
pihat

Release 0.0.6

Dec 11, 2019

Contents:

1	pihat package	3
1.1	Subpackages	3
1.2	Module contents	11
2	Indices and tables	13
	Python Module Index	15
	Index	17

The Raspberry Pi [HAT specification](#) defines the existence of an [ID EEPROM](#) that is connected to pins 27 and 28 of the [40-way connector](#). This ID EEPROM contains a machine-readable description of the HAT hardware in a form that allows the Pi to automatically configure the GPIOs and load appropriate device drivers.

The `pihat` library provides a mechanism for easily extracting and manipulating the contents of the ID EEPROM. For example:

```
from pihat.eeprom import EepromDevice

with EepromDevice() as eeprom:
    print("Board UUID is %s" % eeprom.uuid)
    print("Board uses %d GPIOs" %
          sum(x.used for x in eeprom.pins))
```


1.1 Subpackages

1.1.1 pihat.eeprom package

Submodules

pihat.eeprom.cli module

Pi Hat command line interface

```
class pihat.eeprom.cli.Command(args: dataclasses.InitVar)
    Bases: object
    Extract/merge/replace Pi Hat ID EEPROM
    dump()
        Dump existing EEPROM contents to stdout
    eeprom(**kwargs)
        Construct EEPROM object
    execute()
        Execute command
    extract()
        Extract existing EEPROM content
    merge()
        Merge with existing EEPROM content
    parser()
        Argument parser
    read_desc()
        Read description
```

```
read_fdt ()  
    Read devicetree  
  
replace ()  
    Replace existing EEPROM content  
  
write_desc (desc)  
    Write description  
  
write_fdt (fdt)  
    Write devicetree
```

pihat.eeprom.constants module

Pi Hat EEPROM constants

```
class pihat.eeprom.constants.EepromSignature  
    Bases: enum.IntFlag  
  
    Magic signature  
  
    RPI = 1766862162  
  
class pihat.eeprom.constants.EepromVersion  
    Bases: enum.IntFlag  
  
    Structure version  
  
    V1 = 1  
  
class pihat.eeprom.constants.EepromAtomType  
    Bases: enum.IntFlag  
  
    Atom type  
  
    CUSTOM = 4  
  
    DTBO = 3  
  
    GPIO = 2  
  
    INFO = 1  
  
class pihat.eeprom.constants.EepromGpioDrive  
    Bases: enum.IntEnum  
  
    GPIO bank drive  
  
    DEFAULT = 0  
  
    MA_10 = 5  
  
    MA_12 = 6  
  
    MA_14 = 7  
  
    MA_16 = 8  
  
    MA_2 = 1  
  
    MA_4 = 2  
  
    MA_6 = 3  
  
    MA_8 = 4  
  
    RESERVED_10 = 10
```



```

    RESERVED_11 = 11
    RESERVED_12 = 12
    RESERVED_13 = 13
    RESERVED_14 = 14
    RESERVED_15 = 15
    RESERVED_9 = 9

class pihat.eeprom.constants.EepromGpioSlew
    Bases: enum.IntEnum
    GPIO bank slew
    DEFAULT = 0
    LIMITED = 1
    RESERVED = 3
    UNLIMITED = 2

class pihat.eeprom.constants.EepromGpioHysteresis
    Bases: enum.IntEnum
    GPIO bank hysteresis
    DEFAULT = 0
    DISABLED = 1
    ENABLED = 2
    RESERVED = 3

class pihat.eeprom.constants.EepromGpioBackPower
    Bases: enum.IntEnum
    Board back powering
    MA_1300 = 1
    MA_2000 = 2
    NONE = 0
    RESERVED = 3

class pihat.eeprom.constants.EepromGpioFunction
    Bases: enum.IntEnum
    GPIO pin function
    ALT0 = 4
    ALT1 = 5
    ALT2 = 6
    ALT3 = 7
    ALT4 = 3
    ALT5 = 2
    INPUT = 0

