

10.1. Tehnici de scadere a puterii in sisteme EMBEDDED

Inteligența sporită a sistemelor de comandă și control industriale bazate pe microcontrolere necesita tot mai multe resurse energetice, resurse care de foarte multe ori nu sunt disponibile sau sunt foarte scumpe. Sistemele avansate pentru o reducere considerabilă a puterii consumate sunt astfel proiectate încât să permită conectarea lor facilă la sistemele uzuale de calcul, de stocare și prelucrare a datelor. Comunicația între cele două părți poate fi locală, când acestea se află în imediata vecinătate, sau la distanță (teletransmisie) în cazul în care entitățile se află în incinte diferite sau chiar în localități diferite. Comunicația locală se poate realiza prin fir, optic sau radio, fiecare metodă prezentând avantaje și dezavantaje.

Comunicația locală se realizează în general prin fir, cele mai folosite interfețe fiind cele seriale, RS232 în cazul soluțiilor mai simple, punct la punct, sau RS485 pentru aplicații mai complexe, în care sunt necesare comunicații multipunct (în general magistrale master-slave). În ultimul timp devine tot mai populară interfața USB, care are avantajul largii răspândiri pe calculatoarele personale dar este limitată la folosirea de cabluri scurte, maxim câțiva metri lungime.

Comunicația locală pe cale optică elimină cablurile de legătură dar limitează la câțiva metri distanța între entitățile care trebuie să schimbe între ele date; un dezavantaj suplimentar este viteza redusă de comunicație, uzual sub 9600 Bauds. Printre avantaje se pot aminti posibilitatea etanșării carcasei sistemului electronic, imunitatea la perturbații și separarea electrică foarte bună între cele două sisteme participante la comunicație.

Un alt mod de comunicare locală fără fir are la bază tehnologia radio într-o bandă de frecvență alocată comunicațiilor industriale și de automatizări.

Pentru implementarea simplă și eficientă a comunicației locale între sisteme înglobate cu microcontroler și alte sisteme de calcul am elaborat un protocol simplu de comunicație serială.

Datele provenite de la nivelele superioare ale comunicației sunt divizate în cadre la nivelul legăturii de date și transport (numit în continuare DLT – DataLink and Transport). Fiecare cadru de date DLT conține un octet de start și unul de stop, doi pentru indexul cadrului, unul pentru tipul cadrului, unul pentru lungimea zonei de date utile, 2 pentru codul detector de erori tip CRC16 și până la 256 octeți de date utile. Structura cadrului de date este ilustrată în tabelul următor:

Tabelul 10.1 Structura cadrului de date la protocolul propus

Start 0x7E	Fr. Index_lo	Fr. Index_hi	Fr. Type	Frame Length	B0	B1	...	B255	crc16_lo	crc16_hi	Stop 0x7E
---------------	-----------------	-----------------	-------------	-----------------	----	----	-----	------	----------	----------	--------------

Octeții Start și Stop sunt întotdeauna 0x7E. Acest caracter nu are voie să apară în interiorul cadrului și trebuie mascat. Secvența de mascare (escape) se realizează prin înlocuirea



Structuri hardware si algoritmi specifici microsistemelor EMBEDDED

caracterului cu octetul 0x7D urmat de rezultatul operației XOR dintre caracterul de control respectiv și 0x20. Procedeul se aplică de asemenea caracterului indicator 0x7D. Secvența de mascare se aplică tuturor octeților 0x7D sau 0x7E din cadru cu excepția caracterelor de start și stop.

Indexul de cadru este un număr pe 16 biți ce indică numărul cadrelor ce urmează cadrului curent. Dacă nivelul superior dorește să trimită un bloc de date mai mare de 256 de octeți, nivelul DLT îl va împărți automat în 2 sau mai multe cadre. Primul cadru va avea indexul de valoare numărul de cadre minus unu, al doilea numărul de cadre minus 2 și tot așa până la ultimul (care are indexul 0). Indexul de cadru permite comunicații de blocuri de date fără confirmarea imediată a primirii fiecărui cadru.

