

Kurs Datenbankgrundlagen und Modellierung

Sebastian Maneth, Universität Bremen

smaneth@uni-bremen.de

SoSe 2023

21.06.2023

Tutorial 7: UML State Charts

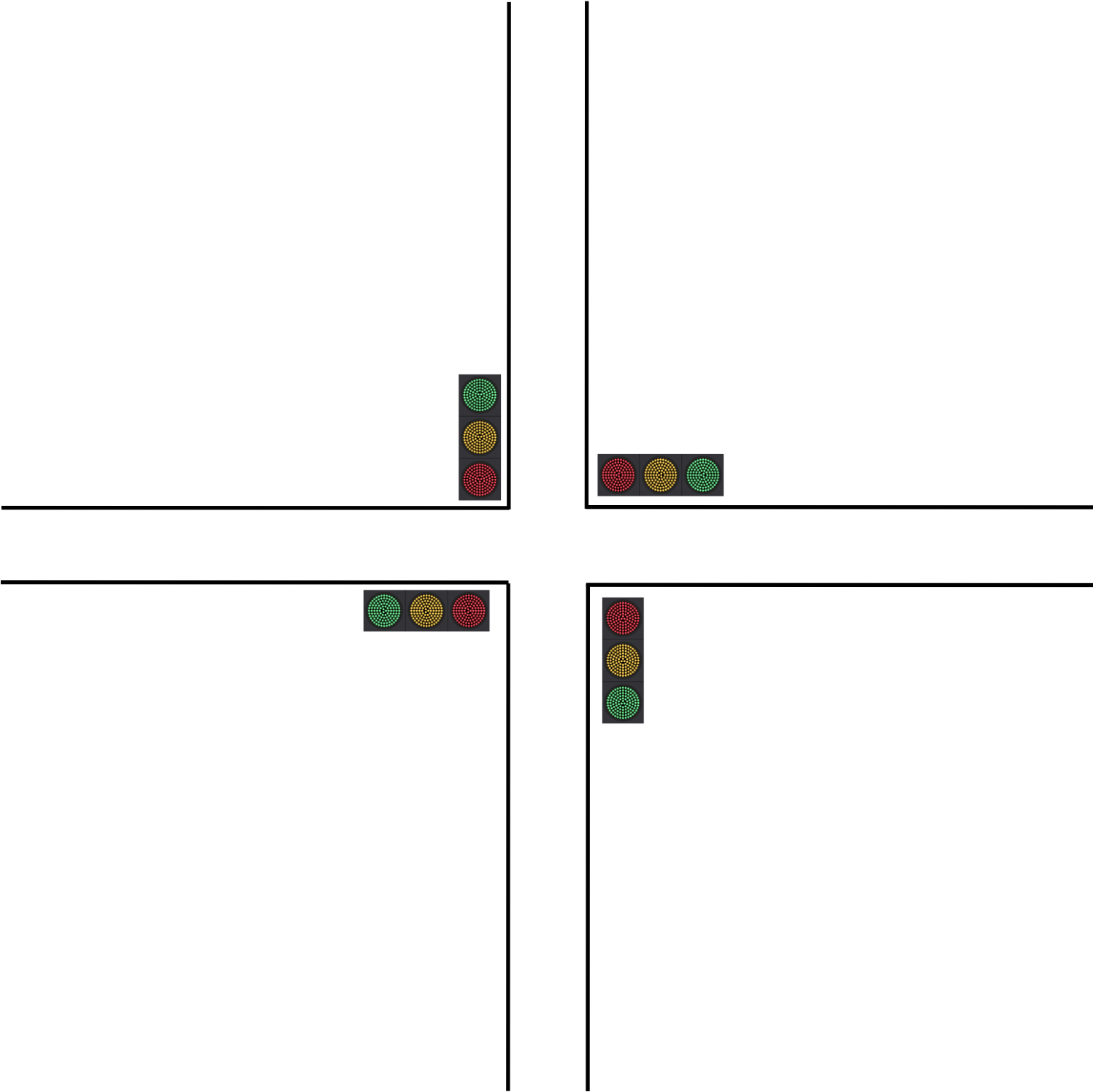
Tutor: Marcel Westenberg

Agenda

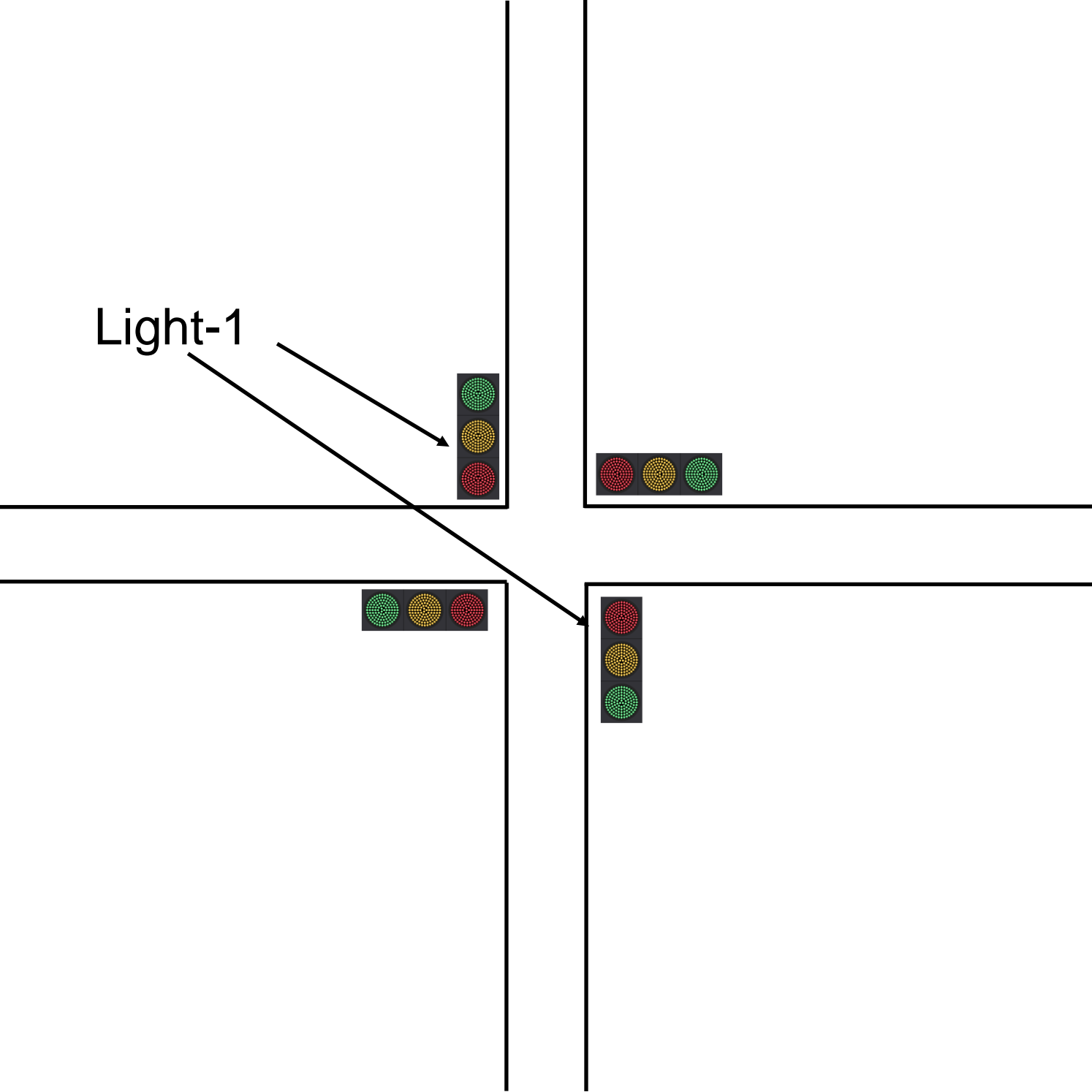
- 1.) State Chart for Traffic Light
- 2.) Anrufbeantworter
- 3.) Fuel Pump
- 4.) Conversion: Regions to “Combined” Automaton

Consider a traffic light system at a four-way crossroads (e.g., two roads intersecting at right angles). Assume the simplest algorithm for cycling through the lights (e.g., all traffic on one road is allowed to go through the crossroad while the other traffic is stopped). Identify the states of this system and draw a statechart describing them. Remember that each individual traffic light has three states (i.e. green, yellow, and red).

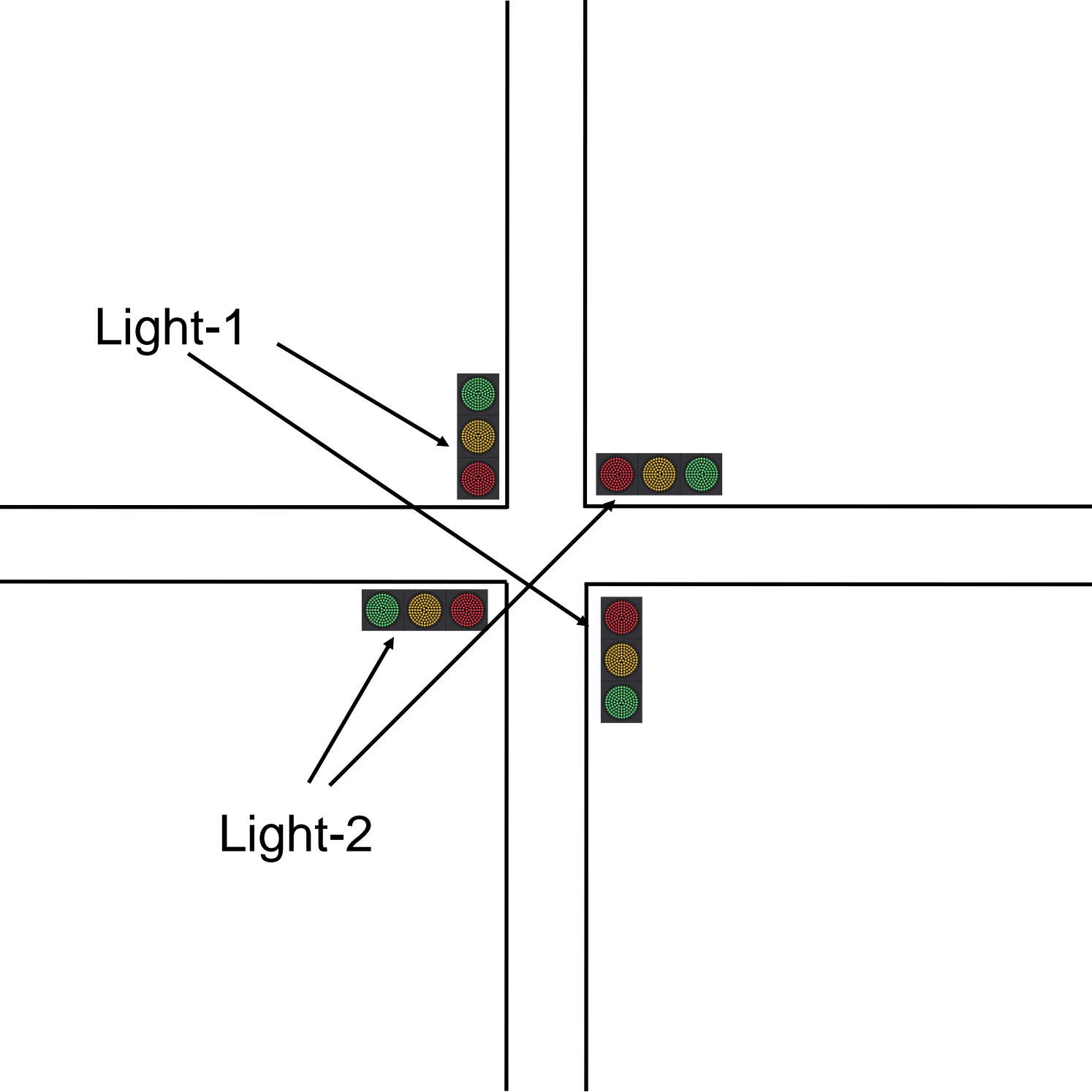
Consider a traffic light system at a four-way crossroads (e.g., two roads intersecting at right angles). Assume the simplest algorithm for cycling through the lights (e.g., all traffic on one road is allowed to go through the crossroad while the other traffic is stopped). Identify the states of this system and draw a statechart describing them. Remember that each individual traffic light has three states (i.e. green, yellow, and red).



