

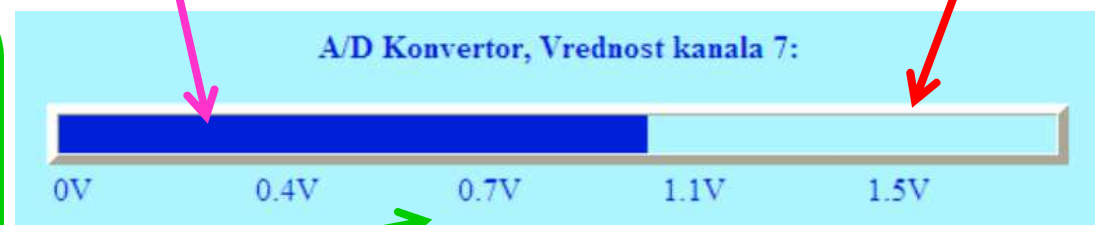
HTML kod i CVI aplikacija

Prikaz napona sa kanala 7 A/D Konvertora

`<p>A/D Konvertor, Vrednost kanala 7:</p>`

```
<table bgcolor=#ADF4FF border=5 cellpadding=0 cellspacing=0 width=500>  
<tr>  
<td>  
<table width=AD7% border=0 cellpadding=0 cellspacing=0>  
<tr><td bgcolor=#0120DD> &nbsp; </td></tr>  
</table>  
</td>  
</tr>  
</table>
```

```
<table border=0 width=500>  
<tr>  
<td width=20%>0V</td>  
<td width=20%>0.4V</td>  
<td width=20%>0.7V</td>  
<td width=20%>1.1V</td>  
<td width=20%>1.5V</td>  
</tr>  
</table>
```



Prikaz temperature sa kanala 10 A/D Konvertora

```
<p><b>Temperatura integrisanog senzora:</b></p>
```

```
<table bgcolor=#ADF4FF border=5 cellpadding=0 cellspacing=0 width=500>
```

```
<tr>
```

40% => 20°C, puna skala 50°C

```
<td>
```

```
<table width=ADA% border=0 cellpadding=0 cellspacing=0>
```

```
<tr><td bgcolor=#0120DD>&nbsp;</td></tr>
```

```
</table>
```

```
</td>
```

```
</tr>
```

```
</table>
```

```
<table border=0 width=500>
```

```
<tr>
```

```
<td width=10%>0°C</td>
```

```
<td width=10%>5°C</td>
```

```
<td width=10%>10°C</td>
```

```
<td width=10%>15°C</td>
```

```
<td width=10%>20°C</td>
```

```
<td width=10%>25°C</td>
```

```
<td width=10%>30°C</td>
```

```
<td width=10%>35°C</td>
```

```
<td width=10%>40°C</td>
```

```
<td width=10%>45°C</td>
```

```
</tr>
```

```
</table>
```



```
<html>
  <head>
    <meta http-equiv=\refresh\ content=\5\>
    <title>Ocitanje senzora preko EasyWEB2 ploce</title>
  </head>

  <body bgcolor=\#ADF4FF\ text=\#0120DD\>
    <center>

      <!tempAD10=ADA%>
      <p><b><font color="\#0120DD" size="6">Ovo je dinamicka web strana koja se <br>
      uploaduje sa web-servera easyWEB</font></b></p>
      <p><b>Hardver:</b></p>
      <ul>
        <li><b>MSP430F149, 8 MHz, 60KB Flash, 2KB SRAM</b></li>
        <li><b>CS8900A Crystal Ethernet Controller</b></li>
      </ul>

      <p><b>A/D Konvertor, Vrednost kanala 7:</b></p>

      <table bgcolor=\#ADF4FF\ border=\5\ cellpadding=\0\ cellspacing=\0\ width=\500\>
        <tr>
          <td>
            <table width=\AD7%\ border=\0\ cellpadding=\0\ cellspacing=\0\>
              <tr><td bgcolor=\#0120DD\>&nbsp;&nbsp;&nbsp;</td></tr>
            </table>
          </td>
        </tr>
      </table>
```