```

OUTPUT = 1

class pihat.eeprom.constants.EepromGpioPull

Bases: enum.IntEnum

GPIO pull direction

DEFAULT = 0

DOWN = 2

NONE = 3

UP = 1

pihat.eeprom.crc module

CRC-16 algorithm

```
pihat.eeprom.crc.crc16(data, crc=0, table=array('H', [0, 49345, 49537, 320, 49921, 960, 640,
49729, 50689, 1728, 1920, 51009, 1280, 50625, 50305, 1088, 52225, 3264,
3456, 52545, 3840, 53185, 52865, 3648, 2560, 51905, 52097, 2880, 51457,
2496, 2176, 51265, 55297, 6336, 6528, 55617, 6912, 56257, 55937, 6720,
7680, 57025, 57217, 8000, 56577, 7616, 7296, 56385, 5120, 54465, 54657,
5440, 55041, 6080, 5760, 54849, 53761, 4800, 4992, 54081, 4352, 53697,
53377, 4160, 61441, 12480, 12672, 61761, 13056, 62401, 62081, 12864,
13824, 63169, 63361, 14144, 62721, 13760, 13440, 62529, 15360, 64705,
64897, 15680, 65281, 16320, 16000, 65089, 64001, 15040, 15232, 64321,
14592, 63937, 63617, 14400, 10240, 59585, 59777, 10560, 60161, 11200,
10880, 59969, 60929, 11968, 12160, 61249, 11520, 60865, 60545, 11328,
58369, 9408, 9600, 58689, 9984, 59329, 59009, 9792, 8704, 58049, 58241,
9024, 57601, 8640, 8320, 57409, 40961, 24768, 24960, 41281, 25344,
41921, 41601, 25152, 26112, 42689, 42881, 26432, 42241, 26048, 25728,
42049, 27648, 44225, 44417, 27968, 44801, 28608, 28288, 44609, 43521,
27328, 27520, 43841, 26880, 43457, 43137, 26688, 30720, 47297, 47489,
31040, 47873, 31680, 31360, 47681, 48641, 32448, 32640, 48961, 32000,
48577, 48257, 31808, 46081, 29888, 30080, 46401, 30464, 47041, 46721,
30272, 29184, 45761, 45953, 29504, 45313, 29120, 28800, 45121, 20480,
37057, 37249, 20800, 37633, 21440, 21120, 37441, 38401, 22208, 22400,
38721, 21760, 38337, 38017, 21568, 39937, 23744, 23936, 40257, 24320,
40897, 40577, 24128, 23040, 39617, 39809, 23360, 39169, 22976, 22656,
38977, 34817, 18624, 18816, 35137, 19200, 35777, 35457, 19008, 19968,
36545, 36737, 20288, 36097, 19904, 19584, 35905, 17408, 33985, 34177,
17728, 34561, 18368, 18048, 34369, 33281, 17088, 17280, 33601, 16640,
33217, 32897, 16448]))
```

Calculate CRC-16

pihat.eeprom.device module

EEPROM device

```
class pihat.eeprom.device.EepromDeviceOverlay(bus: int = 99, autocreate: bool = True,
autoremove: bool = False, timeout: float
= 2.0, interval: float = 0.1)
```

Bases: object

EEPROM devicetree overlay

```

autocreate = True
autoremove = False
bus = 99
data
    Overlay devicetree blob
directory
    Overlay directory
dtbo
    Overlay file
eeprom
    EEPROM device path
install ()
    Install overlay
interval = 0.1
name
    Overlay name
remove ()
    Remove overlay
timeout = 2.0
wait ()
    Wait for EEPROM device to exist

class pihat.eeprom.device.EepromDevice (file: Union[None, os.PathLike, IO] =
    None, mode: str = 'r+b', header: pi-
    hat.eeprom.layout.EepromHeader = <factory>,
    atoms: List[pihat.eeprom.layout.EepromAtom]
    = <factory>, autoloader: bool = True, au-
    tosave: bool = False, autouuid: bool = True,
    bus: dataclasses.InitVar = None, overlay: pi-
    hat.eeprom.device.EepromDeviceOverlay = <fac-
    tory>)

    Bases: pihat.eeprom.file.EepromFile
    EEPROM stored in an i2c EEPROM device

    autouuid = True
    bus = None
    open (file=None, mode=None)
        Open file