Lungimea se referă la numărul de octeți de date coresunzători nivelului superior. O valoare 0 a acestui octet semnifică o lungime de 256 (valoarea maximă) de octeți a blocului de date întrucât nu se acceptă cadre cu 0 octeți în zona blocului de date.

Octetul tip de cadru indică dacă cadrul curent este de date sau cadru de confirmare sau infirmare.

Octeții de date B0 ... B255 provin de la nivelele superioare ale protocolului de comunicație.

Octeții CRC16 reprezintă Frame Check Sequence(FCS) – secvența de verificare a cadrului-și se calculează utilizând același polinom CCIT ca și cadrele protocolului serial HLDC:

$$X^{16} + X^{12} + X^5 + 1 \text{ utilizând valoarea de pornire } FCS=0xFFFF$$

După deschiderea canalului de date sistemul cu microcontroler așteaptă comenzi, comunicația fiind întotdeauna inițiată de softul ce rulează pe PC. Fiecare cadru trebuie confirmat (pozitiv sau negativ) de către cealaltă parte într-un anumit timp. Se pot implementa protocoale cu fereastră alunecătoare pentru confirmări, similare TCP, sau protocoale simple, la care nu se trece la următorul cadru de date până la sosirea confirmării recepționării corecte a cadrului curent.

Codul sursă în limbajul C pentru microcontrolere MSP430 și respectiv Delphi pentru aplicații Windows pentru calcularea acestei secvențe de verificare este prezentat în anexă și se bazează pe un algoritm rapid de calcul tabelat ([TRE01]). Protocolul serial descris mai sus a fost testat și utilizat într-o serie de aplicații dezvoltate de autor pe parcursul ultimilor ani ([DRS07], [DRS07-1], [DRV06], [DSP04], [DVV07])

Comunicația la distanță se realizează folosind rețeaua de telefonie fixă sau mobilă sau prin Internet, accesat prin tehnologii clasice (dial-up, Ethernet), wireless (radio, standard IEEE 802.11) sau de bandă largă (xDSL, cablu TV). Interesantă este conexiunea dialup, care permite realizarea accesului la internet cu resurse hardware modeste.

Sistemele actuale de comandă și control ale proceselor industriale sunt configurabile și oferă suficiente informații legate de starea proceselor controlate. Din păcate, majoritatea acestor informații sunt accesibile doar local, făcând dificilă sau chiar imposibilă integrarea acestor sisteme într-un sistem centralizat de management al informațiilor. Depășirea acestui neajuns se



UNIUNEA EUROPEANĂ

MINISTERUL MUNCII, FAMILIEI ȘI
PROTECȚIEI SOCIALE
AMFOSDRUFONDUL SOCIAL EUROPEAN
POSDRU
2007-2013INSTRUMENTE STRUCTURALE
2007-2013

Structuri hardware si algoritmi specifici microsistemelor EMBEDDED

face prin dotarea sistemului de control al procesului industrial cu capabilități de acces de la distanță. Lucrarea de față analizează opțiunile uzuale de acces la distanță – comunicații seriale prin cablu sau modemuri analogice sau GSM , comunicații TCP/IP prin rețea locală Ethernet, comunicații radio în benzile autorizate aplicațiilor de comandă și control industrial.

Structura unui sistem cu capabilități de acces la distanță este prezentată în Figura 10. 1 ([DSP04]). În partea stângă se plasează sistemul de control al procesului industrial, în partea dreaptă apar blocurile sistemului de monitorizare a acestuia, în general un PC pe care rulează aplicații software specifice.

