softusbduino Documentation

Release 1.1.1

ponty

March 21, 2012

CONTENTS

softusbduino

Date March 21, 2012

PDF softusbduino.pdf

CHAPTER

ONE

ABOUT

softusbduino is a Python package and Arduino firmware library. They can be used together to control the Arduino board over USB in Python.

Links:

- home: https://github.com/ponty/softusbduino
- documentation: http://ponty.github.com/softusbduino

Hierarchy: Python Application -> softusbduino python library -> PyUSB -> libusb -> USB cable -> V-USB hardware -> Arduino -> V-USB library -> softusbduino firmware

Features:

- Possible usage: prototyping or creating simple low speed USB devices.
- firmware should be load only once to the Arduino board.
- 1 low level call takes 2 ms in tests
- **python library functions:**
  - read or write all registers
  - call arduino functions
  - read many defines (example: F_CPU)
- Python USB back-end: PyUSB library (0.4 API)
- Arduino USB back-end: V-USB library

Known problems:

- tested only on Linux + arduino 0022 + ATmega88 board
- pull-up read is not implemented
- PWM read is not implemented
- PWM config is hardcoded

similar projects:

- https://github.com/HashNuke/Python-Arduino-Prototyping-API
- http://code.google.com/p/vusb-for-arduino/
- http://code.google.com/p/pyduino/

CHAPTER

TWO

BASIC USAGE OF PROTOTYPING

```python
from softusbduino.protoapi import *

def setup():
    pinMode(13, OUTPUT);

def loop():
    digitalWrite(13, HIGH);
    delay(1000);
    digitalWrite(13, LOW);
    delay(1000);

sketch = Sketch(setup, loop)
sketch.run()
```

CHAPTER

THREE

INSTALLATION

3.1 General

- install Python
- install pip
- install arduino
- install libusb
- **install SoftUsb subdirectory as arduino library**
  - Manual installation: http://arduino.cc/en/Guide/Environment#libraries
  - **Automatic installation:**
    - install confduino
    - install the library: `python -m confduino.libinstall https://github.com/ponty/softusbduino/zipball/master`
- install python package:

  ```
  # as root
  pip install https://github.com/ponty/softusbduino/zipball/master
  ```

3.2 Ubuntu

```
sudo apt-get install arduino python-pip libusb-1.0-0
sudo pip install confduino
sudo pip install https://github.com/ponty/softusbduino/zipball/master
sudo python -m confduino.libinstall https://github.com/ponty/softusbduino/zipball/master
# optional for examples
sudo pip install matplotlib traits traitsui
```

3.3 Upload firmware

1. start Arduino
2. open examples > SoftUsb > Simple
3. upload to board

CHAPTER

FOUR

USAGE

```
>>> from softusbduino import *
>>>
>>> mcu = Arduino()
>>>
>>> # reset pin directions
>>> mcu.reset()
>>>
>>> # constants in python library
>>> print '0x%X' % mcu.usb.id_vendor
0x16C0
>>> print '0x%X' % mcu.usb.id_product
0x5DF
>>> print mcu.bandgap_voltage
1.1
>>>
>>> # constants in firmware
>>> print mcu.pins.usb_minus_pin
0
>>> print mcu.pins.usb_plus_pin
2
>>> print mcu.pins.count
20
>>> print mcu.pins.count_digital
14
>>> print mcu.pins.count_analog
6
>>> print mcu.pins.range_all
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19]
>>> print mcu.pins.range_digital
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]
>>> print mcu.pins.range_analog
[14, 15, 16, 17, 18, 19]
>>>
>>> # supply voltage
>>> print mcu.vcc.voltage
4.89739130435
>>> print mcu.vcc.u_voltage
4.89739130435+/-0.0212930056711
>>>
>>> # pin
>>> print mcu.pin(8).nr
8
>>> print mcu.pin('D8').nr
8
>>> print mcu.pin('A2').nr
16
>>> print mcu.pin('D13').programming_function
SCK
```

```
>>>
>>> # pin mode
>>> mcu.pins.write_mode(8, OUTPUT)
>>> print mcu.pins.read_mode(8)
1
>>> print mcu.pin('D8').read_mode()
1
>>> print mcu.pin('D8').mode
1
>>> mcu.pin('D8').mode = INPUT
>>> print mcu.pins.read_mode(8)
0
>>>
>>> # analog read
>>> print mcu.pins.read_analog(15)
305
>>> print mcu.pin('A2').read_analog()
355
>>> print mcu.pin('A2').analog
428
>>>
>>> # digital read
>>> print mcu.pins.read_digital(8)
1
>>> print mcu.pin('D8').read_digital()
1
>>> print mcu.pin('D8').digital
1
>>>
>>> # pullup
>>> mcu.pins.write_pullup(8, HIGH)
>>> mcu.pin('D8').write_pullup(HIGH)
>>>
>>> # digital write
>>> mcu.pins.write_mode(8, OUTPUT)
>>> mcu.pins.write_digital(8, HIGH)
>>> mcu.pin('D8').write_digital(HIGH)
>>> mcu.pin('D8').digital = HIGH
>>>
>>> # PWM
>>> print mcu.pin('D9').pwm.available
True
>>> print mcu.pin('D9').pwm.timer_register_name
TCCR1B
>>> print mcu.pin('D9').pwm.frequencies_available
[39062.5, 4882.8125, 610.3515625, 152.587890625, 38.14697265625]
>>> print mcu.pin('D9').pwm.frequency
610.3515625
>>> print mcu.pin('D9').pwm.divisors_available
[1, 8, 64, 256, 1024]
>>> print mcu.pin('D9').pwm.divisor
64
>>> mcu.pin('D9').pwm.divisor = 256
>>> print mcu.pin('D9').pwm.frequency
152.587890625
>>> print mcu.pin('D9').pwm.divisor
256
>>> mcu.pin('D9').pwm.frequency = 38
>>> print mcu.pin('D9').pwm.frequency
38.1469726562
>>> print mcu.pin('D9').pwm.divisor
1024
>>> mcu.pins.pwm.write_value(9, 54)
>>> mcu.pin(’D9’).pwm.write_value(44)
>>> mcu.pin(’D9’).pwm.value = 34
>>>
>>> # read defines
>>> print mcu.define(’F_CPU’)
20000000
>>> print mcu.defines.value(’F_CPU’)
20000000
>>> print mcu.defines.exists(’F_CPU’)
True
>>>
>>> print mcu.define(’MCU_DEFINED’)
__AVR_ATmega88__
>>> print mcu.define(’F_CPU’)
20000000
>>> print mcu.define(’__DATE__’)
Mar 19 2012
>>> print mcu.define(’MOSI’)
11
>>> print mcu.define(’USB_CFG_DMINUS_BIT’)
0
>>> print mcu.define(’ARDUINO’)
22
>>> print mcu.define(’__AVR_LIBC_VERSION__’)
10701
>>> print mcu.define(’A0’)
14
>>>
>>> # read/write register
>>> mcu.register(’DDRB’).value = 0
>>> print mcu.registers.read_value(’DDRB’)
0
>>> print mcu.register(’DDRB’).read_value()
0
>>> print mcu.register(’DDRB’).value
0
>>> print mcu.pin(8).mode
0
>>> mcu.register(’DDRB’).value = 1
>>> print mcu.register(’DDRB’).value
1
>>> print mcu.pin(8).mode
1
>>> mcu.pin(8).mode = INPUT
>>> print mcu.register(’DDRB’).value
0
>>> print mcu.pin(8).mode
0
>>>
>>>
>>> mcu.reset()
```

4.1 Code generation

Integer defines should be listed in softusbduino/intdefs.csv. String defines are hardcoded. Registers and MCU names are read from AVR Libc directory (/usr/lib/avr/include/avr/).