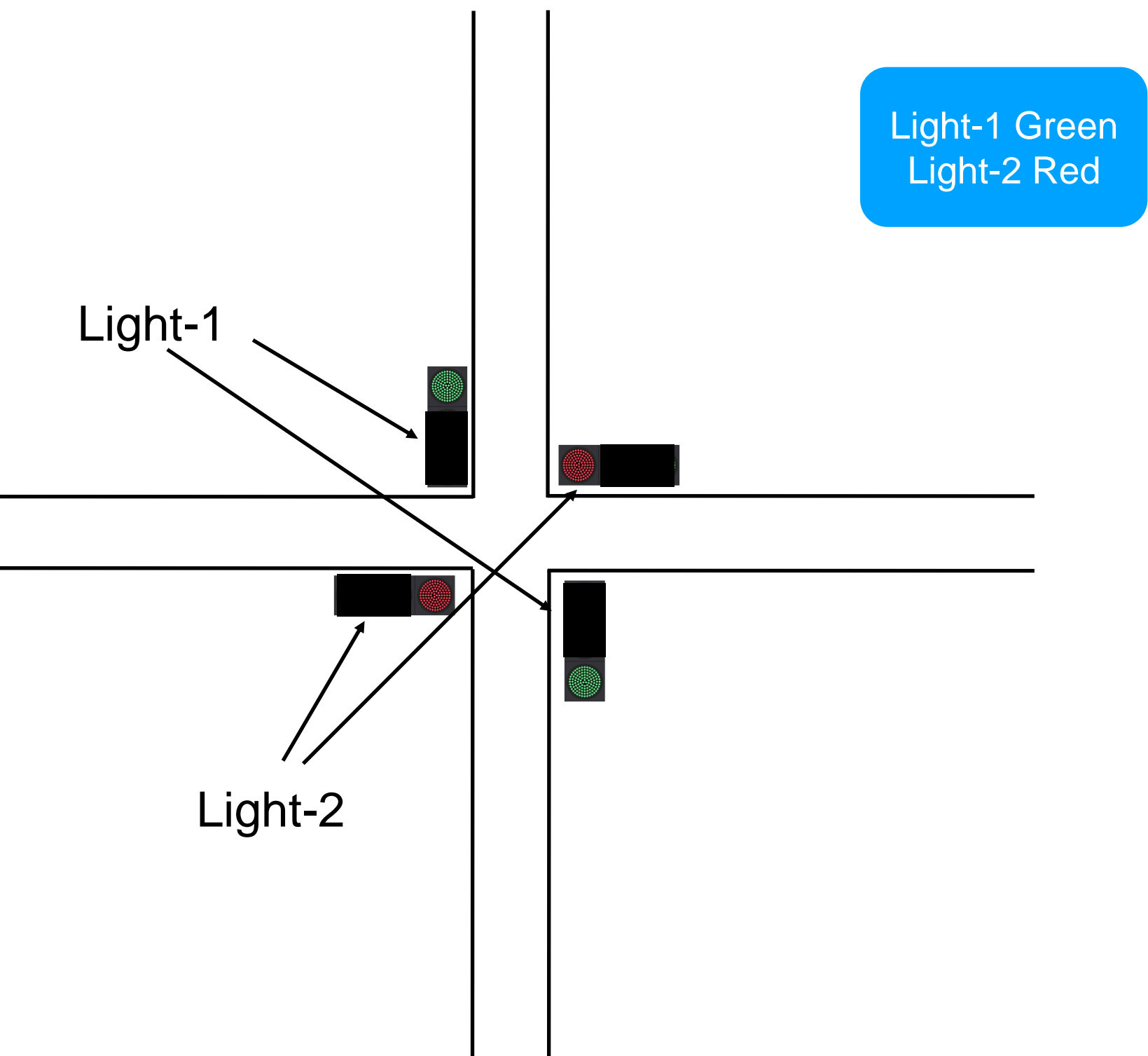
Consider a traffic light system at a four-way crossroads (e.g., two roads intersecting at right angles). Assume the simplest algorithm for cycling through the lights (e.g., all traffic on one road is allowed to go through the crossroad while the other traffic is stopped). Identify the states of this system and draw a statechart describing them. Remember that each individual traffic light has three states (i.e. green, yellow, and red).



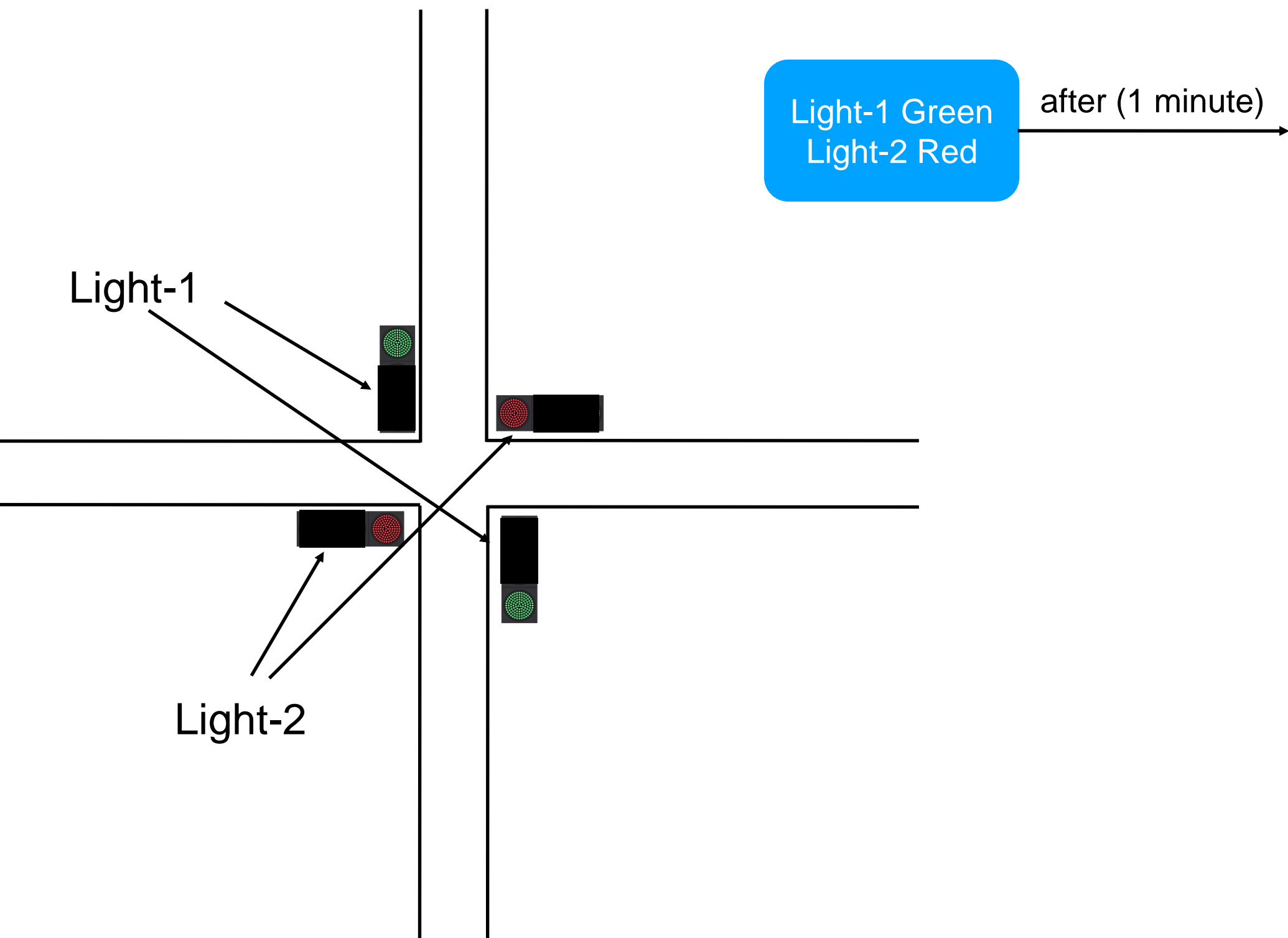
Consider a traffic light system at a four-way crossroads (e.g., two roads intersecting at right angles). Assume the simplest algorithm for cycling through the lights (e.g., all traffic on one road is allowed to go through the crossroad while the other traffic is stopped). Identify the states of this system and draw a statechart describing them. Remember that each individual traffic light has three states (i.e. green, yellow, and red).



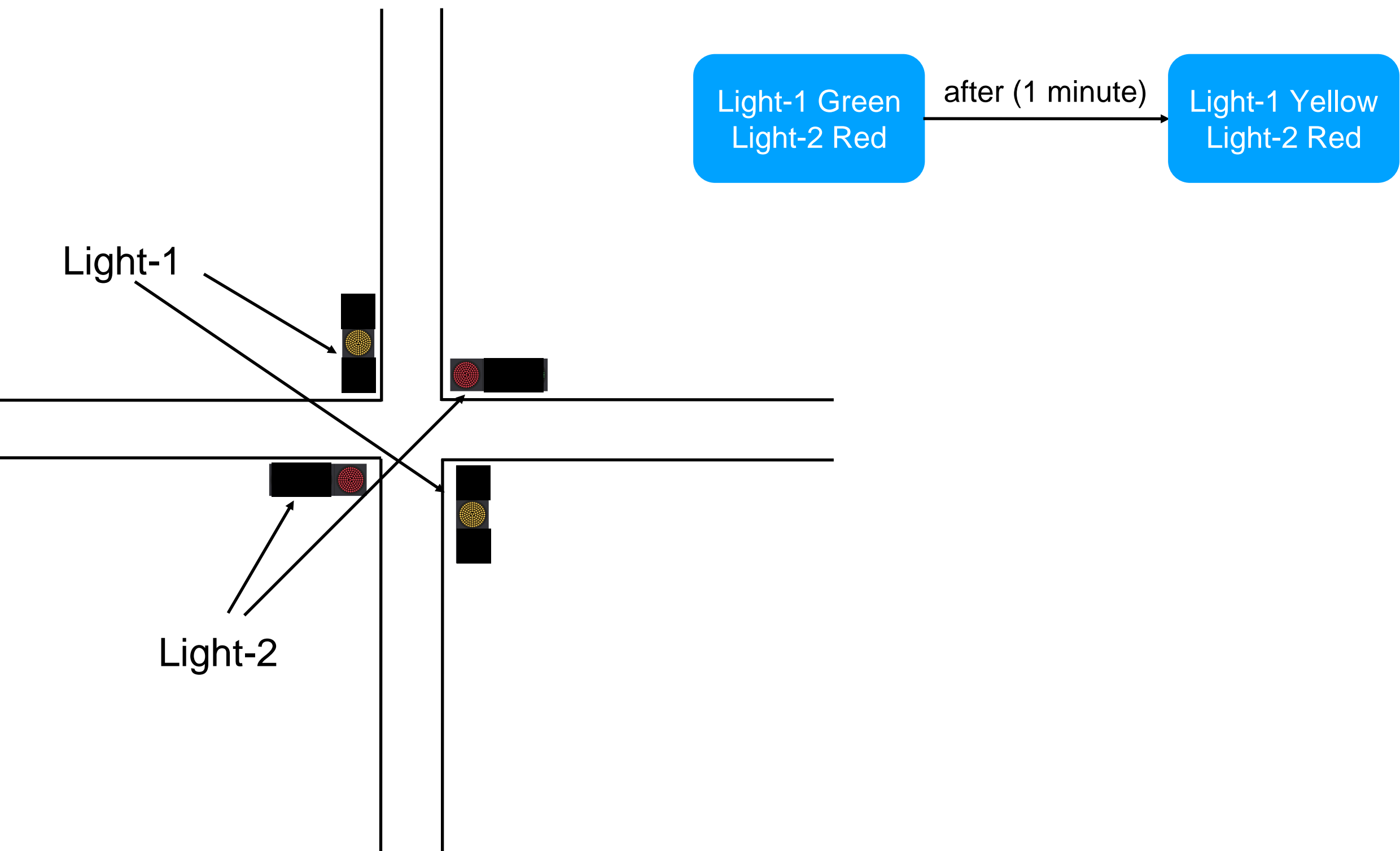
Consider a traffic light system at a four-way crossroads (e.g., two roads intersecting at right angles). Assume the simplest algorithm for cycling through the lights (e.g., all traffic on one road is allowed to go through the crossroad while the other traffic is stopped). Identify the states of this system and draw a statechart describing them. Remember that each individual traffic light has three states (i.e. green, yellow, and red).