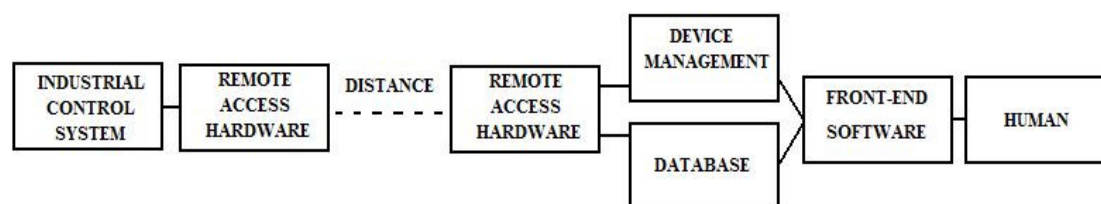


Figura 10. 1 Sistem de control cu capabilități de acces la distanță

În cazul folosirii microcontrollerelor pentru controlul proceselor industriale comunicația serială este cea mai ieftină din punct de vedere al implementării hardware și cea mai facilă din punct de vedere software, pentru că microcontrolerele moderne dispun de porturi seriale hardware integrate.

Mediul fizic la comunicațiile seriale prin cablu poate fi RS422, în cazul existenței unui singur sistem de controlat, sau RS485, în cazul în care se controlează mai multe sisteme industriale. În toate cazurile arhitectura este de tip master-slave, cu un PC acționând ca master la un capăt al cablului și unul sau mai multe sisteme slave de control industrial la celălalt capăt al cablului. Nu se poate folosi interfața RS232 datorită limitei impuse lungimii cablului (circa 10 metri).

Comunicația serială prin cablu are avantajul costului scăzut și al simplității implementării software, dar existența cablului poate constitui un dezavantaj, în primul rând datorită limitării distanței de la care se poate face accesul (de ordinul sutelor de metri).

Utilizarea mediului Ethernet este limitată doar la sistemele de comandă și control cu resurse hardware bogate – memorie suficientă, linii I/O suficiente, putere de procesare mare. Avantajul acestei soluții rezidă în standardizarea sa - ușurința accesului oferit și larga răspândire. Practic, nu este necesar soft special de comunicare cu sistemul de control, orice browser web (Internet Explorer, Netscape, Opera) putând fi folosit. În plus, accesul din Internet al unui asemenea sistem poate fi realizat simplu, utilizând tehnicile și procedurile standard de configurare a rețelelor de calculatoare.

Rețeaua locală Ethernet are mai multe avantaje: larga răspândire, posibilitatea accesului permanent la Internet, costuri de comunicație nule, dar și câteva dezavantaje – necesită hardware scump și complex, utilizarea acestei soluții fiind limitată de existența infrastructurii.

Structuri hardware si algoritmi specifici microsistemelor EMBEDDED

Prin utilizarea unor modemuri analogice cuplate la liniile de telefonie fixă se pot implementa soluții de acces la distanță relativ ieftine și ușor de implementat. Majoritatea modemurilor analogice dispun de interfețe standard RS232 și sunt configurabile prin comenzi standard AT, astfel că efortul de implementare software nu este mare.

Avantajul liniilor telefonice este costul scăzut al comunicației și ușurința în realizarea accesului la distanțe foarte mari. Dezavantajul major este indisponibilitatea liniilor telefonice libere în orice punct.

Utilizarea rețelei GSM ca mediu de comunicație este o soluție scumpă, datorită costurilor ridicate ale modemurilor GSM și ale elementelor anexe (cabluri și conectoare, antenă), precum și datorită costurilor ridicate ale comunicației în sine. În plus, acoperirea GSM nu este 100% iar calitatea semnalului nu este aceeași peste tot, astfel că trebuie efectuate teste la locul instalației deservite pentru a decide dacă soluția accesului prin rețeaua GSM este fezabilă.

Avantajele comunicației GSM față de metodele prin cablu sunt evidente – flexibilitate sporită în amplasare, lipsa cablurilor, posibilitatea implementării facile a unor funcții de alarmare sau a unor comenzi prin mesaje SMS.

Soluția GSM prezintă doua variante, identice din punct de vedere hardware, dar total diferite din punct de vedere al implementării software – transfer de date GSM standard sau prin tehnologie GPRS.

