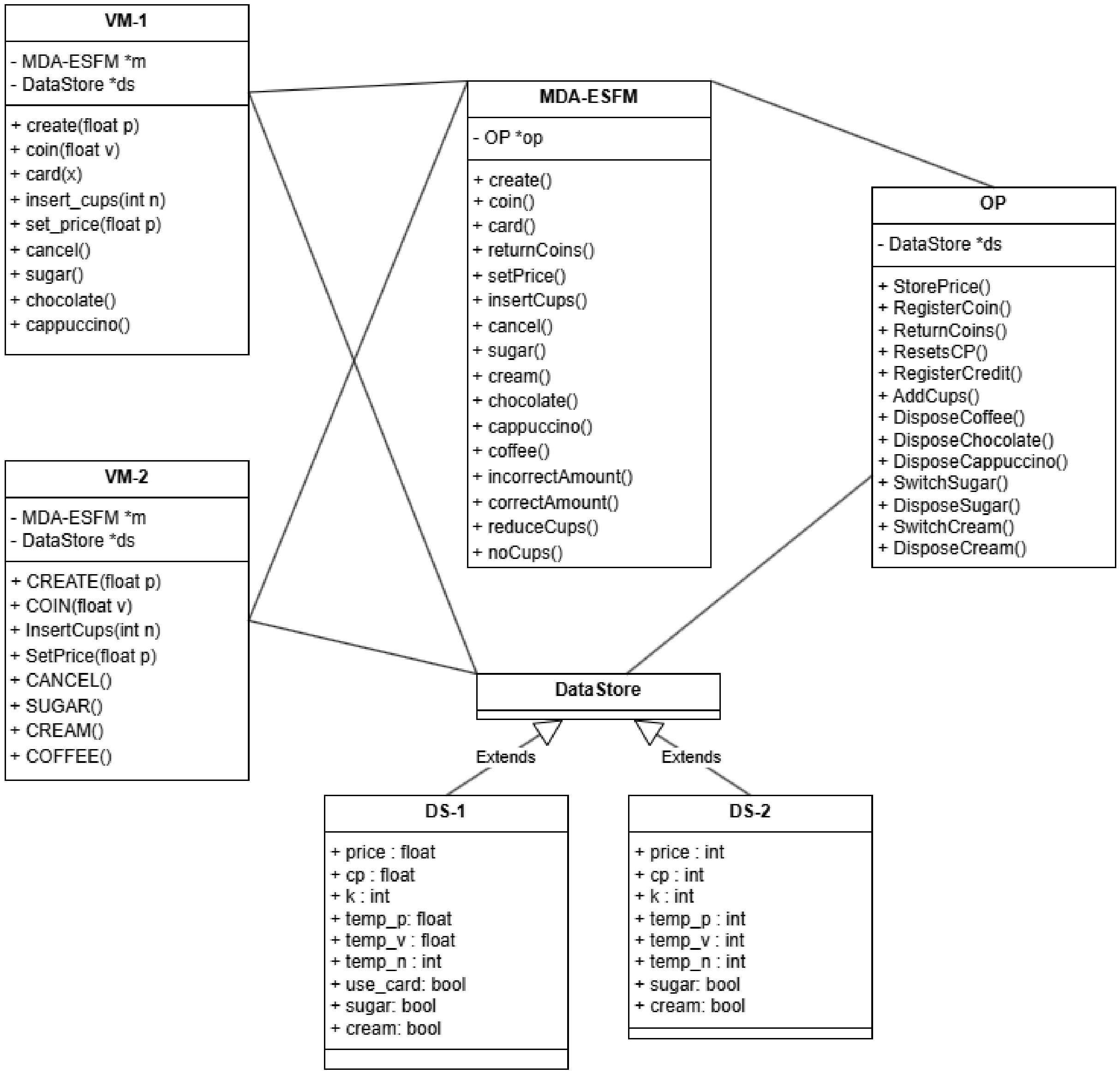
PART 1: Class Diagram



PART 2 AND PART 3:

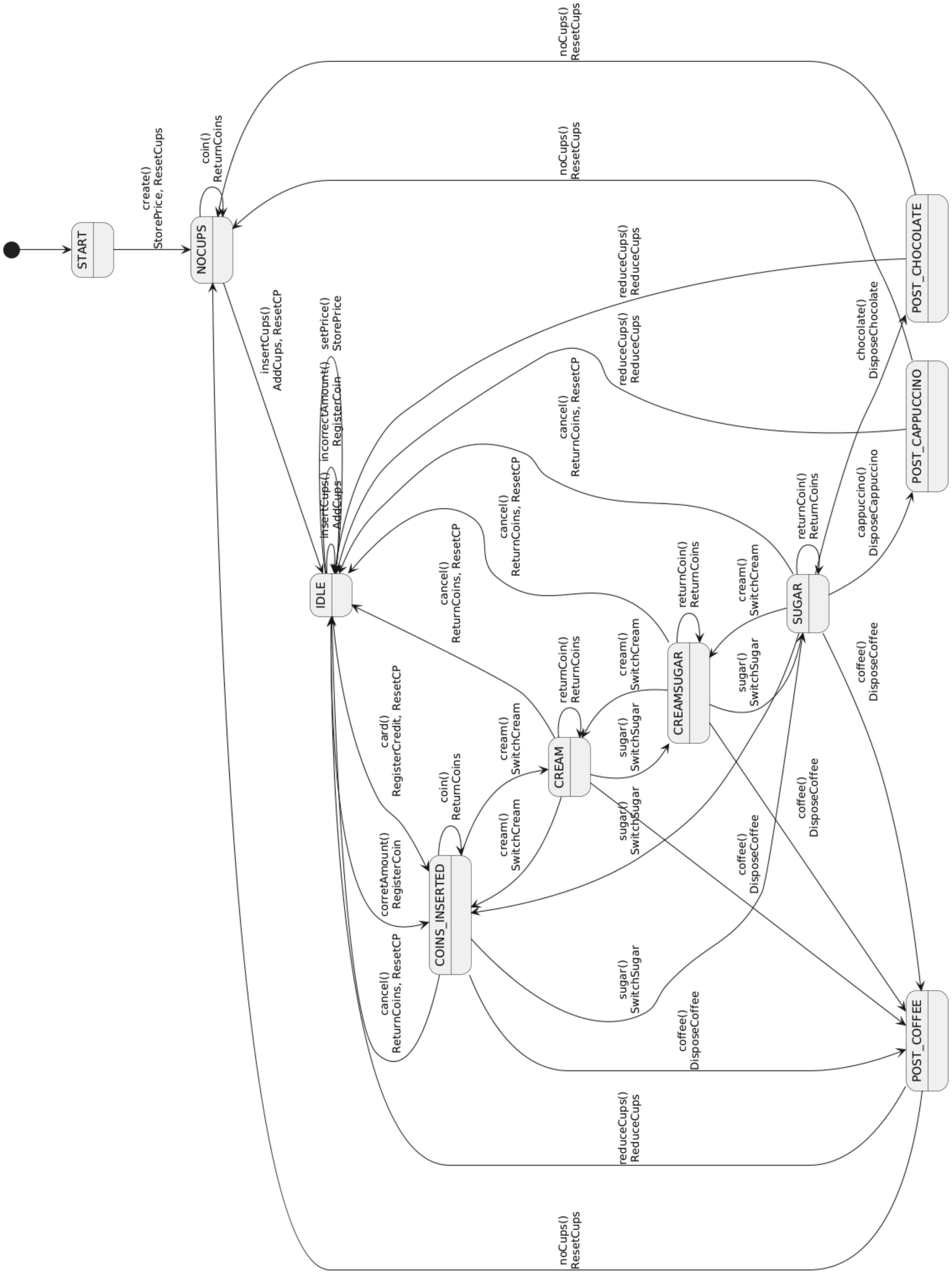
MDA-ESFM Events

m->create()
m->coin()
m->card()
m->returnCoins()
m->setPrice()
m->insertCups()
m->cancel()
m->sugar()
m->cream()
m->chocolate()
m->cappuccino()
m->coffee()
m->incorrectAmount()
m->corretAmount()
m->reduceCups()
m->noCups()

MDA-ESFM Actions

StorePrice	// Saves the temporary price from Start into the price
RegisterCoin	// Increases the amount of money inserted
ReturnCoins	// Returns the amount of money inserted
ReduceCups	// Reduces the amount of cups by n
ResetCups	// Resets the amount of cups to 0
ResetsCP	// Resets the amount of money inserted to 0
RegisterCredit	// Indicates that a credit card is used
AddCups	// Adds cups to the count
DisposeCoffee	// Serves coffee only
DisposeChocolate	// Serves chocolate only
DisposeCappuccino	// Serves cappuccino only
SwitchSugar	// Switches the flag for sugar
DisposeSugar	// Serves sugar only
SwitchCream	// Switches the flag for cream
DisposeCream	// Serves cream only