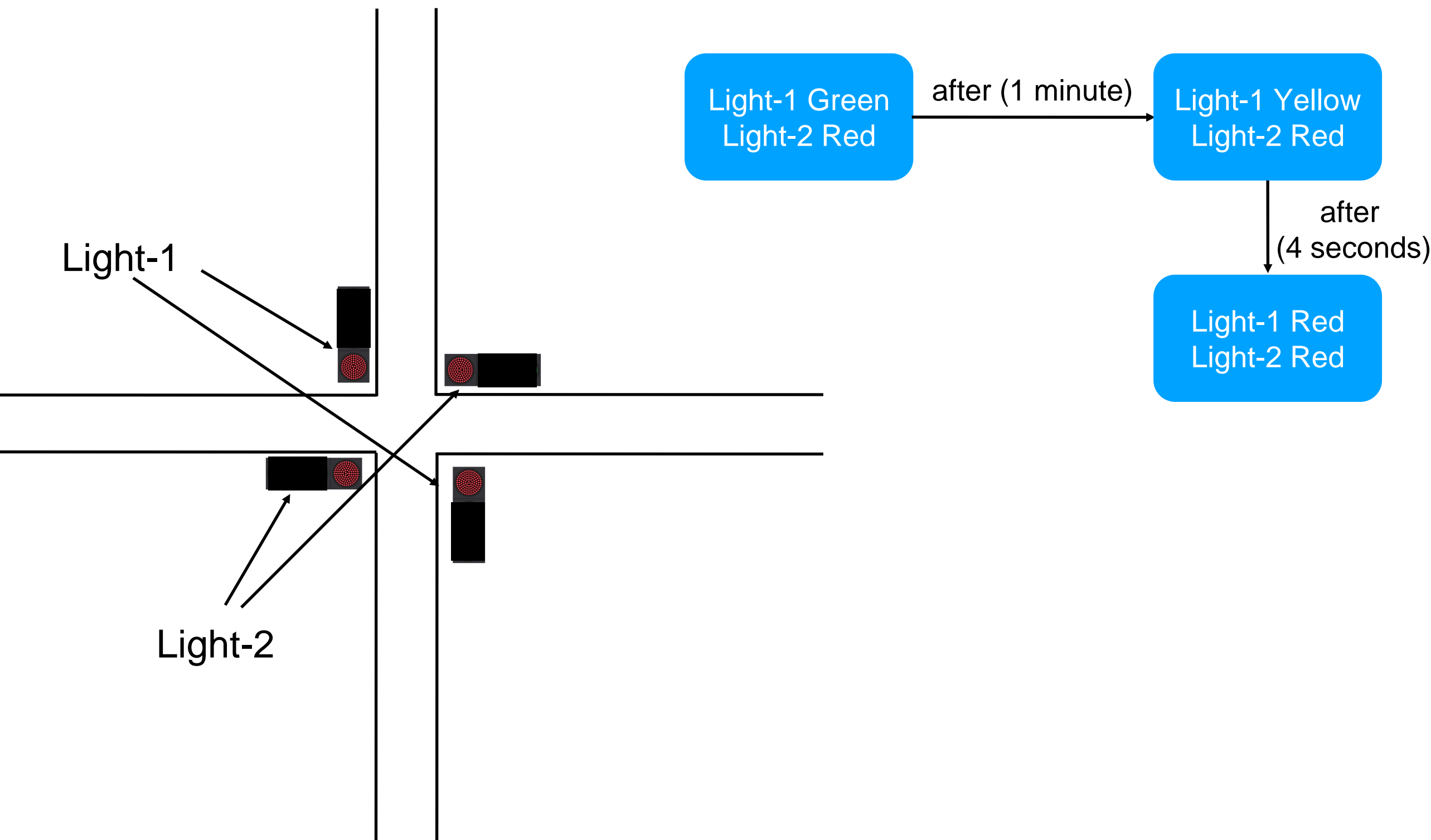
Consider a traffic light system at a four-way crossroads (e.g., two roads intersecting at right angles). Assume the simplest algorithm for cycling through the lights (e.g., all traffic on one road is allowed to go through the crossroad while the other traffic is stopped). Identify the states of this system and draw a statechart describing them. Remember that each individual traffic light has three states (i.e. green, yellow, and red).



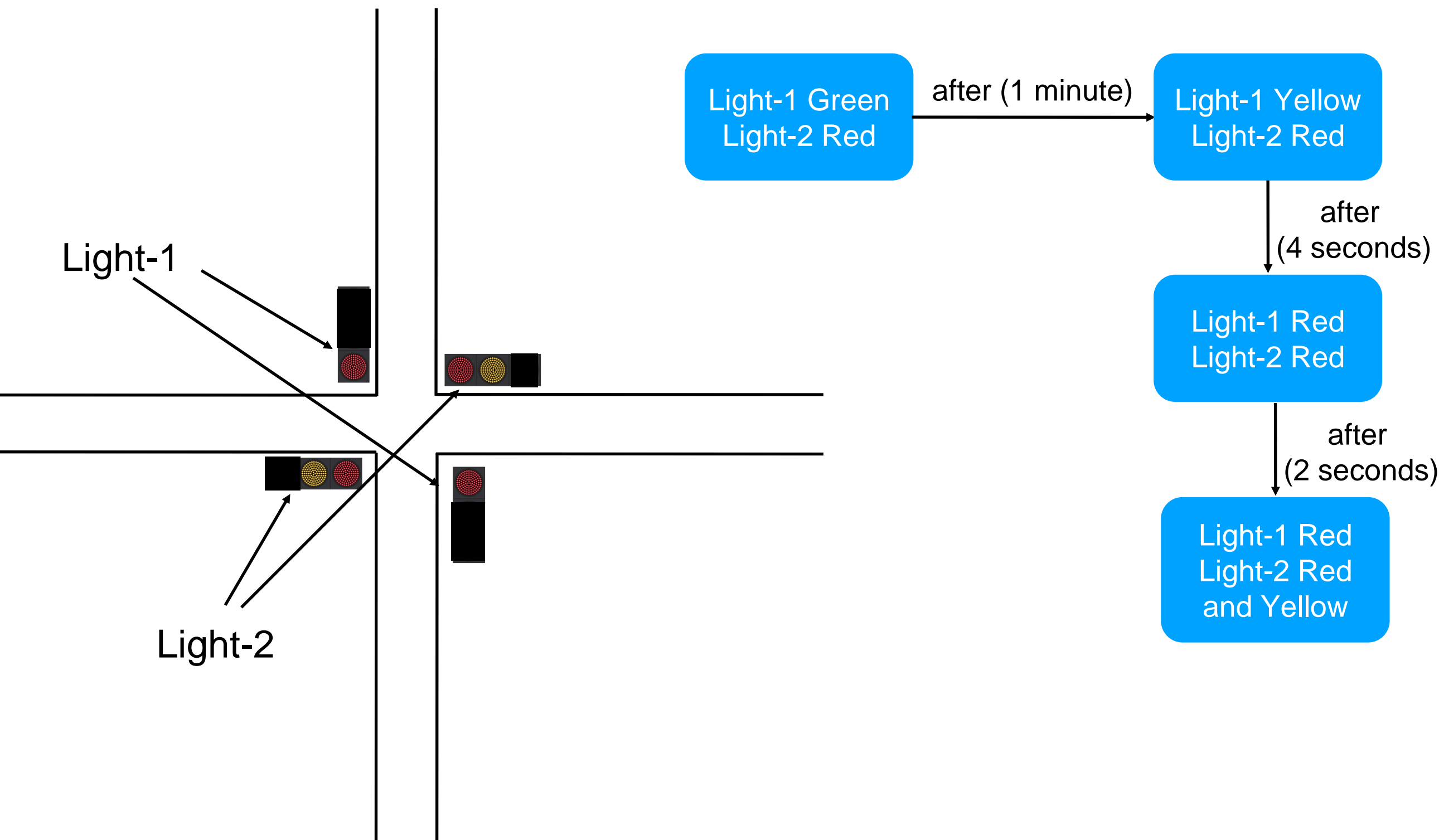
Consider a traffic light system at a four-way crossroads (e.g., two roads intersecting at right angles). Assume the simplest algorithm for cycling through the lights (e.g., all traffic on one road is allowed to go through the crossroad while the other traffic is stopped). Identify the states of this system and draw a statechart describing them. Remember that each individual traffic light has three states (i.e. green, yellow, and red).



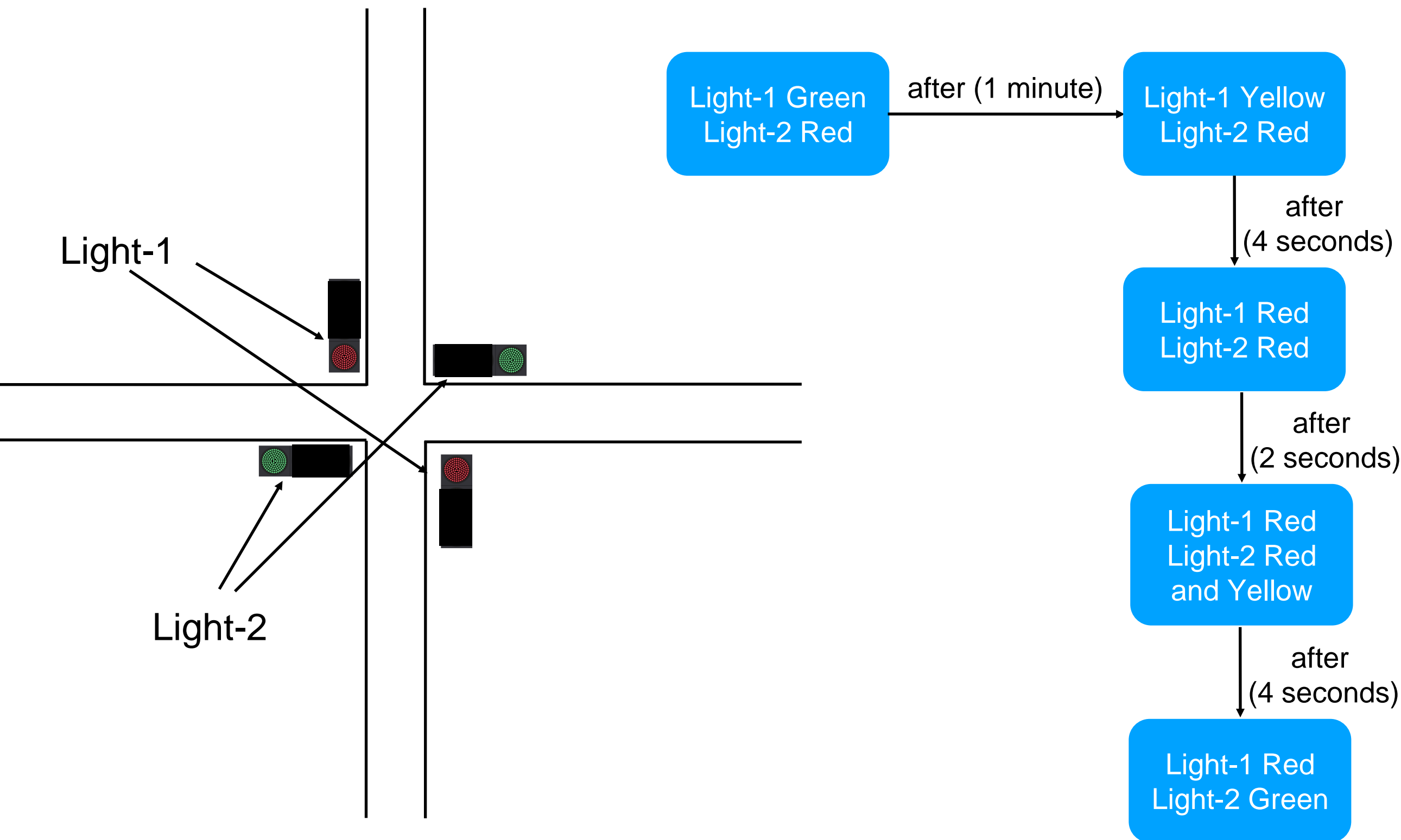
Consider a traffic light system at a four-way crossroads (e.g., two roads intersecting at right angles). Assume the simplest algorithm for cycling through the lights (e.g., all traffic on one road is allowed to go through the crossroad while the other traffic is stopped). Identify the states of this system and draw a statechart describing them. Remember that each individual traffic light has three states (i.e. green, yellow, and red).



