să poată fi citite sau scrise de pe calculatoare diferite conectate la rețea, sau Single-Process dacă variabilele vor fi disponibile doar pe calculatorul pe care au fost create. Datele de configurare ale variabilelor create sunt salvate de LabVIEW într-un fișier cu extensia .lvlib, care conține informații despre toate variabilele din biblioteca asociată proiectului.

Mecanismul Shared Variable Engine (SVE) utilizează protocolul NI Publish-Subscribe Protocol (NI-PSP) pentru transferul datelor corespunzătoare

variabilelor distribuite în rețea. LabVIEW identifică variabilele distribuite prin intermediul unei căi de rețea care conține numele calculatorului, numele bibliotecii proiectului și numele variabilei respective. Aceste biblioteci se numesc procese, și de altfel rulează ca procese în cadrul sistemului de operare. Când aplicația rulează, SVE controlează procesul și transferul datelor variabilelor distribuite.

Pe calculatoarele care rulează sistemul de operare Windows, se poate integra un server OPC prin mecanismul SVE. Orice aplicație client OPC poate citi sau scrie date prin SVE. Astfel, Shared Variable Engine poate fi considerat un server OPC implicit al mediului de programare LabVIEW. Singura deosebire este aceea că instrumentul virtual care creează biblioteca de variabile distribuite poate avea și funcții de achiziții de date sau de control al procesului.

Pentru a putea transmite datele corespunzătoare variabilelor distribuite prin firewall-uri dau routere, acestea trebuie configurate. Astfel, pentru calculatoarele care rulează sistemul de operare Windows XP sau o versiune mai nouă, în lista de excepții a firewall-ului trebuie introduse următoarele fișiere:

- C:\Windows\system32\lkads.exe
- C:\Windows\system32\lktsrv.exe
- C:\Program Files\National Instruments\Shared\Tagger\tagsrv.exe
- C:\Windows\system32\nicid15.exe (dacă se utilizează LabVIEW DSC)
- C:\Program Files\National Instruments\DataSocket\cwdss.exe (dacă se utilizează LabVIEW DataSocket)

În cazul routerelor, trebuie configurată tabela de conversie Network Address Translating (NAT) pentru a realiza corespondența dintre adresele IP locale și adresele IP publice.

Sursele posibile de date pentru variabilele distribuite sunt:

- Canalele de intrare/ieșire ale plăcilor de achiziție de date;
- Itemi de date din cadrul instrumentelor virtuale;
- Itemi de date ale protocolului NI-PSP, de exemplu variabile distribuite ale altei aplicații;
- Itemi de date ale serverelor de intrare/ieșire definite în afara proiectului activ.

Configurarea sursei de date a variabilelor distribuite se face prin bifarea căsuței *Bind to Source* de pe subpagina *Variable* din caseta de dialog *Shared Variable Properties*. În cazul configurării sursei de date pe o altă mașină țintă din rețea, LabVIEW nu transmite în mod automat toate schimbările de configurație de la sursa de date. Astfel, reconfigurările sursei de date pot conduce la întreruperea conexiunii variabilei distribuite.