Soluția GPRS este similară din punct de vedere software soluției Ethernet – se face prin protocolul TCP/IP și oferă conexiune la Internet, cu toate avantajele și dezavantajele ce decurg de aici.

Soluția cu transfer de date standard GSM este similară accesului prin telefonie fixă, fiind preferabilă sistemelor cu resurse hardware modeste.

Arhitectura software a soluției de implementare a accesului la distanță depinde foarte mult de arhitectura hardware. În cazul în care se folosesc sisteme de calcul performante, PC-uri sau sisteme tip Single Board Computer, cu resurse de memorie și de procesare suficiente, protocoalele standard complexe pot fi utilizate – TCP/IP, GPRS; pentru sisteme bazate pe microcontrolere de 8/16 biți cu resurse limitate de memorie se pot folosi numai protocoale seriale simple.

Considerând modelul OSI al protocoalelor de comunicație cu 7 nivele, soluțiile de acces la distanță pentru un sistem de control al proceselor industriale utilizează 3 până la 5 nivele – fizic, legătură de date, eventual nivelele rețea și transport, și nivelul aplicație. Această structură asigură flexibilitate sporită prin posibilitatea reutilizării softului.

De exemplu, dacă se utilizează un modem GPRS, atunci softul trebuie să conțină funcții de lucru cu portul serial, module PPP (Point to Point Protocol), IP (Internet Protocol) și TCP (Transport Control Protocol), și module nivel aplicație; dacă același sistem trebuie să aiba opțiuni Ethernet, va avea aceleași module software pentru nivelele TCP/IP și aplicație, și numai nivelele fizic și legătură de date vor fi diferite Figura 10. 2.



UNIUNEA EUROPEANĂ

MINISTERUL MUNCII, FAMILIEI ȘI
PROTECȚIEI SOCIALE
AMFOSDRUFONDUL SOCIAL EUROPEAN
POSDRU
2007-2013INSTRUMENTE STRUCTURALE
2007-2013

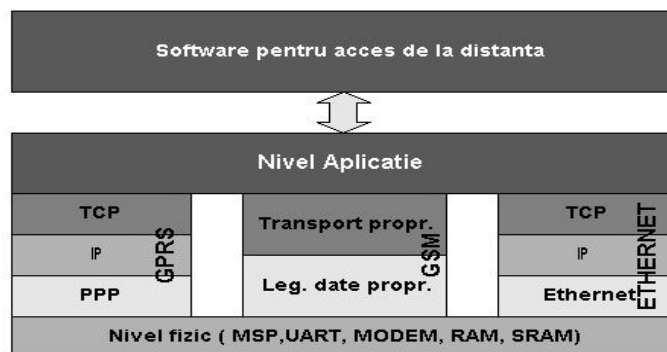
Structuri hardware si algoritmi specifici microsistemelor EMBEDDED

Figura 10. 2. Nivelele protocoalelor de comunicație pentru diferite modalități de acces

Implementarea unei stive TCP/IP este un proces complex; codul sursă este complicat și greu de manipulat, deci susceptibil la erori de programare, iar resursele hardware cerute sunt apreciabile. De aceea, soluția Ethernet sau GPRS se recomandă numai sistemelor tip Single Board Computer cu sisteme de operare ce oferă suport nativ pentru TCP/IP și mai puțin sistemelor bazate pe microcontrolere la care se recomandă utilizarea unor protocoale de comunicație proprietare la nivelul legăturii de date și nivelul transport, adaptate la resursele hardware limitate.

