
confduino Documentation

Release 0.0.11

ponty

February 08, 2012

CONTENTS

1	Basic usage	2
2	Installation	3
2.1	General	3
2.2	Ubuntu	3
2.3	Uninstall	3
3	Usage with libraries	4
3.1	Arduino path	4
3.2	List installed libraries	4
3.3	Install new library	6
3.4	Upgrade existing library	6
3.5	Remove existing library	6
3.6	Create menu item “all” for examples	6
3.7	Removing menu item ‘all’	7
4	Usage with boards	8
4.1	List installed boards	8
4.2	List installed MCUs	12
4.3	Install new board	13
4.4	Remove existing board	13
5	Usage with programmers	14
5.1	List installed programmers	14
5.2	Install new programmer	14
5.3	Remove existing programmer	15
6	Examples	16
6.1	Install libraries	16
6.2	Install USBasp programmer	18
6.3	Install STK200 programmer	18
6.4	Install atmega88 board	19
6.5	remove boards	20
6.6	remove libraries	21
7	command line help	23
7.1	lib	23
7.2	board	24
7.3	programmer	24
7.4	version	25
8	API	26
8.1	lib	26
8.2	board	27

8.3	programmer	27
8.4	version	28
9	Indices and tables	29
	Python Module Index	30
	Index	31

confduino

Date February 08, 2012

PDF [confduino.pdf](#)

Contents:

confduino is an [arduino](#) library configurator

Links:

- home: <https://github.com/ponty/confduino>
- documentation: <http://ponty.github.com/confduino>

Features:

- list, install, remove [arduino](#) libraries
- install libraries from internet or local drive
- fix `examples` directory name before installing
- clean library (`.*_*`) before installing
- move examples under `examples` directory
- list, install, remove [arduino](#) programmers
- list, install, remove [arduino](#) boards
- written in python
- crossplatform
- can be used as a python library or as a console program
- unpacker backend: [pyunpack](#)
- downloader backend: [urllib](#)
- some functionality is based on [arscons](#)

Known problems:

- Python 3 is not supported
- tested only on linux
- tested only with arduino version 0022
- some libraries with unusual structure can not be installed
- not all commands have console interface

arduino libraries: <http://www.arduino.cc/en/Reference/Libraries>

BASIC USAGE

install library:

```
>>> from confduino.libinstall import install_lib
>>> install_lib('http://arduino.cc/playground/uploads/Main/PS2Keyboard002.zip')
```

or on console:

```
python -m confduino.libinstall http://arduino.cc/playground/uploads/Main/PS2Keyboard002.zip
```

install a lot of libraries:

```
python -m confduino.libinstall.examples.upgrademany
```

INSTALLATION

2.1 General

- install `arduino`
- install `python`
- install `pip`
- install backends for `pyunpack` (optional)
- install the program:

```
# as root
pip install confduino
```

2.2 Ubuntu

```
sudo apt-get install arduino
sudo apt-get install python-pip
sudo pip install confduino
sudo apt-get install unzip unrar p7zip-full
```

2.3 Uninstall

```
# as root
pip uninstall confduino
```

USAGE WITH LIBRARIES

3.1 Arduino path

If Arduino can not be found at default path, then ARDUINO_HOME environment variable should be set.

on Ubuntu: in ~/.profile:

```
ARDUINO_HOME=~/.opt/arduino
export ARDUINO_HOME
```

Default path:

- Mac: /Applications/Arduino.app/Contents/Resources/Java
- Linux: /usr/share/arduino/

3.2 List installed libraries

From python:

```
>>> from confduino.liblist import libraries
>>> libraries()
['AdvButton', 'ArduinoTestSuite', 'ArduinoUnit', 'AtTouch', 'Bounce', 'Button', 'ByteBuffer', 'Cap
```

From console:

```
$ python -m confduino.liblist
AdvButton
ArduinoTestSuite
ArduinoUnit
AtTouch
Bounce
Button
ByteBuffer
CapSense
Charlieplex
DB
DallasTemperature
DataFlash
DateTime
DateTimeStrings
DigitalToggle
EDB
EEPROM
EasyTransfer
Enerlib
Ethernet
```

EventFuse
 FancyLED
 Firmata
 Flash
 FreqCounter
 FrequencyTimer2
 LED
 LPM11162
 LedControl
 LedDisplay
 LiquidCrystal
 Matrix
 MatrixMath
 Messenger
 Metro
 Morse
 MorseEnDecoder
 MsTimer2
 NewSoftSerial
 OneWire
 PID_v1
 PS2Keyboard
 PS2X_lib
 PString
 PWMServo
 PinChangeInt
 Ping
 Qtouch1Wire
 QueueArray
 QueueList
 SD
 SPI
 SSerial2Mobile
 SerialIP
 SerialManager
 Servo
 SevenSegment
 SimpleMessageSystem
 SoftEasyTransfer
 SoftwareSerial
 Sprite
 StackArray
 StackList
 Streaming
 TButton
 TimedAction
 TimerOne
 TinyGPS
 Tween
 Twitter
 UComms
 UsbDevice
 UsbKeyboard
 WebServer
 WiShield
 Wire
 arduinode
 bassdll
 libcoll
 morse
 multiCameraIrControl
 myprojects
 osa


```
spline  
tmp  
usb  
x10
```

3.3 Install new library

Existing library will not be changed.

From python:

```
>>> from confduino.libinstall import install_lib  
>>> install_lib('http://arduino.cc/playground/uploads/Main/PS2Keyboard002.zip')
```

From console:

```
python -m confduino.libinstall http://arduino.cc/playground/uploads/Main/PS2Keyboard002.zip
```

3.4 Upgrade existing library

Same as install with *replace_existing* option.

From python:

```
>>> from confduino.libinstall import install_lib  
>>> install_lib('http://arduino.cc/playground/uploads/Main/PS2Keyboard002.zip', replace_existing=True)
```

From console:

```
python -m confduino.libinstall http://arduino.cc/playground/uploads/Main/PS2Keyboard002.zip --replace-existing
```

3.5 Remove existing library

From python:

```
>>> from confduino.libremove import remove_lib  
>>> remove_lib('PS2Keyboard')
```

From console:

```
python -m confduino.libremove PS2Keyboard
```

3.6 Create menu item “all” for examples

If you have a lot of libraries and low screen resolution then all menu items under “examples” can not be accessed.

Bug report: “Long menus don’t scroll” (<http://code.google.com/p/arduino/issues/detail?id=426>)

My workaround creates a 2 level deep menu structure without changing other menu items. Symbolic links are used if possible.