Run codegen.py to update generated files:

- softusbduino/generated_registers.csv

- SoftUsb/generated_registers.h
- SoftUsb/generated_intdefs.h
- SoftUsb/generated_mcu.h
- SoftUsb/generated_version.h

CHAPTER

FIVE

EXAMPLES

5.1 Simple example

```
from entrypoint2 import entrypoint
from softusbduino.arduino import Arduino


@entrypoint
def main():
    mcu = Arduino()
    print 'F_CPU=', mcu.define('F_CPU')
    print 'DDRC=', mcu.register('DDRC').read_value()
```

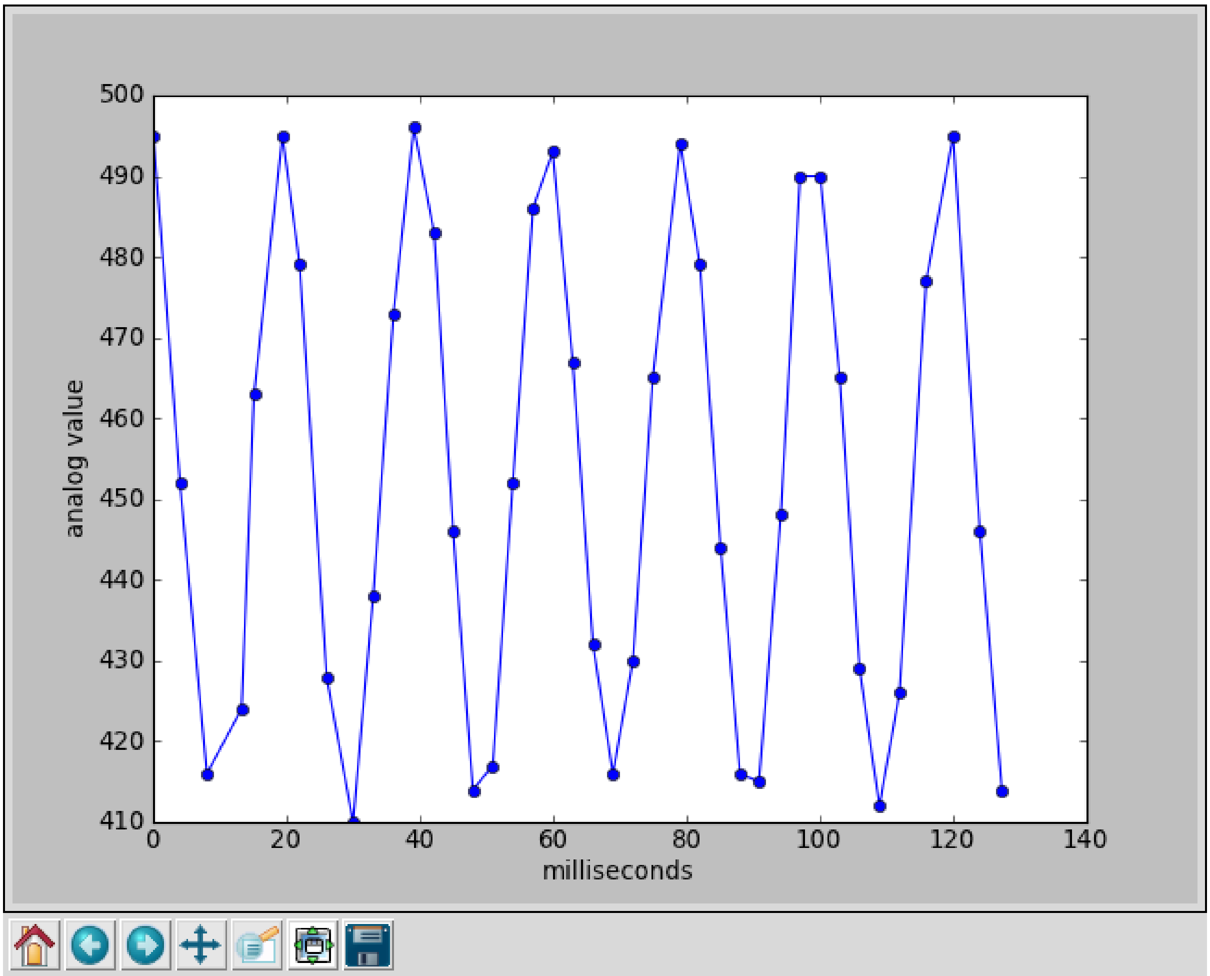
```
$ python -m softusbduino.examples.simple
F_CPU= 20000000
DDRC= 0
```

5.2 Plot

```
from entrypoint2 import entrypoint
from matplotlib.ticker import FuncFormatter
from softusbduino.arduino import Arduino
import matplotlib.pyplot as plt
import time

@entrypoint
def main(n=40, pin_nr=13, reset=False):
    '''
    measuring analog input
    '''
    mcu = Arduino(reset=reset)
    pin = mcu.pin(pin_nr)

    x = []
    y = []
    start = time.time()
    for i in range(n):
        t = time.time() - start
        v = pin.read_analog()
        x.append(t)
        y.append(v)
    fig = plt.figure()
    ax = fig.add_subplot(111)
    ax.plot(x, y, 'b-o')
    ax.yaxis.set_major_formatter(FuncFormatter(lambda x, pos: ('%d') % (x)))
    ax.set_ylabel('analog value')

    ax.xaxis.set_major_formatter(FuncFormatter(lambda x, pos: '%.0f' % (1000 * x)))
    ax.set_xlabel('milliseconds')
    plt.show()
```

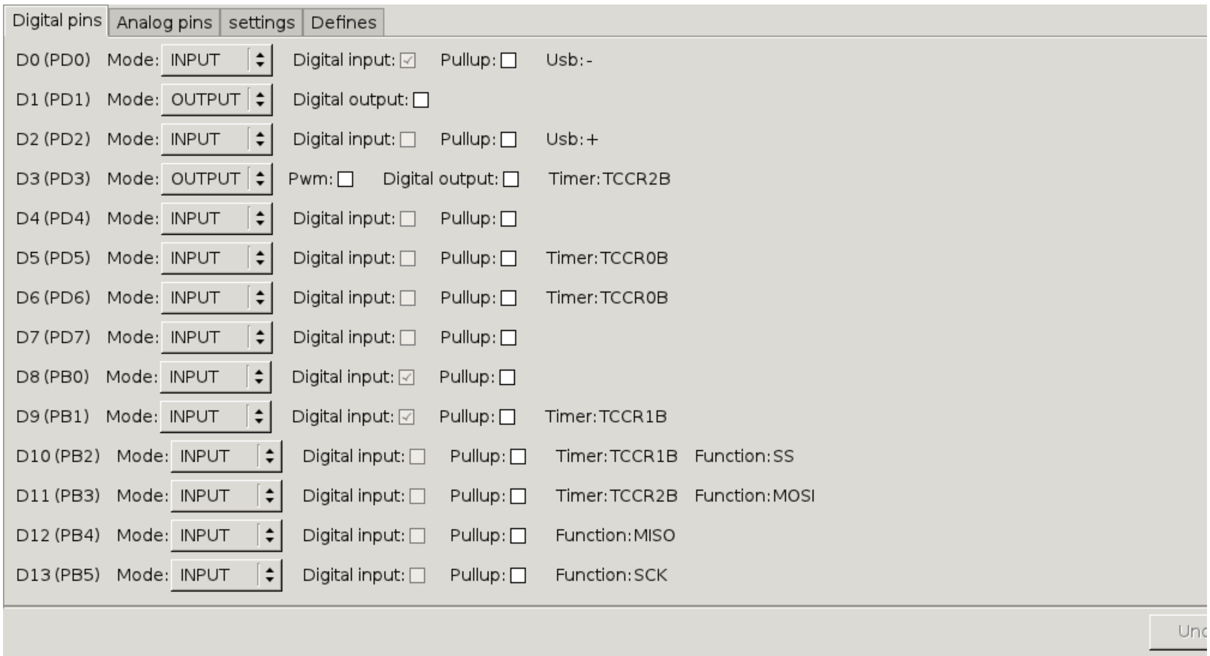
```
$ python -m softusbduino.examples.analogplot --help
usage: analogplot.py [-h] [--n N] [-p PIN_NR] [-r] [--debug]

measuring analog input

optional arguments:
  -h, --help            show this help message and exit
  --n N
  -p PIN_NR, --pin-nr PIN_NR
  -r, --reset
  --debug               set logging level to DEBUG
```

```
$ python -m softusbduino.examples.analogplot
```



5.3 Demo GUI

```
$ python -m softusbduino.examples.guidemo
```



D0 (PD0) Mode: INPUT Digital input: ☑ Pullup: □ Usb: -
D1 (PD1) Mode: OUTPUT Digital output: □
D2 (PD2) Mode: INPUT Digital input: □ Pullup: □ Usb: +
D3 (PD3) Mode: OUTPUT Pwm: □ Digital output: □ Timer: TCCR2B
D4 (PD4) Mode: INPUT Digital input: □ Pullup: □
D5 (PD5) Mode: INPUT Digital input: □ Pullup: □ Timer: TCCR0B
D6 (PD6) Mode: INPUT Digital input: □ Pullup: □ Timer: TCCR0B
D7 (PD7) Mode: INPUT Digital input: □ Pullup: □
D8 (PB0) Mode: INPUT Digital input: ☑ Pullup: □
D9 (PB1) Mode: INPUT Digital input: ☑ Pullup: □ Timer: TCCR1B
D10 (PB2) Mode: INPUT Digital input: □ Pullup: □ Timer: TCCR1B Function: SS
D11 (PB3) Mode: INPUT Digital input: □ Pullup: □ Timer: TCCR2B Function: MOSI
D12 (PB4) Mode: INPUT Digital input: □ Pullup: □ Function: MISO
D13 (PB5) Mode: INPUT Digital input: □ Pullup: □ Function: SCK