Protocolul PPP – Point to Point Protocol – este un mecanism pentru crearea și rularea protocolului internet IP sau a altor protocoale de rețea peste o conexiune serială (utilizând un cablu serial tip null modem), peste o conexiune telnet sau peste o conexiune realizată utilizând modemuri și linii telefoice. Cu ajutorul PPP se poate realiza conectarea unui sistem client Linux (și a altor sisteme) la un server PPP și accesa resursele rețelei în care este plasat serverul PPP ca și cum sistemul client a fi fost conectat direct în rețea. De asemenea se poate configura ușor sistemul Linux pentru rolul de server PPP, caz în care alte calculatoare se pot conecta la acesta și accesa resursele locale sau din rețeaua sa. Protocolul PPP este de tip peer-to-peer și permite interconectarea a două rețele (sau a unei rețele locale la Internet) prin realizarea unei legături PPP între 2 calculatoare plasate fiecare în una din cele 2 rețele.

O deosebire majoră între protocolul PPP bazat pe conexiune serială și protocolul Ethernet este viteza – o conexiune Ethernet standard operează la 10Mbit/s, 100Mbit/s sau 1Gbit/s viteză teoretică maximă, în timp ce un modem analog lucrează la viteze de maxim 56kbps.

Tehnic vorbind, PPP este un protocol peer-to-peer, ceea ce înseamnă că nu există nici o diferență între mașina care inițiază conexiunea și mașina care îi răspunde. Din motive de claritate, este util să se gândească în termeni de client și server: mașina care încearcă să inițieze conexiunea cu cealaltă mașină este clientul, mașina la care se conectează este serverul. Orice sistem Linux poate fi client și server PPP, uneori în același timp dacă sunt disponibile mai multe porturi seriale (eventual modemuri). Din punct de vedere al protocolului PPP, odată ce conexiunea a fost stabilită, nu există nici o diferență reală între client și server. Forma cea mai des întâlnită de utilizare a protocolului PPP este cea de client, și anume pentru a conecta un

Structuri hardware si algoritmi specifici microsistemelor EMBEDDED

sistem la Internet; orice sistem de operare modern (Windows 9x, Me, 2000, XP, Linux, Mac OS etc.) dispune de o implementare de client PPP, absolut necesară conectării prin dial-up la Internet.

Informații necesare pentru configurarea clientului PPP în vederea conectării la un server PPP:

- numărul de telefon al serverului; eventual cu prefixul pentru ton de exterior (trebuie format 0,123456 pentru a apela numărul 123456)
- adresa de IP a serverului, în cazul utilizării adreselor statice; în general se utilizează adrese IP dinamice, comunicate clientului de către server

O conexiune dialup tipică poate fi descrisă conform procesului din Figura 10. 3.

Mecanismul este foarte simplu: clientul PPP formează numărul serverului, acesta răspunde, are loc autentificarea părților, se stabilește conexiunea PPP și apoi TCP/IP (TRE01)].

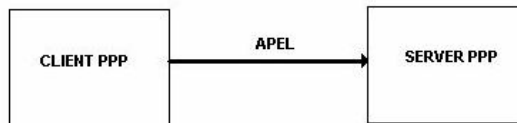


Figura 10. 3. Acces dialup standard

Apare o problemă de securitate în acest caz, oricine poate (teoretic) apela serverul PPP și accesa sistemul. Mecanismele de autentificare a conexiunii PPP, respectiv protocoalele de genul PAP sau CHAP, nu oferă o protecție suficientă contra încercărilor neautorizate de intrare în sistem. Pentru securizarea suplimentară a unei conexiuni dialup se pot folosi diverse metode.

O primă metoda ar fi recunoașterea numărului apelant și acceptarea sau rejectarea apelului în funcție de acest număr. Procedura funcționează în cazul în care modemul suportă protocolul CLIP (Caller Line Identification Protocol) iar softul este capabil să utilizeze acest protocol.