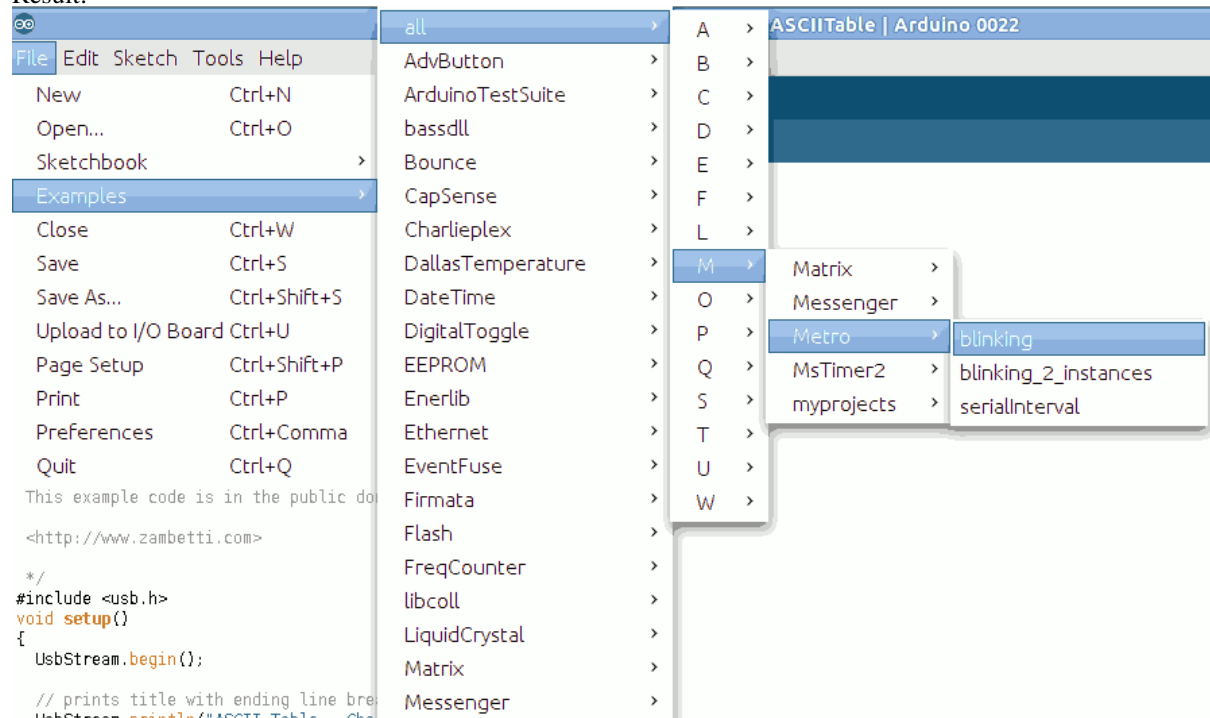
From python:

```
>>> from confduino.exampallcreate import create_examples_all  
>>> create_examples_all()
```

From console:

```
python -m confduino.exampallcreate
```

Result:



3.7 Removing menu item 'all'

From python:

```
>>> from confduino.exampallremove import remove_examples_all
>>> remove_examples_all()
```

From console:

```
python -m confduino.exampallremove
```

USAGE WITH BOARDS

4.1 List installed boards

From python:

```
>>> from confduino.boardlist import boards
>>> boards()
AutoBunch(atmega8=AutoBunch(bootloader=AutoBunch(file='ATmegaBOOT.hex', high_fuses='0xca', lock_b
>>> boards().diecimila.build.f_cpu
'16000000L'
>>> boards()['diecimila']['build']['f_cpu']
'16000000L'
```

From console:

```
$ python -m confduino.boardlist
atmega8
atmega88
bt
bt328
diecimila
fio
lilypad
lilypad328
mega
mega2560
metaboard
mini
pro
pro328
pro5v
pro5v328
uno
```

verbose:

```
$ python -m confduino.boardlist --verbose
{'atmega8': {'bootloader': {'file': 'ATmegaBOOT.hex',
                             'high_fuses': '0xca',
                             'lock_bits': '0x0F',
                             'low_fuses': '0xdf',
                             'path': 'atmega8',
                             'unlock_bits': '0x3F'},
             'build': {'core': 'arduino',
                        'f_cpu': '16000000L',
                        'mcu': 'atmega8'},
             'name': 'Arduino NG or older w/ ATmega8',
             'upload': {'maximum_size': '7168',
                         'protocol': 'stk500',
```

```

        'speed': '19200'}},
'atmega88': {'build': {'core': 'arduino',
                      'f_cpu': '20000000L',
                      'mcu': 'atmega88'},
            'name': 'atmega88@20000000 programmer:usbasp',
            'upload': {'maximum_size': '8192',
                      'using': 'usbasp'}}},
'bt': {'bootloader': {'extended_fuses': '0x00',
                     'file': 'ATmegaBOOT_168.hex',
                     'high_fuses': '0xdd',
                     'lock_bits': '0x0F',
                     'low_fuses': '0xff',
                     'path': 'bt',
                     'unlock_bits': '0x3F'},
      'build': {'core': 'arduino',
                'f_cpu': '16000000L',
                'mcu': 'atmega168'},
      'name': 'Arduino BT w/ ATmega168',
      'upload': {'disable_flushing': 'true',
                 'maximum_size': '14336',
                 'protocol': 'stk500',
                 'speed': '19200'}}},
'bt328': {'bootloader': {'extended_fuses': '0x05',
                        'file': 'ATmegaBOOT_168_atmega328_bt.hex',
                        'high_fuses': '0xd8',
                        'lock_bits': '0x0F',
                        'low_fuses': '0xff',
                        'path': 'bt',
                        'unlock_bits': '0x3F'},
      'build': {'core': 'arduino',
                'f_cpu': '16000000L',
                'mcu': 'atmega328p'},
      'name': 'Arduino BT w/ ATmega328',
      'upload': {'disable_flushing': 'true',
                 'maximum_size': '28672',
                 'protocol': 'stk500',
                 'speed': '19200'}}},
'diecimila': {'bootloader': {'extended_fuses': '0x00',
                             'file': 'ATmegaBOOT_168_diecimila.hex',
                             'high_fuses': '0xdd',
                             'lock_bits': '0x0F',
                             'low_fuses': '0xff',
                             'path': 'atmega',
                             'unlock_bits': '0x3F'},
      'build': {'core': 'arduino',
                'f_cpu': '16000000L',
                'mcu': 'atmega168'},
      'name': 'Arduino Diecimila, Duemilanove, or Nano w/ ATmega168',
      'upload': {'maximum_size': '14336',
                 'protocol': 'stk500',
                 'speed': '19200'}}},
'fio': {'bootloader': {'extended_fuses': '0x05',
                      'file': 'ATmegaBOOT_168_atmega328_pro_8MHz.hex',
                      'high_fuses': '0xDA',
                      'lock_bits': '0x0F',
                      'low_fuses': '0xFF',
                      'path': 'arduino:atmega',
                      'unlock_bits': '0x3F'},
      'build': {'core': 'arduino:arduino',
                'f_cpu': '8000000L',
                'mcu': 'atmega328p'},
      'name': 'Arduino Fio',
      'upload': {'maximum_size': '30720',

