

```

<efsm> ::=
    efsmId
    numStates startStateIndex exitStateIndex
    <transitions>
<transitions> ::=
    <transition> | <transitions> <transition>
<transition> ::=
    transition transitionId
    sourceStateIndex destinationStateIndex
    <requirement>
<requirement> ::=
    [<input>]
    [<enablingPredicate>]
    /
    [<actions>]
<actions> ::=
    <action> | <actions> <action>
<action> ::=
    <output> | <assignment> | <set> | <reset> | <procedureCall>
<input> ::=
    inputId ( [<parameters>] )
<output> ::=
    outputId ( [<parameters>] )
<enablingPredicate> ::=
    <variableIds> [/* booleanExpression */]
<assignment> ::=
    <variableId> := <expression>
<set> ::=
    set ( constant , timerId )
<reset> ::=
    reset ( timerId )
procedureCall ::=
    procedure ( procedureId ( <variableIds> [; <variableIds>]
    ) ) { <pbrDefs> }
<parameters> ::=
    <parameter> { , <parameter> } *
<parameter> ::=
    <variableId> | constant
<variableIds> ::=
    <variableId> { , <variableId> } *
<pbrDefs> ::=
    <pbrDef> | <pbrDefs> <pbrDef>
<pbrDef> ::=
    <variableId> := <expression> ;
<expression> ::=
    function ( <variableIds> ) | constant
<variableId> ::=
    id

```

ATM_System

5 0 4

transition T1

0 1

Card(pin, b) /

Prompt_for_PIN()

attempts := constant

transition T2

1 1

PIN(p)

p, pin, attempts /* [(p!=pin) and (attempts<3)] */ /

Display_error()

attempts := function(attempts)

Prompt_for_PIN()

transition T3

1 4

PIN(p)

p, pin, attempts /* [(p!=pin) and (attempts==3)] */ /

Display_error()

Eject_card()

transition T4

1 2

PIN(p)

p, pin /* [p==pin] */ /

Display_menu()

transition T5

2 3

Withdrawal(w) /

b := function(b, w)

transition T6

2 3

Deposit(d) /

b := function(b, d)

transition T7

2 3

Balance() /

print(b)

transition T8

3 2

Continue() /

Display_menu()

transition T9

2 4

Exit() /

Eject_card()

