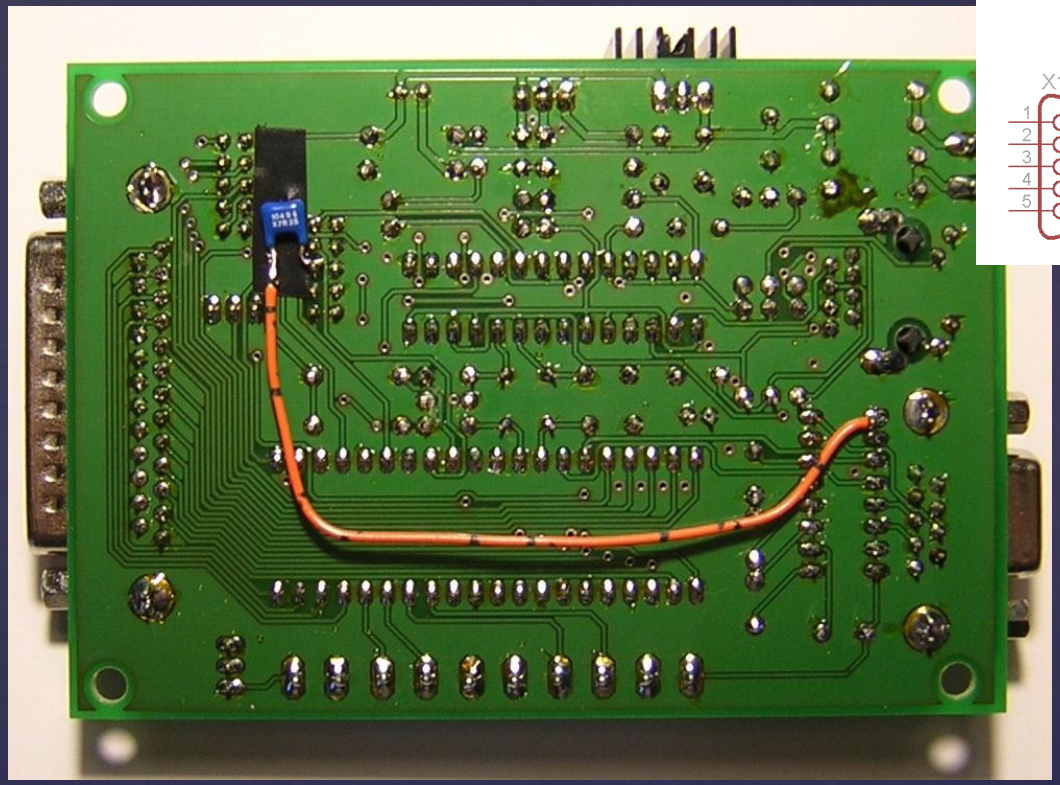
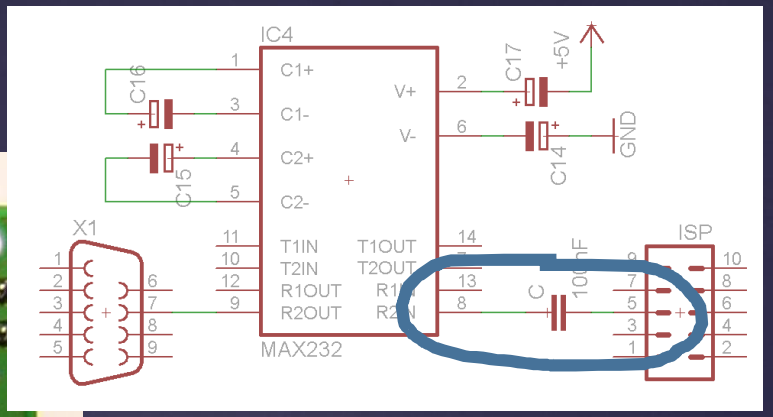
CheapNetDuinoHack