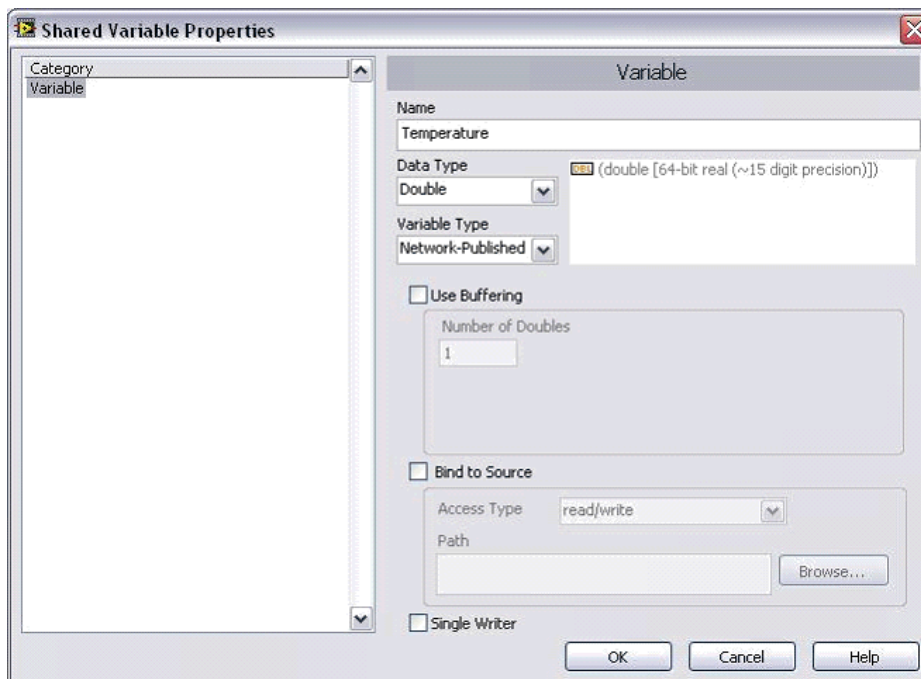


Figura 3. Caseta de dialog *Shared Variable Properties*

Pentru a adăuga o variabilă distribuită într-un proiect, se face click dreapta pe o țintă, pe o bibliotecă a proiectului sau pe un folder din biblioteca proiectului din fereastra *Project Explorer* și se selectează **New»Variable** din meniul contextual pentru a accesa caseta de dialog *Shared Variable Properties*. Aici se setează opțiunile de configurare dorite, apoi se apasă butonul **OK**. Variabilele create se vor găsi într-o bibliotecă a proiectului. Dacă se creează o variabilă pe o țintă sau pe un folder care nu se găsește într-o bibliotecă a proiectului, LabVIEW creează o nouă bibliotecă și plasează noua variabilă creată în interiorul acesteia.

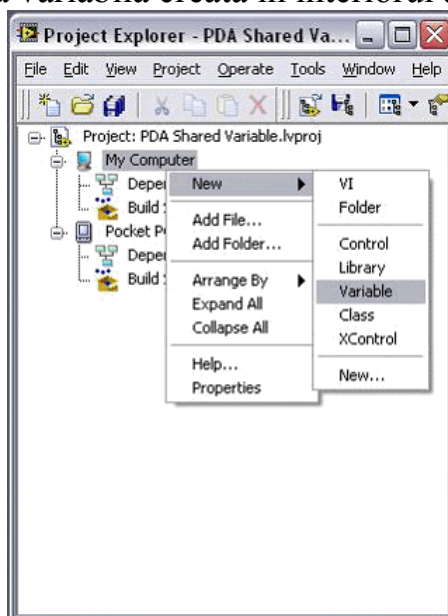


Figura 4. Crearea unei noi variabile în fereastra *Project Explorer*

Fiecare variabilă distribuită de pe o mașină țintă are o adresă pe care protocolul NI-PSP o folosește pentru identificare. Înainte de a putea accesa o variabilă distribuită din alte aplicații cum sunt panouri de control, noduri de

variabile distribuite din cadrul diagramelor bloc, sau alte variabile distribuite, biblioteca de variabile distribuite a proiectului trebuie publicată în rețea. Pentru aceasta, se face click dreapta pe biblioteca respectivă și se selectează *Deploy* din meniul contextual. În cazul folosirii modulului în timp real (RT Module), în meniul contextual apare opțiunea *Deploy All* care publică în rețea toate bibliotecile conținând variabile distribuite de pe mașina țintă, dar și toate instrumentele virtuale care conțin variabile distribuite de pe mașina respectivă.

Se poate utiliza și meniul dialog *Variable Manager* pentru a edita, crea și monitoriza variabile distribuite în afara mediului proiectului. Trebuie avut în vedere faptul că dacă se adaugă noi variabile distribuite unei biblioteci prin acest meniu, noile variabile vor putea exista numai până când biblioteca originală a proiectului va fi publicată din nou.

Configurarea sau reconfigurarea variabilelor distribuite se poate face prin efectuarea unui click dreapta pe variabila respectivă și selectarea opțiunii *Properties* din meniul contextual, care afișează caseta de dialog *Shared Variable Properties*. Ca setare implicită, o variabilă poate fi accesată pentru scriere de multiple aplicații. Prin bifarea opțiunii *Single Writer* se stabilește ca o singură aplicație la un moment dat să poată scrie date în variabila respectivă, protejând astfel operația de scriere să fie afectată de o altă aplicație care accesează variabila în același timp. Mecanismul variabilelor distribuite restricționează operația de scriere a datelor la o singură aplicație de pe aceeași mașină, astfel că prima aplicație care se conectează la variabila respectivă poate efectua scrierea, iar celelalte primesc o notificare sub forma unui mesaj de eroare.