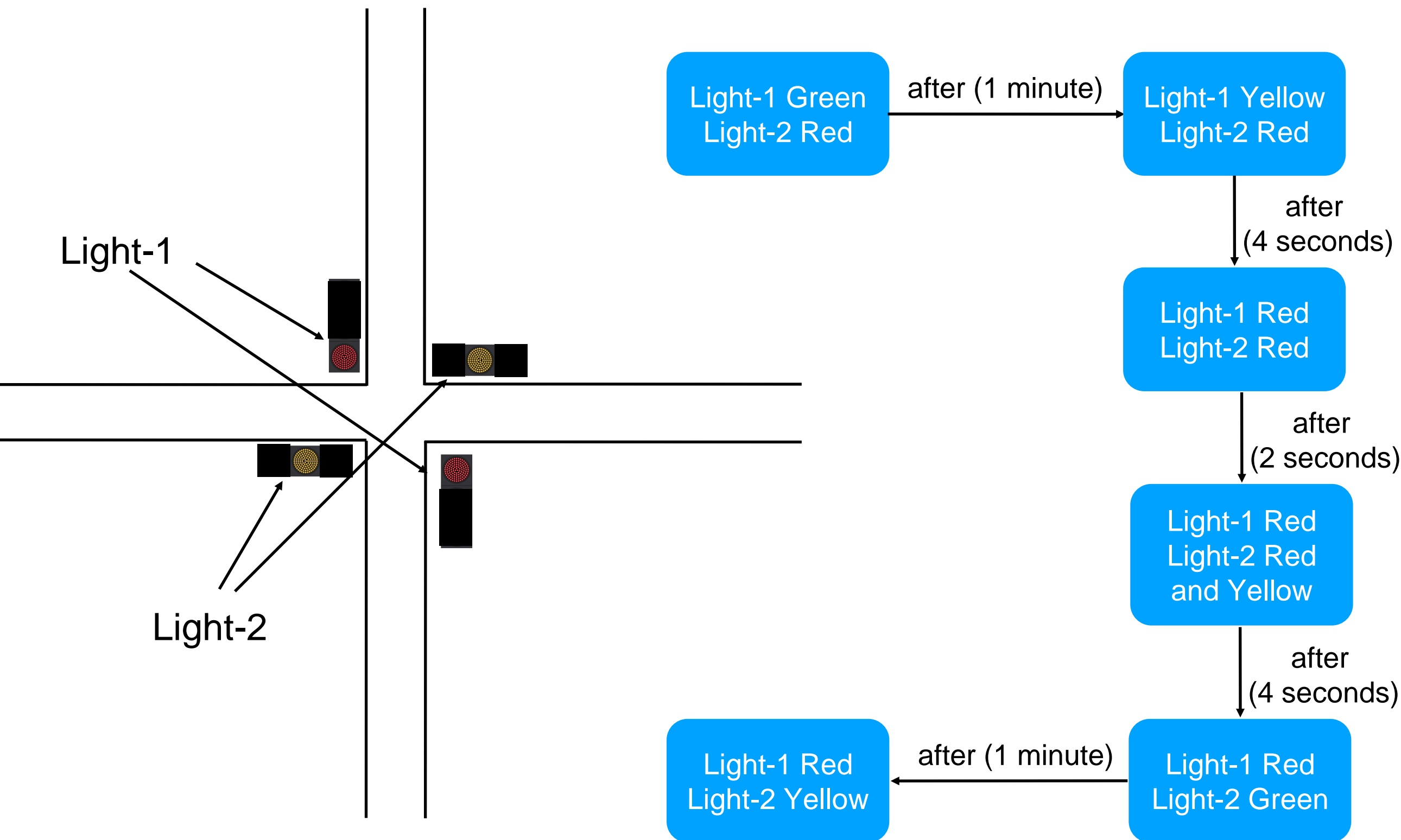
Consider a traffic light system at a four-way crossroads (e.g., two roads intersecting at right angles). Assume the simplest algorithm for cycling through the lights (e.g., all traffic on one road is allowed to go through the crossroad while the other traffic is stopped). Identify the states of this system and draw a statechart describing them. Remember that each individual traffic light has three states (i.e. green, yellow, and red).



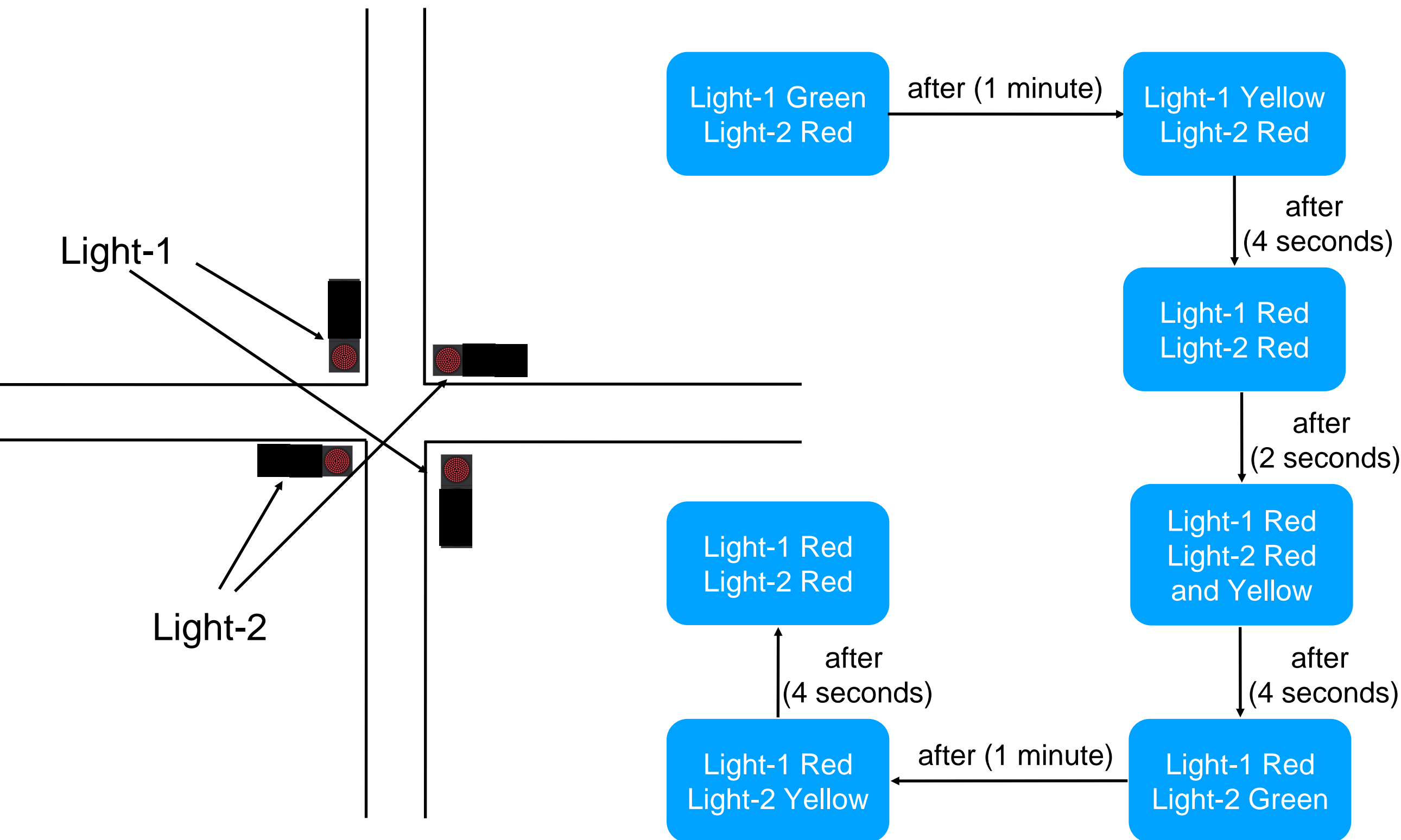
Consider a traffic light system at a four-way crossroads (e.g., two roads intersecting at right angles). Assume the simplest algorithm for cycling through the lights (e.g., all traffic on one road is allowed to go through the crossroad while the other traffic is stopped). Identify the states of this system and draw a statechart describing them. Remember that each individual traffic light has three states (i.e. green, yellow, and red).



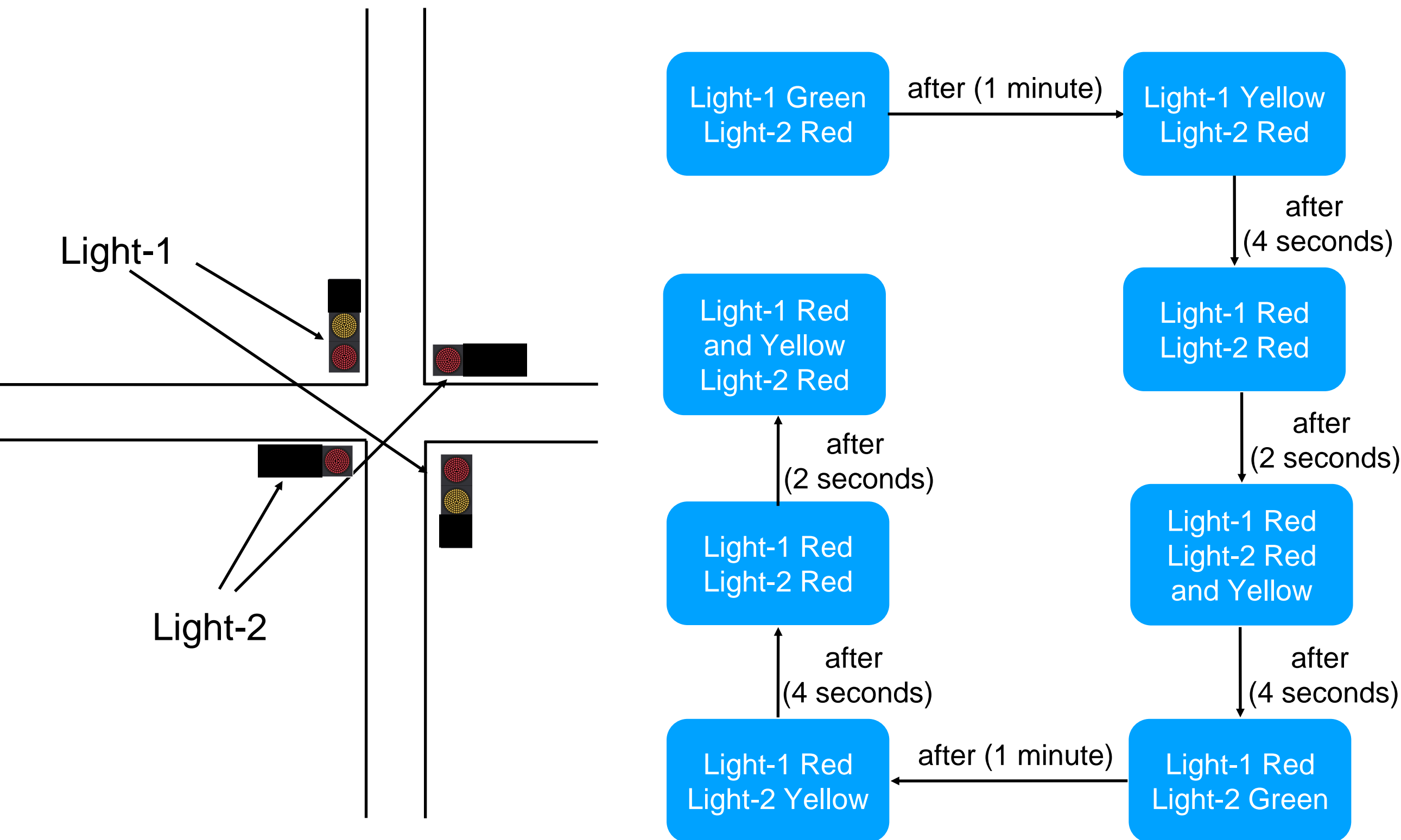
Consider a traffic light system at a four-way crossroads (e.g., two roads intersecting at right angles). Assume the simplest algorithm for cycling through the lights (e.g., all traffic on one road is allowed to go through the crossroad while the other traffic is stopped). Identify the states of this system and draw a statechart describing them. Remember that each individual traffic light has three states (i.e. green, yellow, and red).



