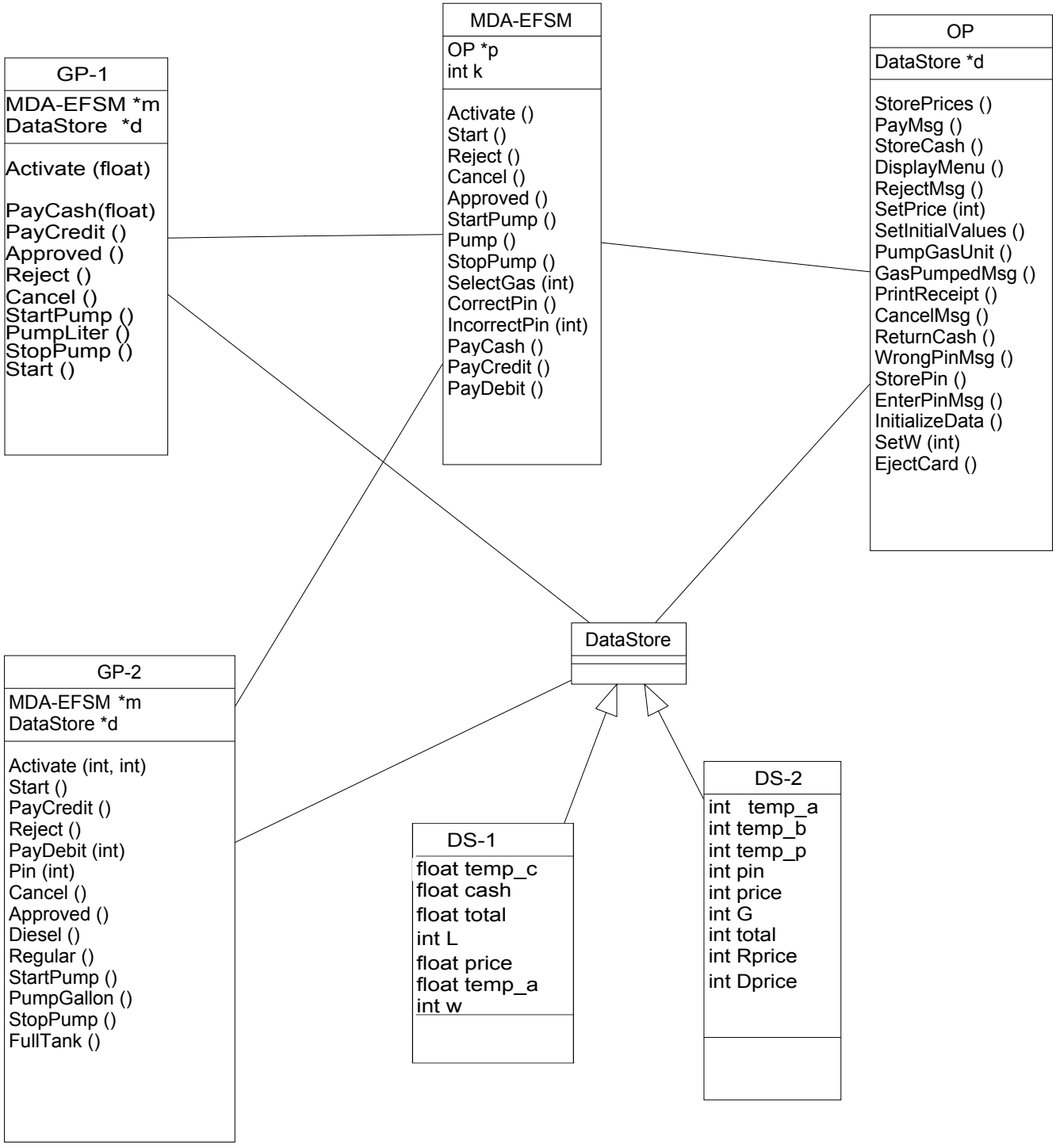
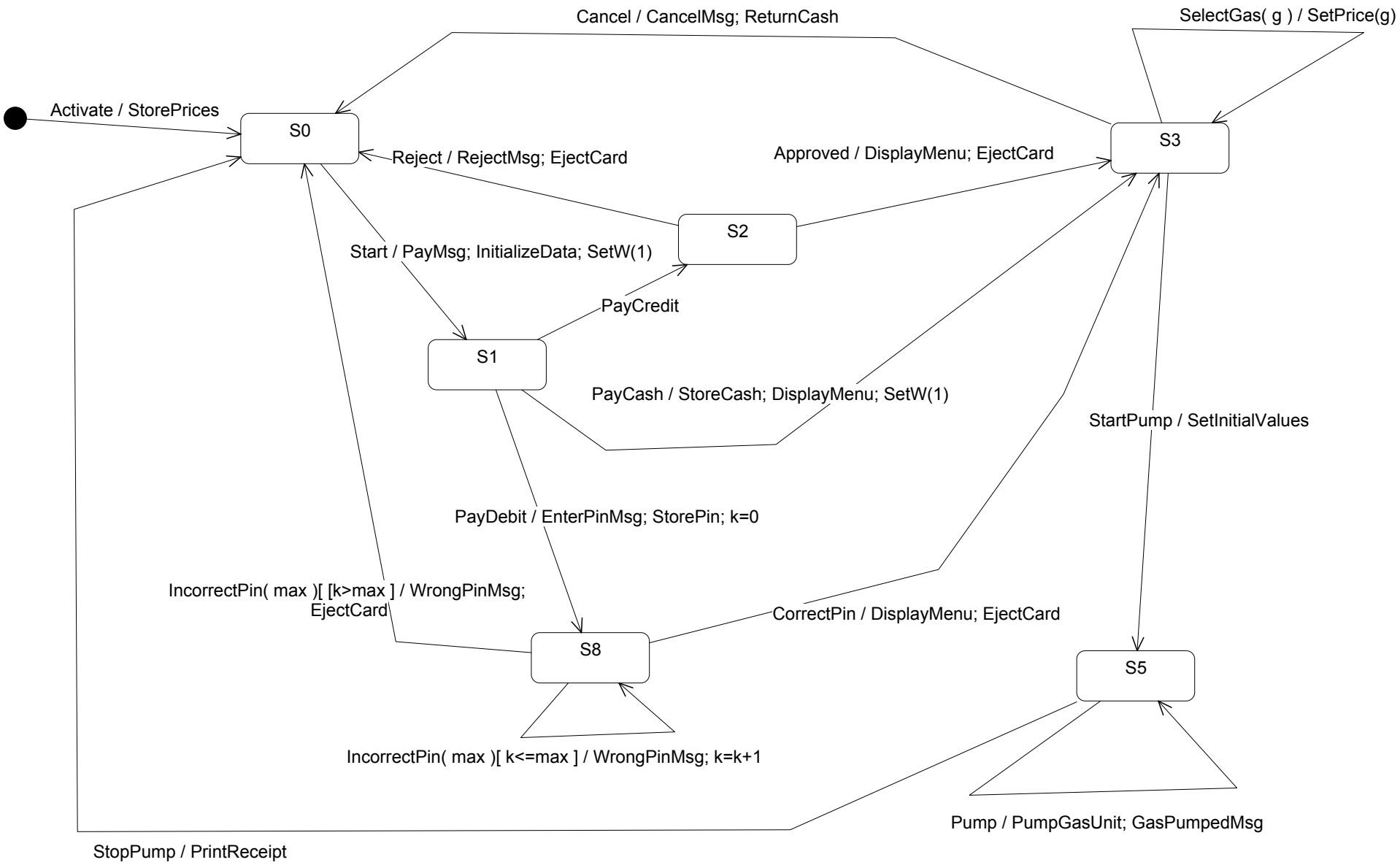