PART 4: State Diagram



Start State	**Transition**	**Side Effect**	**End State**
START	create()	StorePrice ResetCups	NOCUPS
NOCUPS	coin()	ReturnCoins	NOCUPS
NOCUPS	insertCups()	AddCups ResetCP	IDLE
IDLE	insertCups()	AddCups	IDLE
IDLE	incorrectAmount()	RegisterCoin	IDLE
IDLE	corretAmount()	RegisterCoin	COINS_INSERTED
IDLE	setPrice()	StorePrice	IDLE
IDLE	card()	RegisterCredit ResetCP	COINS_INSERTED
COINS_INSERTED	coin()	ReturnCoins	COINS_INSERTED
COINS_INSERTED	coffee()	DisposeCoffee	POST-COFFEE
COINS_INSERTED	cancel()	ReturnCoins ResetCP	IDLE
COINS_INSERTED	sugar()	SwitchSugar	SUGAR
COINS_INSERTED	cream()	SwitchCream	CREAM
POST-COFFEE	reduceCups()	ReduceCups	IDLE
POST-COFFEE	noCups()	ResetCups	NOCUPS
SUGAR	cappuccino()	DisposeCappuccino	POST-CAPPUCCINO
SUGAR	chocolate()	DisposeChocolate	POST-CHOCOLATE
SUGAR	coffee()	DisposeCoffee	POST-COFFEE
SUGAR	cancel()	ReturnCoins ResetCP	IDLE
SUGAR	cream()	SwitchCream	CREAM+SUGAR
SUGAR	sugar()	SwitchSugar	COINS_INSERTED
SUGAR	returnCoin()	ReturnCoins	SUGAR
POST-CAPPUCCINO	reduceCups()	ReduceCups	IDLE
POST-CAPPUCCINO	noCups()	ResetCups	NOCUPS
POST-CHOCOLATE	reduceCups()	ReduceCups	IDLE
POST-CHOCOLATE	noCups()	ResetCups	NOCUPS
CREAM	coffee()	DisposeCoffee	POST-COFFEE

CREAM	cancel()	ReturnCoins ResetCP	IDLE
CREAM	cream()	SwitchCream	COINS_INSERTED
CREAM	sugar()	SwitchSugar	CREAM+SUGAR
CREAM	returnCoin()	ReturnCoins	CREAM
CREAM+SUGAR	coffee()	DisposeCoffee	POST-COFFEE
CREAM+SUGAR	returnCoin()	ReturnCoins	CREAM+SUGAR
CREAM+SUGAR	cancel()	ReturnCoins ResetCP	IDLE
CREAM+SUGAR	cream()	SwitchCream	CREAM
CREAM+SUGAR	sugar()	SwitchSugar	SUGAR

PART 5: Pseudo-code of all operations of Input Processors of VM-1 and VM-2

Vending Machine 1	Vending Machine 2
create(float p)	CREATE(int p)
d->temp_p = p	d->temp_p = p
m->create()	m->create()
coin(float v)	COIN(int v)
d->temp_v = v	d->temp_v = v
if (m->cp + v < m->price)	if (m->cp + v < m->price)
m->incorrectAmount()	m->incorrectAmount()
else	else
m->correctAmount()	m->correctAmount()
card(x)	InsertCups(int n)
d->temp_x = x	d->temp_n = n
if (d->temp_x > d->price)	if (n > 0)
m->card()	m->insertCups()
insert_cups(int n)	SetPrice(int p)
d->temp_n = n	d->temp_p = p
if (n > 0)	m->setPrice()
m->insertCups()	CANCEL()
set_price(float p)	m->cancel()
d->temp_p = p	SUGAR()
m->setPrice()	m->sugar()
cancel()	CREAM()
m->cancel()	m->cream()
sugar()	COFFEE()
m->sugar()	m->coffee()
chocolate()	if (d->k > 1)
m->chocolate()	m->reduceCups()
if (d->k > 1)	else m->noCups()
m->reduceCups()	
else m->noCups()	
cappuccino()	
m->cappuccino()	
if (d->k > 1)	
m->reduceCups()	
else m->noCups()	