```

pihat.eeprom.exceptions module

Pi Hat EEPROM exceptions

```

exception pihat.eeprom.exceptions.EepromValueError
    Bases: ValueError
    EEPROM value error

```

exception `pihat.eeprom.exceptions.EepromSignatureError`

Bases: `pihat.eeprom.exceptions.EepromValueError`

EEPROM magic signature error

exception `pihat.eeprom.exceptions.EepromCrcError`

Bases: `pihat.eeprom.exceptions.EepromValueError`

EEPROM CRC error

exception `pihat.eeprom.exceptions.EepromLengthError`

Bases: `pihat.eeprom.exceptions.EepromValueError`

EEPROM length error

exception `pihat.eeprom.exceptions.EepromVerificationError`

Bases: `pihat.eeprom.exceptions.EepromValueError`

EEPROM verification error

pihat.eeprom.file module

EEPROM file

class `pihat.eeprom.file.EepromFile` (`file: Union[None, os.PathLike, IO] = None, mode: str = 'r+b', header: pihat.eeprom.layout.EepromHeader = <factory>, atoms: List[pihat.eeprom.layout.EepromAtom] = <factory>, autoload: bool = True, autosave: bool = False, autouuid: bool = False`)

Bases: `pihat.eeprom.layout.Eeprom`, `pihat.eeprom.file.OpenableFile`

EEPROM stored in a file

autoload = **True**

autosave = **False**

autouuid = **False**

load (`file=None, mode='rb'`)

Load EEPROM from file

pack (`fixup=True`)

Pack structure to binary data

save (`file=None, mode='wb', verify=False`)

Save EEPROM to file

pihat.eeprom.layout module

Pi Hat EEPROM layout

class `pihat.eeprom.layout.EepromHeader` (`**kwargs`)

Bases: `pihat.eeprom.layout.EepromLittleEndianStructure`

EEPROM header

eeplen

Structure/Union member

numatoms

Structure/Union member

signature
An enumerated type field within an EEPROM structure

version
An enumerated type field within an EEPROM structure

class pihat.eeprom.layout.**EepromVendorInfo** (**kwargs)
 Bases: pihat.eeprom.layout.EepromAtomData, pihat.eeprom.layout.EepromLittleEndianStructure
 EEPROM vendor information data

fixup ()
Fix up fields for consistency

pack (fixup=True)
Pack structure to binary data

pid
Structure/Union member

pstr
A string field within an EEPROM structure

pver
Structure/Union member

type = 1

unpack (raw)
Unpack structure from binary data

uuid
A UUID field within an EEPROM structure

vstr
A string field within an EEPROM structure

class pihat.eeprom.layout.**EepromGpioBank** (**kwargs)
 Bases: pihat.eeprom.layout.EepromLittleEndianStructure
 EEPROM GPIO bank configuration

drive
An enumerated type field within an EEPROM structure

hysteresis
An enumerated type field within an EEPROM structure

slew
An enumerated type field within an EEPROM structure

class pihat.eeprom.layout.**EepromGpioPower** (**kwargs)
 Bases: pihat.eeprom.layout.EepromLittleEndianStructure
 EEPROM GPIO powering