După efectuarea oricărei modificări de configurație a unei variabile distribuite, biblioteca în care este conținută aceasta trebuie republicată în rețea. De asemenea se poate utiliza proprietatea *Variable Reference* pentru a modifica configurațiile din interiorul unui instrument virtual, prin programare.

Utilizarea variabilelor distribuite în cadrul instrumentelor virtuale pentru dispozitive portabile de tip PDA sau Touch Panel

Variabilele distribuite pot fi utilizate pentru a utiliza în comun datele de către instrumente virtuale care rulează pe aceeași mașină sau pe mașini diferite conectate în rețea. Modulele PDA și Touch Panel nu suportă protocolul DataSocket Transport Protocol (DSTP) sau găzduirea de variabile distribuite în rețea, ceea ce înseamnă că pe aceste dispozitive se pot crea numai aplicații client, care pot accesa pentru citire/scriere variabilele distribuite puse la dispoziție de aplicații server care rulează pe alte tipuri de echipamente.

Mecanismul Shared Variable Engine (SVE) nu rulează pe mașini de tip PDA sau Touch Panel, iar înainte de a încerca să se ruleze aplicații care accesează variabile publicate în rețea trebuie instalat suportul LabVIEW pentru variabile distribuite, altfel se va primi un mesaj de eroare. Pentru instalare, având PDA-ul conectat la calculator și sincronizat prin aplicația ActiveSync se rulează programul de instalare aflat în locația labview\PDA\Utilities\Variables\Setup.exe, sau prin

click dreapta pe ținta PDA sau Touch Panel din fereastra *Project Explorer* de unde se selectează *Install»Support for Shared Variables*.

Crearea aplicațiilor wireless pentru dispozitivele portabile de tip PDA sau Touch Panel

Aplicațiile wireless pentru PDA sau Touch Panel se constituie dintr-un client și un serviciu. Clientul este aplicația PDA sau Touch Panel care monitorizează și/sau controlează servicii la distanță prin rețea și care comunică direct cu un serviciu de pe un server sau de pe un alt dispozitiv. Serviciile sunt instrumente virtuale sau alte aplicații care realizează procese pe care le accesează clienții.

Pentru exemplificare, am realizat un experiment în care instalația de automatizare este compusă dintr-o placa de achiziții NI USB-6009 prin care se comandă tensiunea de alimentare a unei grupări în serie LED și un rezistor, și se măsoară tensiunea de pe rezistor. Placa este conectată prin USB la un laptop, pe care rulează aplicația server. Fereastra *Project Explorer* corespunzătoare experimentului este prezentată în figura 5, în care se observă biblioteca proiectului *PDA Remote Library 1.lvlib* și aplicația server *PDA Remote.vi* aflate pe *My Computer*, iar aplicația *PDA Client.vi* aflată pe PDA (*Windows Mobile 5.0 Pocket PC Device*).

