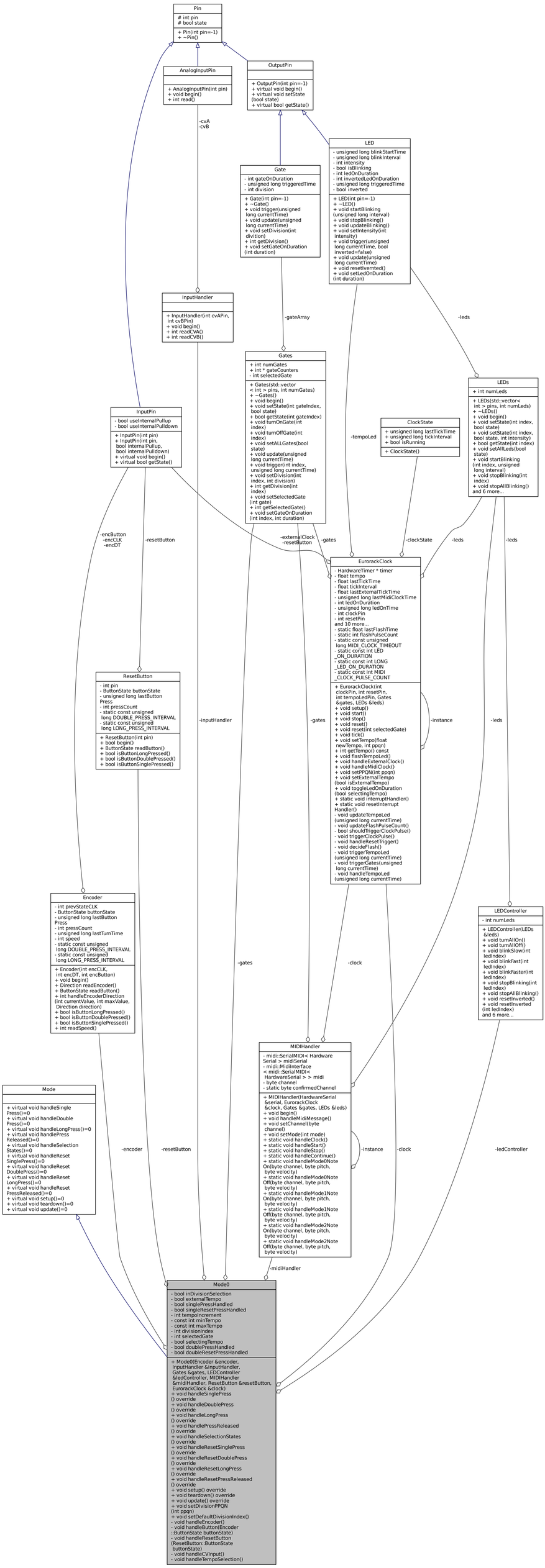
Pin
int pin
bool state
+ Pin(int pin=-1)
+ ~Pin()
AnalogInputPin
+ AnalogInputPin(int pin)
+ void begin()
+ int read()
OutputPin
+ OutputPin(int pin=-1)
+ virtual void begin()
+ virtual void setState
(bool state)
+ virtual bool getState()
LED
- unsigned long blinkStartTime
- unsigned long blinkInterval
- int intensity
- bool isBlinking
- int ledOnDuration
- int invertedLedOnDuration
- unsigned long triggeredTime
- bool inverted
+ LED(int pin=-1)
+ ~LED()
+ void startBlinking
(unsigned long interval)
+ void stopBlinking()
+ void updateBlinking()
+ void setIntensity(int
intensity)
+ void trigger(unsigned
long currentTime, bool
inverted=false)
+ void update(unsigned
long currentTime)
+ void resetInverted()
+ void setLedOnDuration
(int duration)
Gate
- int gateOnDuration
- unsigned long triggeredTime
- int division
+ Gate(int pin=-1)
+ ~Gate()
+ void trigger(unsigned
long currentTime)
+ void update(unsigned
long currentTime)
+ void setDivision(int
division)
+ int getDivision()
+ void setGateOnDuration
(int duration)
InputHandler
+ InputHandler(int cvAPin,
int cvBPin)
+ void begin()
+ int readCVA()
+ int readCVB()
Gates
+ int numGates
+ int * gateCounters
- int selectedGate
+ Gates(std::vector
< int > pins, int numGates)
+ ~Gates()
+ void begin()
+ void setState(int gateIndex,
bool state)
+ bool getState(int gateIndex)
+ void turnOnGate(int
index)
+ void turnOffGate(int
index)
+ void setALLGates(bool
state)
+ void update(unsigned
long currentTime)
+ void trigger(int index,
unsigned long currentTime)
+ void setDivision(int
index, int division)
+ int getDivision(int
index)
+ void setSelectedGate
(int gate)
+ int getSelectedGate()
+ void setGateOnDuration
(int index, int duration)
LEDs
+ int numLeds
+ LEDs(std::vector<
int > pins, int numLeds)
+ ~LEDs()
+ void begin()
+ void setState(int index,
bool state)
+ void setState(int index,
bool state, int intensity)
+ bool getState(int index)
+ void setAllLeds(bool
state)
+ void startBlinking
(int index, unsigned
long interval)
+ void stopBlinking(int
index)
+ void stopAllBlinking()
and 6 more...
InputPin
- bool useInternalPullup
- bool useInternalPulldown
+ InputPin(int pin)
+ InputPin(int pin,
bool internalPullup,
bool internalPulldown)
+ virtual void begin()
+ virtual bool getState()
ClockState
+ unsigned long lastTickTime
+ unsigned long tickInterval
+ bool isRunning
+ ClockState()
EurorackClock
- HardwareTimer * timer
- float tempo
- float lastTickTime