Metoda recunoașterii apelului oferă o modalitate simplă și eficientă de limitare a accesului în sistem. Apariția modemurilor GSM a complicat puțin problema accesului prin conexiuni dialup din cauza politicii operatorilor de telefonie mobilă – nu permit întotdeauna apeluri de date către cartele SIM prepaid, ci numai apeluri de voce; apelurile de date dinspre cartela prepaid spre exterior sunt permise

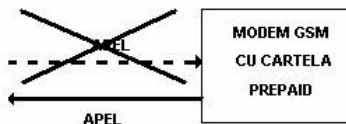


Figura 10. 4. Acces dialup nepermis spre prepaid

În aceste condiții s-ar părea că nu se pot folosi cartele GSM prepaid pentru serverul PPP, cu toate că acestea prezintă avantaje evidente (costuri mult mai reduse, în cazul pierderii/furtului cartelei se poate pierde cel mult creditul de pe aceasta). Există totuși o soluție pentru a utiliza cardurile prepaid, mai complicată, dar care oferă un plus de securitate sistemului – procedeul Callback.

Până acum am considerat că pe sistemul embedded rulează serverul PPP și pe sistemul utilizator (de obicei PC cu sistem de operare Windows) clientul PPP.

Structuri hardware si algoritmi specifici microsistemelor EMBEDDED

La mecanismul Callback (apelează înapoi) rolurile sunt inversate: sistemul utilizator este configurat ca server PPP iar sistemul embedded este clientul PPP. Pentru inițierea unei conexiuni sistemul utilizator apelează sistemul embedded; acesta, pe baza mecanismului CLIP recunoaște numărul apelant, respinge apelul și decide să sune înapoi (CallBack) sistemul utilizator al cărui număr este salvat în memoria sa nevolatilă.

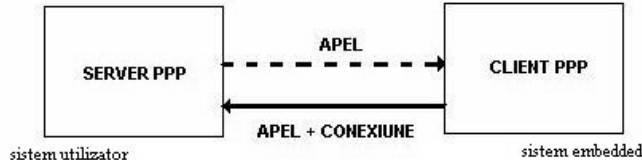


Figura 10. 5 Acces dialup tip call-back

În felul acesta se asigură o securitate sporită, pentru că sistemul embedded (clientul PPP) apelează întotdeauna un singur număr (al serverului PPP) și numai ca urmare a unui apel recunoscut de la acesta. Orice apel de la orice alt număr este rejectat fără să se mai întâmple altceva, iar mecanismul Callback este declanșat doar de apelurile de la numărul sistemului utilizator. Evident, întrucât sistemul embedded este cel care inițiază apelul de date, se pot folosi și cartele prepaid.

Mecanismul callback l-am implementat și testat cu succes în aplicații de acces la distanță prin comunicație GSM/GPRS ([DPS04], [DPS05], [DSP04]), în acest sens elaborând o bibliotecă de funcții și rutine în limbajul C și de asamblare pentru microcontrolere MSP430.

În cadrul activităților de cercetare-dezvoltare efectuate în timpul stagiului la firma Siemens Building Technologies din Germania am conceput, realizat și testat o implementare a protocoalelor GPRS pe microcontrolere de 16 biți Texas Instruments MSP430 pentru a oferi acces la distanță ([DSP04], [DPS04]) prin modem GSM/GPRS pentru un concentrator de date([QVE08]). Accesul la distanță prin GPRS presupune însă implementarea protocoalelor PPP, TCP/IP și eventual HTTP, FTP sau SMTP (trimitere de email).

Greutatea problemei rezidă în complexitatea acestor protocoale corelată cu resursele limitate de memorie ale microcontrolerului folosit (2Kbytes RAM pentru modelul ales MSP430F149 [TEX06]).

Funcțiile realizate în limbajul C asigură implementarea protocoalelor de inițializare și configurare modem, gestiunea comenzilor AT, formarea numerelor și conectarea la un server de date (modem.c, mc35.c), urmată de stabilirea protocolului PPP (ppp.c) și în final a celui TCP/IP (tcpip.c).