5.4 prototyping

softusbduino/examples/proto/Blink.py

```
'''
  Blink
  Turns on an LED on for one second, then off for one second, repeatedly.

  Converted from Arduino example.
'''

from softusbduino.protoapi import *

def setup():
    pinMode(13, OUTPUT);

def loop():
    digitalWrite(13, HIGH);
    delay(1000);
    digitalWrite(13, LOW);
    delay(1000);



sketch = Sketch(setup, loop)
sketch.run()
```

softusbduino/examples/proto/AnalogInOutSerial.py

```
'''
  Analog input, analog output, serial output

 Reads an analog input pin, maps the result to a range from 0 to 255
 and uses the result to set the pulsewidth modulation (PWM) of an output pin.
 Also prints the results to the serial monitor.

 The circuit:
 * potentiometer connected to analog pin 0.
```

```
   Center pin of the potentiometer goes to the analog pin.
   side pins of the potentiometer go to +5V and ground
 * LED connected from digital pin 9 to ground

  Converted from Arduino example.
'''

from softusbduino.protoapi import *

# These constants won't change.  They're used to give names
# to the pins used:
analogInPin = A0;  # Analog input pin that the potentiometer is attached to
analogOutPin = 9; # Analog output pin that the LED is attached to

sensorValue = 0;        # value read from the pot
outputValue = 0;        # value output to the PWM (analog out)

def setup():
    # initialize serial communications at 9600 bps:
    Serial.begin(9600);


def loop():
    # read the analog in value:
    sensorValue = analogRead(analogInPin);
    # map it to the range of the analog out:
    outputValue = map(sensorValue, 0, 1023, 0, 255);
    # change the analog out value:
    analogWrite(analogOutPin, outputValue);

    # print the results to the serial monitor:
    Serial.print_("sensor = " );
    Serial.print_(sensorValue);
    Serial.print_("\t output = ");
    Serial.println(outputValue);

    # wait 10 milliseconds before the next loop
    # for the analog-to-digital converter to settle
    # after the last reading:
    delay(10);

sketch = Sketch(setup, loop)
sketch.run()
```

CHAPTER

SIX

1WIRE EXAMPLES

6.1 Reading temperature

```
from entrypoint2 import entrypoint
from softusbduino.arduino import Arduino
import time

@entrypoint
def main(
         pin='D9',
         timeout=10,
         ):
    mcu = Arduino()
    bus = mcu.bus1wire(pin)
    devs = bus.search()
    d = devs[0]
    print 'address=', d.address_str
    print 'address_valid=', d.address_valid
    print 'chip=', d.chip
    print 'resolution=', d.resolution, 'bit'

    start = time.time()
    while 1:
        x = d.scratchpad()
        print x.celsius, 'C', time.ctime(x.t), x.data
        time.sleep(0.1)
        if timeout > 0:
            if timeout < time.time() - start:
                break
```

```
$ python -m softusbduino.examples.onewire_demo
address= 28.89.E1.2E.03.00.00.F4
address_valid= True
chip= DS18B20
resolution= 12 bit
22.625 C Wed Mar 21 20:14:47 2012 [106, 1, 75, 70, 127, 255, 6, 16, 95]
22.625 C Wed Mar 21 20:14:50 2012 [106, 1, 75, 70, 127, 255, 6, 16, 95]
22.625 C Wed Mar 21 20:14:54 2012 [106, 1, 75, 70, 127, 255, 6, 16, 95]
22.625 C Wed Mar 21 20:14:57 2012 [106, 1, 75, 70, 127, 255, 6, 16, 95]
```

CHAPTER SEVEN

TESTS

Test system versions:

```
$ python -m softusbduino.lsversion
platform        Linux-3.0.0-16-generic-i686-athlon-with-LinuxMint-12-lisa
python          2.7.2+
```

Performance test:

```
$ python -m softusbduino.check.performance
performance test
n= 100

pins.read_analog(0)                     3.53 ms per call,   283 call per second
pins.write_mode(8,0)                    3.38 ms per call,   296 call per second
pins.read_digital(8)                    3.22 ms per call,   311 call per second
defines.value("__TIME__")               0.07 ms per call, 14993 call per second
defines.exists("__TIME__")              0.03 ms per call, 29446 call per second
defines.exists("xx")                    0.04 ms per call, 23776 call per second
define("A0")                            0.07 ms per call, 14990 call per second
registers.read_value("DDRB")            3.73 ms per call,   268 call per second
registers.exists("DDRB")                0.02 ms per call, 42539 call per second
registers.exists("xx")                  0.02 ms per call, 41218 call per second
register("DDRB").value                  3.74 ms per call,   267 call per second
register("DDRB").read_value()           3.38 ms per call,   296 call per second
register("DDRB").exists                 0.03 ms per call, 37190 call per second
register("xx").exists                   0.03 ms per call, 33638 call per second
vcc.voltage                             0.37 ms per call,  2681 call per second
vcc.read_voltage()                     21.12 ms per call,    47 call per second
read_vcc()                             22.30 ms per call,    45 call per second
pins.count                              0.02 ms per call, 53160 call per second
pins.usb_minus_pin                      0.02 ms per call, 64065 call per second
pins.usb_plus_pin                       0.02 ms per call, 43800 call per second
firmware_test()                         0.02 ms per call, 40864 call per second
pins.read_mode(0)                       3.48 ms per call,   288 call per second
reset()                               121.88 ms per call,     8 call per second
```