back_power
An enumerated type field within an EEPROM structure

class pihat.eeprom.layout.**EepromGpioPin** (**kwargs)
 Bases: pihat.eeprom.layout.EepromLittleEndianStructure
 EEPROM GPIO pin description

```

function
    An enumerated type field within an EEPROM structure

pull
    An enumerated type field within an EEPROM structure

used
    An enumerated type field within an EEPROM structure

class pihat.eeprom.layout.EepromGpioPins
    Bases: pihat.eeprom.layout.EepromArray
    EEPROM GPIO pin descriptions

class pihat.eeprom.layout.EepromGpioMap (**kwargs)
    Bases: pihat.eeprom.layout.EepromAtomData, pihat.eeprom.layout.
            EepromLittleEndianStructure
    EEPROM GPIO map

bank
    Structure/Union member

pins
    Structure/Union member

power
    Structure/Union member

type = 2

class pihat.eeprom.layout.EepromAtom (**kwargs)
    Bases: pihat.eeprom.layout.EepromLittleEndianStructure
    EEPROM atom

count
    Structure/Union member

data = b''

fixup()
    Fix up fields for consistency

pack (fixup=True)
    Pack structure to binary data

type
    An enumerated type field within an EEPROM structure

unfixed_len
    Recorded length (ignoring any potential data changes)

unpack (raw)
    Unpack structure from binary data

class pihat.eeprom.layout.Eeprom (header: pihat.eeprom.layout.EepromHeader = <factory>,
                                     atoms: List[pihat.eeprom.layout.EepromAtom] = <factory>)
    Bases: pihat.eeprom.layout.EepromStructure
    EEPROM content

atom (type)
    Find first atom of a specified type

```

bank
An EEPROM attribute located inside the GPIO map atom

dtbo
Device tree overlay atom

fdt
An EEPROM attribute located inside the device tree blob atom

fixup ()
Fix up fields for consistency

gpio
GPIO map atom

has_dtbo
Check for presence of device tree overlay atom

info
Vendor information atom

pack (fixup=True)
Pack structure to binary data

pid
An EEPROM attribute located inside the vendor information atom

pins
An EEPROM attribute located inside the GPIO map atom

power
An EEPROM attribute located inside the GPIO map atom

pstr
An EEPROM attribute located inside the vendor information atom

pver
An EEPROM attribute located inside the vendor information atom

unpack (raw)
Unpack structure from binary data

uuid
An EEPROM attribute located inside the vendor information atom

vstr
An EEPROM attribute located inside the vendor information atom

Module contents

Pi Hat EEPROM

1.2 Module contents

Pi Hat

CHAPTER 2

Indices and tables

- `genindex`
- `modindex`
- `search`

p

- `pihat`, [11](#)
- `pihat.eeprom`, [11](#)
- `pihat.eeprom.cli`, [3](#)
- `pihat.eeprom.constants`, [4](#)
- `pihat.eeprom.crc`, [6](#)
- `pihat.eeprom.device`, [6](#)
- `pihat.eeprom.exceptions`, [7](#)
- `pihat.eeprom.file`, [8](#)
- `pihat.eeprom.layout`, [8](#)

A

ALT0 (*pihat.eeprom.constants.EepromGpioFunction* attribute), 5
 ALT1 (*pihat.eeprom.constants.EepromGpioFunction* attribute), 5
 ALT2 (*pihat.eeprom.constants.EepromGpioFunction* attribute), 5
 ALT3 (*pihat.eeprom.constants.EepromGpioFunction* attribute), 5
 ALT4 (*pihat.eeprom.constants.EepromGpioFunction* attribute), 5
 ALT5 (*pihat.eeprom.constants.EepromGpioFunction* attribute), 5
 atom() (*pihat.eeprom.layout.Eeprom* method), 10
 autocreate (*pihat.eeprom.device.EepromDeviceOverlay* attribute), 6
 autoload (*pihat.eeprom.file.EepromFile* attribute), 8
 autoremove (*pihat.eeprom.device.EepromDeviceOverlay* attribute), 7
 autosave (*pihat.eeprom.file.EepromFile* attribute), 8
 autouuid (*pihat.eeprom.device.EepromDevice* attribute), 7
 autouuid (*pihat.eeprom.file.EepromFile* attribute), 8

B

back_power (*pihat.eeprom.layout.EepromGpioPower* attribute), 9
 bank (*pihat.eeprom.layout.Eeprom* attribute), 10
 bank (*pihat.eeprom.layout.EepromGpioMap* attribute), 10
 bus (*pihat.eeprom.device.EepromDevice* attribute), 7
 bus (*pihat.eeprom.device.EepromDeviceOverlay* attribute), 7