- float tickInterval
- float lastExternalTickTime
- unsigned long lastMidiClockTime
- int ledOnDuration
- unsigned long ledOnTime
- int clockPin
- int resetPin
and 10 more...
- static float lastFlashTime
- static int flashPulseCount
- static const unsigned
long MIDI_CLOCK_TIMEOUT
- static const int LED
_ON_DURATION
- static const int LONG
_LED_ON_DURATION
- static const int MIDI
_CLOCK_PULSE_COUNT
+ EurorackClock(int
clockPin, int resetPin,
int tempoLedPin, Gates
&gates, LEDs &leds)
+ void setup()
+ void start()
+ void stop()
+ void reset()
+ void reset(int selectedGate)
+ void tick()
+ void setTempo(float
newTempo, int ppqn)
+ int getTempo() const
+ void flashTempoLed()
+ void handleExternalClock()
+ void handleMidiClock()
+ void setPPQN(int ppqn)
+ void setExternalTempo
(bool isExternalTempo)
+ void toggleLedOnDuration
(bool selectingTempo)
+ static void interruptHandler()
+ static void resetInterrupt
Handler()
- void updateTempoLed
(unsigned long currentTime)
- void updateFlashPulseCount()
- bool shouldTriggerClockPulse()
- void triggerClockPulse()
- void handleResetTrigger()
- void decideFlash()
- void triggerTempoLed
(unsigned long currentTime)
- void triggerGates(unsigned
long currentTime)
- void handleTempoLed
(unsigned long currentTime)
ResetButton
- int pin
- ButtonState buttonState
- unsigned long lastButton
Press
- int pressCount
- static const unsigned
long DOUBLE_PRESS_INTERVAL
- static const unsigned
long LONG_PRESS_INTERVAL
+ ResetButton(int pin)
+ bool begin()
+ ButtonState readButton()
+ bool isButtonLongPressed()
+ bool isButtonDoublePressed()
+ bool isButtonSinglePressed()
Encoder
- int prevStateClk
- ButtonState buttonState
- unsigned long lastButton
Press
- int pressCount
- unsigned long lastTurnTime
- int speed
- static const unsigned
long DOUBLE_PRESS_INTERVAL
- static const unsigned
long LONG_PRESS_INTERVAL
+ Encoder(int encCLK,
int encDT, int encButton)
+ void begin()
+ Direction readEncoder()
+ ButtonState readButton()
+ int handleEncoderDirection
(int currentValue, int maxValue,
Direction direction)
+ bool isButtonLongPressed()
+ bool isButtonDoublePressed()
+ bool isButtonSinglePressed()
+ int readSpeed()
LEDController
- int numLeds
+ LEDController(LEDs
&leds)
+ void turnAllOn()
+ void turnAllOff()
+ void blinkSlow(int
ledIndex)
+ void blinkFast(int
ledIndex)
+ void blinkFaster(int
ledIndex)
+ void stopBlinking(int
ledIndex)
+ void stopAllBlinking()
+ void resetInverted()
+ void resetInverted
(int ledIndex)
and 6 more...