Dump state:

```
$ python -m softusbduino.check.dump

================================
Arduino() attributes:
================================
Rout            =              15
_bus1wire       = <functools.partial object at 0x8b9d0a4>
_pin            = <functools.partial object at 0x8b9d0a4>
adc_accuracy    =               1
analog_range    =       (0, 1023)
bandgap_voltage =             1.1
```

```
define          = <functools.partial object at 0x8b9d0a4>
defines         = <softusbduino.defines.Defines object at 0x8bfcf4c>
lowlevel_1wire  = <softusbduino.onewire.LowLevel1Wire object at 0x8c03aec>
lowlevel_defines = <softusbduino.defines.DefinesLowLevel object at 0x8c0304c>
lowlevel_delaytest = <softusbduino.delaytest.DelayTestLowLevel object at 0x8c03b0c>
lowlevel_pins   = <softusbduino.pin.PinsLowLevel object at 0x8c0328c>
lowlevel_pwm    = <softusbduino.pwmpin.PwmLowLevel object at 0x8c03b2c>
lowlevel_registers = <softusbduino.registers.RegistersLowLevel object at 0x8c03b4c>
pins            = <softusbduino.pin.Pins object at 0x8bfce6c>
pwm             = <softusbduino.pwmpin.Pwm object at 0x8c03b6c>
register        = <functools.partial object at 0x8b9d11c>
registers       = <softusbduino.registers.Registers object at 0x8c03bcc>
serializer      = <softusbduino.ser.Serializer object at 0x8c030ec>
usb             = <softusbduino.usbdevice.UsbDevice object at 0x8c0312c>
vcc             = <softusbduino.vcc.Vcc object at 0x8c03bac>

=================================
Arduino().usb attributes:
=================================
auto_reconnect  =            True
device          = <usb.Device object at 0x8c468e8>
device_handle   = <usb.DeviceHandle object at 0xb7747210>
id_product      =            1503
id_vendor       =            5824
manufacturer    = o\b\d\e\v\.\a\t\
productName     = L\E\D\C\t\l\H\I\D\

=================================
Arduino().pins attributes:
=================================
avr_bit         = <functools.partial object at 0x8b9d0a4>
avr_port        = <functools.partial object at 0x8b9d0a4>
base            = <softusbduino.pin.PinsLowLevel object at 0x8c0328c>
count           =              20
count_analog    =               6
count_digital   =              14
defines         = <softusbduino.defines.Defines object at 0x8bfcf4c>
mcu             = <softusbduino.arduino.Arduino object at 0x8ba96ac>
range_all       = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19]
range_analog    = [14, 15, 16, 17, 18, 19]
range_digital   = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]
usb_minus_pin   =               0
usb_neighbours  =          [1, 3]
usb_plus_pin    =               2

=================================
Arduino().pin(nr) attributes:
=================================
---------- nr=0 ---------------
A0              =              14
analog          =             596
analog_obj      = AnalogInputValue<value:560 voltage:2.66666666667>
avr_bit         =               0
avr_pin         =             PD0
avr_port        =               D
base            = <softusbduino.pin.Pins object at 0x8bfce6c>
digital         =               1
digital_in      =               1
digital_out     =            None
is_analog       =           False
is_digital      =            True
is_usb_minus    =            True
is_usb_plus     =           False
```

```
mcu             = <softusbduino.arduino.Arduino object at 0x8ba96ac>
mode            =               0
name            =              D0
nr              =               0
nr_analog       =            None
programming_function =            None
pwm             = <softusbduino.pwmpin.PwmPin object at 0x8ae272c>
---------- nr=1 ---------------
A0              =              14
analog          =             535
analog_obj      = AnalogInputValue<value:580 voltage:2.7619047619>
avr_bit         =               1
avr_pin         =             PD1
avr_port        =               D
base            = <softusbduino.pin.Pins object at 0x8bfce6c>
digital         =               0
digital_in      =            None
digital_out     =               0
is_analog       =           False
is_digital      =            True
is_usb_minus    =           False
is_usb_plus     =           False
mcu             = <softusbduino.arduino.Arduino object at 0x8ba96ac>
mode            =               1
name            =              D1
nr              =               1
nr_analog       =            None
programming_function =            None
pwm             = <softusbduino.pwmpin.PwmPin object at 0x8ae244c>
---------- nr=2 ---------------
A0              =              14
analog          =             466
analog_obj      = AnalogInputValue<value:580 voltage:2.7619047619>
avr_bit         =               2
avr_pin         =             PD2
avr_port        =               D
base            = <softusbduino.pin.Pins object at 0x8bfce6c>
digital         =               0
digital_in      =               0
digital_out     =            None
is_analog       =           False
is_digital      =            True
is_usb_minus    =           False
is_usb_plus     =            True
mcu             = <softusbduino.arduino.Arduino object at 0x8ba96ac>
mode            =               0
name            =              D2
nr              =               2
nr_analog       =            None
programming_function =            None
pwm             = <softusbduino.pwmpin.PwmPin object at 0x8ae260c>
---------- nr=3 ---------------
A0              =              14
analog          =             490
analog_obj      = AnalogInputValue<value:493 voltage:2.34761904762>
avr_bit         =               3
avr_pin         =             PD3
avr_port        =               D
base            = <softusbduino.pin.Pins object at 0x8bfce6c>
digital         =               0
digital_in      =            None
digital_out     =               0
is_analog       =           False
```

```
is_digital      =            True
is_usb_minus    =           False
is_usb_plus     =           False
mcu             = <softusbduino.arduino.Arduino object at 0x8ba96ac>
mode            =               1
name            =              D3
nr              =               3
nr_analog       =            None
programming_function =            None
pwm             = <softusbduino.pwmpin.PwmPin object at 0x8ae29ec>
--- pwm ---
available       =            True
base            = <softusbduino.pwmpin.Pwm object at 0x8c03b6c>
base_divisor    =             512
divisor         =              64
divisors_available = [1, 8, 32, 64, 128, 256, 1024]
frequencies_available = [39062.5, 4882.8125, 1220.703125, 610.3515625, 305.17578125, 152.587890625
frequency       =     610.3515625
pin             = <softusbduino.pin.Pin object at 0x8ae2e6c>
timer_mode      =               4
timer_register_name =          TCCR2B
---------- nr=4 ---------------
A0              =              14
analog          =             412
analog_obj      = AnalogInputValue<value:487 voltage:2.31904761905>
avr_bit         =               4
avr_pin         =             PD4
avr_port        =               D
base            = <softusbduino.pin.Pins object at 0x8bfce6c>
digital         =               0
digital_in      =               0
digital_out     =            None
is_analog       =           False
is_digital      =            True
is_usb_minus    =           False
is_usb_plus     =           False
mcu             = <softusbduino.arduino.Arduino object at 0x8ba96ac>
mode            =               0
name            =              D4
nr              =               4
nr_analog       =            None
programming_function =            None
pwm             = <softusbduino.pwmpin.PwmPin object at 0x8ae2ccc>
---------- nr=5 ---------------
A0              =              14
analog          =             407
analog_obj      = AnalogInputValue<value:351 voltage:1.67142857143>
avr_bit         =               5
avr_pin         =             PD5
avr_port        =               D
base            = <softusbduino.pin.Pins object at 0x8bfce6c>
digital         =               0
digital_in      =               0
digital_out     =            None
is_analog       =           False
is_digital      =            True
is_usb_minus    =           False
is_usb_plus     =           False
mcu             = <softusbduino.arduino.Arduino object at 0x8ba96ac>
mode            =               0
name            =              D5
nr              =               5
nr_analog       =            None
```

```
programming_function =            None
pwm             = <softusbduino.pwmpin.PwmPin object at 0x8ae2fcc>
--- pwm ---
available       =            True
base            = <softusbduino.pwmpin.Pwm object at 0x8c03b6c>
base_divisor    =             256
divisor         =              64
divisors_available = [1, 8, 64, 256, 1024]
frequencies_available = [78125.0, 9765.625, 1220.703125, 305.17578125, 76.2939453125]
frequency       =     1220.703125
pin             = <softusbduino.pin.Pin object at 0x8ae2b0c>
timer_mode      =               3
timer_register_name =          TCCR0B
---------- nr=6 ---------------
A0              =              14
analog          =             363
analog_obj      = AnalogInputValue<value:363 voltage:1.72857142857>
avr_bit         =               6
avr_pin         =             PD6
avr_port        =               D
base            = <softusbduino.pin.Pins object at 0x8bfce6c>
digital         =               0
digital_in      =               0
digital_out     =            None
is_analog       =           False
is_digital      =            True
is_usb_minus    =           False
is_usb_plus     =           False
mcu             = <softusbduino.arduino.Arduino object at 0x8ba96ac>
mode            =               0
name            =              D6
nr              =               6
nr_analog       =            None
programming_function =            None
pwm             = <softusbduino.pwmpin.PwmPin object at 0x8ae606c>
--- pwm ---
available       =            True
base            = <softusbduino.pwmpin.Pwm object at 0x8c03b6c>
base_divisor    =             256
divisor         =              64
divisors_available = [1, 8, 64, 256, 1024]
frequencies_available = [78125.0, 9765.625, 1220.703125, 305.17578125, 76.2939453125]
frequency       =     1220.703125
pin             = <softusbduino.pin.Pin object at 0x8ae278c>
timer_mode      =               3
timer_register_name =          TCCR0B
---------- nr=7 ---------------
A0              =              14
analog          =             379
analog_obj      = AnalogInputValue<value:379 voltage:1.80476190476>
avr_bit         =               7
avr_pin         =             PD7
avr_port        =               D
base            = <softusbduino.pin.Pins object at 0x8bfce6c>
digital         =               0
digital_in      =               0
digital_out     =            None
is_analog       =           False
is_digital      =            True
is_usb_minus    =           False
is_usb_plus     =           False
mcu             = <softusbduino.arduino.Arduino object at 0x8ba96ac>
mode            =               0
```

```
name            =              D7
nr              =               7
nr_analog       =            None
programming_function =            None
pwm             = <softusbduino.pwmpin.PwmPin object at 0x8ae2eac>
---------- nr=8 ---------------
A0              =              14
analog          =             539
analog_obj      = AnalogInputValue<value:628 voltage:2.99047619048>
avr_bit         =               0
avr_pin         =             PB0
avr_port        =               B
base            = <softusbduino.pin.Pins object at 0x8bfce6c>
digital         =               1
digital_in      =               1
digital_out     =            None
is_analog       =           False
is_digital      =            True
is_usb_minus    =           False
is_usb_plus     =           False
mcu             = <softusbduino.arduino.Arduino object at 0x8ba96ac>
mode            =               0
name            =              D8
nr              =               8
nr_analog       =            None
programming_function =            None
pwm             = <softusbduino.pwmpin.PwmPin object at 0x8ae854c>
---------- nr=9 ---------------
A0              =              14
analog          =             589
analog_obj      = AnalogInputValue<value:500 voltage:2.38095238095>
avr_bit         =               1
avr_pin         =             PB1
avr_port        =               B
base            = <softusbduino.pin.Pins object at 0x8bfce6c>
digital         =               1
digital_in      =               1
digital_out     =            None
is_analog       =           False
is_digital      =            True
is_usb_minus    =           False
is_usb_plus     =           False
mcu             = <softusbduino.arduino.Arduino object at 0x8ba96ac>
mode            =               0
name            =              D9
nr              =               9
nr_analog       =            None
programming_function =            None
pwm             = <softusbduino.pwmpin.PwmPin object at 0x8ae81cc>
--- pwm ---
available       =            True
base            = <softusbduino.pwmpin.Pwm object at 0x8c03b6c>
base_divisor    =             512
divisor         =              64
divisors_available = [1, 8, 64, 256, 1024]
frequencies_available = [39062.5, 4882.8125, 610.3515625, 152.587890625, 38.14697265625]
frequency       =     610.3515625
pin             = <softusbduino.pin.Pin object at 0x8ae6fec>
timer_mode      =               3
timer_register_name =          TCCR1B
---------- nr=10 ---------------
A0              =              14
analog          =             594
```

```
analog_obj      = AnalogInputValue<value:495 voltage:2.35714285714>
avr_bit         =               2
avr_pin         =             PB2
avr_port        =               B
base            = <softusbduino.pin.Pins object at 0x8bfce6c>
digital         =               0
digital_in      =               0
digital_out     =            None
is_analog       =           False
is_digital      =            True
is_usb_minus    =           False
is_usb_plus     =           False
mcu             = <softusbduino.arduino.Arduino object at 0x8ba96ac>
mode            =               0
name            =             D10
nr              =              10
nr_analog       =            None
programming_function =              SS
pwm             = <softusbduino.pwmpin.PwmPin object at 0x8ae8b6c>
--- pwm ---
available       =            True
base            = <softusbduino.pwmpin.Pwm object at 0x8c03b6c>
base_divisor    =             512