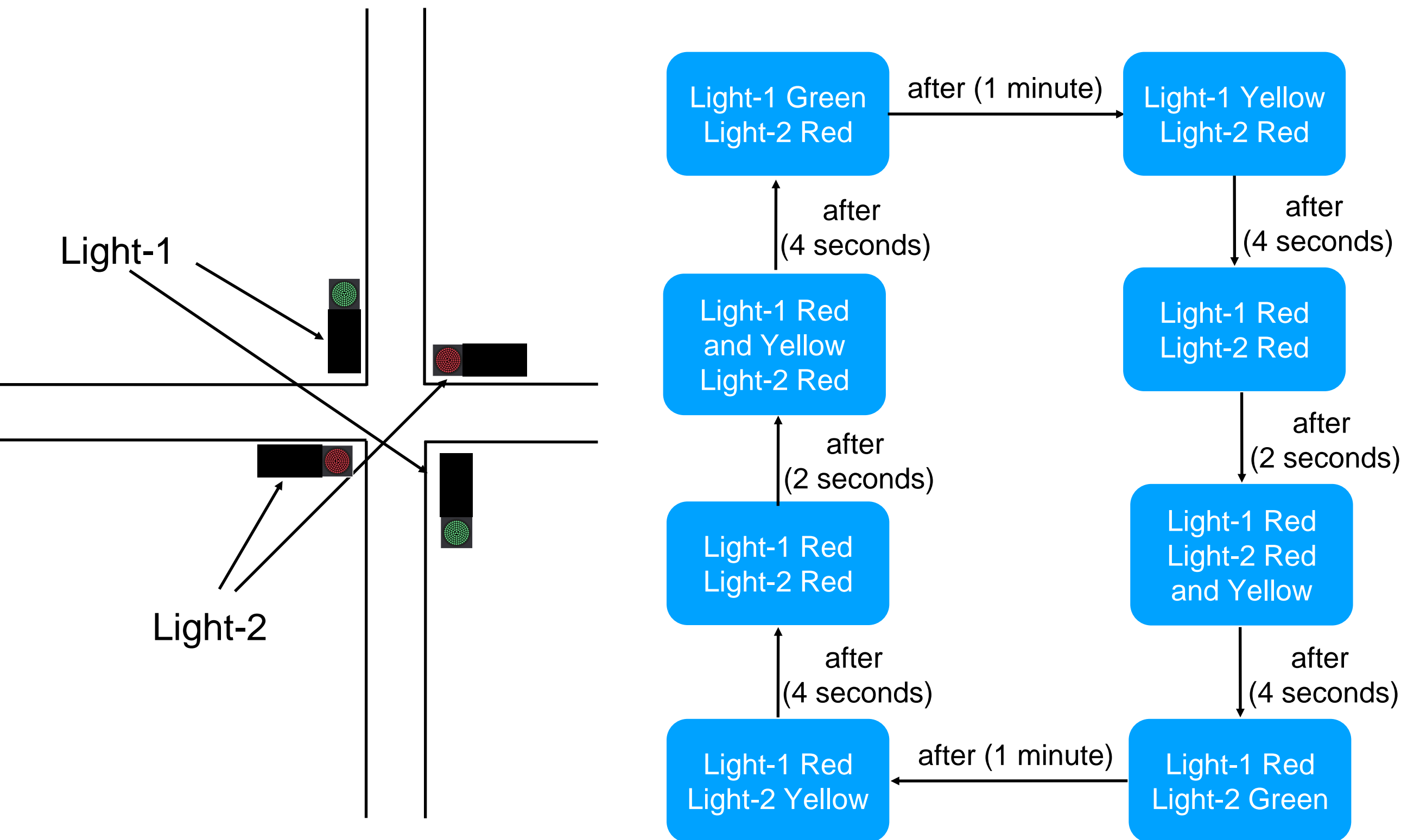
Consider a traffic light system at a four-way crossroads (e.g., two roads intersecting at right angles). Assume the simplest algorithm for cycling through the lights (e.g., all traffic on one road is allowed to go through the crossroad while the other traffic is stopped). Identify the states of this system and draw a statechart describing them. Remember that each individual traffic light has three states (i.e. green, yellow, and red).



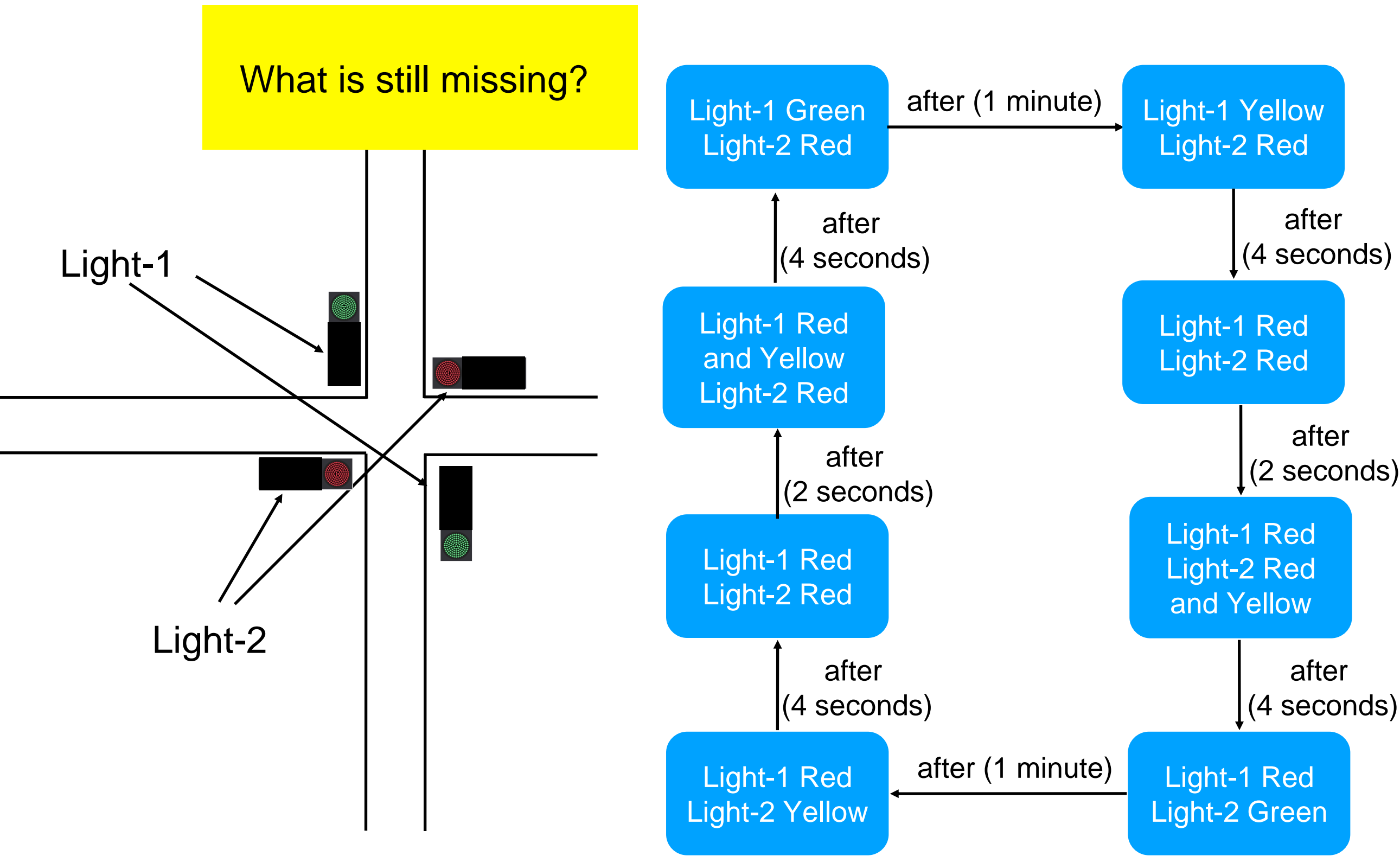
Consider a traffic light system at a four-way crossroads (e.g., two roads intersecting at right angles). Assume the simplest algorithm for cycling through the lights (e.g., all traffic on one road is allowed to go through the crossroad while the other traffic is stopped). Identify the states of this system and draw a statechart describing them. Remember that each individual traffic light has three states (i.e. green, yellow, and red).



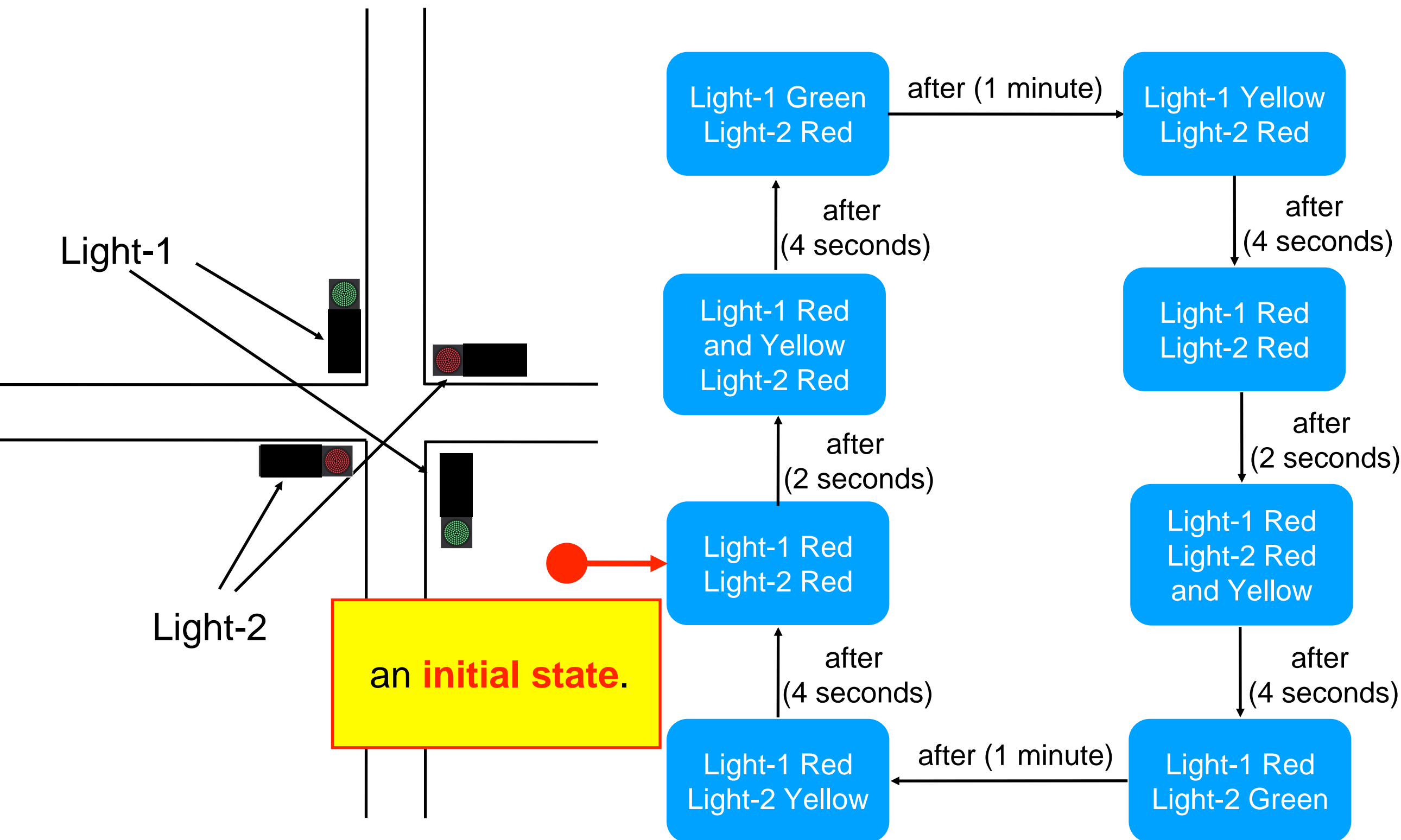
Consider a traffic light system at a four-way crossroads (e.g., two roads intersecting at right angles). Assume the simplest algorithm for cycling through the lights (e.g., all traffic on one road is allowed to go through the crossroad while the other traffic is stopped). Identify the states of this system and draw a statechart describing them. Remember that each individual traffic light has three states (i.e. green, yellow, and red).



