



Python Fork API

Revised Jan 31, 2020

Table of Contents

Introduction.....	3
Install on Windows.....	4
Quick start.....	5
Digital Input (DI) commands.....	6
Digital Output (DO) commands.....	8
Analog Input (AI) commands.....	10
Analog Output (AO) commands.....	12
Digital Interface Lines (I) commands.....	15
UART.....	15
SPI.....	17
I2C.....	20
Other commands.....	23
Stream.....	24

Introduction

Python Fork API is a Python library to ease using Fork. It gives you all handy tools for work.

Install on Windows

For library installation launch file “install.cmd”.

Quick start

You can just run this code and receive digital input(DI) state.

```
from forkapi.fork import Fork
dev1 = Fork('192.168.0.205')
di1 = dev1.diGet(1)
print(di1)
```

Congratulations! You've read state of DI1.

Digital Input (DI) commands

Command: `forkapi.fork.Fork.diGet(pin)`

Description: read digital input channel state

Input values: pin – channel number (0-7)

Result: 0 / 1 / False

Command: `forkapi.fork.Fork.diGetAll()`

Description: read all digital input channel state

Input values: –

Result: list of values / False

Command: `forkapi.fork.Fork.diGetPullup(pin)`

Description: read pullup state of digital input channel

Input values: pin – channel number (0-7)

Result: 0 / 1 / False

Command: `forkapi.fork.Fork.diGetPullupAll()`

Description: read all pullup states of digital input channels

Input values: –

Result: list of values / False

Command: `forkapi.fork.Fork.diCntStart(pin)`

Description: start impulse counter on channel

Input values: pin – channel number (0-7)

Result: value(int) / False

Command: forkapi.fork.Fork.diCnt(pin)

Description: read impulse counter on channel

Input values: pin – channel number (0-7)

Result: value(int) / False

Command: forkapi.fork.Fork.diCntReset(pin)

Description: reset impulse counter on channel

Input values: pin – channel number (0-7)

Result: value(int) / False

Command: forkapi.fork.Fork.diFreq(pin)

Description: start / read value of frequency on channel

Input values: pin – channel number (0-7)

Result: value(int) / False

Command: forkapi.fork.Fork.diStop(pin)

Description: stop impulse counter on channel

Input values: pin – channel number (0-7)

Result: value(int) / False

Digital Output (DO) commands

Command: `forkapi.fork.Fork.doGet(pin)`

Description: read digital output channel state

Input values: pin – channel number (0-7)

Result: 0 / 1 / False

Command: `forkapi.fork.Fork.doSet(pin, value)`

Description: writes digital output channel state

Input values: pin – channel number (0-7)

value – value to write (0 / 1)

Result: True / False

Command: `forkapi.fork.Fork.doGetAll()`

Description: read all digital output channels state

Input values: –

Result: list of values[array of 8] / False

Command: `forkapi.fork.Fork.doSetPwm(pin, en, freq, duty)`

Description: use digital output in PWM mode

Input values: pin – channel number (0-7)

en – enable or disable PWM (0 / 1)

freq – PWM frequency (0-1000000)

duty – PWM duty cycle (0-100)

Result: True / False

Command: `forkapi.fork.Fork.doSetPwmEn(pin, enable)`

Description: set digital output PWM enable / disable

Input values: pin – channel number (0-7)

en – enable or disable PWM (0 / 1)

Result: True / False

Command: `forkapi.fork.Fork.doSetPwmFreq(pin, freq)`

Description: set digital output PWM frequency

Input values: pin – channel number (0-7)

freq – PWM frequency (0-1000000)

Result: True / False

Command: `forkapi.fork.Fork.doSetPwmDuty(pin, duty)`

Description: set digital output PWM duty cycle

Input values: pin – channel number (0-7)

duty – PWM duty cycle (0-100)

Result: True / False

Command: `forkapi.fork.Fork.doGetPwmAll()`

Description: set digital output PWM duty cycle

Input values: pin – channel number (0-7)

duty – PWM duty cycle (0-100)

Result: list of values[en[array of 8], freq[array of 8], duty[array of 8]] / False

Command: `forkapi.fork.Fork.doLevel3V3()`

Description: set voltage level of all digital outputs to 3.3V

Input values: –

Result: True / False

Command: `forkapi.fork.Fork.doLevel5V()`

Description: set voltage level of all digital outputs to 5V

Input values: –

Result: True / False

Analog Input (AI) commands

Command `forkapi.fork.Fork.aiGet(pin)`

Description: Read state of AI channel.