divisor         =              64
divisors_available = [1, 8, 64, 256, 1024]
frequencies_available = [39062.5, 4882.8125, 610.3515625, 152.587890625, 38.14697265625]
frequency       =     610.3515625
pin             = <softusbduino.pin.Pin object at 0x8ae654c>
timer_mode      =               3
timer_register_name =          TCCR1B
---------- nr=11 ---------------
A0              =              14
analog          =             519
analog_obj      = AnalogInputValue<value:584 voltage:2.78095238095>
avr_bit         =               3
avr_pin         =             PB3
avr_port        =               B
base            = <softusbduino.pin.Pins object at 0x8bfce6c>
digital         =               0
digital_in      =               0
digital_out     =            None
is_analog       =           False
is_digital      =            True
is_usb_minus    =           False
is_usb_plus     =           False
mcu             = <softusbduino.arduino.Arduino object at 0x8ba96ac>
mode            =               0
name            =             D11
nr              =              11
nr_analog       =            None
programming_function =            MOSI
pwm             = <softusbduino.pwmpin.PwmPin object at 0x8ae8a2c>
--- pwm ---
available       =            True
base            = <softusbduino.pwmpin.Pwm object at 0x8c03b6c>
base_divisor    =             512
divisor         =              64
divisors_available = [1, 8, 32, 64, 128, 256, 1024]
frequencies_available = [39062.5, 4882.8125, 1220.703125, 610.3515625, 305.17578125, 152.587890625
frequency       =     610.3515625
pin             = <softusbduino.pin.Pin object at 0x8ae2b2c>
timer_mode      =               4
timer_register_name =          TCCR2B
```

```
---------- nr=12 ---------------
A0              =              14
analog          =             507
analog_obj      = AnalogInputValue<value:442 voltage:2.10476190476>
avr_bit         =               4
avr_pin         =             PB4
avr_port        =               B
base            = <softusbduino.pin.Pins object at 0x8bfce6c>
digital         =               0
digital_in      =               0
digital_out     =            None
is_analog       =           False
is_digital      =            True
is_usb_minus    =           False
is_usb_plus     =           False
mcu             = <softusbduino.arduino.Arduino object at 0x8ba96ac>
mode            =               0
name            =             D12
nr              =              12
nr_analog       =            None
programming_function =            MISO
pwm             = <softusbduino.pwmpin.PwmPin object at 0x8ae842c>
---------- nr=13 ---------------
A0              =              14
analog          =             374
analog_obj      = AnalogInputValue<value:451 voltage:2.14761904762>
avr_bit         =               5
avr_pin         =             PB5
avr_port        =               B
base            = <softusbduino.pin.Pins object at 0x8bfce6c>
digital         =               0
digital_in      =               0
digital_out     =            None
is_analog       =           False
is_digital      =            True
is_usb_minus    =           False
is_usb_plus     =           False
mcu             = <softusbduino.arduino.Arduino object at 0x8ba96ac>
mode            =               0
name            =             D13
nr              =              13
nr_analog       =            None
programming_function =             SCK
pwm             = <softusbduino.pwmpin.PwmPin object at 0x8ae8c4c>
---------- nr=14 ---------------
A0              =              14
analog          =             574
analog_obj      = AnalogInputValue<value:481 voltage:2.29047619048>
avr_bit         =               0
avr_pin         =             PC0
avr_port        =               C
base            = <softusbduino.pin.Pins object at 0x8bfce6c>
digital         =               1
digital_in      =               1
digital_out     =            None
is_analog       =            True
is_digital      =           False
is_usb_minus    =           False
is_usb_plus     =           False
mcu             = <softusbduino.arduino.Arduino object at 0x8ba96ac>
mode            =               0
name            =              A0
nr              =              14
```

```
nr_analog       =               0
programming_function =            None
pwm             = <softusbduino.pwmpin.PwmPin object at 0x8ae8c2c>
---------- nr=15 ---------------
A0              =              14
analog          =             626
analog_obj      = AnalogInputValue<value:503 voltage:2.39523809524>
avr_bit         =               1
avr_pin         =             PC1
avr_port        =               C
base            = <softusbduino.pin.Pins object at 0x8bfce6c>
digital         =               1
digital_in      =               1
digital_out     =            None
is_analog       =            True
is_digital      =           False
is_usb_minus    =           False
is_usb_plus     =           False
mcu             = <softusbduino.arduino.Arduino object at 0x8ba96ac>
mode            =               0
name            =              A1
nr              =              15
nr_analog       =               1
programming_function =            None
pwm             = <softusbduino.pwmpin.PwmPin object at 0x8ae8aac>
---------- nr=16 ---------------
A0              =              14
analog          =             625
analog_obj      = AnalogInputValue<value:512 voltage:2.4380952381>
avr_bit         =               2
avr_pin         =             PC2
avr_port        =               C
base            = <softusbduino.pin.Pins object at 0x8bfce6c>
digital         =               1
digital_in      =               1
digital_out     =            None
is_analog       =            True
is_digital      =           False
is_usb_minus    =           False
is_usb_plus     =           False
mcu             = <softusbduino.arduino.Arduino object at 0x8ba96ac>
mode            =               0
name            =              A2
nr              =              16
nr_analog       =               2
programming_function =            None
pwm             = <softusbduino.pwmpin.PwmPin object at 0x8ae940c>
---------- nr=17 ---------------
A0              =              14
analog          =             537
analog_obj      = AnalogInputValue<value:537 voltage:2.55714285714>
avr_bit         =               3
avr_pin         =             PC3
avr_port        =               C
base            = <softusbduino.pin.Pins object at 0x8bfce6c>
digital         =               1
digital_in      =               1
digital_out     =            None
is_analog       =            True
is_digital      =           False
is_usb_minus    =           False
is_usb_plus     =           False
mcu             = <softusbduino.arduino.Arduino object at 0x8ba96ac>
```

```
mode            =               0
name            =              A3
nr              =              17
nr_analog       =               3
programming_function =            None
pwm             = <softusbduino.pwmpin.PwmPin object at 0x8ae922c>
---------- nr=18 ---------------
A0              =              14
analog          =             488
analog_obj      = AnalogInputValue<value:445 voltage:2.11904761905>
avr_bit         =               4
avr_pin         =             PC4
avr_port        =               C
base            = <softusbduino.pin.Pins object at 0x8bfce6c>
digital         =               1
digital_in      =               0
digital_out     =            None
is_analog       =            True
is_digital      =           False
is_usb_minus    =           False
is_usb_plus     =           False
mcu             = <softusbduino.arduino.Arduino object at 0x8ba96ac>
mode            =               0
name            =              A4
nr              =              18
nr_analog       =               4
programming_function =            None
pwm             = <softusbduino.pwmpin.PwmPin object at 0x8ae994c>
---------- nr=19 ---------------
A0              =              14
analog          =             383
analog_obj      = AnalogInputValue<value:453 voltage:2.15714285714>
avr_bit         =               5
avr_pin         =             PC5
avr_port        =               C
base            = <softusbduino.pin.Pins object at 0x8bfce6c>
digital         =               0
digital_in      =               0
digital_out     =            None
is_analog       =            True
is_digital      =           False
is_usb_minus    =           False
is_usb_plus     =           False
mcu             = <softusbduino.arduino.Arduino object at 0x8ba96ac>
mode            =               0
name            =              A5
nr              =              19
nr_analog       =               5
programming_function =            None
pwm             = <softusbduino.pwmpin.PwmPin object at 0x8ae990c>

================================
Arduino().vcc attributes:
================================
_u_voltage      = 4.87619047619+/-0.0211090496805
base            = <softusbduino.arduino.Arduino object at 0x8ba96ac>
t               =   1332357256.75
u_voltage       = 4.87619047619+/-0.0211090496805
voltage         =   4.87619047619

================================
Arduino().defines attributes:
================================
base            = <softusbduino.defines.DefinesLowLevel object at 0x8c0304c>
intdef_ids      = Bunch(A0=14, ARDUINO=9, E2END=2, E2PAGESIZE=5, FLASHEND=3, F_CPU=8, MISO=12, MO
special_defines = Bunch(MAGIC_NUMBER=42, MCU_DEFINED=26, USBDRV_VERSION=25, USB_CFG_IOPORT=34, __
value           = <functools.partial object at 0x8c1361c>

================================
Arduino().registers attributes:
================================
address         = <functools.partial object at 0x8c1361c>
base            = <softusbduino.registers.RegistersLowLevel object at 0x8c03b4c>
exists          = <functools.partial object at 0x8c1361c>
register_id_map = Bunch(AC0CON=0, AC1CON=1, AC1ECON=2, AC2CON=3, AC2ECON=4, AC3CON=5, AC3ECON=6,


================================
defines:
================================
A0                   =                 14
ARDUINO              =                 22
E2END                =                511
E2PAGESIZE           =                  4
FLASHEND             =               8191
F_CPU                =           20000000
MAGIC_NUMBER         =                 42
MCU_DEFINED          =   __AVR_ATmega88__
MISO                 =                 12
MOSI                 =                 11
ONEWIRE_BUS_COUNT    =                 10
RAMEND               =               1279
SCK                  =                 13
SOFTUSBDUINO_FIRMWARE_VERSION =                  3
SOFTUSBDUINO_VERSION =              10101
SPM_PAGESIZE         =                 64
SS                   =                 10
USBDRV_VERSION       =           20100715
USB_CFG_DMINUS_BIT   =                  0
USB_CFG_DPLUS_BIT    =                  2
USB_CFG_IOPORT       =                  4
XRAMEND              =               1279
__AVR_LIBC_DATE_     =           20110216
__AVR_LIBC_VERSION__ =              10701
__DATE__             =        Mar 19 2012
__TIME__             =           15:08:20

================================
registers:
================================
ACSR                 = 0x30 @0x50
ADCH                 = 0x01 @0x79
ADCL                 = 0xC5 @0x78
ADCSRA               = 0x97 @0x7A
ADCSRB               = 0x00 @0x7B
ADMUX                = 0x45 @0x7C
ASSR                 = 0x00 @0xB6
CLKPR                = 0x00 @0x61
DDRB                 = 0x00 @0x24
DDRC                 = 0x00 @0x27
DDRD                 = 0x0A @0x2A
DIDR0                = 0x00 @0x7E
DIDR1                = 0x00 @0x7F
EEAR                 = 0x9B @0x41
EEARH                = 0x01 @0x42
EEARL                = 0x9B @0x41
```

```
EECR                 = 0x00 @0x3F
EEDR                 = 0x00 @0x40
EICRA                = 0x02 @0x69
EIFR                 = 0x00 @0x3C
EIMSK                = 0x01 @0x3D
GPIOR0               = 0x00 @0x3E
GPIOR1               = 0x00 @0x4A
GPIOR2               = 0x00 @0x4B
GTCCR                = 0x00 @0x43
ICR1                 = 0xA5 @0x86
ICR1H                = 0x00 @0x87
ICR1L                = 0xA5 @0x86
MCUCR                = 0x00 @0x55
MCUSR                = 0x01 @0x54
MONDR                = 0x35 @0x51
OCR0A                = 0x00 @0x47
OCR0B                = 0x00 @0x48
OCR1A                = 0x2D @0x88
OCR1AH               = 0x00 @0x89
OCR1AL               = 0x2D @0x88
OCR1B                = 0x00 @0x8A
OCR1BH               = 0x00 @0x8B
OCR1BL               = 0x00 @0x8A
OCR2A                = 0x00 @0xB3
OCR2B                = 0x00 @0xB4
OSCCAL               = 0x98 @0x66
PCICR                = 0x00 @0x68
PCIFR                = 0x00 @0x3B
PCMSK0               = 0x00 @0x6B
PCMSK1               = 0x00 @0x6C
PCMSK2               = 0x00 @0x6D
PINB                 = 0x03 @0x23
PINC                 = 0x00 @0x26
PIND                 = 0x01 @0x29
PORTB                = 0x00 @0x25
PORTC                = 0x00 @0x28
PORTD                = 0x00 @0x2B
PRR                  = 0x00 @0x64
SMCR                 = 0x00 @0x53
SP                   = 0xE4 @0x5D
SPCR                 = 0x00 @0x4C
SPDR                 = 0xB9 @0x4E
SPH                  = 0x04 @0x5E
SPL                  = 0xE4 @0x5D
SPMCSR               = 0x00 @0x57
SPSR                 = 0x00 @0x4D
SREG                 = 0x82 @0x5F
TCCR0A               = 0x03 @0x44
TCCR0B               = 0x03 @0x45
TCCR1A               = 0x01 @0x80
TCCR1B               = 0x03 @0x81
TCCR1C               = 0x00 @0x82
TCCR2A               = 0x01 @0xB0
TCCR2B               = 0x04 @0xB1
TCNT0                = 0x24 @0x46
TCNT1                = 0xB8 @0x84
TCNT1H               = 0x00 @0x85
TCNT1L               = 0xBA @0x84
TCNT2                = 0x05 @0xB2
TIFR0                = 0x07 @0x35
TIFR1                = 0x27 @0x36
TIFR2                = 0x07 @0x37
TIMSK0               = 0x00 @0x6E
```

```
TIMSK1                    = 0x00 @0x6F
TIMSK2                    = 0x00 @0x70
TWAMR                     = 0x00 @0xBD
TWAR                      = 0xFE @0xBA
TWBR                      = 0x00 @0xB8
TWCR                      = 0x00 @0xBC
TWDR                      = 0xFF @0xBB
TWSR                      = 0xF8 @0xB9
UBRR0                     = 0x00 @0xC4
UBRR0H                    = 0x00 @0xC5
UBRR0L                    = 0x00 @0xC4
UCSR0A                    = 0x20 @0xC0
UCSR0B                    = 0x00 @0xC1
UCSR0C                    = 0x06 @0xC2
UDR0                      = 0x00 @0xC6
WDTCSR                    = 0x0E @0x60
```