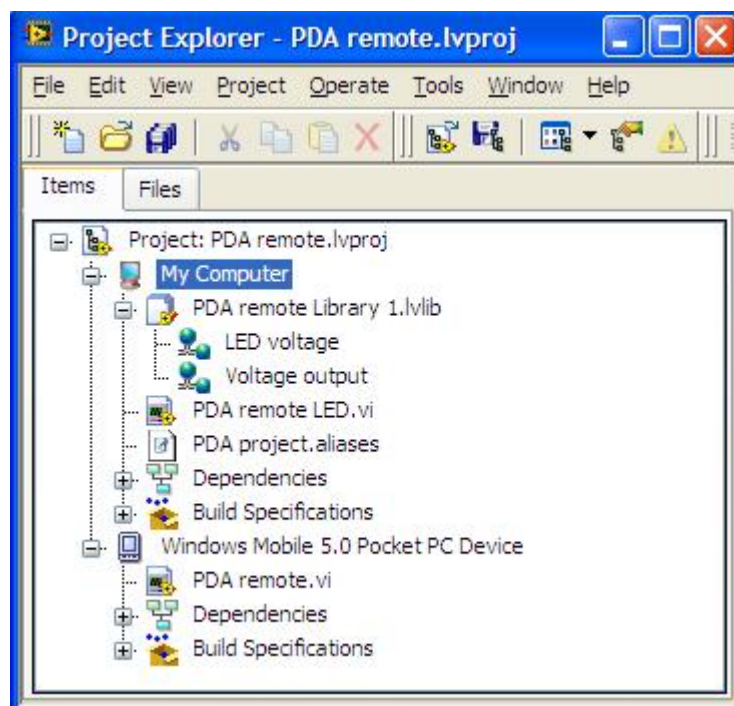


Figura 5. Fereastra *Project Explorer* a experimentului

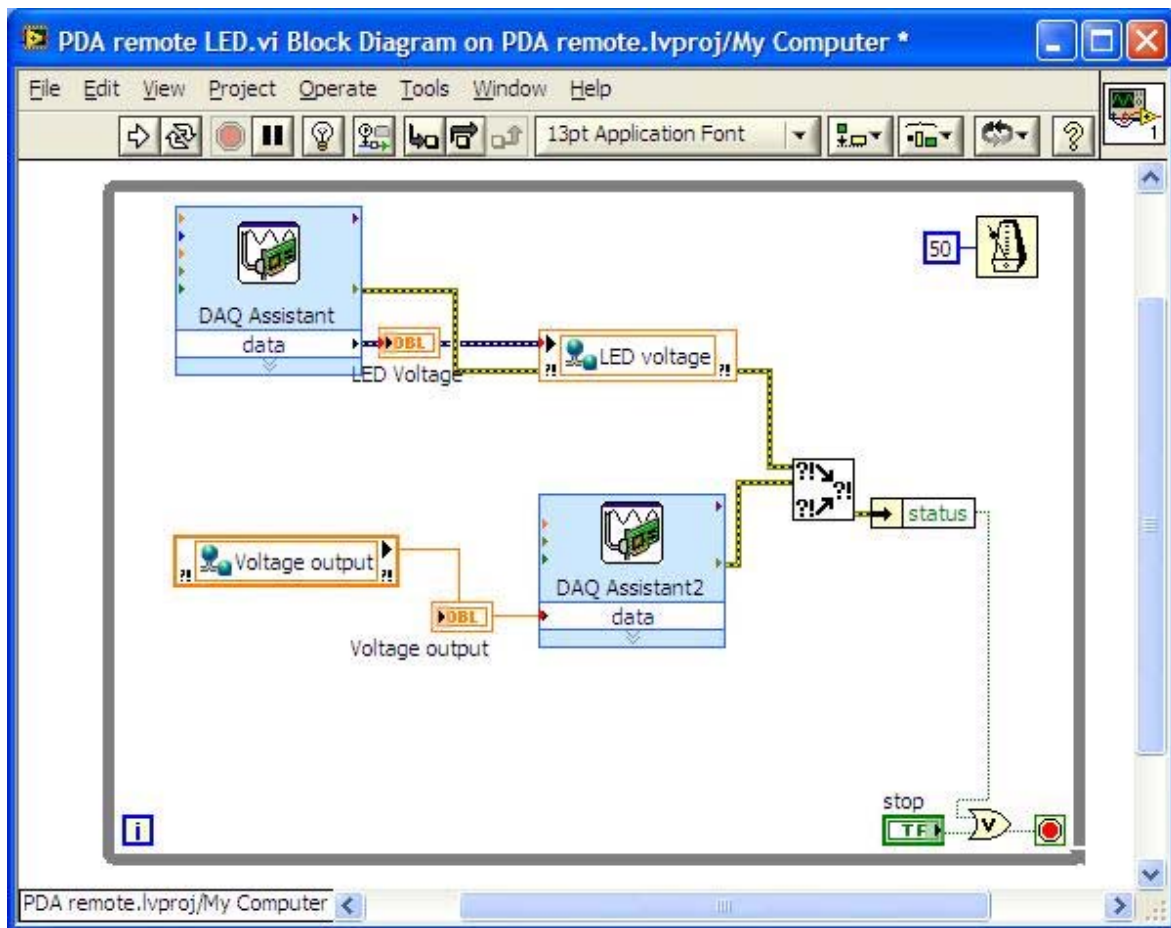


Figura 6. Diagrama bloc a aplicației server

Prin aplicația server a cărei diagramă bloc este prezentată în figura 6, se accesează prima ieșire analogică a plăcii de achiziție, la care se transmite valoarea tensiunii de ieșire conținută de variabila distribuită *Voltage Output*, și se citește valoarea tensiunii furnizate de prima intrare analogică a plăcii, care măsoară tensiunea de pe rezistor, valoare transferată variabilei distribuite în rețea *LED Voltage*. Mesajele de eroare furnizate de cele două operații de intrare/ieșire se concatenează, și împreună cu o comanda *stop* de la un buton dispus pe panoul frontal al aplicației server, pot întrerupe funcționarea serverului. Actualizarea informațiilor este controlată de un timer, la fiecare 50 milisecunde.