{ Using a Pollin AVR NetIO as Arduino w/ Ethernet

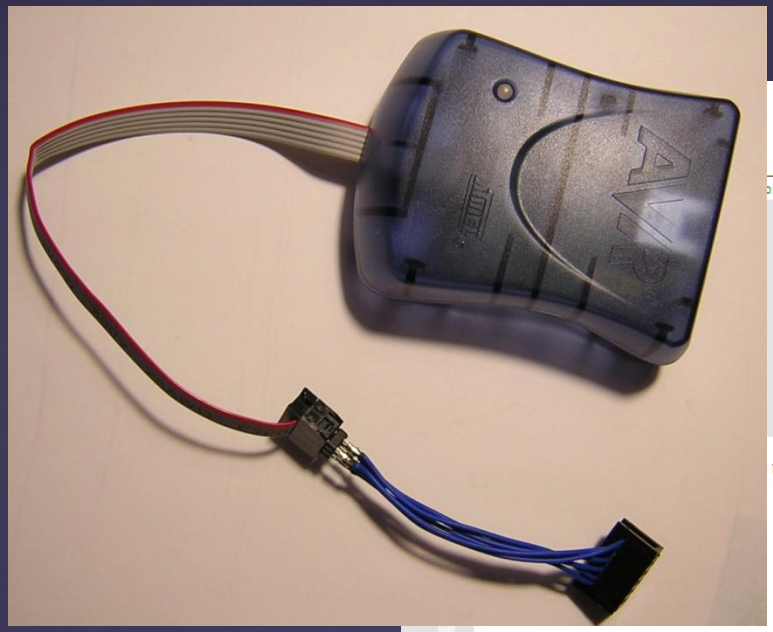
Markus Gebhard, Karlsruhe @ elektro:camp 2013:04





Add a RESET line for the Arduino IDE

Pollin AVR NetIO arduinofied

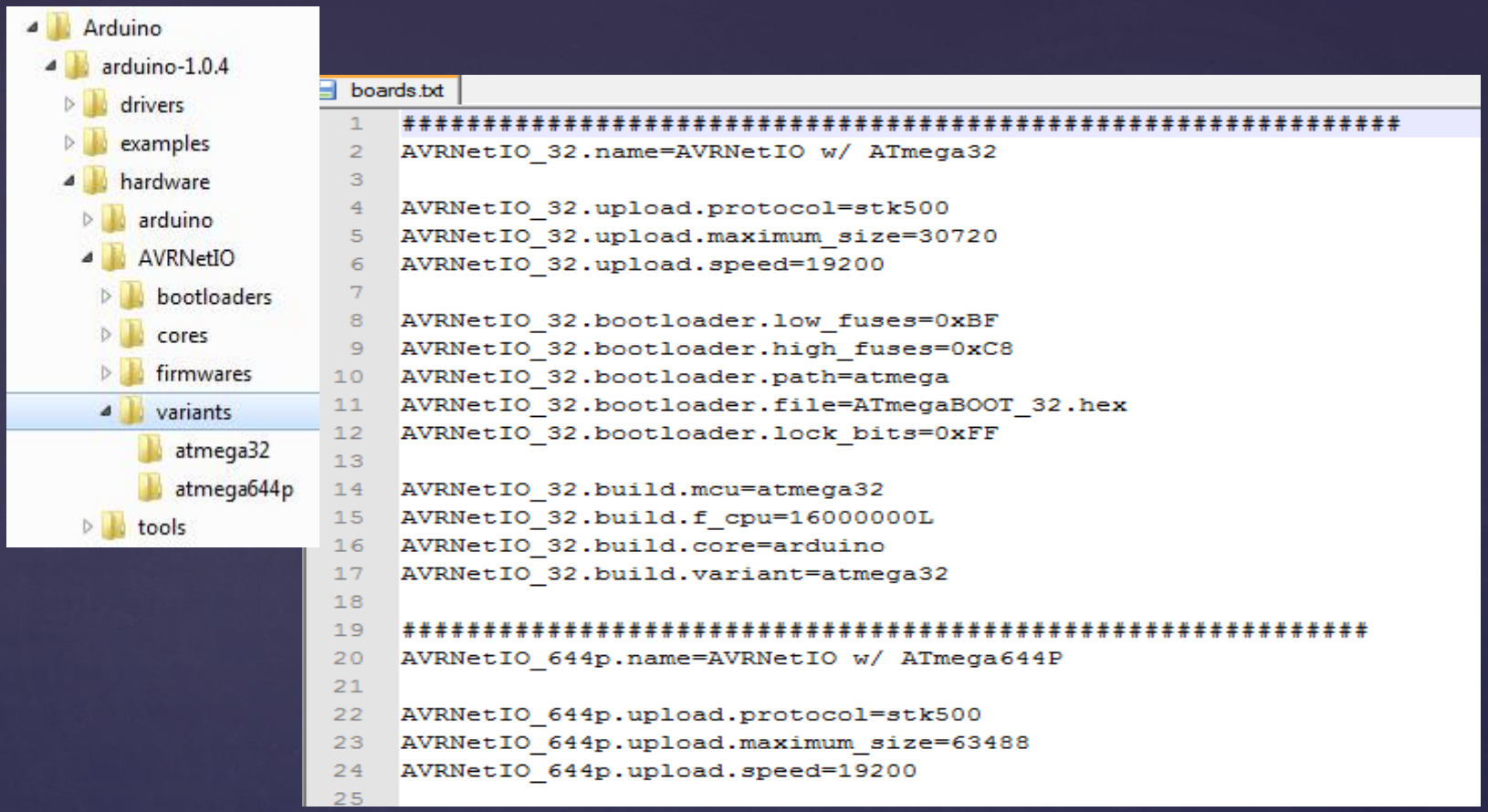


```
41  /* onboard LED is used to indicate, that the bootloade
42  /* if monitor functions are included, LED goes on afte
43  #if defined __AVR_ATmega128__ || defined __AVR_ATmega1
44  /* Onboard LED is connected to pin PB7 (e.g. Crumb128,
45  #define LED_DDR  DDRB
46  #define LED_PORT PORTB
47  #define LED_PIN  PINB
48  #define LED      PINB7
```