CHAPTER

EIGHT

HARDWARE

http://vusb.wikidot.com/hardware

I use Solution B:

“Solution B: Level conversion on D+ and D- Level conversion with Zener diodes.

Instead of reducing the AVR’s power supply, we can limit the output voltage on D+ and D- with Zener diodes. We recommend 3.6 V low power types, those that look like 1N4148 (usually 500 mW or less). Low power types are required because they have less capacitance and thus cause less distortion on the data lines. And 3.6 V is better than 3.3 V because 3.3 V diodes yield only ca. 2.7 V in conjunction with an 1.5 kâ„¦ (or more exactly 10 kâ„¦) pull-up resistor. With 3.3 V diodes, the device may not be detected reliably.

If you use Zener diodes for level conversion, please measure the voltage levels to make sure that the diodes you have chosen match the requirements.

Advantages of the Zener diode approach:

- Low cost.
- Easy to obtain.
- Entire design can be at 5 V.
- AVR can be clocked at high rates.

Disadvantages:

- Not a clean solution, a compromise between all parameters must be found.
- Zener diodes come with a broad range of characteristics, especially at low currents, results may not be reproducible.
- High currents when sending high-level.
- High level is different for signaling and in idle state because signaling uses high currents to drive the diodes while idle state is driven by a 1.5 kâ„¦ pull-up resistor.”

8.1 Pins

USB pins are defined in `pinconfig.h`:

```
#define USB_CFG_IOPORTNAME      D
/* This is the port where the USB bus is connected. When you configure it to
 * "B", the registers PORTB, PINB and DDRB will be used.
 */
#define USB_CFG_DMINUS_BIT      0
/* This is the bit number in USB_CFG_IOPORT where the USB D- line is connected.
 * This may be any bit in the port.
 */
#define USB_CFG_DPLUS_BIT       2
```

```
/* This is the bit number in USB_CFG_IOPORT where the USB D+ line is connected.
 * This may be any bit in the port. Please note that D+ must also be connected
 * to interrupt pin INT0! [You can also use other interrupts, see section
 * "Optional MCU Description" below, or you can connect D- to the interrupt, as
 * it is required if you use the USB_COUNT_SOF feature. If you use D- for the
 * interrupt, the USB interrupt will also be triggered at Start-Of-Frame
 * markers every millisecond.]
 */
```

Pin mapping depends on board. Example:

http://arduino.cc/hu/Hacking/PinMapping

CHAPTER

NINE

BUILD TESTS

9.1 Results

9.1.1 Arduino version 0022

index	board	Basic	OneWire	Blink
1	atmega8	OK (P:3776 D:152)	OK (P:5964 D:191)	OK (P:6160 D:201)
2	atmega88	OK (P:4086 D:152)	OK (P:6274 D:191)	OK (P:6470 D:201)
3	bt	OK (P:4230 D:154)	OK (P:6484 D:193)	OK (P:6688 D:203)
4	bt328	OK (P:4226 D:154)	OK (P:6480 D:193)	OK (P:6684 D:203)
5	diecimila	OK (P:4230 D:154)	OK (P:6484 D:193)	OK (P:6688 D:203)
6	fio	ERR	ERR	ERR
7	lilypad	ERR	ERR	ERR
8	lilypad328	ERR	ERR	ERR
9	mega	OK (P:5476 D:154)	OK (P:7730 D:193)	OK (P:7934 D:203)
10	mega2560	OK (P:5480 D:154)	OK (P:7734 D:193)	OK (P:7938 D:203)
11	metaboard	OK (P:4230 D:154)	OK (P:6484 D:193)	OK (P:6688 D:203)
12	mini	OK (P:4230 D:154)	OK (P:6484 D:193)	OK (P:6688 D:203)
13	pro	ERR	ERR	ERR
14	pro328	ERR	ERR	ERR
15	pro5v	OK (P:4230 D:154)	OK (P:6484 D:193)	OK (P:6688 D:203)
16	pro5v328	OK (P:4226 D:154)	OK (P:6480 D:193)	OK (P:6684 D:203)
17	uno	OK (P:4226 D:154)	OK (P:6480 D:193)	OK (P:6684 D:203)
18	arduino_OrangutanSVP1284	OK (P:4564 D:156)	OK (P:6818 D:195)	OK (P:7022 D:205)
19	arduino_amber128	ERR	ERR	ERR
20	arduino_android2561	ERR	ERR	ERR
21	arduino_android2561_16	OK (P:5246 D:154)	OK (P:7500 D:193)	OK (P:7704 D:203)
22	arduino_at90can128	OK (P:4776 D:154)	OK (P:7030 D:193)	OK (P:7234 D:203)
23	arduino_at90can32	OK (P:4766 D:154)	OK (P:7020 D:193)	OK (P:7224 D:203)
24	arduino_at90can64	OK (P:4766 D:154)	OK (P:7020 D:193)	OK (P:7224 D:203)
25	arduino_at90usb162	OK (P:4170 D:154)	OK (P:6424 D:193)	OK (P:6628 D:203)
26	arduino_at90usb646	OK (P:4794 D:154)	OK (P:7048 D:193)	OK (P:7252 D:203)
27	arduino_at90usb647	OK (P:4898 D:154)	OK (P:7152 D:193)	OK (P:7356 D:203)
28	arduino_at90usbkey	OK (P:4914 D:156)	OK (P:7168 D:195)	OK (P:7372 D:205)
29	arduino_atmega16	ERR	ERR	ERR
30	arduino_atmega165	ERR	ERR	ERR
31	arduino_atmega3290p	OK (P:4494 D:156)	OK (P:6748 D:195)	OK (P:6952 D:205)
32	arduino_atmega8515	OK (P:3778 D:154)	OK (P:5966 D:193)	OK (P:6162 D:203)
33	arduino_atmega8535	OK (P:3890 D:154)	OK (P:6078 D:193)	OK (P:6274 D:203)
34	arduino_attiny2313	ERR	ERR	ERR
35	arduino_attiny26	ERR	ERR	ERR