C

Command (class in *pihat.eeprom.cli*), 3
 count (*pihat.eeprom.layout.EepromAtom* attribute), 10
 crc16() (in module *pihat.eeprom.crc*), 6

CUSTOM (*pihat.eeprom.constants.EepromAtomType* attribute), 4

D

data (*pihat.eeprom.device.EepromDeviceOverlay* attribute), 7
 data (*pihat.eeprom.layout.EepromAtom* attribute), 10
 DEFAULT (*pihat.eeprom.constants.EepromGpioDrive* attribute), 4
 DEFAULT (*pihat.eeprom.constants.EepromGpioHysteresis* attribute), 5
 DEFAULT (*pihat.eeprom.constants.EepromGpioPull* attribute), 6
 DEFAULT (*pihat.eeprom.constants.EepromGpioSlew* attribute), 5
 directory (*pihat.eeprom.device.EepromDeviceOverlay* attribute), 7
 DISABLED (*pihat.eeprom.constants.EepromGpioHysteresis* attribute), 5
 DOWN (*pihat.eeprom.constants.EepromGpioPull* attribute), 6
 drive (*pihat.eeprom.layout.EepromGpioBank* attribute), 9
 DTBO (*pihat.eeprom.constants.EepromAtomType* attribute), 4
 dtbo (*pihat.eeprom.device.EepromDeviceOverlay* attribute), 7
 dtbo (*pihat.eeprom.layout.Eeprom* attribute), 11
 dump() (*pihat.eeprom.cli.Command* method), 3

E

eeplen (*pihat.eeprom.layout.EepromHeader* attribute), 8
 Eeprom (class in *pihat.eeprom.layout*), 10
 eeprom (*pihat.eeprom.device.EepromDeviceOverlay* attribute), 7
 eeprom() (*pihat.eeprom.cli.Command* method), 3
 EepromAtom (class in *pihat.eeprom.layout*), 10
 EepromAtomType (class in *pihat.eeprom.constants*), 4
 EepromCrcError, 8

EepromDevice (class in *pihat.eeprom.device*), 7
EepromDeviceOverlay (class in *pihat.eeprom.device*), 6
EepromFile (class in *pihat.eeprom.file*), 8
EepromGpioBackPower (class in *pihat.eeprom.constants*), 5
EepromGpioBank (class in *pihat.eeprom.layout*), 9
EepromGpioDrive (class in *pihat.eeprom.constants*), 4
EepromGpioFunction (class in *pihat.eeprom.constants*), 5
EepromGpioHysteresis (class in *pihat.eeprom.constants*), 5
EepromGpioMap (class in *pihat.eeprom.layout*), 10
EepromGpioPin (class in *pihat.eeprom.layout*), 9
EepromGpioPins (class in *pihat.eeprom.layout*), 10
EepromGpioPower (class in *pihat.eeprom.layout*), 9
EepromGpioPull (class in *pihat.eeprom.constants*), 6
EepromGpioSlew (class in *pihat.eeprom.constants*), 5
EepromHeader (class in *pihat.eeprom.layout*), 8
EepromLengthError, 8
EepromSignature (class in *pihat.eeprom.constants*), 4
EepromSignatureError, 7
EepromValueError, 7
EepromVendorInfo (class in *pihat.eeprom.layout*), 9
EepromVerificationError, 8
EepromVersion (class in *pihat.eeprom.constants*), 4
ENABLED (*pihat.eeprom.constants.EepromGpioHysteresis* attribute), 5
execute () (*pihat.eeprom.cli.Command* method), 3
extract () (*pihat.eeprom.cli.Command* method), 3

F

fdt (*pihat.eeprom.layout.Eeprom* attribute), 11
fixup () (*pihat.eeprom.layout.Eeprom* method), 11
fixup () (*pihat.eeprom.layout.EepromAtom* method), 10
fixup () (*pihat.eeprom.layout.EepromVendorInfo* method), 9
function (*pihat.eeprom.layout.EepromGpioPin* attribute), 9

