

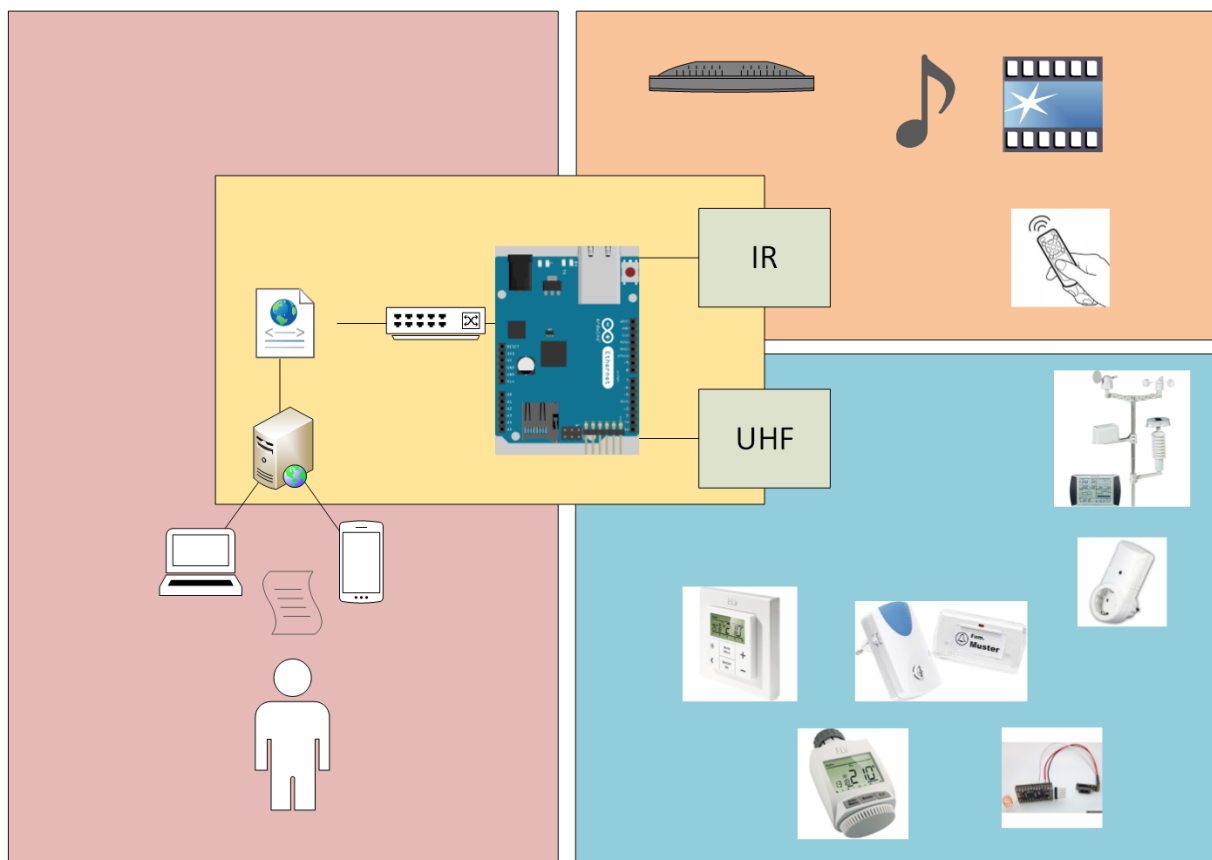
Arduino Projekt

David Arenz & Matthias Lehmann

Ziel des Projektes:

Schaffung einer universellen Plattform zur Hausautomatisierung

1. Einbinden verschiedener Steuerungssystemen aus dem Konsumerbereich
 - Verwirklichung eines Universeller Infrarot Sender/Empfängers
 - TV, Hifi, Lampen usw.
 - Beliebige Fernbedienungen als Bedienelemente
 - Einbindung von UHF Funktransceivern (433/866 MHz)
 - Funksteckdosen, Funkdimmer
 - Empfang von Wetterdaten
2. Ansteuerung verschiedener Systeme bündeln
 - API ähnliche Befehle
 - Abarbeiten von Befehlsketten
 - Ggf. Überwachung und Regelung von Parametern
3. HMI Schnittstelle per Webserver



