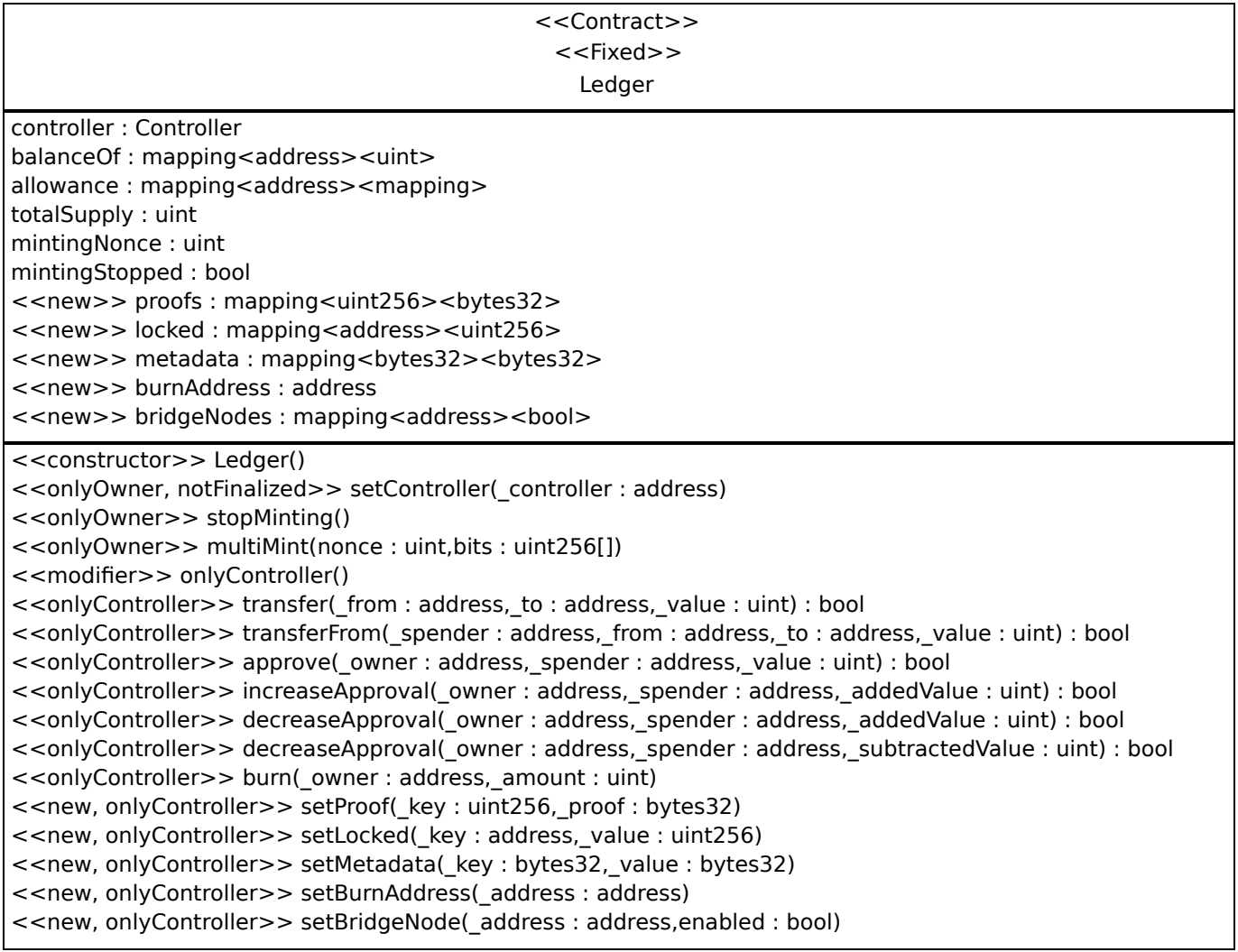
<<Contract>>
<<Fixed>>
Ledger
controller : Controller
balanceOf : mapping<address><uint>
allowance : mapping<address><mapping>
totalSupply : uint
mintingNonce : uint
mintingStopped : bool
<<new>> proofs : mapping<uint256><bytes32>
<<new>> locked : mapping<address><uint256>
<<new>> metadata : mapping<bytes32><bytes32>
<<new>> burnAddress : address
<<new>> bridgeNodes : mapping<address><bool>
<<constructor>> Ledger()
<<onlyOwner, notFinalized>> setController(_controller : address)
<<onlyOwner>> stopMinting()
<<onlyOwner>> multiMint(nonce : uint,bits : uint256[])
<<modifier>> onlyController()
<<onlyController>> transfer(_from : address,_to : address,_value : uint) : bool
<<onlyController>> transferFrom(_spender : address,_from : address,_to : address,_value : uint) : bool
<<onlyController>> approve(_owner : address,_spender : address,_value : uint) : bool
<<onlyController>> increaseApproval(_owner : address,_spender : address,_addedValue : uint) : bool
<<onlyController>> decreaseApproval(_owner : address,_spender : address,_addedValue : uint) : bool
<<onlyController>> decreaseApproval(_owner : address,_spender : address,_subtractedValue : uint) : bool
<<onlyController>> burn(_owner : address,_amount : uint)
<<new, onlyController>> setProof(_key : uint256,_proof : bytes32)
<<new, onlyController>> setLocked(_key : address,_value : uint256)
<<new, onlyController>> setMetadata(_key : bytes32,_value : bytes32)
<<new, onlyController>> setBurnAddress(_address : address)
<<new, onlyController>> setBridgeNode(_address : address,enabled : bool)