Aplicația client, a cărei diagramă bloc este prezentată în figura 7, preia tensiunea de la variabila distribuită *LED Voltage* și o afișează în două moduri (grafic și numeric) și furnizează tensiunea de comandă preluată de la un control grafic variabilei distribuite *Voltage Output*. De asemenea, mesajele de eroare sau butonul *Stop* prezent pe panoul frontal (figura 8) pot întrerupe funcționarea buclei de citire/scriere a datelor din aplicația client.

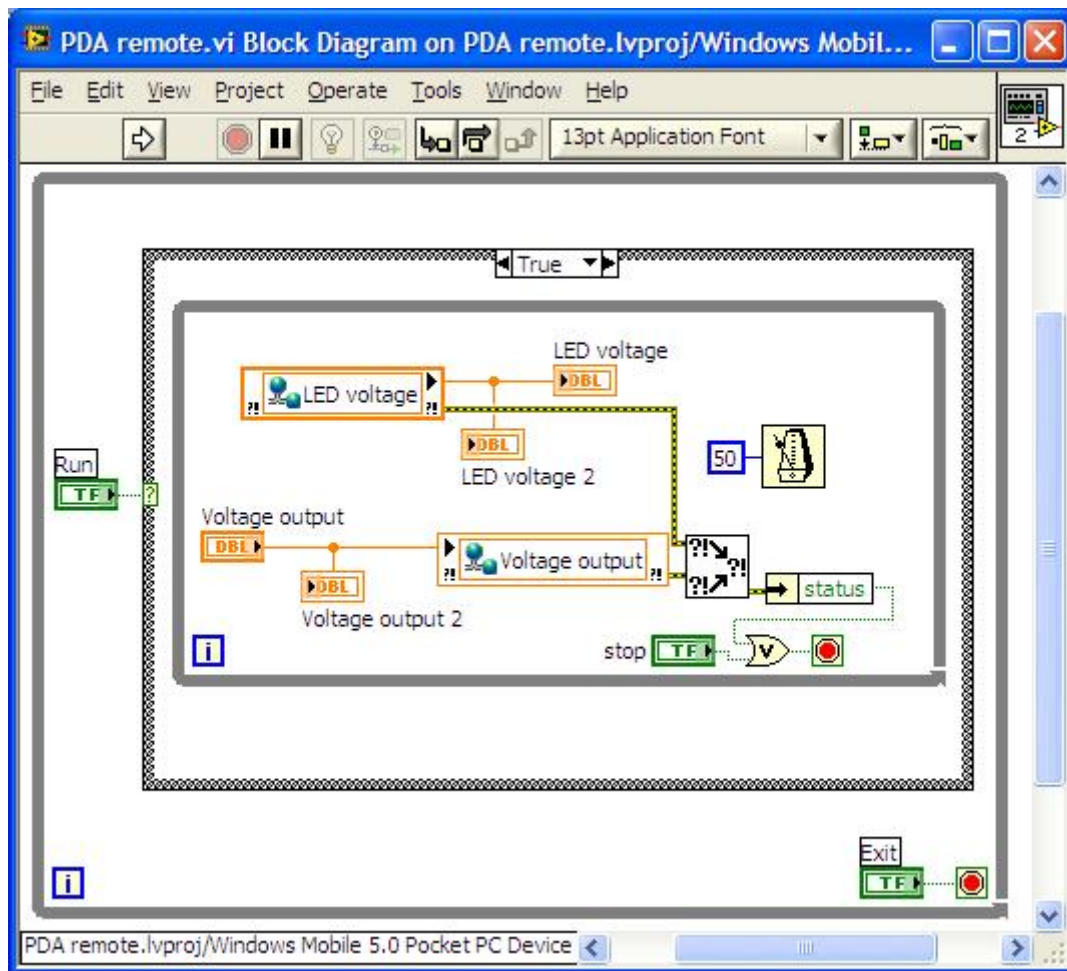


Figura 7. Diagrama bloc a aplicației client

Specific aplicațiilor pentru PDA sunt butoanele *Run* și *Exit*, care controlează pornirea și închiderea aplicației prin butoanele corespunzătoare de pe panoul de control al aplicației de pe PDA.