Fiecare protocol de nivel inferior adaugă propriul header de informație la datele provenite de la nivelele superioare (Figura 10. 6)

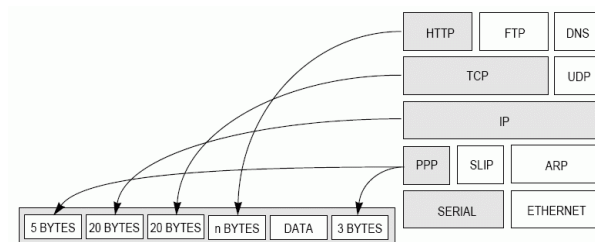
Structuri hardware si algoritmi specifici microsistemelor EMBEDDED

Figura 10. 6. Iferite protocoale standard pentru comunicația de date

Protocolul PPP se stabilește în 3 etape. Mai întâi are loc configurarea legăturii, prin Link Control Protocol, urmată de autentificarea părților, care se poate realiza conform unui anumit protocol (PAP – Password Authentication Protocol, CHAP – Challenge Handshaking Authentication Protocol etc.) și, în final, se schimbă datele de rețea, prin care părțile își comunică adresele IP proprii. Toate aceste subprotocoale se realizează prin trimiteri succesive de cadre de interogare și cadre de confirmare și răspuns, astfel că la sfârșit se poate trece la protocolul TCPIP

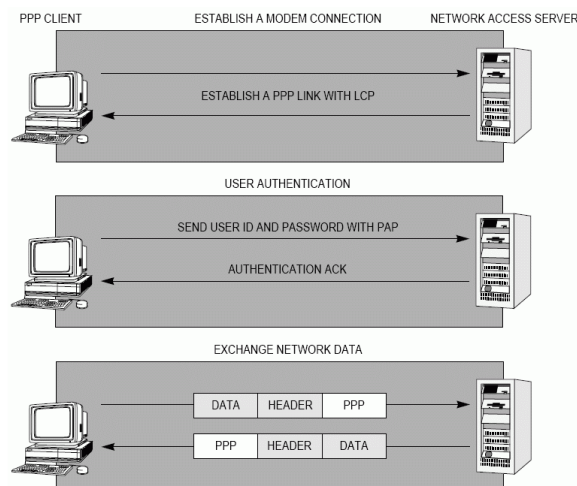


Figura 10. 7. Etapele realizării protocolului PPP

Protocolul TCPIP are o structură deosebit de complexă ce pune probleme deosebite implementărilor pe sisteme cu resurse limitate de memorie. Pentru implementarea efectivă s-a ales soluția easyWeb ([DAN04]) care realizează protocolul TCPIP pe sisteme cu MSP430F149 și controler Ethernet CS8900. S-a eliminat partea de Ethernet și s-a modificat codul TCPIP pentru a-l integra cu protocolul PPP. Procesarea unui cadru TCP se face conform diagramei alăturate.

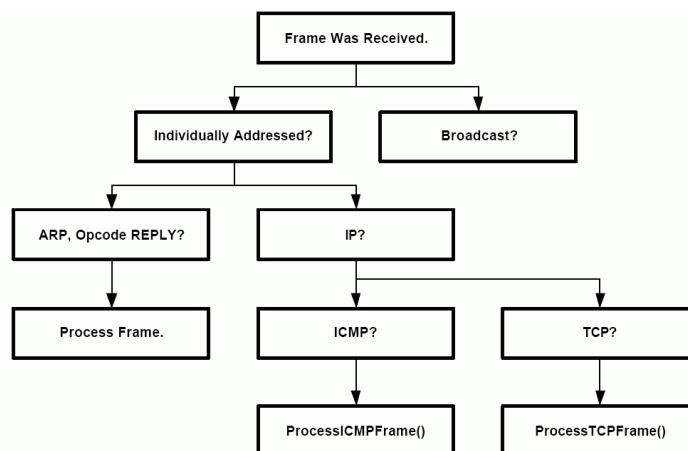


Figura 10. 8. Procesarea cadrelor TCP

Un siste embedded trebuie sa functioneze si dupa o cadere temporara a tensiunii retelei. Pentru aceasta este necesar ca datele sa fie in permanenta in stare consistenta, deci sistemul de fisiere utilizat de sistemul de operare trebuie sa asigure acest lucru.