```

```

        'protocol': 'stk500',
        'speed': '57600'}},
'lilypad': {'bootloader': {'extended_fuses': '0x00',
                           'file': 'LilyPadBOOT_168.hex',
                           'high_fuses': '0xdd',
                           'lock_bits': '0x0F',
                           'low_fuses': '0xe2',
                           'path': 'lilypad',
                           'unlock_bits': '0x3F'},
            'build': {'core': 'arduino',
                      'f_cpu': '8000000L',
                      'mcu': 'atmega168'},
            'name': 'LilyPad Arduino w/ ATmega168',
            'upload': {'maximum_size': '14336',
                      'protocol': 'stk500',
                      'speed': '19200'}}},
'lilypad328': {'bootloader': {'extended_fuses': '0x05',
                              'file': 'ATmegaBOOT_168_atmega328_pro_8MHz.hex',
                              'high_fuses': '0xDA',
                              'lock_bits': '0x0F',
                              'low_fuses': '0xFF',
                              'path': 'atmega',
                              'unlock_bits': '0x3F'},
              'build': {'core': 'arduino',
                        'f_cpu': '8000000L',
                        'mcu': 'atmega328p'},
              'name': 'LilyPad Arduino w/ ATmega328',
              'upload': {'maximum_size': '30720',
                        'protocol': 'stk500',
                        'speed': '57600'}}},
'mega': {'bootloader': {'extended_fuses': '0xF5',
                        'file': 'ATmegaBOOT_168_atmega1280.hex',
                        'high_fuses': '0xDA',
                        'lock_bits': '0x0F',
                        'low_fuses': '0xFF',
                        'path': 'atmega',
                        'unlock_bits': '0x3F'},
        'build': {'core': 'arduino',
                  'f_cpu': '16000000L',
                  'mcu': 'atmega1280'},
        'name': 'Arduino Mega (ATmega1280)',
        'upload': {'maximum_size': '126976',
                  'protocol': 'stk500',
                  'speed': '57600'}}},
'mega2560': {'bootloader': {'extended_fuses': '0xFD',
                            'file': 'stk500boot_v2_mega2560.hex',
                            'high_fuses': '0xD8',
                            'lock_bits': '0x0F',
                            'low_fuses': '0xFF',
                            'path': 'stk500v2',
                            'unlock_bits': '0x3F'},
            'build': {'core': 'arduino',
                      'f_cpu': '16000000L',
                      'mcu': 'atmega2560'},
            'name': 'Arduino Mega 2560',
            'upload': {'maximum_size': '258048',
                      'protocol': 'stk500v2',
                      'speed': '115200'}}},
'metaboard': {'build': {'core': 'arduino',
                        'f_cpu': '16000000L',
                        'mcu': 'atmega168'},
              'name': 'Metaboard',
              'upload': {'disable_flushing': 'true',
                        'protocol': 'stk500',
                        'speed': '57600'}}},

```

```

        'maximum_size': '14336',
        'protocol': 'usbasp',
        'speed': '19200'}},
'mini': {'bootloader': {'extended_fuses': '0x00',
                        'file': 'ATmegaBOOT_168_ng.hex',
                        'high_fuses': '0xdd',
                        'lock_bits': '0x0F',
                        'low_fuses': '0xff',
                        'path': 'atmega',
                        'unlock_bits': '0x3F'},
        'build': {'core': 'arduino',
                  'f_cpu': '16000000L',
                  'mcu': 'atmega168'},
        'name': 'Arduino Mini',
        'upload': {'maximum_size': '14336',
                  'protocol': 'stk500',
                  'speed': '19200'}},
'pro': {'bootloader': {'extended_fuses': '0x00',
                      'file': 'ATmegaBOOT_168_pro_8MHz.hex',
                      'high_fuses': '0xdd',
                      'lock_bits': '0x0F',
                      'low_fuses': '0xc6',
                      'path': 'atmega',
                      'unlock_bits': '0x3F'},
        'build': {'core': 'arduino',
                  'f_cpu': '8000000L',
                  'mcu': 'atmega168'},
        'name': 'Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega168',
        'upload': {'maximum_size': '14336',
                  'protocol': 'stk500',
                  'speed': '19200'}},
'pro328': {'bootloader': {'extended_fuses': '0x05',
                        'file': 'ATmegaBOOT_168_atmega328_pro_8MHz.hex',
                        'high_fuses': '0xDA',
                        'lock_bits': '0x0F',
                        'low_fuses': '0xFF',
                        'path': 'atmega',
                        'unlock_bits': '0x3F'},
        'build': {'core': 'arduino',
                  'f_cpu': '8000000L',
                  'mcu': 'atmega328p'},
        'name': 'Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega328',
        'upload': {'maximum_size': '30720',
                  'protocol': 'stk500',
                  'speed': '57600'}},
'pro5v': {'bootloader': {'extended_fuses': '0x00',
                        'file': 'ATmegaBOOT_168_diecimila.hex',
                        'high_fuses': '0xdd',
                        'lock_bits': '0x0F',
                        'low_fuses': '0xff',
                        'path': 'atmega',
                        'unlock_bits': '0x3F'},
        'build': {'core': 'arduino',
                  'f_cpu': '16000000L',
                  'mcu': 'atmega168'},
        'name': 'Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega168',
        'upload': {'maximum_size': '14336',
                  'protocol': 'stk500',
                  'speed': '19200'}},
'pro5v328': {'bootloader': {'extended_fuses': '0x05',
                          'file': 'ATmegaBOOT_168_atmega328.hex',
                          'high_fuses': '0xDA',
                          'lock_bits': '0x0F',