Consider a traffic light system at a four-way crossroads (e.g., two roads intersecting at right angles). Assume the simplest algorithm for cycling through the lights (e.g., all traffic on one road is allowed to go through the crossroad while the other traffic is stopped). Identify the states of this system and draw a statechart describing them. Remember that each individual traffic light has three states (i.e. green, yellow, and red).

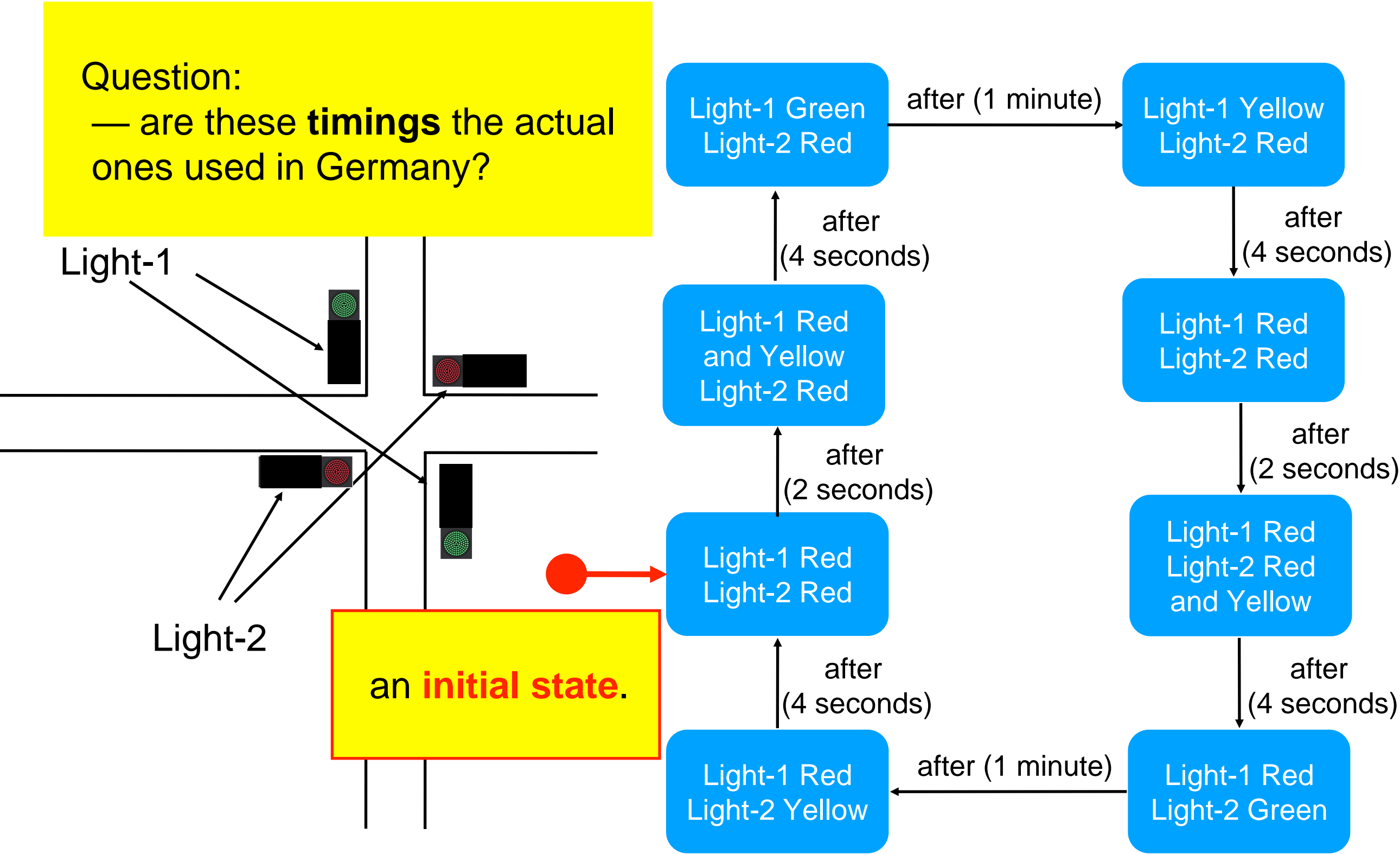


Consider a traffic light system at a four-way crossroads (e.g., two roads intersecting at right angles). Assume the simplest algorithm for cycling through the lights (e.g., all traffic on one road is allowed to go through the crossroad while the other traffic is stopped). Identify the states of this system and draw a statechart describing them. Remember that each individual traffic light has three states (i.e. green, yellow, and red).



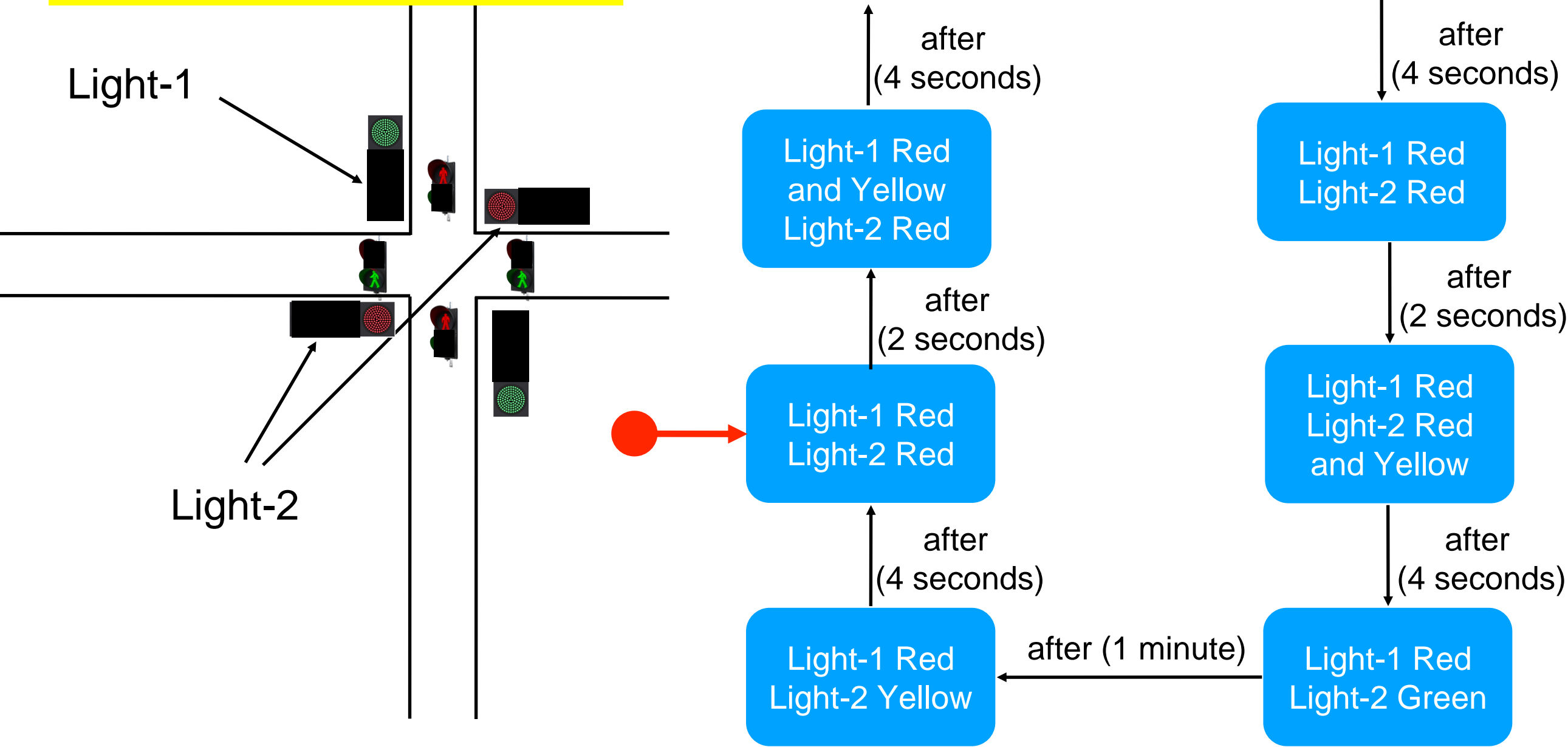
Consider a traffic light system at a four-way crossroads (e.g., two roads intersecting at right angles). Assume the simplest algorithm for cycling through the lights (e.g., all traffic on one road is allowed to go through the crossroad while the other traffic is stopped). Identify the states of this system and draw a statechart describing them. Remember that each individual traffic light has three states (i.e. green, yellow, and red).

Question:
— are these **timings** the actual ones used in Germany?



Consider a traffic light system at a four-way crossroads (e.g., two roads intersecting at right angles). Assume the simplest algorithm for cycling through the lights (e.g., all traffic on one road is allowed to go through the crossroad while the other traffic is stopped). Identify the states of this system and draw a statechart describing them. Remember that each individual traffic light has three states (i.e. green, yellow, and red).

Task:
— add pedestrian lights.