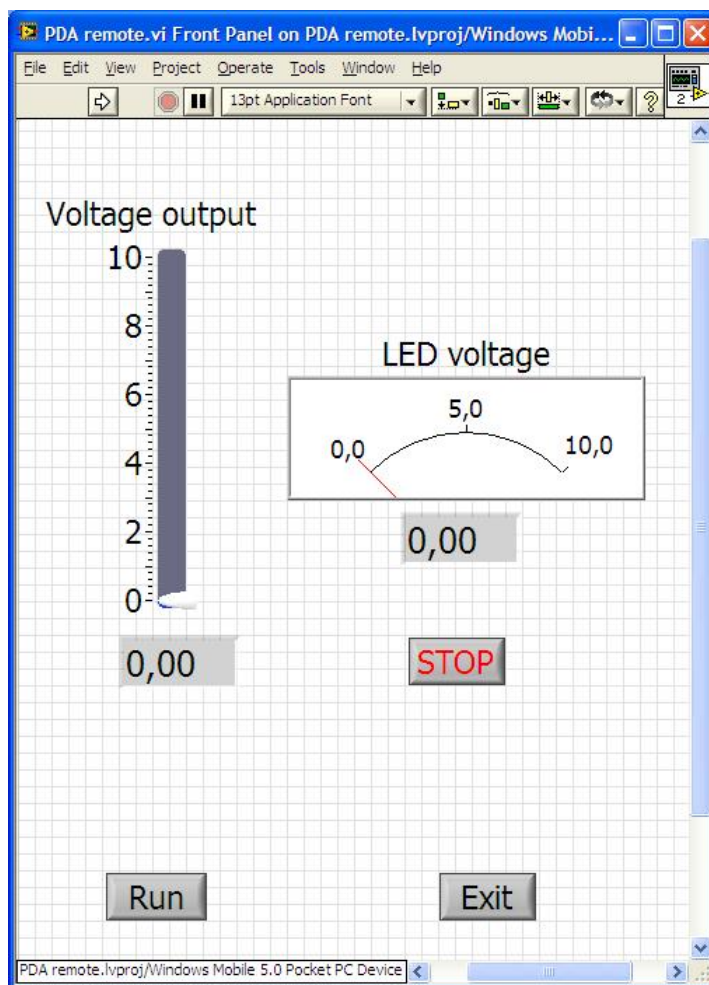


Figura 8. Panoul de control al aplicației client de pe PDA

Concluzii

Prin experimentul prezentat am exemplificat o foarte simplă aplicație de comandă/monitorizare a unei instalații de automatizare industrială în care se poate utiliza un PDA (sau un Smartphone) conectat prin rețeaua fără fir locală sau prin Internet la echipamente mai complexe, compuse din multiple instalații de automatizare controlate de calculatoare locale. Dimensiunile reduse ale PDA-urilor sau prezența permanentă a unui telefon inteligent asupra personalului de deservire justifică această abordare prin care se evită utilizarea unui laptop ca și echipament mobil de monitorizare și/sau control în cadrul procesului de producție, permițând astfel reducerea costurilor echipamentelor, dar și îmbunătățirea ergonomiei proceselor.

Aplicațiile de monitorizare/control realizate pentru PDA trebuie completate cu configurarea corespunzătoare a setărilor de securitate a rețelei, astfel încât să se respecte principiile acesteia: *Confidențialitate – Integritate – Disponibilitate*, în scopul asigurării calității comunicațiilor în rețea, incluzând elementele de autentificare-autorizare a potențialilor utilizatori în conformitate cu nivelul de acces necesar îndeplinirii atribuțiilor acestora.

Bibliografie

- [13] <http://www.opcfoundation.org>
- [14] <http://www.ni.com/labview/>
- [15] NI LabVIEW 8.6 Help - <http://zone.ni.com/reference/en-XX/help/371361E-01/>
- [16] LabVIEW 8.6 Datalogging and Supervisory Control Module Help - <http://zone.ni.com/reference/en-XX/help/371618D-01/>
- [17] <http://sine.ni.com/nips/cds/view/p/lang/en/nid/10418>
- [18] <http://www.ni.com/lookout/>
- [19] <http://www.ni.com/fieldpoint/>
- [20] <http://www.ni.com/dataacquisition/>
- [21] <http://sine.ni.com/nips/cds/view/p/lang/en/nid/1318>
- [22] <http://www.wikipedia.org>