```

```

        'low_fuses': '0xFF',
        'path': 'atmega',
        'unlock_bits': '0x3F'},
    'build': {'core': 'arduino',
              'f_cpu': '16000000L',
              'mcu': 'atmega328p'},
    'name': 'Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega328',
    'upload': {'maximum_size': '30720',
               'protocol': 'stk500',
               'speed': '57600'}}},
    'uno': {'bootloader': {'extended_fuses': '0x05',
                          'file': 'optiboot_atmega328.hex',
                          'high_fuses': '0xde',
                          'lock_bits': '0x0F',
                          'low_fuses': '0xff',
                          'path': 'optiboot',
                          'unlock_bits': '0x3F'},
            'build': {'core': 'arduino',
                      'f_cpu': '16000000L',
                      'mcu': 'atmega328p'},
            'name': 'Arduino Uno',
            'upload': {'maximum_size': '32256',
                      'protocol': 'stk500',
                      'speed': '115200'}}}}

```

4.2 List installed MCUs

From python:

```

>>> from confduino.mculist import mcus
>>> mcus()
['at90can128', 'at90can32', 'at90can64', 'at90usb1286', 'at90usb1287', 'at90usb162', 'at90usb646',

```

From console:

```

$ python -m confduino.mculist
at90can128
at90can32
at90can64
at90usb1286
at90usb1287
at90usb162
at90usb646
at90usb647
atmega128
atmega1280
atmega1281
atmega1284p
atmega16
atmega165
atmega165p
atmega168
atmega169
atmega2560
atmega2561
atmega32
atmega324p
atmega328p
atmega3290p
atmega32u4
atmega48

```

```

atmega64
atmega640
atmega644
atmega644p
atmega645
atmega8
atmega8515
atmega8535
atmega88
atmega88p
atmega8u2
attiny2313
attiny26
attiny45
attiny85
attiny861

```

4.3 Install new board

Existing board will not be changed.

From python:

```

from confduino.boardinstall import install_board
from confduino.util import AutoBunch
from entrypoint2 import entrypoint

@entrypoint
def install(id='atmega88', mcu='atmega88', f_cpu=20000000, upload='usbasp', core='arduino', replace_existing=True):
    'install atmega88 board'
    board = AutoBunch()
    board.name = '{mcu}@{f_cpu} programmer:{upload}'.format(mcu=mcu, f_cpu=f_cpu, upload=upload)

    board.upload.using = upload
    board.upload.maximum_size = 8*1024

    board.build.mcu = mcu
    board.build.f_cpu = str(f_cpu) + 'L'
    board.build.core = core

    install_board(id, board, replace_existing=replace_existing)

```

console is not implemented

4.4 Remove existing board

From python:

```

>>> from confduino.boardremove import remove_board
>>> remove_board('diecimila')

```

From console:

```
python -m confduino.boardremove diecimila
```


USAGE WITH PROGRAMMERS

5.1 List installed programmers

From python:

```
>>> from confduino.proglist import programmers
>>> programmers()
AutoBunch(arduinoisp=AutoBunch(communication='serial', name='Arduino as ISP', protocol='stk500v1',
>>> programmers().arduinoisp.speed
'19200'
>>> programmers()['arduinoisp']['speed']
'19200'
```

From console:

```
$ python -m confduino.proglist
arduinoisp
avrisp
avrispmkii
parallel
stk200
usbasp
usbtinyisp
```

verbose:

```
$ python -m confduino.proglist --verbose
{'arduinoisp': {'communication': 'serial',
                'name': 'Arduino as ISP',
                'protocol': 'stk500v1',
                'speed': '19200'},
 'avrisp': {'communication': 'serial',
            'name': 'AVR ISP',
            'protocol': 'stk500v1'},
 'avrispmkii': {'communication': 'usb',
                'name': 'AVRISP mkII',
                'protocol': 'stk500v2'},
 'parallel': {'force': 'true',
              'name': 'Parallel Programmer',
              'protocol': 'dapa'},
 'stk200': {'force': 'true', 'name': 'STK200', 'protocol': 'dapa'},
 'usbasp': {'communication': 'usb', 'name': 'USBasp', 'protocol': 'usbasp'},
 'usbtinyisp': {'name': 'USBtinyISP', 'protocol': 'usbtiny'}}
```

5.2 Install new programmer

From python:

```
from confduino.proginstall import install_programmer
from confduino.util import AutoBunch
from entrypoint2 import entrypoint

@entrypoint
def install(replace_existing=False):
    'install usbasp programmer'
    usbasp = AutoBunch()
    usbasp.name = 'USBasp'
    usbasp.communication = 'usb'
    usbasp.protocol = 'usbasp'

    install_programmer('usbasp', usbasp, replace_existing=replace_existing)
```

console is not implemented

5.3 Remove existing programmer

From python:

```
>>> from confduino.progremove import remove_programmer
>>> remove_programmer('parallel')
```

From console:

```
python -m confduino.progremove parallel
```

EXAMPLES

6.1 Install libraries

Many libraries are upgraded in examples/upgrademany.py, this can be started:

```
python -m confduino.examples.upgrademany
```