Input values: pin – channel number (0-7)

Result: value (float) / False

Command `forkapi.fork.Fork.aiGetAll()`

Description: Read state of all AI channels /

Input values: –

Result: list of values (float) / False

Command `forkapi.fork.Fork.aiSetTrig(aiPin, diPin)`

Description: Set the source of trigger signal. DI channels is using as trigger signal.

Input values: aiPin – channel number (0-7)

diPin – trigger source channel number (0-7)

Result: True / False

Command `forkapi.fork.Fork.aiResetTrig (pin)`

Description: Turn off trigger signal.

Input values: pin – channel number (0-7)

Result: True / False

Command `forkapi.fork.Fork.aiGetTrig (pin)`

Description: Read value latched on trigger. If there was no trigger signal return None

Input values: pin – channel number (0-7)

Result: float / None / False

Command forkapi.fork.Fork.aiSetTrigPol (pin)

Description: Set trigger edge polarity.

Input values: pin – channel number (0-7)

edge – 0 - falling / 1 - rising

Result: True / False

Command forkapi.fork.Fork.aiSetSingleTrig (pin)

Description: Set the source of trigger signal for single trigger.

Input values: pin – channel number (0-7)

Result: True / False

Command forkapi.fork.Fork.aiResetSingleTrig (pin)

Description: Turn off trigger signal.

Input values: pin – channel number (0-7)

Result: True / False

Command forkapi.fork.Fork.aiSetSingleTrigW (pin)

Description: Set the source of single trigger and wait answer.

Input values: pin – channel number (0-7)

Result: True / False

Command forkapi.fork.Fork.aiSetFilter(pin)

Description: Enable or disable filter on AI channel.

Input values: pin – channel number (0-7)

enable – enable or disable filter(0 / 1)

Result: True / False

Analog Output (AO) commands

Command `forkapi.fork.Fork.aoSet(pin, value)`

Description: set the voltage on AO channel.

Input values: pin – channel number (0-7)

value – voltage on AO channel (0-10)

Result: True / False

Command `forkapi.fork.Fork.aoGet (pin, value)`

Description: read the voltage on AO channel.

Input values: pin – channel number (0-7)

Result: Float / False

Command `forkapi.fork.Fork.aoGetAll (pin, value)`

Description: read the voltage on all AO channels.

Input values: –

Result: list of values(floats) / False

Command `forkapi.fork.Fork.aoSetGenEn(pin, en)`

Description: enable generator on AO channel

Input values: pin – channel number (0-7)

enable – enable or disable (0 / 1)

Result: True / False

Command forkapi.fork.Fork.aoSetGenFreq(pin, freq)

Description: set frequency of generator on AO channel

Input values: pin – channel number (0-7)

freq – frequency(0-100000)

Result: True / False

Command forkapi.fork.Fork.aoSetGenMin(pin, min)

Description: set min voltage of generator on AO channel

Input values: pin – channel number (0-7)

min – voltage (0-10)

Result: True / False

Command forkapi.fork.Fork.aoSetGenMax(pin, max)

Description: set max voltage of generator on AO channel

Input values: pin – channel number (0-7)

max – voltage (0-10)

Result: True / False

Command forkapi.fork.Fork.aoSetGenPhase(pin, phase)

Description: set starting phase of generator on AO channel

Input values: pin – channel number (0-7)

phase – degrees (0-360)

Result: True / False

Command `forkapi.fork.Fork.aoSetGenPhase2(phase)`

Description: set delta of starting phase of generator on AO channel

Input values: pin – channel number (0-7)

enable – enable or disable (0 / 1)

Result: True / False

Command `forkapi.fork.Fork.aoSetGenDC(pin, dc)`

Description: set dc voltage of generator on AO channel

Input values: pin – channel number (0-7)

dc – voltage (0-10)

Result: True / False

Command `forkapi.fork.Fork.aoSetGenAmp(pin, amp)`

Description: set dc amplitude of generator on AO channel

Input values: pin – channel number (0-7)

amp – voltage (0-10)

Result: True / False

Command `forkapi.fork.Fork.aoSetGenCnt(pin, cnt)`

Description: set count of waves which generating on AO channel

Input values: pin – channel number (0-7)

cnt – count(0-1000)

Result: True / False

Command `forkapi.fork.Fork.aoGetGenAll()`

Description: read all parameters of generators

Input values: –

Result: list of values [en[0..1], mode[0..1], freq[0..1], dc[0..1], amp[0..1]] / False.