G

GPIO (*pihat.eeprom.constants.EepromAtomType* attribute), 4
gpio (*pihat.eeprom.layout.Eeprom* attribute), 11

H

has_dtbo (*pihat.eeprom.layout.Eeprom* attribute), 11
hysteresis (*pihat.eeprom.layout.EepromGpioBank* attribute), 9

I

INFO (*pihat.eeprom.constants.EepromAtomType* attribute), 4
info (*pihat.eeprom.layout.Eeprom* attribute), 11
INPUT (*pihat.eeprom.constants.EepromGpioFunction* attribute), 5
install () (*pihat.eeprom.device.EepromDeviceOverlay* method), 7
interval (*pihat.eeprom.device.EepromDeviceOverlay* attribute), 7

L

LIMITED (*pihat.eeprom.constants.EepromGpioSlew* attribute), 5
load () (*pihat.eeprom.file.EepromFile* method), 8

M

MA_10 (*pihat.eeprom.constants.EepromGpioDrive* attribute), 4
MA_12 (*pihat.eeprom.constants.EepromGpioDrive* attribute), 4
MA_1300 (*pihat.eeprom.constants.EepromGpioBackPower* attribute), 5
MA_14 (*pihat.eeprom.constants.EepromGpioDrive* attribute), 4
MA_16 (*pihat.eeprom.constants.EepromGpioDrive* attribute), 4
MA_2 (*pihat.eeprom.constants.EepromGpioDrive* attribute), 4
MA_2000 (*pihat.eeprom.constants.EepromGpioBackPower* attribute), 5
MA_4 (*pihat.eeprom.constants.EepromGpioDrive* attribute), 4
MA_6 (*pihat.eeprom.constants.EepromGpioDrive* attribute), 4
MA_8 (*pihat.eeprom.constants.EepromGpioDrive* attribute), 4
merge () (*pihat.eeprom.cli.Command* method), 3

N

name (*pihat.eeprom.device.EepromDeviceOverlay* attribute), 7
NONE (*pihat.eeprom.constants.EepromGpioBackPower* attribute), 5
NONE (*pihat.eeprom.constants.EepromGpioPull* attribute), 6
numatoms (*pihat.eeprom.layout.EepromHeader* attribute), 8

O

open () (*pihat.eeprom.device.EepromDevice* method), 7
OUTPUT (*pihat.eeprom.constants.EepromGpioFunction* attribute), 5

P

[pack\(\)](#) (*pihat.eeprom.file.EepromFile* method), 8
[pack\(\)](#) (*pihat.eeprom.layout.Eeprom* method), 11
[pack\(\)](#) (*pihat.eeprom.layout.EepromAtom* method), 10
[pack\(\)](#) (*pihat.eeprom.layout.EepromVendorInfo* method), 9
[parser\(\)](#) (*pihat.eeprom.cli.Command* method), 3
[pid](#) (*pihat.eeprom.layout.Eeprom* attribute), 11
[pid](#) (*pihat.eeprom.layout.EepromVendorInfo* attribute), 9
[pihat](#) (*module*), 11
[pihat.eeprom](#) (*module*), 11
[pihat.eeprom.cli](#) (*module*), 3
[pihat.eeprom.constants](#) (*module*), 4
[pihat.eeprom.crc](#) (*module*), 6
[pihat.eeprom.device](#) (*module*), 6
[pihat.eeprom.exceptions](#) (*module*), 7
[pihat.eeprom.file](#) (*module*), 8
[pihat.eeprom.layout](#) (*module*), 8
[pins](#) (*pihat.eeprom.layout.Eeprom* attribute), 11
[pins](#) (*pihat.eeprom.layout.EepromGpioMap* attribute), 10
[power](#) (*pihat.eeprom.layout.Eeprom* attribute), 11
[power](#) (*pihat.eeprom.layout.EepromGpioMap* attribute), 10
[pstr](#) (*pihat.eeprom.layout.Eeprom* attribute), 11
[pstr](#) (*pihat.eeprom.layout.EepromVendorInfo* attribute), 9
[pull](#) (*pihat.eeprom.layout.EepromGpioPin* attribute), 10
[pver](#) (*pihat.eeprom.layout.Eeprom* attribute), 11
[pver](#) (*pihat.eeprom.layout.EepromVendorInfo* attribute), 9