Code:

```
from confduino import exampallcreate
from confduino.libinstall import install_lib
from confduino.util import ConfduinoError
from entrypoint2 import entrypoint

@entrypoint
def upgrade_many(upgrade=True, create_examples_all=True):
    '''upgrade many libs

    source: http://arduino.cc/playground/Main/LibraryList

    you can set your arduino path if it is not default
    os.environ['ARDUINO_HOME'] = '/home/...'
    '''
    urls=set()
    def inst(url):
        print 'upgrading ' + url
        assert url not in urls
        urls.add(url)
        try:
            lib = install_lib(url, upgrade)
            print ' -> ', lib
        except ConfduinoError as e:
            print e

    #####
    # github.com
    #####
    inst('https://github.com/madscil016/Arduino-EasyTransfer/zipball/master')
    inst('https://github.com/madscil016/Arduino-SoftEasyTransfer/zipball/master')
    inst('https://github.com/madscil016/Arduino-PS2X/zipball/master')
    # inst('http://github.com/wimleers/flexitimer2/zipball/v1.0')# can't install
    inst('https://github.com/kerinin/arduino-splines/zipball/master')
    inst('https://github.com/asynclabs/WiShield/zipball/master')
    inst('https://github.com/asynclabs/dataflash/zipball/master')
    inst('https://github.com/slugmobile/AtTouch/zipball/master')
    inst('https://github.com/carlynorama/Arduino-Library-Button/zipball/master')
```

```

inst('https://github.com/carlynoroma/Arduino-Library-FancyLED/zipball/master')
inst('https://github.com/markfickett/arduinomorse/zipball/master')

#####
# arduiniana.org
#####
# TODO: how to get latest version??
inst('http://arduiniana.org/PString/PString2.zip')
inst('http://arduiniana.org/Flash/Flash3.zip')
inst('http://arduiniana.org/NewSoftSerial/NewSoftSerial10c.zip')
inst('http://arduiniana.org/Streaming/Streaming4.zip')
inst('http://arduiniana.org/PWMServo/PWMServo.zip')
inst('http://arduiniana.org/TinyGPS/TinyGPS10.zip')

#####
# google
#####
# TODO: how to get latest version??
# parse http://code.google.com/p/arduino-pinchangeint/downloads/list

# inst('http://rogue-code.googlecode.com/files/Arduino-Library-Tone.zip') # already in core!
inst('http://arduino-playground.googlecode.com/files/LedDisplay03.zip')
inst('http://sserial2mobile.googlecode.com/files/SSerial2Mobile-1.1.0.zip')
inst('http://webduino.googlecode.com/files/webduino-1.4.1.zip') # can't install
inst('http://arduino-pid-library.googlecode.com/files/PID_v1.0.1.zip')
inst('http://ideoarduinolibraries.googlecode.com/files/Qtouch1Wire.zip')
inst('http://arduino-timerone.googlecode.com/files/TimerOne-v8.zip')
inst('http://arduinounit.googlecode.com/files/arduinounit-1.4.2.zip')
inst('http://arduinode.googlecode.com/files/arduinode_0.1.zip')
inst('http://arduino-edb.googlecode.com/files/EDB_r7.zip')
inst('http://arduino-dblib.googlecode.com/files/DB.zip')
inst('http://morse-endecoder.googlecode.com/files/Morse_EnDecoder_2010.12.06.tar.gz')
inst('http://arduino-pinchangeint.googlecode.com/files/PinChangeInt.zip')

#####
# others
#####
inst('http://download.milesburton.com/Arduino/MaximTemperature/DallasTemperature_370Beta.zip')
inst('http://www.pjrc.com/teensy/arduino_libraries/OneWire.zip')
inst('http://interface.khm.de/wp-content/uploads/2009/01/FreqCounter1.zip')
# inst('http://www.state-machine.com/arduino/qp_arduino.zip') # too big
inst('ftp://momjian.us/pub/arduino/TButton.zip') # AdvButton is better
inst('http://johnmchilton.com/media/UComms.zip')
inst('http://www.shikadi.net/files/arduino/SerialIP-1.0.zip')
inst('http://siggiorn.com/wp-content/uploads/libraries/ArduinoByteBuffer.zip')
inst('http://siggiorn.com/wp-content/uploads/libraries/ArduinoSerialManager.zip')
inst('http://arduino-tweet.appspot.com/Library-Twitter-1.2.2.zip')
# inst('http://gkaindl.com/php/download.php?key=ArduinoEthernet') # can't install
inst('http://geekcowboy.net/downloads/x10.zip')
inst('http://sebastian.setz.name/wp-content/uploads/2011/01/multiCameraIrControl_1-5.zip')
inst('http://www.familjenlinder.se/Morse.7z')
inst('http://www.pjrc.com/teensy/arduino_libraries/FrequencyTimer2.zip')
inst('http://alexandre.quessey.net/static/avr/Tween_01.zip')
inst('http://www.lpelettronica.it/images/stories/LPM11162_images/Arduino/LPM11162_ArduinoLib_')

#####
# arduino.cc
#####
inst('http://arduino.cc/playground/uploads/Main/PS2Keyboard002.zip')
inst('http://arduino.cc/playground/uploads/Code/Metro.zip')
inst('http://www.arduino.cc/playground/uploads/Main/MsTimer2.zip')

```

```
# inst('http://www.arduino.cc/playground/uploads/Code/Time.zip')# can't install
inst('http://arduino.cc/playground/uploads/Main/LedControl.zip')
# inst('http://www.arduino.cc/playground/uploads/Code/ks0108GLCD.zip')# can't install
inst('http://arduino.cc/playground/uploads/Code/Bounce.zip')
inst('http://arduino.cc/playground/uploads/Main/CapacitiveSense003.zip')
inst('http://arduino.cc/playground/uploads/Main/PinChangeInt.zip')
# inst('http://arduino.cc/playground/uploads/Code/TimerThree.zip')# can't install
inst('http://arduino.cc/playground/uploads/Code/TimedAction-1_6.zip')
# inst('http://www.arduino.cc/playground/uploads/Code/Time.zip')# can't install
inst('http://arduino.cc/playground/uploads/Code/EventFuse.zip')
inst('http://arduino.cc/playground/uploads/Code/Charlieplex.zip')
inst('http://arduino.cc/playground/uploads/Code/DigitalToggle.zip')
inst('http://arduino.cc/playground/uploads/Code/Enerlib.zip')

inst('http://arduino.cc/playground/uploads/Code/AdvButton_11.zip')
#inst('http://arduino.cc/playground/uploads/Code/AdvButton.zip') # old version

# inst('http://arduino.cc/playground/uploads/Code/SerialDebugger.zip') # can't install
inst('http://arduino.cc/playground/uploads/Code/MatrixMath.zip')

inst('http://arduino.cc/playground/uploads/Code/StackArray.zip')
inst('http://arduino.cc/playground/uploads/Code/StackList.zip')
inst('http://arduino.cc/playground/uploads/Code/QueueArray.zip')
inst('http://arduino.cc/playground/uploads/Code/QueueList.zip')
inst('http://arduino.cc/playground/uploads/Code/Ping-1_3.zip')
inst('http://www.arduino.cc/playground/uploads/Code/LED.zip')

# inst('')
if create_examples_all:
    print 'create "all" menu item'
    exampallcreate.create_examples_all()
print 'install finished'
```

6.2 Install USBasp programmer

```
python -m confduino.examples.usbasp
```

Code:

```
from confduino.progininstall import install_programmer
from confduino.util import AutoBunch
from entrypoint2 import entrypoint

@entrypoint
def install(replace_existing=False):
    'install usbasp programmer'
    usbasp = AutoBunch()
    usbasp.name = 'USBasp'
    usbasp.communication = 'usb'
    usbasp.protocol = 'usbasp'

    install_programmer('usbasp', usbasp, replace_existing=replace_existing)
```

6.3 Install STK200 programmer

```
python -m confduino.examples.stk200
```

Code:

```

from confduino.proginstall import install_programmer
from confduino.util import AutoBunch
from entrypoint2 import entrypoint

@entrypoint
def install(replace_existing=False):
    'install stk200 programmer'
    bunch = AutoBunch()
    bunch.name = 'STK200'
    bunch.protocol = 'stk200'
    #bunch.force = 'true'
    # bunch.delay=200

    install_programmer('stk200', bunch, replace_existing=replace_existing)