MIDIHandler
- midi::SerialMIDI< Hardware
Serial > midiSerial
- midi::MidiInterface
< midi::SerialMIDI<
HardwareSerial > > midi
- byte channel
- static byte confirmedChannel
+ MIDIHandler(HardwareSerial
&serial, EurorackClock
&clock, Gates &gates, LEDs &leds)
+ void begin()
+ void handleMidiMessage()
+ void setChannel(byte
channel)
+ void setMode(int mode)
+ static void handleClock()
+ static void handleStart()
+ static void handleStop()
+ static void handleContinue()
+ static void handleMode0Note
On(byte channel, byte pitch,
byte velocity)
+ static void handleMode0Note
Off(byte channel, byte pitch,
byte velocity)
+ static void handleMode1Note
On(byte channel, byte pitch,
byte velocity)
+ static void handleMode1Note
Off(byte channel, byte pitch,
byte velocity)
+ static void handleMode2Note
On(byte channel, byte pitch,
byte velocity)
+ static void handleMode2Note
Off(byte channel, byte pitch,
byte velocity)
Mode
+ virtual void handleSingle
Press()=0
+ virtual void handleDouble
Press()=0
+ virtual void handleLongPress()=0
+ virtual void handlePress
Released()=0
+ virtual void handleSelection
States()=0
+ virtual void handleReset
SinglePress()=0
+ virtual void handleReset
DoublePress()=0
+ virtual void handleReset
LongPress()=0
+ virtual void handleReset
PressReleased()=0
+ virtual void setup()=0
+ virtual void teardown()=0
+ virtual void update()=0
Mode0
- bool inDivisionSelection
- bool externalTempo
- bool singlePressHandled
- bool singleResetPressHandled
- int tempoIncrement
- const int minTempo
- const int maxTempo
- int divisionIndex
- int selectedGate
- bool selectingTempo
- bool doublePressHandled
- bool doubleResetPressHandled
+ Mode0(Encoder &encoder,
InputHandler &inputHandler,
Gates &gates, LEDController
&ledController, MIDIHandler
&midiHandler, ResetButton &resetButton,
EurorackClock &clock)
+ void handleSinglePress
() override
+ void handleDoublePress
() override
+ void handleLongPress
() override
+ void handlePressReleased
() override
+ void handleSelectionStates
() override
+ void handleResetSinglePress
() override
+ void handleResetDoublePress
() override
+ void handleResetLongPress
() override
+ void handleResetPressReleased
() override
+ void setup() override
+ void teardown() override
+ void update() override
+ void setDivisionPPQN
(int ppqn)
+ void setDefaultDivisionIndex()
- void handleEncoder()
- void handleButton(Encoder
::ButtonState buttonState)
- void handleResetButton
(ResetButton::ButtonState
buttonState)
- void handleCVInput()
- void handleTempoSelection()
-cvA
-cvB
-gateArray
-leds
-tempoLed
-encButton
-encCLK
-encDT
-resetButton
-externalClock
-resetButton
-gates
-clockState
-leds
-leds
-inputHandler
-gates
-instance
-leds
-gates
-clock
-encoder
-resetButton
-instance
-clock
-ledController
-midiHandler