R

[read_desc\(\)](#) (*pihat.eeprom.cli.Command* method), 3
[read_fdt\(\)](#) (*pihat.eeprom.cli.Command* method), 3
[remove\(\)](#) (*pihat.eeprom.device.EepromDeviceOverlay* method), 7
[replace\(\)](#) (*pihat.eeprom.cli.Command* method), 4
[RESERVED](#) (*pihat.eeprom.constants.EepromGpioBackPower* attribute), 5
[RESERVED](#) (*pihat.eeprom.constants.EepromGpioHysteresis* attribute), 5
[RESERVED](#) (*pihat.eeprom.constants.EepromGpioSlew* attribute), 5
[RESERVED_10](#) (*pihat.eeprom.constants.EepromGpioDrive* attribute), 4
[RESERVED_11](#) (*pihat.eeprom.constants.EepromGpioDrive* attribute), 4
[RESERVED_12](#) (*pihat.eeprom.constants.EepromGpioDrive* attribute), 5
[RESERVED_13](#) (*pihat.eeprom.constants.EepromGpioDrive* attribute), 5

[RESERVED_14](#) (*pihat.eeprom.constants.EepromGpioDrive* attribute), 5
[RESERVED_15](#) (*pihat.eeprom.constants.EepromGpioDrive* attribute), 5
[RESERVED_9](#) (*pihat.eeprom.constants.EepromGpioDrive* attribute), 5
[RPI](#) (*pihat.eeprom.constants.EepromSignature* attribute), 4

S

[save\(\)](#) (*pihat.eeprom.file.EepromFile* method), 8
[signature](#) (*pihat.eeprom.layout.EepromHeader* attribute), 8
[slew](#) (*pihat.eeprom.layout.EepromGpioBank* attribute), 9

T

[timeout](#) (*pihat.eeprom.device.EepromDeviceOverlay* attribute), 7
[type](#) (*pihat.eeprom.layout.EepromAtom* attribute), 10
[type](#) (*pihat.eeprom.layout.EepromGpioMap* attribute), 10
[type](#) (*pihat.eeprom.layout.EepromVendorInfo* attribute), 9

U

[unfixed_len](#) (*pihat.eeprom.layout.EepromAtom* attribute), 10
[UNLIMITED](#) (*pihat.eeprom.constants.EepromGpioSlew* attribute), 5
[unpack\(\)](#) (*pihat.eeprom.layout.Eeprom* method), 11
[unpack\(\)](#) (*pihat.eeprom.layout.EepromAtom* method), 10
[unpack\(\)](#) (*pihat.eeprom.layout.EepromVendorInfo* method), 9
[UP](#) (*pihat.eeprom.constants.EepromGpioPull* attribute), 6
[used](#) (*pihat.eeprom.layout.EepromGpioPin* attribute), 10
[uuid](#) (*pihat.eeprom.layout.Eeprom* attribute), 11
[uuid](#) (*pihat.eeprom.layout.EepromVendorInfo* attribute), 9

V

[V1](#) (*pihat.eeprom.constants.EepromVersion* attribute), 4
[version](#) (*pihat.eeprom.layout.EepromHeader* attribute), 9
[vstr](#) (*pihat.eeprom.layout.Eeprom* attribute), 11
[vstr](#) (*pihat.eeprom.layout.EepromVendorInfo* attribute), 9

W

[wait\(\)](#) (*pihat.eeprom.device.EepromDeviceOverlay* method), 7

```
write_desc() (pihat.eeprom.cli.Command method),  
4  
write_fdt() (pihat.eeprom.cli.Command method), 4
```