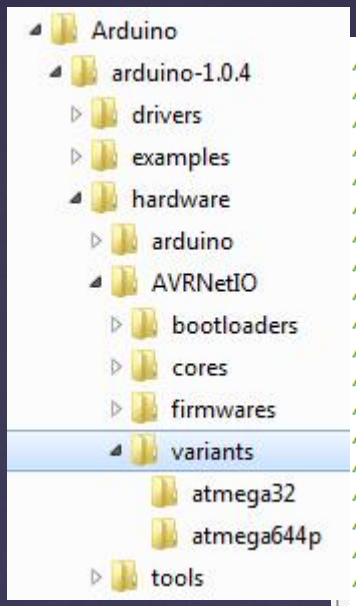
```
116   #define BL_PIN  PINF
117   #define BL0     PINF7
118   #define BL1     PINF6
119   #elif defined __AVR_ATmega1280__
120   /* we just don't do anything for the MEGA and enter bo
121   #elif defined __AVR_ATmega644P__  // the ATmega644P ha
122   #define BL_DDR  DDRD
123   #define BL_PORT PORTD
124   #define BL_PIN  PIND
125   #define BL0     PIND0
126   #define BL1     PIND2
127   #elif defined __AVR_ATmega32__
128   #define BL_DDR  DDRD
129   #define BL_PORT PORTD
130   #define BL_PIN  PIND
131   #define BL      PIND0
132   #else
133   /* other ATmegas have only one UART, so only one pin
134   #define BL_DDR  DDRD
135   #define BL_PORT PORTD
136   #define BL_PIN  PIND
137   #define BL      PIND6
138   #endif
139
140
141   /* onboard LED is used to indicate, that the bootload
142   /* if monitor functions are included, LED goes on aft
143   #if defined __AVR_ATmega128__ || defined __AVR_ATmega1
144   /* Onboard LED is connected to pin PB7 (e.g. Crumb128,
145   #define LED_DDR  DDRB
146   #define LED_PORT PORTB
147   #define LED_PIN  PINB
148   #define LED      PINB7
149   #elif defined __AVR_ATmega644P__ || defined __AVR_ATm
150   /* use an LED on pin B1 (AVRNetIO jumper to select pr
151   #define LED_DDR  DDRB
152   #define LED_PORT PORTB
```

Flash Arduino Bootloader

Adapt standard ATmegaBOOT_168.c for ATmega32/644 – basically PINs, chip IDs and #defines – no serious changes required…