GP-1
MDA-EFSM *m
DataStore *d
Activate (float)
PayCash(float)
PayCredit ()
Approved ()
Reject ()
Cancel ()
StartPump ()
PumpLiter ()
StopPump ()
Start ()
MDA-EFSM
OP *p
int k
Activate ()
Start ()
Reject ()
Cancel ()
Approved ()
StartPump ()
Pump ()
StopPump ()
SelectGas (int)
CorrectPin ()
IncorrectPin (int)
PayCash ()
PayCredit ()
PayDebit ()
OP
DataStore *d
StorePrices ()
PayMsg ()
StoreCash ()
DisplayMenu ()
RejectMsg ()
SetPrice (int)
SetInitialValues ()
PumpGasUnit ()
GasPumpedMsg ()
PrintReceipt ()
CancelMsg ()
ReturnCash ()
WrongPinMsg ()
StorePin ()
EnterPinMsg ()
InitializeData ()
SetW (int)
EjectCard ()
DataStore
GP-2
MDA-EFSM *m
DataStore *d
Activate (int, int)
Start ()
PayCredit ()
Reject ()
PayDebit (int)
Pin (int)
Cancel ()
Approved ()
Diesel ()
Regular ()
StartPump ()
PumpGallon ()
StopPump ()
FullTank ()
DS-1
float temp_c
float cash
float total
int L
float price
float temp_a
int w
DS-2
int temp_a
int temp_b
int temp_p
int pin
int price
int G
int total
int Rprice
int Dprice



