<<Contract>>
Controller

ledger : Ledger
token : Token
<<new>> burnAddress : address

<<onlyOwner>> setToken(_token : address)
<<onlyOwner>> setLedger(_ledger : address)
<<modifier>> onlyToken()
<<modifier>> onlyLedger()
<<const>> totalSupply() : uint
<<const>> balanceOf(_a : address) : uint
<<const>> allowance(_owner : address,_spender : address) : uint
<<onlyLedger>> ledgerTransafer(from : address,to : address,val : uint)
<<onlyToken>> transfer(_from : address,_to : address,_value : uint) : bool
<<onlyToken>> transferFrom(_spender : address, from : address, to : address,_value : uint) : bool
<<onlyToken>> approve(_owner : address,_spender : address,_value : uint) : bool
<<onlyToken>> increaseApproval(_owner : address,_spender : address,_addedValue : uint) : bool
<<onlyToken>> decreaseApproval(_owner : address,_spender : address,_subtractedValue : uint) : bool
<<onlyToken>> burn(_owner : address,_amount : uint)
<<new, onlyOwner>> setBurnAddress(_address : address)
<<new, onlyOwner>> enableBurning()
<<new, onlyOwner>> disableBurning()
<<new, onlyToken>> burn(_from : address,_to : bytes32,_amount : uint) : bool
<<new, onlyToken>> claimByProof(_claimer : address,data : bytes32[],proofs : bytes32[],number) : bool
<<new, onlyToken>> claim(_claimer : address) : bool