FPGA

-SIZE: int

- graph: Graph

- cells:Block[][]

+ FPGA(Graph, int)

+ WriteToFile(String, FPGA): void

+ rippleMove(int, int): void

+ validatePos(Pos2D): boolean

+ getBestFreePosNearBy(Block, Pos2D, Set<Pos2D>): Pos2D

+ getBestAvailPosNearBy(Block, Pos2D, Set<Pos2D>): Pos2D

+ getBestPosNearBy(Block, Pos2D, Set<Pos2D>, boolean): List<Pos2D>

+ placeRandom(): void

+ initPlace(): void

+ getCellByPos(Pos2D): Block

+ setCell(Block, Pos2D): void

+ removeCell(Pos2D): void

+ isCellEmpty(Pos2D): boolean

+ moveCell(Pos2D, Pos2D): void

+ getSize(): int

ParsedBlock

+ item: attribute

- SUBBLOCK_LUT_SIZE : int

- name: String

- in: List<String>

- out: List<String>

- type: BlockType

+ ParsedBlock(BlockType, String, List<String>, List<String>)

+ Parse(String): List<ParsedBlock>

+getBlocksThatAreConnectedTo
(List<ParsedBlock>, List<String>, List<String>): List<ParsedBlock>

+ getName(): String

+ getType(): BlockType

+ getInputs(): List<String>

+ getOutputs: List<String>

ParsedPlacement

- name:String

- x: int

- y: int

- subBlock: int

+ ParsedPlacement(String, int, int, int)

+ Parse(String): Map<String, ParsedPlacement>

+ getName(): String

+ getX(): int

+ getY(): int

+ getSubBlock(): int

Graph

- blocks: Map<String, Block>

+ Graph(List<ParsedBlock>, Map<String, ParsedPlacement>)

- createBlocks(List<ParsedBlock>, Map<String, ParsedPlacement>): void

- connectBlocks(List<ParsedBlock>): void

+ getBlock(String): Block

+ getSortedLogicBlocks(): List<LogicBlock>

+ getSortedPads(): List<Pad>

+ getSortedPadsByLogicBlocks(): List<Pad>

graph 1

blocks *

Pos2D

- x: int

- y: int

+ Pos2D(Pos2D)

+ Pos2D(int, int)

+ setPosition(int,in): void

+ getX(): int

+ getY(): int

position

Block

name: String

position:Pos2D

+ Block(String)

+ Block(String, Pos2D)

+ getName(): String

- connectedBlocks: List<Block>

+ getPosition(): Pos2D

+ setPosition(Pos2D): void

+ addConnectedBlock(Block): void

+ getConnectedBlocks() : List<Block>

getBlockType:BlockType

<<enumeration>>
Pad

PAD
LOGIC_BLOCK

LogicBlock

+ LogicBlock(String)

+ LogicBlock(String, Pos2D)

+ getZFT(): Pos2D

+ getBlockType(): BlockType

Pad

- subBlock: int

+ Pad(String, Pos2D, int)

+ getBlockType(): BlockType