Hardware for Arduino IDE → boards.txt




```
// ATMEL ATMEGA32
//
//                                  +---\/---+
//  Ext7         T0       D0   PB0 1 |        |40  PA0 AI0/D24 Eingang1/SubD10
//  J11          T1       D1   PB1 2 |        |39  PA1 AI1/D25 Eingang2/SubD11
//  ENC-INT      INT2     D2   PB2 3 |        |38  PA2 AI2/D26 Eingang3/SubD12
//  Ext8         PWM      D3   PB3 4 |        |37  PA3 AI3/D27 Eingang4/SubD13
//  ENC-SPI/ISP  SS       D4   PB4 5 |        |36  PA4 AI4/D28 ADC1
//  ENC-SPI/ISP  MOSI     D5   PB5 6 |        |35  PA5 AI5/D29 ADC2
//  ENC-SPI/ISP  MISO     D6   PB6 7 |        |34  PA6 AI6/D30 ADC3
//  ENC-SPI/ISP  SCK      D7   PB7 8 |        |33  PA7 AI7/D31 ADC4
//                                RST 9 |        |32  AREF
//                                VCC 10|        |31  GND
//                                GND 11|        |30  AVCC
//                              XTAL2 12|        |29  PC7 D23      Ausgang8/SubD9
//                              XTAL1 13|        |28  PC6 D22      Ausgang7/SubD8
//  MAX232-RX    RX0      D8   PD0 14|        |27  PC5 D21 TDI  Ausgang6/SubD7
//  MAX232-TX    TX0      D9   PD1 15|        |26  PC4 D20 TDO  Ausgang5/SubD6
//  Ext1         INT0     D10  PD2 16|        |25  PC3 D19 TMS  Ausgang4/SubD5
//  Ext2         INT1     D11  PD3 17|        |24  PC2 D18 TCK  Ausgang3/SubD4
//  Ext3         PWM      D12  PD4 18|        |23  PC1 D17 SDA  Ausgang2/SubD3
//  Ext4         PWM      D13  PD5 19|        |22  PC0 D16 SCL  Ausgang1/SubD2
//  Ext5                  D14  PD6 20|        |21  PD7 D15 PWM  Ext6
//                                   +--------+
//
//

#define NOT_A_PIN 0
#define NOT_A_PORT 0

#define NOT_ON_TIMER 0
#define TIMER0 1
#define TIMER1A 2
```

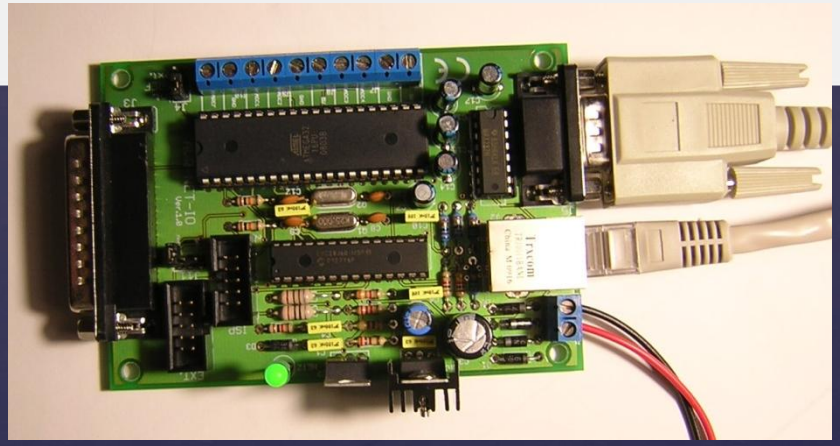
Hardware for Arduino IDE → pins_arduino.h variants

Use Jean-Claude‘s brilliant ethercard driver…

Enjoy a cheap Arduino w/ ethernet capability

Received Data

```
1    5    10    15    20    25    30    35    40    45    50    55    60    65    70    75    80    85
AVRNetIO Test\r\n
\r\n
Assigned IP address 192.168.0.128\r\n
GET /?o1=1&o4=1&o5=1&o8=1 HTTP/1.1\r\n
Host: 192.168.0.128\r\n
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:20.0) Gecko/20100101 Firefox/20.0\r\n
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n
Accept-Language: de-de,de;q=0.8,en-us;q=0.5,en;q=0.3\r\n
Accept-Encoding: gzip, deflate\r\n
DNT: 1\r\n
Referer: http://192.168.0.128/?o1=1&o4=1&o5=1&o8=1\r\n
Connection: keep-alive\r\n
\r\n
\r\n
```



Simple AVRNetIO webserver

☑ Output 1	☐ Output 2
☐ Output 3	☑ Output 4
☑ Output 5	☐ Output 6
☐ Output 7	☑ Output 8
Input 1: 1	Input 2: 1
Input 3: 1	Input 4: 1
ADC1: 824	ADC2: 666
ADC3: 435	ADC4: 331

Daten absenden

Easy Server

https://github.com/gebhardm/energyhacks/tree/master/AVRNetIOduino

With ideas taken from
http://sanguino.cc/
http://son.ffdf-clan.de/?path=start

Resources