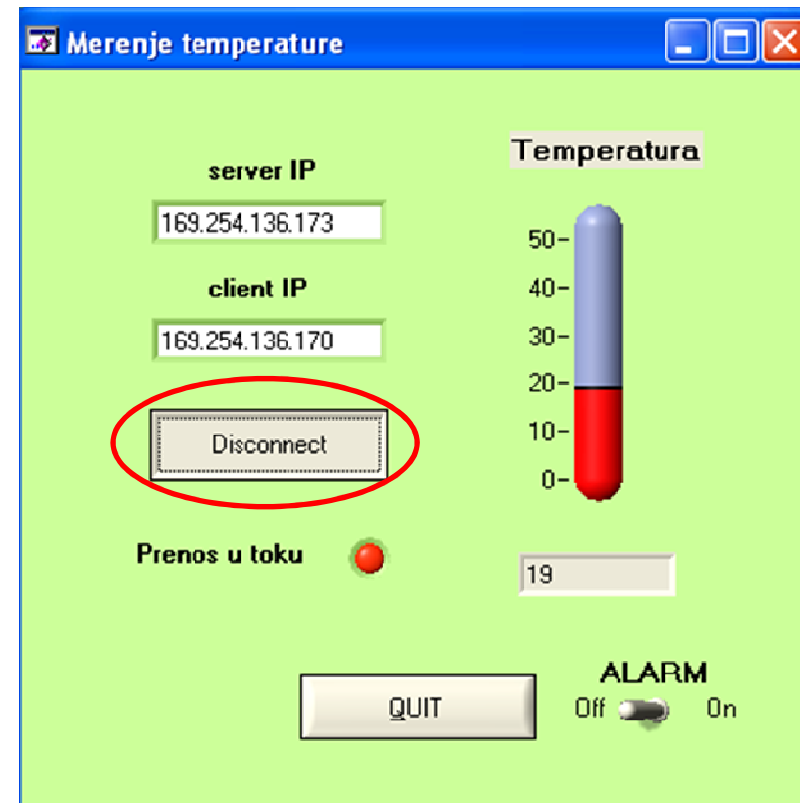
CVI aplikacija – *main* funkcija

```
26 int main (int argc, char *argv[])
27 {
28     if (InitCVIRTE (0, argv, 0) == 0)
29         return -1; /* out of memory */
30     if ((panelHandle = LoadPanel (0, "TCPIPtemp.uir", PANEL)) < 0)
31         return -1;
32
33     PromptPopup ("Server Name?",
34                 "Unesite IP adresu servera "
35                 "\n\n(primer: abc.xyz.com or xxx.xxx.xxx.xxx) ",
36                 ip_adress_str, 24);
37     SetCtrlVal (panelHandle, PANEL_STRING, ip_adress_str);
38     GetTCPHostAddr (ip_client_adress_str, 25);
39     SetCtrlVal (panelHandle, PANEL_STRING_2, ip_client_adress_str);
40     DisplayPanel (panelHandle);
41     RunUserInterface ();
42     DiscardPanel (panelHandle);
43     return 0;
44 }
```



CVI aplikacija – *ButtonPush* funkcija

```
61 int CVICALLBACK ButtonPush (int panel, int control, int event,  
62     void *callbackData, int eventData1, int eventData2)  
63 {  
64     switch (event)  
65     {  
66     case EVENT_COMMIT:  
67         GetCtrlVal (panelHandle, PANEL_TOGGLEBUTTON, &pritisnuto);  
68  
69         break;  
70     case EVENT_VAL_CHANGED:  
71  
72         break;  
73     case EVENT_RIGHT_CLICK:  
74  
75         break;  
76     }  
77     return 0;  
78 }
```



CVI aplikacija – *TimerTick* funkcija

```
80 int CVICALLBACK TimerTick (int panel, int control, int event,  
81     void *callbackData, int eventData1, int eventData2)  
82 {  
83     switch (event)  
84     {  
85         case EVENT_TIMER_TICK:  
86             if (pritisnuto == 1) {  
87                 //konektovanje  
88                 GetTCPHostAddr (temp, 100);  
89                 //MessagePopup ("Moj IP", temp);  
90  
91                 status_veze = ConnectToTCPServer (&conv_handle, 80, ip_adress_str, TCP_Callback, NULL, 1000);  
92                 if (status_veze < 0) {  
93                     //obrada greske  
94                     tcp_greska = GetTCPSystemErrorString ();  
95                     MessagePopup ("Greska", tcp_greska);  
96                 }  
97                 else {  
98                     //konektovani smo  
99                     //int temp;  
100                    SetCtrlVal (panelHandle, PANEL_LED, 1);  
101  
102                    GetCtrlVal (panelHandle, PANEL_BINARYSWITCH, &alarm_state);  
103                    if (alarm_state==1) strcpy(getmetod,"1");  
104                    else strcpy(getmetod,"0");  
105  
106                    //sprintf(getmetod,"x"); // prenos stranice se zapocinje slanjem bilo cega,  
107                    //ne mora GET zahtev TCP /index.html HTTP/1.0  
108                    ClientTCPWrite (conv_handle, getmetod, strlen(getmetod), 1000);  
109                    // temp += 100;  
110                }  
111            }  
112            break;  
113        }  
114        return 0;  
}
```



CVI aplikacija – *TCP_Callback* funkcija

```
134 switch (event)
135 {
136     case TCP_DATAREADY:
137         bytes_read = ClientTCPRead (conv_handle, data, MAXBYTES-1, 0);
138         if (bytes_read <= 0) {
139             //jbg
140             goto Down;
141         }
142         else {
143             //cool
144             data[bytes_read]='\0';
145             // MessagePopup ("Info", data);
146         }
147
148         temp_string=strstr(data,text_to_find);
149         if (temp_string == NULL) break;
150         temp_string=&temp_string[strlen(text_to_find)];
151         for(i=0;i<4;i++)
152             {
153                 {
154                     if (temp_string[i]=='%') break;
155                 }
156             }
157         strcat(temp_value,temp_string,i);
158         int_temp_val=atoi(temp_value);
159         int_temp_val=int_temp_val/2;
160         // MessagePopup ("vrednost temperature", temp_value);
161         SetCtrlVal (panelHandle, PANEL_NUMERICTHERM, int_temp_val);
162
```

```
char text_to_find[10]="tempAD10=";
char *temp_string=NULL;
char temp_value[10]="";
int int_temp_val;
int i;

char* data = NULL;
int MAXBYTES = 15000;
int bytes_read;
data = malloc(MAXBYTES*sizeof(char));
if (data==NULL) return -1;
```