```

6.4 Install atmega88 board

```
python -m confduino.examples.atmega88
```

Code:

```

from confduino.boardinstall import install_board
from confduino.util import AutoBunch
from entrypoint2 import entrypoint

@entrypoint
def install(id='atmega88', mcu='atmega88', f_cpu=20000000, upload='usbasp', core='arduino', replace_existing=False):
    'install atmega88 board'
    board = AutoBunch()
    board.name = '{mcu}@{f_cpu} programmer:{upload}'.format(mcu=mcu, f_cpu=f_cpu, upload=upload)

    board.upload.using = upload
    board.upload.maximum_size = 8*1024

    board.build.mcu = mcu
    board.build.f_cpu = str(f_cpu) + 'L'
    board.build.core = core

    install_board(id, board, replace_existing=replace_existing)

```

options:

```

$ python -m confduino.examples.atmega88 --help
usage: atmega88.py [-h] [-i ID] [-m MCU] [-f F_CPU] [-u UPLOAD] [-c CORE] [-r]
                  [--debug]

```

```
install atmega88 board
```

optional arguments:

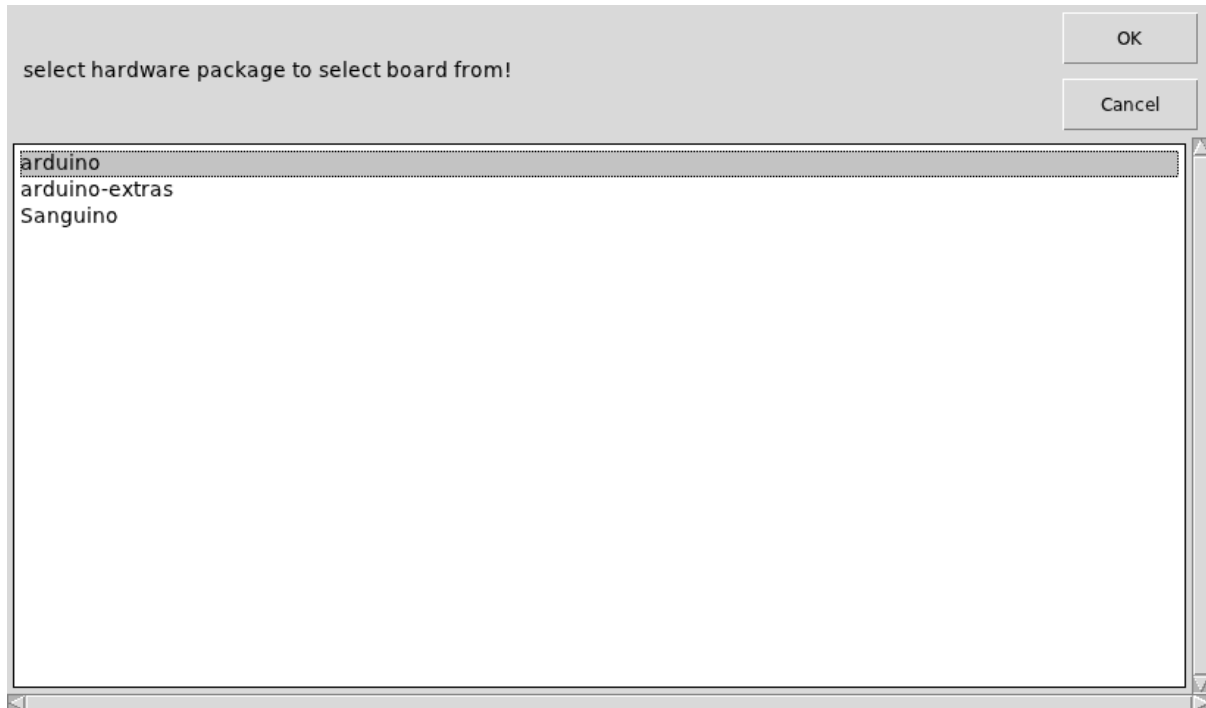
```

-h, --help            show this help message and exit
-i ID, --id ID
-m MCU, --mcu MCU
-f F_CPU, --f-cpu F_CPU
-u UPLOAD, --upload UPLOAD
-c CORE, --core CORE
-r, --replace-existing
--debug              set logging level to DEBUG

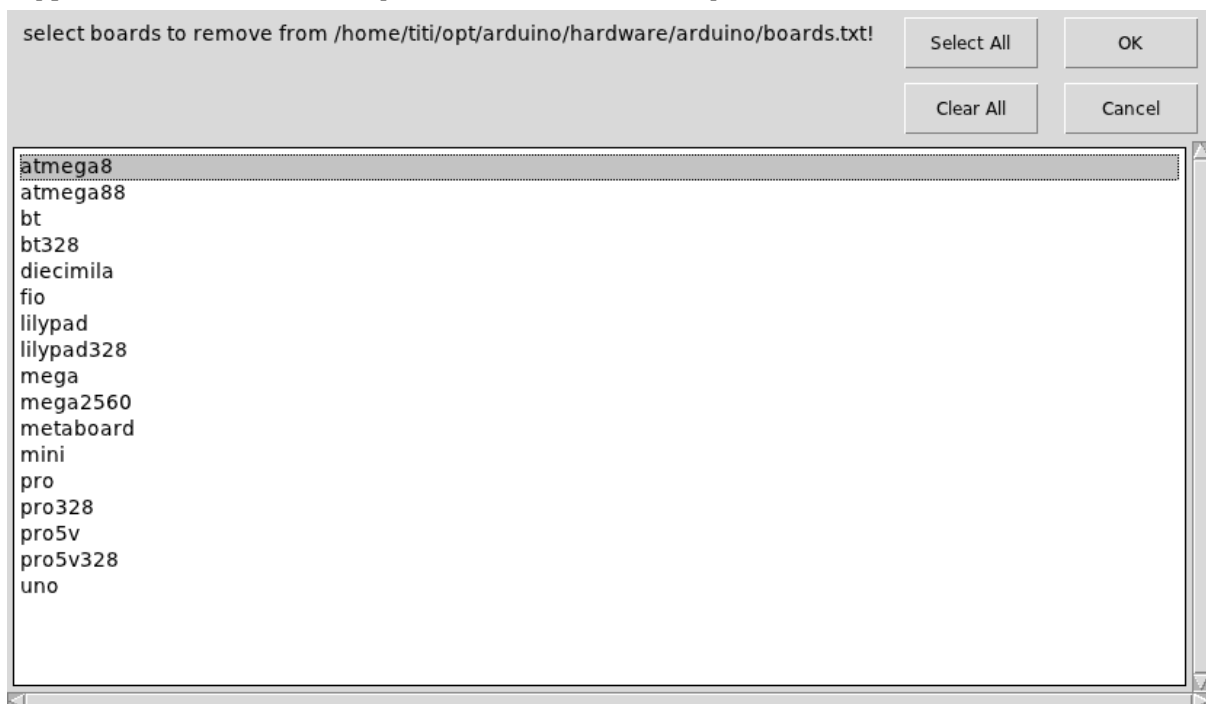
```