MDA-EFSM for Gas Pumps

MDA-EFSM Events:

Activate()
Start()
PayCredit()
PayCash()
PayDebit()
Reject()
Cancel()
Approved()
StartPump()
Pump()
StopPump()
SelectGas(int g)
CorrectPin()
IncorrectPin(int max)

MDA-EFSM Actions:

StorePrices	// stores price(s) for the gas from the temporary data store
PayMsg	// displays a type of payment method
StoreCash	// stores cash from the temporary data store
DisplayMenu	// display a menu with a list of selections
RejectMsg	// displays credit card not approved message
SetPrice(int g)	// set the price for the gas identified by *g* identifier as in SelectGas(int g)
SetInitialValues	// set *G* (or *L*) and *total* to 0;
PumpGasUnit	// disposes unit of gas and counts # of units disposed
GasPumpedMsg	// displays the amount of disposed gas
PrintReceipt	// print a receipt
CancelMsg	// displays a cancellation message
ReturnCash	// returns the remaining cash
WrongPinMsg	// displays incorrect pin message
StorePin	// stores the pin from the temporary data store
EnterPinMsg	// displays a message to enter pin
InitializeData	// set the value of *price* to 0 for GP-2; do nothing for GP-1
EjectCard()	// card is ejected
SetW(int w)	// set value for cash flag

Operations of the Input Processor
(GasPump-1)

```
Activate(float a) {
        if (a>0) {
            d->temp_a=a;
            m->Activate()
        }
}

Start() {
        m->Start();
}
PayCash(float c) {
        if (c>0) {
            d->temp_c=c;
            m->PayCash()
        }
}

PayCredit() {
        m->PayCredit();
}

Reject() {
        m->Reject();
}

Approved() {
        m-> Approved();
}
Cancel() {
        m->Cancel();
}

StartPump() {
        m->StartPump();
}
```

```
PumpLiter() {
if (d->w==1) m->Pump()
else if (d->cash>0)&&(d->cash < d->price*(d->L+1))
            m->StopPump();
else m->Pump()
}

StopPump() {
        m->StopPump();
}
```

Notice:
cash: contains the value of cash deposited
price: contains the price of the selected gas
L: contains the number of liters already pumped
w: cash flag (cash: w=0; otherwise: w=1)
cash, *L*, *price, w* are in the data store
m: is a pointer to the MDA-EFSM object
d: is a pointer to the Data Store object

Operations of the Input Processor
(GasPump-2)

```
Activate(int a, int b) {
        if ((a>0)&&(b>0)) {
              d->temp_a=a;
              d->temp_b=b;
              m->Activate()
        }
}

Start() {
        m->Start();
}

PayCredit() {
        m->PayCredit();
}

Reject() {
        m->Reject();
}

PayDebit(int p) {
        d->temp_p=p;
        m->PayDebit();
}

Pin(int x) {
        if (d->pin==x) m->CorrectPin()
        else m->InCorrectPin(1);
}

Cancel() {
        m->Cancel();
}

Approved() {
        m->Approved();
}

Diesel() {
        m->SelectGas(2)
}

Regular() {
        m->SelectGas(1)
}


StartPump() {
        if (d->price>0) m->StartPump();
}

PumpGallon() {
        m->Pump();

StopPump() {
        m->StopPump();
}

FullTank() {
        m->StopPump();
}
```

Notice:
pin: contains the pin in the data store
m: is a pointer to the MDA-EFSM object
d: is a pointer to the Data Store object
SelectGas(g): Regular: g=1; Diesel: g=2