De aceea, la proiectarea unui sistem embedded un aspect important este reprezentat de alegerea sistemului de fisiere ce va fi utilizat. Sistemul de operare Linux ofera mai multe sisteme de fisiere dar numai unele pot fi utilizate eficient in cadrul unui sistem embedded.

Criteriile care influenteaza alegerea sint:

- Aplicatia are nevoie sa scrie in sistemul de fisiere?
- Performanta (viteza) este cel mai important aspect al sistemului, dau se poate accepta o performanta ceva mai scazuta in schimbul unui cod mai redus sau al utilizarii mai eficiente a dispozitivului de memorare ?

Sisteme de fisiere orientate pe disc

Desi numai o minoritate restrinsa a sistemelor embedded dispun de un hardisk, sistemele de fisiere prezentate in continuare sint cele familiare din lumea sistemelor de calcul obisnuite (PC) si reprezinta un bun element de comparatie.

- *ext2* este sistemul de fisiere traditional in sistemele desktop si server Linux. Este rapid, prezinta multe facilitati si utilizeaza eficient spatiul pe disc, dar necesita o cantitate destul de mare de cod. Prezinta insa un mare dezavantaj pentru un sistem embedded – daca are loc o cadere de tensiune inainte ca sistemul de fisiere *ext2* sa fi fost demontat integritatea datelor de pe disc poate fi afectata (disk corruption) astfel ca la repornirea sistemului trebuie neaparat efectuata verificarea sitemului de fisiere, operatie mare consumatoare de timp in general.
- Sistemele de fisiere jurnalizate sint tot mai utilizate pe versiunile recente de Linux. Acestea prezinta o serie de avantaje, in particular viteza ridicata de transfer si evitarea necesitatii efectuarii verificarii consistentei datelor la pornirea sistemului. Cele mai utilizate sisteme de fisiere jurnalizate pentru Linux sint *ext3* si *Reiser FS*. De remarcat este insa ca aceste sisteme de fisiere adauga chiar mai mult cod imaginii nucleului Linux decit sistemul de fisiere *ext2*.

Structuri hardware si algoritmi specifici microsistemelor EMBEDDED

- Cu toate ca nu este foarte utilizat, sistemul de fisiere *minix* prezinta citeva avantaje: utilizeaza eficient spatiul pe disc, cu pretul eliminarii citorva facilitati utilizate mai rar (anumite atribute pentru fisiere si directoare) si ocupa un spatiu destul de redus in imaginea nucleului. Necesita insa verificarea consistentei datelor la pornirea sistemului.

Bibliografie:

1. Istvan Sztojanov, Sever Pașca, Elisabeta Buzoianu, Aplicații hardware și software cu microcontrolerul PIC12F675, Editura Cavallioti, ISBN 978-973-7622-54-9, Bucuresti 2008
2. Istvan Sztojanov, Alexandru Vasile, Elisabeta Buzoianu, Sever Pașca, *Programarea microcontrolerelor din familia Intel, Aplicații practice hardware cu 80C552*, Editura Man-Dely, ISBN 973-85681-5-3, București 2004.
3. <http://vega.unitbv.ro/~romanca/EmbSys/>
4. <http://facultate.regielive.ro/cursuri/electronica/>
5. www.microcip.com
6. Andrei Drumea, Teza de doctorat, UPB 2009
7. BARUCH Zoltan, Translatarea limbajelor de descriere a unităților hardware, Referat de doctorat Cluj Napoca 2008



UNIUNEA EUROPEANĂ



MINISTERUL MUNCII, FAMILIEI ȘI
PROTECȚIEI SOCIALE
AMFOSDRU



FONDUL SOCIAL EUROPEAN
POSDRU
2007-2013



INSTRUMENTE STRUCTURALE
2007-2013