6.5 remove boards

```
$ python -m confduino.examples.remove_boards
```



```
$ python -m confduino.examples.remove_boards --hwpack arduino
```



Code:

```
from confduino.boardlist import boards, boards_txt, board_names
from confduino.boardremove import remove_board
from confduino.hwpacklist import hwpack_names
from entrypoint2 import entrypoint
import psdialog
```

```
@entrypoint
def remove_boards_gui(hwpack=''):
    'remove boards by GUI'
    if not hwpack:
        if len(hwpack_names()) > 1:
            hwpack = psidialogs.choice(hwpack_names(),
                                      'select hardware package to select board from!',
                                      title='select')

        else:
            hwpack = hwpack_names()[0]
    print hwpack, 'selected'

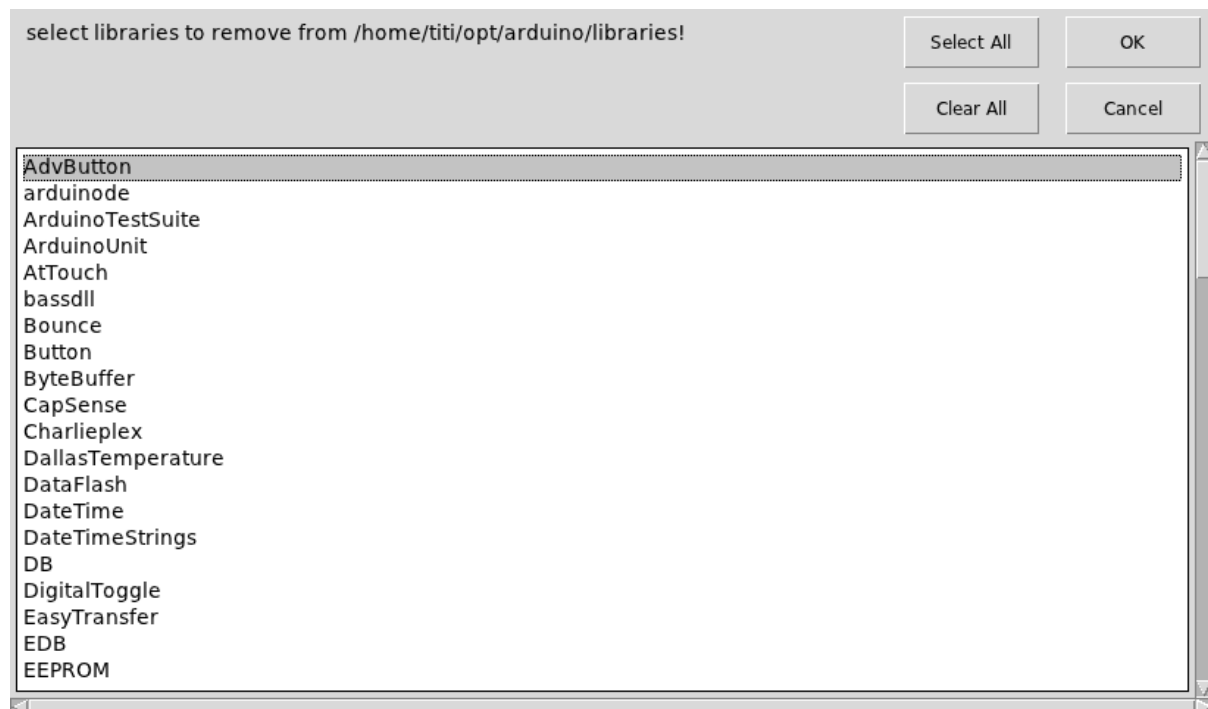
    if hwpack:
        sel = psidialogs.multi_choice(board_names(hwpack),
                                      'select boards to remove from %s!' % boards_txt(hwpack),
                                      title='remove boards')

        print sel, 'selected'

        if sel:
            for x in sel:
                remove_board(x)
            print x + ' was removed'
```

6.6 remove libraries

```
$ python -m confduino.examples.remove_libraries
```



Code:

```
from confduino.liblist import libraries, libraries_dir
from confduino.libremove import remove_lib
from entrypoint2 import entrypoint
import psidialogs
```

```
@entrypoint
```



```
def gui():
    'remove libraries by GUI'

    sel = psidialogs.multi_choice(libraries(),
                                  'select libraries to remove from %s!' % libraries_dir(),
                                  title='remove boards')

    print sel, 'selected'

    if sel:
        if psidialogs.ask_yes_no('Do you really want to remove selected libraries?\n'+'\n'.join(s
        for x in sel:
            remove_lib(x)
            print x + ' was removed')
```

COMMAND LINE HELP

7.1 lib

7.1.1 list

```
$ python -m confduino.liblist --help
usage: liblist.py [-h] [--debug]

print installed arduino libraries

optional arguments:
  -h, --help  show this help message and exit
  --debug     set logging level to DEBUG
```

7.1.2 install

```
$ python -m confduino.libinstall --help
usage: libinstall.py [-h] [-r] [--debug] url

install library from web or local files system

positional arguments:
  url                web address or file path

optional arguments:
  -h, --help          show this help message and exit
  -r, --replace-existing
                        bool
  --debug             set logging level to DEBUG
```

7.1.3 remove

```
$ python -m confduino.libremove --help
usage: libremove.py [-h] [--debug] lib_name

remove library

positional arguments:
  lib_name    library name (e.g. 'PS2Keyboard')

optional arguments:
  -h, --help  show this help message and exit
  --debug     set logging level to DEBUG
```

7.2 board

7.2.1 list

```
$ python -m confduino.boardlist --help
usage: boardlist.py [-h] [--hwpack HWPACK] [-v] [--debug]
```

print boards from boards.txt

optional arguments:

```
-h, --help      show this help message and exit
--hwpack HWPACK
-v, --verbose
--debug        set logging level to DEBUG
```

