

File "/SumpPumpMonitor-WIZwiki-W7500/main_SPM.cpp" printed from mbed.org on Sunday, December 18, 2016

```

1  /*
2  =====
3  Project:      Sump Pump Monitor
4  Description:  Measure pump current to indicate run status. (10amp pump)
5  Date:        11-6-16
6  Author:      Jim Abraham
7  Board:       WIZwiki-W7500
8  Rev:
9  =====
10 */
11 #include "mbed.h"
12 #include "EthernetInterface.h"
13 #include <stdio.h>
14 #include <string.h>
15
16 #define USE_DHCP    0 // 0= no dhcp
17 #define MAC        "\x00\x08\xDC\x11\x34\x78"
18 #define IP         "192.168.0.12"
19 #define MASK       "255.255.255.0"
20 #define GATEWAY    "192.168.0.254"
21 #define HTTPD_SERVER_PORT    80
22 #define HTTPD_MAX_REQ_LENGTH  1023
23 #define HTTPD_MAX_HDR_LENGTH  255
24 #define HTTPD_MAX_FNAME_LENGTH  127
25 #define HTTPD_MAX_DNAME_LENGTH  127
26
27 #if defined(TARGET_WIZwiki_W7500)
28 Serial uart(USBTX, USBRX);
29 #include "static_colors.h"
30 // LED R      : server listing status
31 // LED GREEN  : socket connecting status Ok
32 // LED BLUE   : socket connecting status Busy
33 #endif
34
35 EthernetInterface eth;
36 TCPSocketServer server;
37 TCPSocketConnection client;
38
39 //=====
40 // Data Variables.
41 //=====
42 int LP1,LP2,LP3;
43 float AMP_Reading_on_AO, AMPreadingHI, AMPreading;
44 char buffer[HTTPD_MAX_REQ_LENGTH+1];
45 char httpHeader[HTTPD_MAX_HDR_LENGTH+1];
46
47 //-----
48 // Initialize a pins to perform analog input
49 //-----
50 AnalogIn ain0(A0);
51
52 //-----
53 void Serial_Interface_init(void)
54 //-----
55 {
56     // Serial Interface eth;
57     // Serial port configuration:
58

```

```

59     // 9600 baud, 8-bit data, no parity, stop bit
60     uart.baud(9600);
61     uart.format(8, SerialBase::None, 1);
62     uart.printf("Initializing\n");
63     wait(1.0);
64 }
65
66 //-----
67 void ethernet_init(void)
68 //-----
69 {
70     // EthernetInterface eth;
71     uart.printf("Initializing Ethernet\n");
72     #if USE_DHCP
73     //eth.init Use DHCP
74     // Use DHCP for WIZnetInterface
75     int ret = eth.init((uint8_t*)MAC);
76     uart.printf("Connecting DHCP\n");
77     #else
78     // IP,mask,Gateway
79     int ret = eth.init((uint8_t*)MAC, IP, MASK, GATEWAY);
80     uart.printf("Connecting (IP,mask,Gateway)\n");
81     #endif
82     wait(1.0);
83     // Check Ethernet Link-Done
84     uart.printf("Check Ethernet Link\r\n");
85
86     if(eth.link() == true)
87     {
88         uart.printf("- Ethernet PHY Link - Done\r\n");
89         COLOR(_RED_);
90     }
91     else
92     {
93         uart.printf("- Ethernet PHY Link - Fail\r\n");
94         COLOR(_BLACK_);
95     }
96     wait(1.0);
97     if(!ret)
98     {
99         uart.printf("Initialized, MAC: %s\r\n", eth.getMACAddress());
100        ret = eth.connect();
101
102        if(!ret)
103        {
104            uart.printf("IP: %s, MASK: %s, GW: %s\r\n",
105                eth.getIPAddress(), eth.getNetworkMask(), eth.getGateway());
106            COLOR(_CYAN_);
107        }
108        else
109        {
110            uart.printf("Error ethernet.connect() - ret = %d\r\n", ret);
111            COLOR(_YELLOW_);
112            exit(0);
113        }
114    }
115    else
116    {
117        uart.printf("Error ethernet.init() - ret = %d\r\n", ret);
118        COLOR(_BLACK_);

```

```

119     exit(0);
120 }
121
122     wait(1.0);
123     // TCPsocketServer server;
124     server.bind(HTTPD_SERVER_PORT);
125     server.listen();
126     uart.printf("Server Listening\n");
127 }
128
129 //-----
130 void VAC_Zero_Cross(void)
131 //-----
132 {
133     LP1=0;           // Indicator to exit while loop
134     if (LP2==1)
135     {
136         //Waits for the waveform to be close to 'zero' (500 adc)
137         while(LP1==0)
138         {
139             wait_us(100);
140             AMPreading=ain0.read();
141
142             //check its within range. (+/- 5)
143             if ((AMPreading < 0.5) && (AMPreading > 0.4))
144             {
145                 LP1=1;
146             }
147         }
148     }
149     LP2=1;
150 }
151
152 //-----
153 void Measure_AMPS(void)
154 //-----
155 // Measure AC Current.
156 // 12bit ADC, 10MHz = 0.1usec conversion time.
157 // 12bit ADC (0-3.3v = 0-4095 values) 3.3/4096 = 0.81mv
158 // 1.65v = 2047 = 0 amps.
159 //-----
160 {
161     AMPreadingHI=0.0;
162     for (LP3=0; LP3<100; LP3++)
163     {
164         wait_us(200);
165         AMPreading=ain0.read();
166         if (AMPreadingHI < AMPreading) AMPreadingHI = AMPreading;
167     }
168
169     if (AMPreadingHI <0.51)
170     {
171         COLOR(_BLUE_);
172     }
173     else
174     {
175         COLOR(_GREEN_);
176     }
177
178     AMP_Reading_on_AO = AMPreadingHI;

```

```

179     uart.printf("$%2.2f!", AMPreading);
180     sprintf(httpHeader, "$%2.2f!", AMP_Reading_on_A0);
181     client.send(httpHeader, strlen(httpHeader)); //send amps to PC.
182 }
183
184
185 //=====
186 int main(void)
187 //=====
188 {
189     //RGB LED:
190     //WHITE = program running.
191     //RED   = not connected to PC.
192     //GRN   = pump running.
193     //BLUE  = pump not running.
194
195     Serial_Interface_init();
196     ethernet_init();
197     LP1=0; // Indicator to exit while loop
198     LP2=0;
199 //-----
200     while(true)
201 //-----
202     {
203         uart.printf("\nWait for new connection...\r\n");
204         server.accept(client);
205         client.set_blocking(false, 1500); // Timeout after (1.5)s
206         uart.printf("Connection from: %s\r\n", client.get_address());
207
208 //-----
209         while(true)
210 //-----
211         {
212             if(!client.is_connected())
213             {
214                 COLOR(_RED_);
215                 break; //exit while
216             }
217             // Mesure ADC 0 - Check Sump Pump Run Status
218             VAC_Zero_Cross();
219             Measure_AMPS();
220             wait(1.0);
221             COLOR(_WHITE_);
222         }
223 //-----
224         client.close(); //close connection to pc app.
225         ethernet_init();
226     }
227 }

```

File "/SumpPumpMonitor-WIZwiki-W7500/main_SPM.cpp" printed from mbed.org on Sunday, December 18, 2016