Digital Interface Lines (I) commands

UART

Command `forkapi.fork.UARTenable(ch)`

Description: Enable RX and TX UART channel.

Input values: ch – UART channel number (1-4)

Result: True / False

Command `forkapi.fork.UARTenableRX(ch)`

Description: Enable RX UART channel.

Input values: ch – UART channel number (1-4)

Result: True / False

Command `forkapi.fork.UARTenableTX(ch)`

Description: Enable TX UART channel.

Input values: ch – UART channel number (1-3)

Result: True / False

Command `forkapi.fork.UARTdisable(ch)`

Description: Disable RX and TX UART channel.

Input values: ch – UART channel number (1-4)

Result: True / False

Command `forkapi.fork.UARTdisableRX(ch)`

Description: Disable RX UART channel.

Input values: ch – UART channel number (1-4)

Result: True / False

Command `forkapi.fork.UARTDisableTX(ch)`

Description: Disable TX UART channel.

Input values: ch – UART channel number (1-3)

Result: True / False

Command `forkapi.fork.UARTtransmit(ch,data)`

Description: UART channel transmit data

Input values: ch – UART channel number (1-3)

data – (bytes)

Result: True / False

Command `forkapi.fork.UARTreceive(ch)`

Description: Read data from UART channel.

Input values: ch – UART channel number (1-4)

Result: bytes / None / False

Command `forkapi.fork.UARTSetStopbits(ch,sb)`

Description: set stopbits of UART channel

Input values: ch – UART channel number (1-4)

sb – stopbits (1,2)

Result: True / False

Command `forkapi.fork.UARTGetStopbits(ch)`

Description: get stopbits of UART channel

Input values: ch – UART channel number (1-4)

Result: value (int) / False

Command forkapi.fork.UARTSetBaudrate(ch,br)

Description: Set baudrate of UART channel

Input values: ch – UART channel number (1-4)

br – baudrate (0-2812500)

Result: True / False

Command forkapi.fork.UARTGetBaudrate(ch)

Description: Get baudrate of UART channel

Input values: ch – UART channel number (1-4)

Result: value (int) / False

SPI

Command forkapi.fork.SPIEnable()

Description: Enable SPI.

Input values: –

Result: True / False

Command forkapi.fork.SPIDisable()

Description: Disable SPI.

Input values: –

Result: True / False

Command forkapi.fork.SPIWrite(data)

Description: Write to SPI.

Input values: data – (bytes)

Result: True / False

Command forkapi.fork.SPIRead(cnt)

Description: Read from SPI.

Input values: cnt – count of bytes to read (1-512)

Result: bytes / False

Command `forkapi.fork.SPIWriteRead(data)`

Description: Read while write from SPI.

Input values: data – data bytes to write (bytes)

Result: bytes / False

Command `forkapi.fork.SPISetBaudrate(baudrate)`

Description: Set baudrate of SPI

Input values: baudrate – baudrate (175781, 351563, 703125, 1406250, 2812500, 5625000, 11250000, 22500000)

Result: True / False

Command `forkapi.fork.SPIGetBaudrate()`

Description: Get baudrate of SPI

Input values: –

Result: value (int) / False

Command `forkapi.fork.SPISetDatasize(datasize)`

Description: Set datasize of SPI

Input values: datasize – datasize (8,16)

Result: True / False

Command `forkapi.fork.SPIGetDatasize()`

Description: Get datasize of SPI

Input values: –

Result: value (int) / False

Command `forkapi.fork.SPISetPolarity(polarity)`

Description: Set polarity of SPI

Input values: polarity – polarity of SPI clock ("low", "high")