7.2.2 install

not implemented

7.2.3 remove

```
$ python -m confduino.boardremove --help
usage: boardremove.py [-h] [--debug] board_id
```

remove board

positional arguments:

```
board_id      board id (e.g. 'diecimila')
```

optional arguments:

```
-h, --help      show this help message and exit
--debug        set logging level to DEBUG
```

7.3 programmer

7.3.1 list

```
$ python -m confduino.proglist --help
usage: proglist.py [-h] [-v] [--debug]
```

print programmers from programmers.txt

optional arguments:

```
-h, --help      show this help message and exit
-v, --verbose
--debug        set logging level to DEBUG
```

7.3.2 install

not implemented

7.3.3 remove

```
$ python -m confduino.progremove --help
usage: progremove.py [-h] [--debug] programmer_id

remove programmer

positional arguments:
  programmer_id  programmer id (e.g. 'avrisp')

optional arguments:
  -h, --help      show this help message and exit
  --debug         set logging level to DEBUG
```

7.4 version

```
$ python -m confduino.version --help
usage: version.py [-h] [--debug] [--version]

print arduino version

optional arguments:
  -h, --help      show this help message and exit
  --debug         set logging level to DEBUG
  --version       show program's version number and exit
```

API

8.1 lib

```
confduino.liblist.lib_dir(lib)  
    return library directory  
    $ARDUINO/libraries/$LIB  
  
confduino.liblist.lib_example_dir(lib, example)  
    return library example directory  
    $ARDUINO/libraries/$LIB/examples/$EXAMPLE  
  
confduino.liblist.lib_examples(lib)  
    return library examples  
    EXAMPLE1,EXAMPLE2,..  
  
confduino.liblist.lib_examples_dir(lib)  
    return library examples directory  
    $ARDUINO/libraries/$LIB/examples  
  
confduino.liblist.libraries()  
    return installed library names  
  
confduino.liblist.libraries_dir()  
    return library root path  
    $ARDUINO/libraries  
  
confduino.liblist.print_libraries()  
    print installed arduino libraries  
  
confduino.libinstall.find_lib_dir(root)  
    search for lib dir under root  
  
confduino.libinstall.fix_examples_dir(lib_dir)  
    rename examples dir to examples  
  
confduino.libinstall.install_lib(url, replace_existing=False)  
    install library from web or local files system
```

Parameters

- **url** – web address or file path
- **replace_existing** – bool

Return type

 None

```
confduino.libinstall.move_examples(root, lib_dir)  
    find examples not under lib dir, and move into examples
```

`confduino.libremove.remove_lib (lib_name)`
remove library

Parameters `lib_name` – library name (e.g. 'PS2Keyboard')

Return type None

8.2 board

`confduino.boardlist.board_names (hwpack='arduino')`
return installed board names

`confduino.boardlist.boards (hwpack='arduino')`
read boards from boards.txt

Parameters `core_package` – 'all','arduino',..

`confduino.boardlist.boards_txt (hwpack='arduino')`
path of boards.txt

`confduino.boardlist.print_boards (hwpack='arduino', verbose=False)`
print boards from boards.txt

`confduino.boardinstall.install_board (board_id, board_options, hwpack='arduino', replace_existing=False)`
install board in boards.txt

Parameters

- `board_id` – string identifier
- `board_options` – dict like
- `replace_existing` – bool

Return type None

`confduino.boardremove.remove_board (board_id)`
remove board

Parameters `board_id` – board id (e.g. 'diecimila')

Return type None

8.3 programmer

`confduino.proglist.print_programmers (verbose=False)`
print programmers from programmers.txt

`confduino.proglist.programmer_names (hwpack='arduino')`
return installed board names

`confduino.proglist.programmers ()`
read programmers from programmers.txt

`confduino.proglist.programmers_txt ()`
path of programmers.txt

`confduino.proginstall.install_programmer (programmer_id, programmer_options, replace_existing=False)`
install programmer in programmers.txt

Parameters

- `programmer_id` – string identifier

- **programmer_options** – dict like
- **replace_existing** – bool

Return type None

`confduino.progremove.remove_programmer(programmer_id)`
 remove programmer

Parameters **programmer_id** – programmer id (e.g. ‘avrisp’)

Return type None

8.4 version

`confduino.version.print_version()`
 print arduino version

example: 0022

`confduino.version.version()`
 return version

example: 0022

`confduino.version.version_txt()`
 return version.txt path

\$ARDUINO/lib/version.txt

INDICES AND TABLES

- *genindex*
- *modindex*
- *search*

PYTHON MODULE INDEX

C

- `confduino.boardinstall`, 27
- `confduino.boardlist`, 27
- `confduino.boardremove`, 27
- `confduino.libinstall`, 26
- `confduino.liblist`, 26
- `confduino.libremove`, 26
- `confduino.proginstall`, 27
- `confduino.proglist`, 27
- `confduino.progremove`, 28
- `confduino.version`, 28

INDEX

B

board_names() (in module confduino.boardlist), 27
boards() (in module confduino.boardlist), 27
boards_txt() (in module confduino.boardlist), 27

C

confduino.boardinstall (module), 27
confduino.boardlist (module), 27
confduino.boardremove (module), 27
confduino.libinstall (module), 26
confduino.liblist (module), 26
confduino.libremove (module), 26
confduino.proginstall (module), 27
confduino.proglist (module), 27
confduino.progremove (module), 28
confduino.version (module), 28

F

find_lib_dir() (in module confduino.libinstall), 26
fix_examples_dir() (in module confduino.libinstall), 26

I

install_board() (in module confduino.boardinstall), 27
install_lib() (in module confduino.libinstall), 26
install_programmer() (in module confduino.proginstall), 27

L

lib_dir() (in module confduino.liblist), 26
lib_example_dir() (in module confduino.liblist), 26
lib_examples() (in module confduino.liblist), 26
lib_examples_dir() (in module confduino.liblist), 26
libraries() (in module confduino.liblist), 26
libraries_dir() (in module confduino.liblist), 26

M

move_examples() (in module confduino.libinstall), 26

P

print_boards() (in module confduino.boardlist), 27
print_libraries() (in module confduino.liblist), 26
print_programmers() (in module confduino.proglist), 27
print_version() (in module confduino.version), 28

programmer_names() (in module confduino.proglist), 27

programmers() (in module confduino.proglist), 27
programmers_txt() (in module confduino.proglist), 27

R

remove_board() (in module confduino.boardremove), 27
remove_lib() (in module confduino.libremove), 26
remove_programmer() (in module confduino.progremove), 28

V

version() (in module confduino.version), 28
version_txt() (in module confduino.version), 28