Continued on next page

Table 9.1 – continued from previous page

index	board	Basic	OneWire	Blink
36	arduino_attiny45	ERR	ERR	ERR
37	arduino_attiny85	ERR	ERR	ERR
38	arduino_bahbots1284p	ERR	ERR	ERR
39	arduino_butterfly	ERR	ERR	ERR
40	arduino_cerebot_plus	ERR	ERR	ERR
41	arduino_cerebotii	ERR	ERR	ERR
42	arduino_digilent_explorer	ERR	ERR	ERR
43	arduino_duino644	OK (P:4296 D:154)	OK (P:6550 D:193)	OK (P:6754 D:203)
44	arduino_duino644p	OK (P:4336 D:154)	OK (P:6590 D:193)	OK (P:6794 D:203)
45	arduino_gator	OK (P:4308 D:154)	OK (P:6562 D:193)	OK (P:6766 D:203)
46	arduino_illuminato	OK (P:4238 D:154)	OK (P:6492 D:193)	OK (P:6696 D:203)
47	arduino_penguino_avr	OK (P:3964 D:152)	OK (P:6218 D:191)	OK (P:6422 D:201)
48	arduino_teensy2_ser	OK (P:5044 D:154)	OK (P:7298 D:193)	OK (P:7502 D:203)
49	arduino_teensypp2_ser	OK (P:4810 D:156)	OK (P:7064 D:195)	OK (P:7268 D:205)
50	arduino_wiring1281	OK (P:5256 D:154)	OK (P:7510 D:193)	OK (P:7714 D:203)
51	atmega168	OK (P:4228 D:154)	OK (P:6482 D:193)	OK (P:6686 D:203)
52	atmega328	OK (P:4224 D:154)	OK (P:6478 D:193)	OK (P:6682 D:203)
53	atmega48	BIG (P:4164 D:152)	BIG (P:6352 D:191)	BIG (P:6548 D:201)
54	atmega640	OK (P:5518 D:154)	OK (P:7772 D:193)	OK (P:7976 D:203)
55	atmega8	OK (P:3774 D:152)	OK (P:5962 D:191)	OK (P:6158 D:201)
56	atmega88	OK (P:4162 D:154)	OK (P:6350 D:193)	OK (P:6546 D:203)
57	bt	OK (P:4228 D:154)	OK (P:6482 D:193)	OK (P:6686 D:203)
58	bt328	OK (P:4224 D:154)	OK (P:6478 D:193)	OK (P:6682 D:203)
59	diecimila	OK (P:4228 D:154)	OK (P:6482 D:193)	OK (P:6686 D:203)
60	dvk90can1	ERR	ERR	ERR
61	ecavr_atmega32	ERR	ERR	ERR
62	fio	ERR	ERR	ERR
63	lilypad	ERR	ERR	ERR
64	lilypad328	ERR	ERR	ERR
65	mega	OK (P:5474 D:154)	OK (P:7728 D:193)	OK (P:7932 D:203)
66	mega1280stk500v2	OK (P:5474 D:154)	OK (P:7728 D:193)	OK (P:7932 D:203)
67	mega2560stk500v2	OK (P:5478 D:154)	OK (P:7732 D:193)	OK (P:7936 D:203)
68	mini	OK (P:4228 D:154)	OK (P:6482 D:193)	OK (P:6686 D:203)
69	pro	ERR	ERR	ERR
70	pro328	ERR	ERR	ERR
71	pro5v	OK (P:4228 D:154)	OK (P:6482 D:193)	OK (P:6686 D:203)
72	pro5v328	OK (P:4224 D:154)	OK (P:6478 D:193)	OK (P:6682 D:203)
73	stk502	ERR	ERR	ERR
74	stk525	ERR	ERR	ERR
75	stk525_647	ERR	ERR	ERR

9.1.2 Arduino version 0023

index	board	Basic	OneWire	Blink
76	atmega168	OK (P:4230 D:154)	ERR	ERR
77	atmega328	OK (P:4226 D:154)	ERR	ERR
78	atmega8	OK (P:3776 D:152)	ERR	ERR
79	bt	OK (P:4230 D:154)	ERR	ERR
80	bt328	OK (P:4226 D:154)	ERR	ERR
81	diecimila	OK (P:4230 D:154)	ERR	ERR
82	fio	ERR	ERR	ERR
83	lilypad	ERR	ERR	ERR
84	lilypad328	ERR	ERR	ERR
85	mega	OK (P:5476 D:154)	ERR	ERR
86	mega2560	OK (P:5480 D:154)	ERR	ERR
87	mini	OK (P:4230 D:154)	ERR	ERR
88	pro	ERR	ERR	ERR
89	pro328	ERR	ERR	ERR
90	pro5v	OK (P:4230 D:154)	ERR	ERR
91	pro5v328	OK (P:4226 D:154)	ERR	ERR
92	uno	OK (P:4226 D:154)	ERR	ERR

9.1.3 Arduino version 1.0

index	board	Basic	OneWire	Blink
93	atmega168	OK (P:4242 D:154)	ERR	ERR
94	atmega328	OK (P:4238 D:154)	ERR	ERR
95	atmega8	OK (P:3786 D:152)	ERR	ERR
96	bt	OK (P:4242 D:154)	ERR	ERR
97	bt328	OK (P:4238 D:154)	ERR	ERR
98	diecimila	OK (P:4242 D:154)	ERR	ERR
99	ethernet	OK (P:4238 D:154)	ERR	ERR
100	fio	ERR	ERR	ERR
101	lilypad	ERR	ERR	ERR
102	lilypad328	ERR	ERR	ERR
103	mega	OK (P:5292 D:154)	ERR	ERR
104	mega2560	OK (P:5338 D:154)	ERR	ERR
105	mini	OK (P:4242 D:154)	ERR	ERR
106	mini328	OK (P:4238 D:154)	ERR	ERR
107	nano	OK (P:4242 D:154)	ERR	ERR
108	nano328	OK (P:4238 D:154)	ERR	ERR
109	pro	ERR	ERR	ERR
110	pro328	ERR	ERR	ERR
111	pro5v	OK (P:4242 D:154)	ERR	ERR
112	pro5v328	OK (P:4238 D:154)	ERR	ERR
113	uno	OK (P:4238 D:154)	ERR	ERR

9.2 Board configurations

9.2.1 Arduino version 0022

index	package	id	name	MCU
1	arduino	atmega8	Arduino NG or older w/ ATmega8	atmega8