Result: True / False

Command `forkapi.fork.SPIGetPolarity()`

Description: Get polarity of SPI

Input values: –

Result: "low" / "high" / False

Command `forkapi.fork.SPISetFirstbit(firstbit)`

Description: Set firstbit of SPI

Input values: firstbit – firstbit of SPI ("msb", "lsb")

Result: True / False

Command `forkapi.fork.SPIGetPolarity()`

Description: Get firstbit of SPI

Input values: –

Result: "msb" / "lsb" / False

Command `forkapi.fork.SPISetPhase(phase)`

Description: Set phase of SPI

Input values: phase – phase of SPI (1, 2)

Result: True / False

Command `forkapi.fork.SPIGetPhase()`

Description: Get phase of SPI

Input values: –

Result: value(1,2) / False

I2C

Command `forkapi.fork.I2CEnable()`

Description: Enable I2C.

Input values: –

Result: True / False

Command `forkapi.fork.I2CDisable()`

Description: Disable I2C.

Input values: –

Result: True / False

Command `forkapi.fork.I2CWrite(addr, data)`

Description: Write to I2C.

Input values: `addr` – address of I2C device (1 or 2 bytes)

`data` – (bytes)

Result: True / False

Command `forkapi.fork.I2CRead(addr, cnt)`

Description: Read from I2C.

Input values: `addr` – address of I2C device (1 or 2 bytes)

`cnt` – count of bytes to read (1-65536)

Result: bytes / False

Command `forkapi.fork.I2CWriteRead(addr,cnt,data)`

Description: Read after write to I2C.

Input values: `addr` – address of I2C device(1 or 2 bytes)

cnt – count of bytes to read(1-65536)

data – data bytes to write (bytes)

Result: bytes / False

Command `forkapi.fork.I2CSetAddrsize(addrsize)`

Description: Set addrsize of I2C

Input values: addrsize – addrsize of I2C (7, 10)

Result: True / False

Command `forkapi.fork.I2CGetAddrsize()`

Description: Get addrsize of I2C

Input values: –

Result: 7 / 10 / False

Command `forkapi.fork.I2CSetBaudrate(baudrate)`

Description: Set baudrate of I2C

Input values: baudrate – baudrate (0 - 400000)

Result: True / False

Command `forkapi.fork.I2CGetBaudrate()`

Description: Get baudrate of I2C

Input values: –

Result: value (int) / False

Command `forkapi.fork.I2CSetTimeout(timeout)`

Description: Set timeout of I2C

Input values: timeout – timeout (0 - 400000)

Result: True / False

Command `forkapi.fork.I2CGetTimeout()`

Description: Get timeout of I2C

Input values: –

Result: value (int) / "DISABLE" / False

Other commands

Command `forkapi.fork.info()`

Description: Get list of general device information: firmware version, hardware version, bootloader version, serial number, production date, loading date.

Input values: –

Result: list of values [firmware version, hardware version, bootloader version, serial number, production date, loading date] / False

Stream

Command: `forkapi.fork.Fork.startStreamBlocking(ch_set, stream_callback, reply_format=RAW_BINARY, data_rate=100000)`

`ch_set` – set of channels which will use. Number of channels may be 1,2,4,6,8 Example

`ch_set=[Fork.AI3, Fork.AI2]`

`stream_callback` — callback function, which calling every time, when packet received. Data passes to callback function.

Example

```
def ppp(data):
    ppp.i += 1
    f.write(data)

    if (ppp.i > 10000):
        print('stop stream')
        dev1.stopStream()
```

`reply_format` – format of data, which passes to callback function

`RAW_BINARY` – data of stream returns as raw packets of bytes.

`RAW_ADC` - data of stream returns as list of lists every channel. Every item of lists is ADC code `[[2100,2103,2110], [3500,3530, 3600]]`.

`data_rate` – count of total data from all channels

forkapi.fork.Fork.startStreamPlotting(self, ch_set, data_rate=100000, window=10)

`ch_set` – set of channels which will use. Number of channels may be 1,2,4,6,8 Example

`ch_set=[Fork.AI3, Fork.AI2]`

`data_rate` – count of total data from all channels

`window` – size of plot window in seconds

forkapi.fork.Fork.stopStream() – stop the stream if using `startStreamBlocking()`