1 Aufgabe 1

Listing 1: ../code/InfraredProxy.ino

```
1 #include <SPI.h>
2 #include <Ethernet.h>
3 #include <TextFinder.h>
4 #include <SD.h>
5
6 byte mac[] = { 0x90, 0xA2, 0xDA, 0x0E, 0xDB, 0xAE }; // MAC-Adresse des Ethernet-Shield
7 //byte ip[] = { 192, 168, 2, 102 }; // IP zum aufrufen des
   // Webservers
8 byte sdPin = 4; // Pin der SD-Karte
9
10 EthernetServer server(80); // Server port
11
12 File webFile;
13
14 void setup()
15 {
16     // Open serial communications and wait for port to open:
17     Serial.begin(115200) ;
18     // start the Ethernet connection and the server:
19     Ethernet.begin(mac) ;
20     Serial.print("server is at ") ;
21     Serial.println(Ethernet.localIP()) ;
22     Serial.println("ARDUINO - STEUERUNG");
23
24     Serial.println("Initialisiere SD-Karte...");
25     if (!SD.begin(sdPin))
26     {
27         Serial.println(" - Initialisierung der SD-Karte fehlgeschlagen!");
28         return;
29     }
30     Serial.println(" - SD-Karte erfolgreich initialisiert.");
31
32     if (!SD.exists("index.htm"))
33     {
34         Serial.println(" - Datei (index.htm) wurde nicht gefunden!");
35         return;
36     }
37     Serial.println(" - Datei (index.htm) wurde gefunden.");
38
39     Serial.println();
40     Serial.println("Verbraucher schalten");
41 }
42
43 void loop()
44 {
45     EthernetClient client = server.available(); // Auf Anfrage warten
46
47     if(client)
48     {
49         /*
50          * Ausg nge  ber das Webformular steuern *
51          */
52         TextFinder finder(client);
53
54         if(finder.find("GET"))
55         {
56             while(finder.findUntil("pin", "\n\r"))
```

```

57     {
58         char typ = client.read();
59         int pin = finder.getValue();
60         int val = finder.getValue();
61
62         if(typ == 'D')
63         {
64             pinMode(pin, OUTPUT);
65             digitalWrite(pin, val);
66             Serial.print(" - D"+String(pin));
67         }
68         else if(typ == 'A')
69         {
70             analogWrite(pin, val);
71             Serial.print(" - A"+String(pin));
72         }
73         else Serial.print(" - Falscher Typ");
74
75         if(val==1) Serial.println(" ein"); else Serial.println(" aus");
76     }
77 }
78
79 /*****
80  Webformular anzeigen *
81  *****/
82 boolean current_line_is_blank = true; // eine HTTP-Anfrage endet mit einer
83   Leerzeile und einer neuen Zeile
84
85 while (client.connected())
86 {
87     if (client.available()) // Wenn Daten vom Server empfangen
88         werden
89     {
90         char c = client.read(); // empfangene Zeichen einlesen
91         if (c == '\n' && current_line_is_blank) // wenn neue Zeile und Leerzeile
92             empfangen
93         { // Standard HTTP Header senden
94             client.println("HTTP/1.1 200 OK");
95             client.println("Content-Type: text/html");
96             client.println("Connection: close");
97             client.println();
98
99             // Website von SD-Karte laden
100             webFile = SD.open("index.htm"); // Website laden
101             if (webFile)
102             {
103                 while(webFile.available())
104                 {
105                     client.write(webFile.read()); // Website an Client schicken
106                 }
107                 webFile.close();
108             }
109             break;
110         }
111         if (c == '\n')
112         {
113             current_line_is_blank = true;
114         }
115         else if (c != '\r')
116         {
117             current_line_is_blank = false;
118         }
119     }
120 }

```

```
115         }  
116     }  
117 }  
118     delay(1);  
119     client.stop();  
120 }  
121 }
```