Con

Table 9.2 – continued from previous page

index	package	id	name	MCU
2	arduino	atmega88	atmega88@20000000 programmer:usbasp	atmega8
3	arduino	bt	Arduino BT w/ ATmega168	atmega1
4	arduino	bt328	Arduino BT w/ ATmega328	atmega3
5	arduino	diecimila	Arduino Diecimila, Duemilanove, or Nano w/ ATmega168	atmega1
6	arduino	fio	Arduino Fio	atmega3
7	arduino	lilypad	LilyPad Arduino w/ ATmega168	atmega1
8	arduino	lilypad328	LilyPad Arduino w/ ATmega328	atmega3
9	arduino	mega	Arduino Mega (ATmega1280)	atmega1
10	arduino	mega2560	Arduino Mega 2560	atmega2
11	arduino	metaboard	Metaboard	atmega1
12	arduino	mini	Arduino Mini	atmega1
13	arduino	pro	Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega168	atmega1
14	arduino	pro328	Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega328	atmega3
15	arduino	pro5v	Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega168	atmega1
16	arduino	pro5v328	Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega328	atmega3
17	arduino	uno	Arduino Uno	atmega3
18	arduino-extras	arduino_OrangutanSVP1284	Arduino-Orangutan SVP-1284	atmega1
19	arduino-extras	arduino_amber128	Arduino-Amber 128 14.7456 Mhz	atmega1
20	arduino-extras	arduino_android2561	Arduino-Android 2561 8Mhz	atmega2
21	arduino-extras	arduino_android2561_16	Arduino-Android 2561 16Mhz	atmega2
22	arduino-extras	arduino_at90can128	AT90CAN128 development board NHL (arduino core)	at90can
23	arduino-extras	arduino_at90can32	at90can32 (arduino core)	at90can3
24	arduino-extras	arduino_at90can64	at90can64 (arduino core)	at90can6
25	arduino-extras	arduino_at90usb162	Arduino-at90usb162	at90usb
26	arduino-extras	arduino_at90usb646	Arduino-at90usb646	at90usb6
27	arduino-extras	arduino_at90usb647	Arduino-at90usb647	at90usb6
28	arduino-extras	arduino_at90usbkey	Arduino-at90usbkey	at90usb
29	arduino-extras	arduino_atmega16	Arduino-Atmega16	atmega1
30	arduino-extras	arduino_atmega165	Arduino-Atmega165	atmega1
31	arduino-extras	arduino_atmega3290p	Arduino-Atmega3290p	atmega3
32	arduino-extras	arduino_atmega8515	Arduino-ATmega8515	atmega8
33	arduino-extras	arduino_atmega8535	Arduino-Test-Atmega8535	atmega8
34	arduino-extras	arduino_attiny2313	Arduino-ATtiny2313	attiny23
35	arduino-extras	arduino_attiny26	Arduino-ATtiny26	attiny26
36	arduino-extras	arduino_attiny45	Arduino-ATtiny45	attiny45
37	arduino-extras	arduino_attiny85	Arduino-ATtiny85	attiny85
38	arduino-extras	arduino_bahbots1284p	Arduino-BahBots 1284p	atmega1
39	arduino-extras	arduino_butterfly	Arduino-Butterfly stk500	atmega1
40	arduino-extras	arduino_cerebot_plus	Arduino-Cerebot Plus	atmega2
41	arduino-extras	arduino_cerebotii	Arduino-Cerebot II atemga64	atmega6
42	arduino-extras	arduino_digilent_explorer	Arduino-Digilent I/O Explorer USB	atmega1
43	arduino-extras	arduino_duino644	Arduino-Duino 644	atmega6
44	arduino-extras	arduino_duino644p	Arduino-Duino 644P	atmega6
45	arduino-extras	arduino_gator	Arduino-Rugged Circuits Gator Board	atmega3
46	arduino-extras	arduino_illuminato	Arduino-illuminato	atmega6
47	arduino-extras	arduino_penguino_avr	Arduino-Penguino AVR	atmega3
48	arduino-extras	arduino_teensy2_ser	Arduino-Teensy 2.0 (USB Serial)	atmega3
49	arduino-extras	arduino_teensypp2_ser	Arduino-Teensy++ 2.0 (USB Serial)	at90usb
50	arduino-extras	arduino_wiring1281	Arduino-Wiring 1281	atmega1
51	arduino-extras	atmega168	Arduino NG or older w/ ATmega168	atmega1
52	arduino-extras	atmega328	Arduino Duemilanove or Nano w/ ATmega328	atmega3
53	arduino-extras	atmega48	Arduino Atmega48	atmega4
54	arduino-extras	atmega640	Arduino atmega640	atmega6

Con

Table 9.2 – continued from previous page

index	package	id	name	MCU
55	arduino-extras	atmega8	Arduino NG or older w/ ATmega8	atmega8
56	arduino-extras	atmega88	Atmega88	atmega8
57	arduino-extras	bt	Arduino BT w/ ATmega168	atmega1
58	arduino-extras	bt328	Arduino BT w/ ATmega328	atmega3
59	arduino-extras	diecimila	Arduino Diecimila, Duemilanove, or Nano w/ ATmega168	atmega1
60	arduino-extras	dvk90can1	STK500 w/DVK90CAN1 - AT90can128 (Arduino Core)	at90can
61	arduino-extras	ecavr_atmega32	Embedded market atmega32	atmega3
62	arduino-extras	fio	Arduino Fio	atmega3
63	arduino-extras	lilypad	LilyPad Arduino w/ ATmega168	atmega1
64	arduino-extras	lilypad328	LilyPad Arduino w/ ATmega328	atmega3
65	arduino-extras	mega	Arduino Mega	atmega1
66	arduino-extras	mega1280stk500v2	Arduino Mega1280 stk500v2	atmega1
67	arduino-extras	mega2560stk500v2	Arduino Mega2560 stk500v2	atmega2
68	arduino-extras	mini	Arduino Mini	atmega1
69	arduino-extras	pro	Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega168	atmega1
70	arduino-extras	pro328	Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega328	atmega3
71	arduino-extras	pro5v	Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega168	atmega1
72	arduino-extras	pro5v328	Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega328	atmega3
73	arduino-extras	stk502	STK500 w/STKk502 - ATmega169 (Arduino Core)	atmega1
74	arduino-extras	stk525	STK500 w/STK525 - at90usb1287 (Arduino Core)	at90usb
75	arduino-extras	stk525_647	STK500 w/STK525 - at90usb647 (Arduino Core)	at90usb

9.2.2 Arduino version 0023

index	package	id	name	MCU	F_CPU
76	arduino	atmega168	Arduino NG or older w/ ATmega168	atmega168	16000000L
77	arduino	atmega328	Arduino Duemilanove or Nano w/ ATmega328	atmega328p	16000000L
78	arduino	atmega8	Arduino NG or older w/ ATmega8	atmega8	16000000L
79	arduino	bt	Arduino BT w/ ATmega168	atmega168	16000000L
80	arduino	bt328	Arduino BT w/ ATmega328	atmega328p	16000000L
81	arduino	diecimila	Arduino Diecimila, Duemilanove, or Nano w/ ATmega168	atmega168	16000000L
82	arduino	fio	Arduino Fio	atmega328p	8000000L
83	arduino	lilypad	LilyPad Arduino w/ ATmega168	atmega168	8000000L
84	arduino	lilypad328	LilyPad Arduino w/ ATmega328	atmega328p	8000000L
85	arduino	mega	Arduino Mega (ATmega1280)	atmega1280	16000000L
86	arduino	mega2560	Arduino Mega 2560	atmega2560	16000000L
87	arduino	mini	Arduino Mini	atmega168	16000000L
88	arduino	pro	Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega168	atmega168	8000000L
89	arduino	pro328	Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega328	atmega328p	8000000L
90	arduino	pro5v	Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega168	atmega168	16000000L
91	arduino	pro5v328	Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega328	atmega328p	16000000L
92	arduino	uno	Arduino Uno	atmega328p	16000000L

9.2.3 Arduino version 1.0

index	package	id	name	MCU	F_CPU
93	arduino	atmega168	Arduino NG or older w/ ATmega168	atmega168	16000000L
94	arduino	atmega328	Arduino Duemilanove w/ ATmega328	atmega328p	16000000L
95	arduino	atmega8	Arduino NG or older w/ ATmega8	atmega8	16000000L
96	arduino	bt	Arduino BT w/ ATmega168	atmega168	16000000L
97	arduino	bt328	Arduino BT w/ ATmega328	atmega328p	16000000L
98	arduino	diecimila	Arduino Diecimila or Duemilanove w/ ATmega168	atmega168	16000000L
99	arduino	ethernet	Arduino Ethernet	atmega328p	16000000L
100	arduino	fio	Arduino Fio	atmega328p	8000000L
101	arduino	lilypad	LilyPad Arduino w/ ATmega168	atmega168	8000000L
102	arduino	lilypad328	LilyPad Arduino w/ ATmega328	atmega328p	8000000L
103	arduino	mega	Arduino Mega (ATmega1280)	atmega1280	16000000L
104	arduino	mega2560	Arduino Mega 2560 or Mega ADK	atmega2560	16000000L
105	arduino	mini	Arduino Mini w/ ATmega168	atmega168	16000000L
106	arduino	mini328	Arduino Mini w/ ATmega328	atmega328p	16000000L
107	arduino	nano	Arduino Nano w/ ATmega168	atmega168	16000000L
108	arduino	nano328	Arduino Nano w/ ATmega328	atmega328p	16000000L
109	arduino	pro	Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega168	atmega168	8000000L
110	arduino	pro328	Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega328	atmega328p	8000000L
111	arduino	pro5v	Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega168	atmega168	16000000L
112	arduino	pro5v328	Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega328	atmega328p	16000000L
113	arduino	uno	Arduino Uno	atmega328p	16000000L

CHAPTER

TEN

DOXYGEN DOCUMENTATION

Files