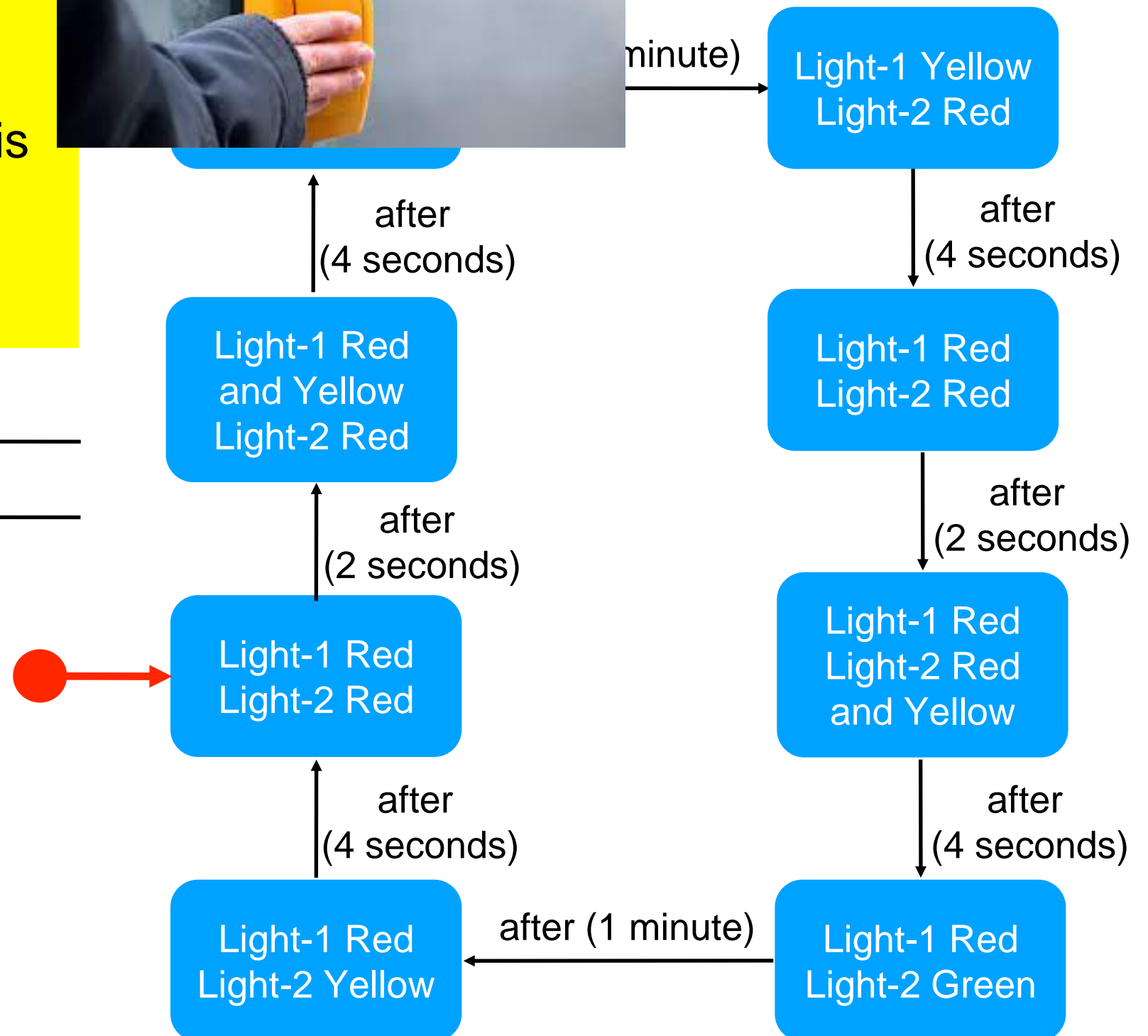
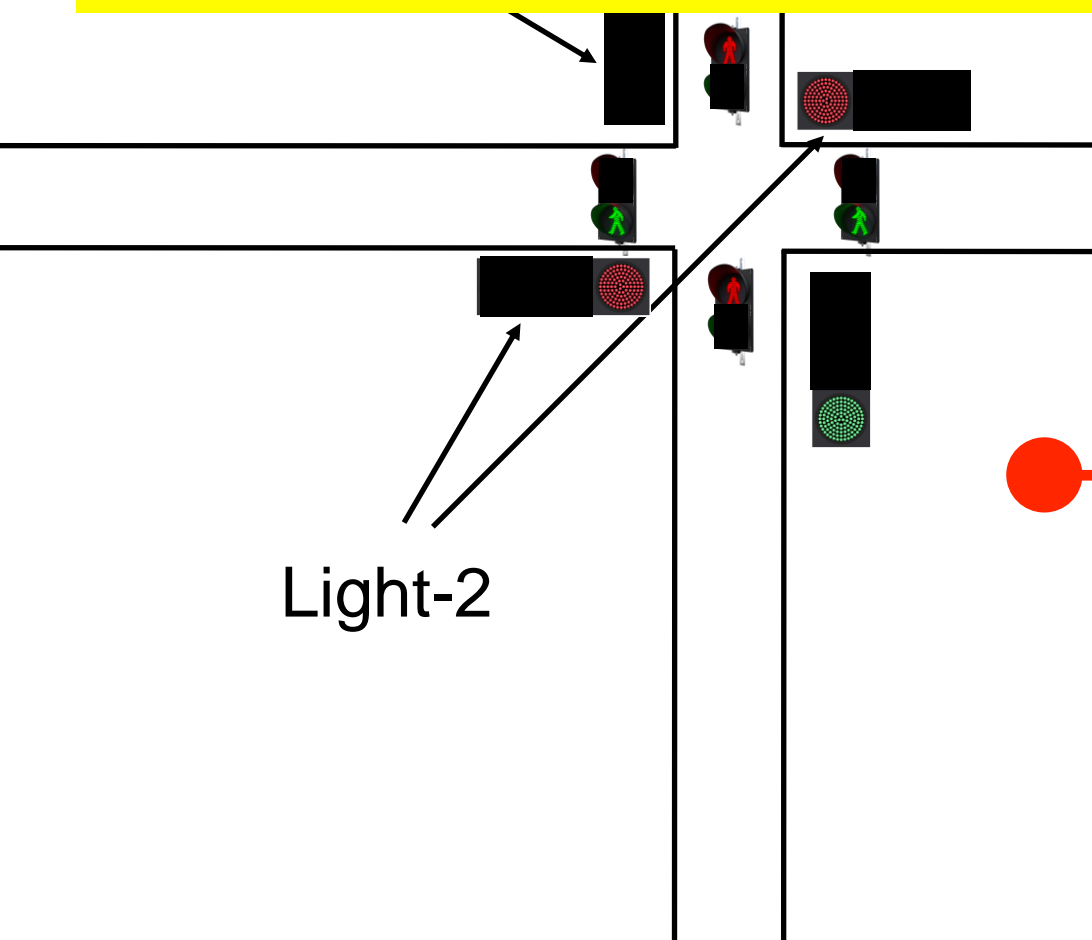


Consider a traffic light system at a four-way crossroads (e.g., two roads intersecting at right angles). Assume the simplest algorithm for cycling through the lights (e.g., all traffic on one road is allowed to go through the crossroads). Identify the states of this system and draw a statechart describing the states (green, yellow, and red).

Task:

- add pedestrian lights.
- with “request” button.

What if the request button is pressed to frequently?



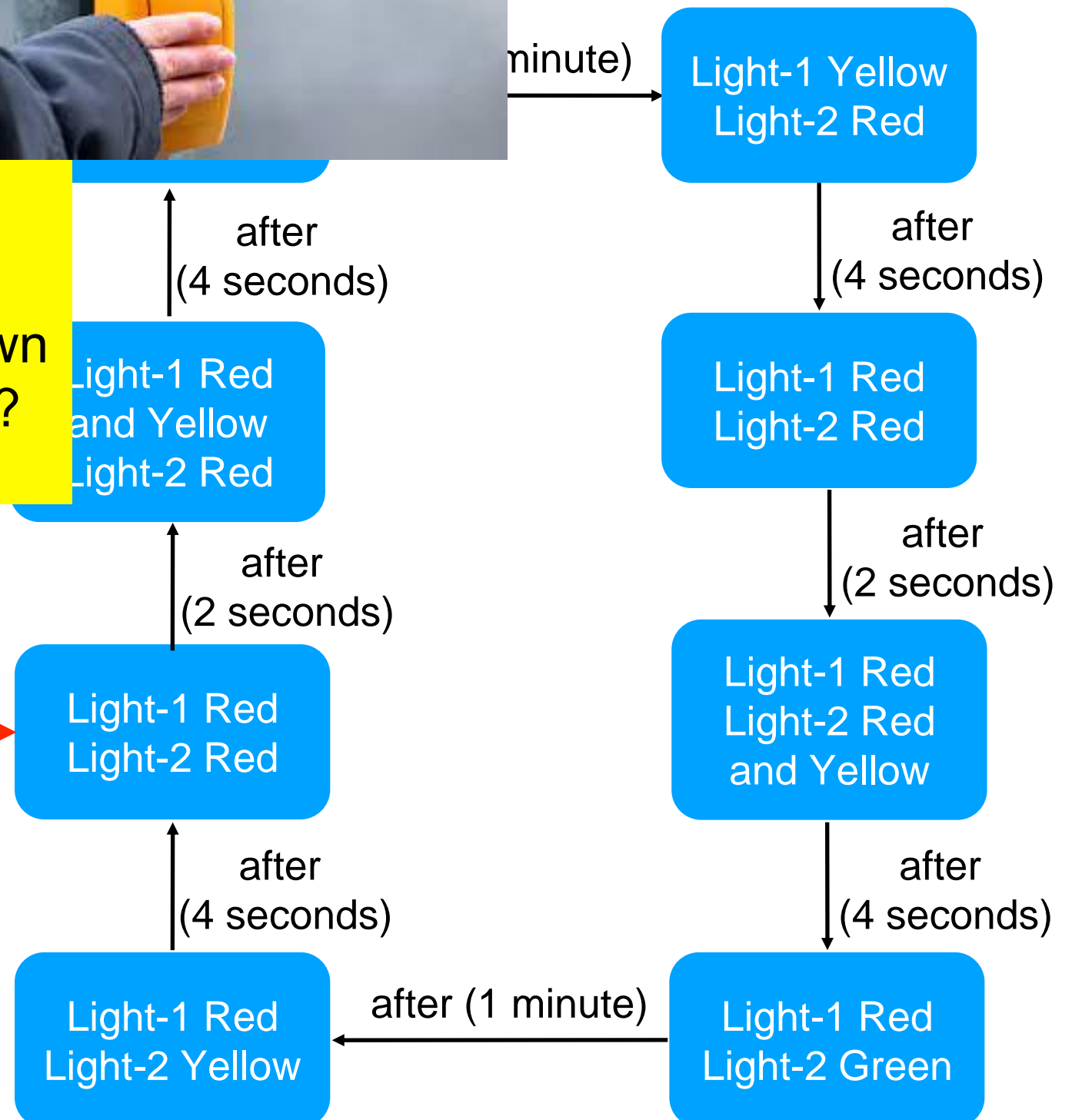
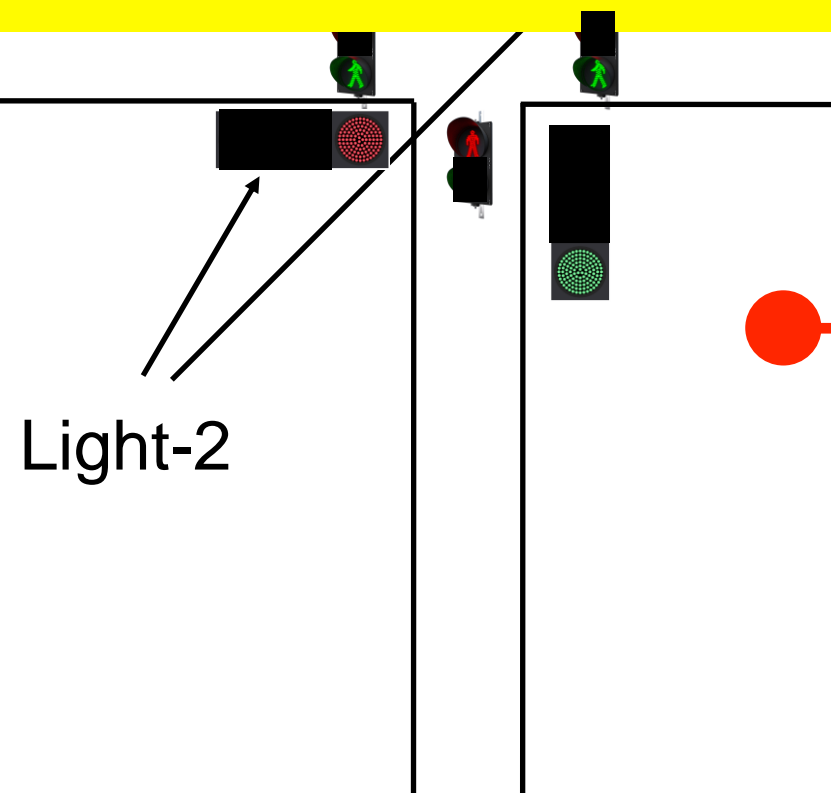
Consider a traffic light system at a four-way crossroads (e.g., two roads intersecting at right angles). Assume the simplest algorithm for cycling through the lights (e.g., all traffic on one road is allowed to go through the crossroads). List the states of this system and draw a statechart describing the states (green, yellow, and red).

Task:

- add pedestrian lights.
- with “request” button.

What if the request button is pressed to frequently?

