

```
1
2
3 srv=net.createServer(net.TCP)
4 srv:listen(80,function(conn)
5     conn:on("receive", function(client,request)
6         local buf = "";
7         local head = "HTTP/1.1 200 OK\n\n!DOCTYPE HTML>\n<html>\n<head></head>\n<body>\n";
8         local _, _, method, path, vars = string.find(request, "([A-Z]+) (.+)?(.+) HTTP");
9         if(method == nil)then
10             _, _, method, path = string.find(request, "([A-Z]+) (.+) HTTP");
11         end
12         local _GET = {}
13         if (vars ~= nil)then
14             for k, v in string.gmatch(vars, "(%w+)=(%w+)&*" ) do
15                 _GET[k] = v
16             end
17         end
18         buf = head;
19         buf = buf.."<h1> LOControl";
20         buf = buf.."<p>Motion <a href=\"?pin=START\"><button>START</button></a>&nbsp;<a href=\"?pin=STOP\"><button>STOP</button></a></p>";
21         buf = buf.."<p>Direction <a href=\"?pin=REW\"><button>REW</button></a>&nbsp;<a href=\"?pin=HLT\"><button>HALT!</button></a></p>";
22         buf = buf.."<p>Speed <a href=\"?pin=SLOW\"><button>-</button></a>&nbsp;<a href=\"?pin=FAST\"><button>+</button></a></p>"
23
24         if(_GET.pin == "START")then
25             start_handler();
26
27         elseif(_GET.pin == "STOP")then
28             stop_handler();
29
30         elseif(_GET.pin == "REW")then
31
32             if (direction == 0) then
33                 dreverse ();
34             elseif (direction == 1) then
35                 dforward ();
36             end
37
38         elseif(_GET.pin == "HLT")then
39             halt ();
40         elseif(_GET.pin == "SLOW")then
41             slow_maker ();
42         elseif(_GET.pin == "FAST")then
```

```

43         fast_maker ();
44
45     end
46
47     local levelp = 0;
48     local directionp = "";
49
50     if (level > 0) then
51         levelp = level/100
52
53     end
54
55     if (direction == 0) then
56         directionp = "F"
57     elseif (direction == 1) then
58         directionp = "R"
59     end
60
61     buf = buf.."<tr at=\"le\">D: "..directionp.." L: "..levelp.."</tr>";
62
63     buf = buf.."<form src=\"/\"><br><button type=\"submit\" name=\"set\" value=\"RST\">Reset WiFi</button></form>\n";
64     buf=buf.."</body></html>";
65     client:send(buf);
66     client:close();
67     collectgarbage();
68
69     if(_GET.set == "RST")then
70         file.remove("wificnf");
71         node.restart();
72     end
73 end)
74 end)
75
76 -- functions -----
77
78 function start_handler ()
79
80     if (level == 0)then
81
82         level=200
83

```

```
84     if (direction == 0) then
85
86
87         pwm.stop (ib)
88         delay_ms (1000)
89         pwm.start (ia)
90         direction=0
91
92         gpio.write (led2, gpio.LOW);
93         gpio.write (led1, gpio.HIGH);
94
95
96
97     elseif (direction == 1) then
98
99
100         pwm.stop (ia)
101         delay_ms (1000)
102         pwm.start (ib)
103         direction=1
104
105         gpio.write (led1, gpio.LOW);
106         gpio.write (led2, gpio.HIGH);
107
108     end
109
110     deltaspeed_handler_up ()
111
112 end
113
114 end
115
116 function stop_handler ()
117
118     level=0
119     deltaspeed_handler_down ()
120     gpio.write (led2, gpio.LOW);
121     gpio.write (led1, gpio.LOW);
122
123 end
124
```

```

125 function fast_maker ()
126
127     if (level<800) then
128         level=level+100
129         deltaspeed_handler_up ()
130     end
131
132 end
133
134 function slow_maker ()
135
136     if (level>=100) then
137         level=level-100
138         deltaspeed_handler_down ()
139     end
140
141 end
142
143 function deltaspeed_handler_up ()
144
145     local dutyval = 0
146     local speednow = level
147     local direction1 = ia
148
149     if (direction == 1)
150     then direction1=ib;
151     end
152
153     dutyval = pwm.getduty(direction1)
154
155
156     if (speednow > dutyval) then
157
158         while (speednow > dutyval) do
159
160             dutyval = dutyval + 100
161
162             pwm.setduty(direction1, dutyval)
163
164             delay_ms (500)
165

```

```

167         end
168     end
169
170 function deltaspeed_handler_down ()
171
172     local dutyval = 0
173     local speednow = level
174     local directionl = ia
175
176     if (direction == 1) then
177         directionl=ib;
178     end
179
180     dutyval = pwm.getduty(directionl)
181
182
183     if (speednow < dutyval) then
184
185         while (speednow < dutyval) do
186
187             dutyval = dutyval - 100
188
189             pwm.setduty(directionl, dutyval)
190
191             delay_ms (500)
192
193         end
194     end
195 end
196
197 function dforward ()
198
199     if (direction == 1)then
200
201         local levelmemo = level
202
203         stop_handler ()
204
205         pwm.stop (ib)
206         delay_ms (1000)
207         pwm.start (ia)

```

```
209 direction=0
210 gpio.write (led2, gpio.LOW);
211 gpio.write (led1, gpio.HIGH);
212
213 level=levelmemo
214 deltaspeed_handler_up ()
215
216 end
217 end
218
219 function dreverse ()
220
221     if (direction == 0)then
222
223         local levelmemo = level
224
225         stop_handler ()
226
227         pwm.stop (ia)
228         delay_ms (1000)
229         pwm.start (ib)
230         direction=1
231
232         gpio.write (led1, gpio.LOW);
233         gpio.write (led2, gpio.HIGH);
234
235         level=levelmemo
236         deltaspeed_handler_up ()
237
238     end
239 end
240
241 function halt ()
242
243     level=0
244     pwm.setduty(ia, 0)
245     pwm.setduty(ib, 0)
246
247 end
248
249 function delay_ms (milli_secs)
```

```
250 local ms = milli_secs * 1000
251 local timestart = tmr.now ()
252
253 while (tmr.now () - timestart < ms) do
254     tmr.wdclr ()
255 end
256 end
```