Should the system “adapt” to known traffic flow (rush hours etc.)? How?



b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (AL) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (VT) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)
--	---	---

b.1) Modellieren Sie das folgende Verhalten des Anrufbeantworters: „Der Anrufbeantworter wartet auf Anrufe. Wenn ein Anruf ankommt, wird der Anruf angenommen. Danach wird entweder *Ansage 1* oder *Ansage 2* abgespielt: Wenn noch nicht die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 1* abgespielt. Wenn die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 2* abgespielt. Nach dem Ende von *Ansage 1* wird die Aufnahme gestartet. Wenn danach der Anrufer auflegt, wird die Verbindung getrennt. Nach dem Ende von *Ansage 2* wird die Verbindung ohne Aufnahme getrennt.“

b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (AL) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (VT) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)
--	---	---

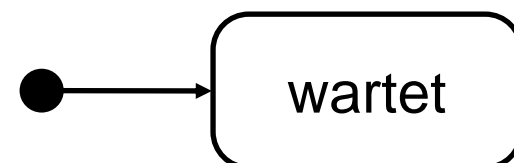
b.1) Modellieren Sie das folgende Verhalten des Anrufbeantworters: „Der Anrufbeantworter wartet auf Anrufe. Wenn ein Anruf ankommt, wird der Anruf angenommen. Danach wird entweder *Ansage 1* oder *Ansage 2* abgespielt: Wenn noch nicht die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 1* abgespielt. Wenn die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 2* abgespielt. Nach dem Ende von *Ansage 1* wird die Aufnahme gestartet. Wenn danach der Anrufer auflegt, wird die Verbindung getrennt. Nach dem Ende von *Ansage 2* wird die Verbindung ohne Aufnahme getrennt.“

b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (AL) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (VT) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)
--	---	---

b.1) Modellieren Sie das folgende Verhalten des Anrufbeantworters: „Der Anrufbeantworter wartet auf Anrufe. Wenn ein Anruf ankommt, wird der Anruf angenommen. Danach wird entweder *Ansage 1* oder *Ansage 2* abgespielt: Wenn noch nicht die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 1* abgespielt. Wenn die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 2* abgespielt. Nach dem Ende von *Ansage 1* wird die Aufnahme gestartet. Wenn danach der Anrufer auflegt, wird die Verbindung getrennt. Nach dem Ende von *Ansage 2* wird die Verbindung ohne Aufnahme getrennt.“

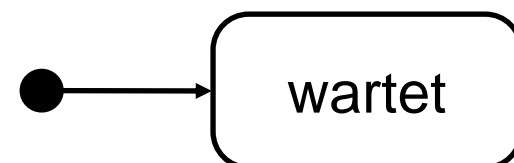


b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (AL) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (VT) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)
--	---	---

b.1) Modellieren Sie das folgende Verhalten des Anrufbeantworters: „Der Anrufbeantworter wartet auf Anrufe. Wenn ein Anruf ankommt, wird der Anruf angenommen. Danach wird entweder *Ansage 1* oder *Ansage 2* abgespielt: Wenn noch nicht die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 1* abgespielt. Wenn die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 2* abgespielt. Nach dem Ende von *Ansage 1* wird die Aufnahme gestartet. Wenn danach der Anrufer auflegt, wird die Verbindung getrennt. Nach dem Ende von *Ansage 2* wird die Verbindung ohne Aufnahme getrennt.“



b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (<u>AK</u>) • Anrufer legt auf (AL) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (<u>AA</u>) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (VT) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)
---	---	--

b.1) Modellieren Sie das folgende Verhalten des Anrufbeantworters: „Der Anrufbeantworter wartet auf Anrufe. Wenn ein Anruf ankommt, wird der Anruf angenommen. Danach wird entweder *Ansage 1* oder *Ansage 2* abgespielt: Wenn noch nicht die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 1* abgespielt. Wenn die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 2* abgespielt. Nach dem Ende von *Ansage 1* wird die Aufnahme gestartet. Wenn danach der Anrufer auflegt, wird die Verbindung getrennt. Nach dem Ende von *Ansage 2* wird die Verbindung ohne Aufnahme getrennt.“



b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (<u>AK</u>) • Anrufer legt auf (AL) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (<u>AA</u>) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (VT) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)
---	---	--

b.1) Modellieren Sie das folgende Verhalten des Anrufbeantworters: „Der Anrufbeantworter wartet auf Anrufe. Wenn ein Anruf ankommt, wird der Anruf angenommen. Danach wird entweder *Ansage 1* oder *Ansage 2* abgespielt: Wenn noch nicht die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 1* abgespielt. Wenn die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 2* abgespielt. Nach dem Ende von *Ansage 1* wird die Aufnahme gestartet. Wenn danach der Anrufer auflegt, wird die Verbindung getrennt. Nach dem Ende von *Ansage 2* wird die Verbindung ohne Aufnahme getrennt.“

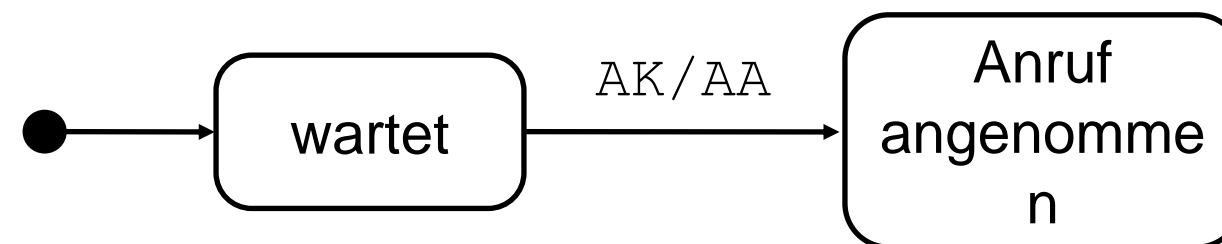


b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (<u>AK</u>) • Anrufer legt auf (AL) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (<u>AA</u>) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (VT) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)
---	---	--

b.1) Modellieren Sie das folgende Verhalten des Anrufbeantworters: „Der Anrufbeantworter wartet auf Anrufe. Wenn ein Anruf ankommt, wird der Anruf angenommen. Danach wird entweder *Ansage 1* oder *Ansage 2* abgespielt: Wenn noch nicht die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 1* abgespielt. Wenn die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 2* abgespielt. Nach dem Ende von *Ansage 1* wird die Aufnahme gestartet. Wenn danach der Anrufer auflegt, wird die Verbindung getrennt. Nach dem Ende von *Ansage 2* wird die Verbindung ohne Aufnahme getrennt.“

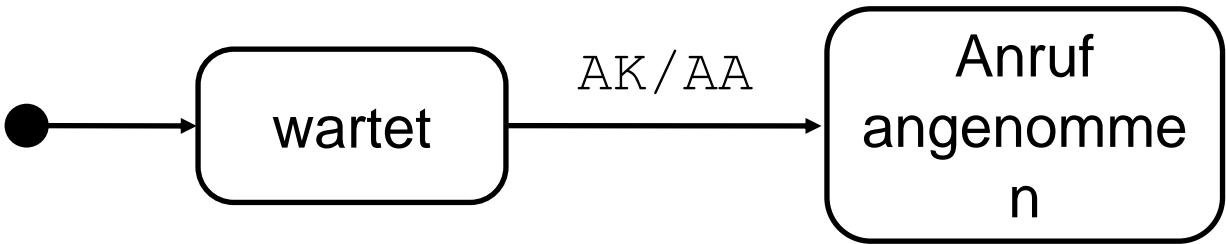


b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<i>Ereignisse:</i>	<i>Bedingungen:</i>	<i>Aktionen:</i>
<ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (AL) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (VT) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)

b.1) Modellieren Sie das folgende Verhalten des Anrufbeantworters: „Der Anrufbeantworter wartet auf Anrufe. Wenn ein Anruf ankommt, wird der Anruf angenommen. Danach wird entweder *Ansage 1* oder *Ansage 2* abgespielt: Wenn noch nicht die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 1* abgespielt. Wenn die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 2* abgespielt. Nach dem Ende von *Ansage 1* wird die Aufnahme gestartet. Wenn danach der Anrufer auflegt, wird die Verbindung getrennt. Nach dem Ende von *Ansage 2* wird die Verbindung ohne Aufnahme getrennt.“

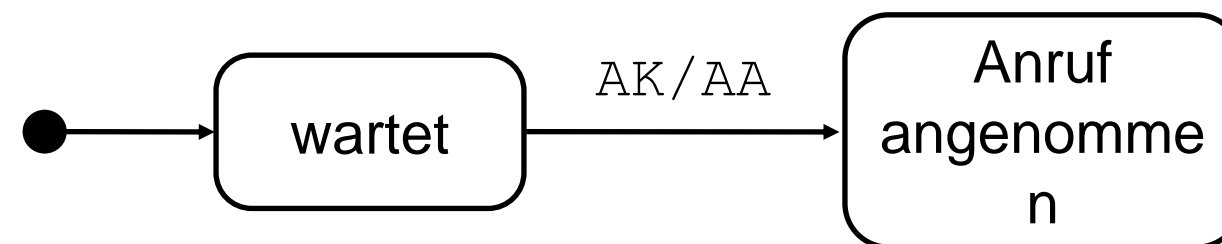


b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (AL) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<p><i>Aktionen:</i></p> <ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (VT) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)
--	---	---

b.1) Modellieren Sie das folgende Verhalten des Anrufbeantworters: „Der Anrufbeantworter wartet auf Anrufe. Wenn ein Anruf ankommt, wird der Anruf angenommen. Danach wird entweder Ansage 1 oder Ansage 2 abgespielt: Wenn noch nicht die maximale Anzahl an Nachrichten gespeichert ist, wird Ansage 1 abgespielt. Wenn die maximale Anzahl an Nachrichten gespeichert ist, wird Ansage 2 abgespielt. Nach dem Ende von Ansage 1 wird die Aufnahme gestartet. Wenn danach der Anrufer auflegt, wird die Verbindung getrennt. Nach dem Ende von Ansage 2 wird die Verbindung ohne Aufnahme getrennt.“

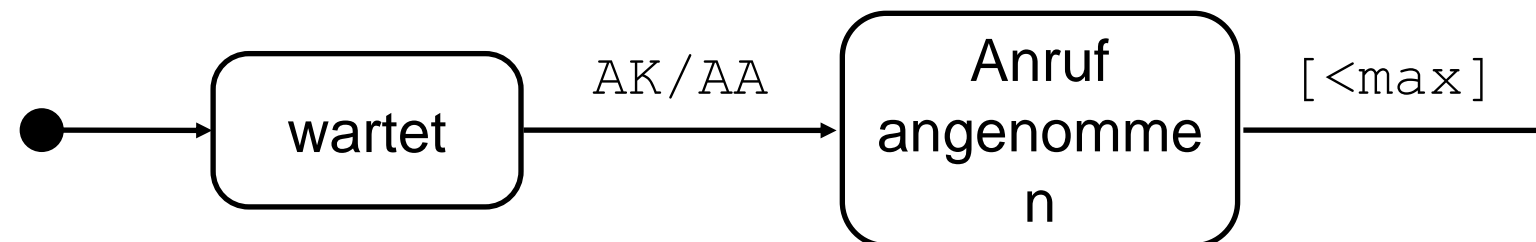


b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (AL) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<u><max</u>) • gespeicherte Nachricht = max (<u>=max</u>) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (<u>A1</u>) • Ansage 2 abspielen (<u>A2</u>) • Aufnahme starten (AS) • Verbindung trennen (VT) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)
--	---	---

b.1) Modellieren Sie das folgende Verhalten des Anrufbeantworters: „Der Anrufbeantworter wartet auf Anrufe. Wenn ein Anruf ankommt, wird der Anruf angenommen. Danach wird entweder Ansage 1 oder Ansage 2 abgespielt: Wenn noch nicht die maximale Anzahl an Nachrichten gespeichert ist, wird Ansage 1 abgespielt. Wenn die maximale Anzahl an Nachrichten gespeichert ist, wird Ansage 2 abgespielt. Nach dem Ende von Ansage 1 wird die Aufnahme gestartet. Wenn danach der Anrufer auflegt, wird die Verbindung getrennt. Nach dem Ende von Ansage 2 wird die Verbindung ohne Aufnahme getrennt.“

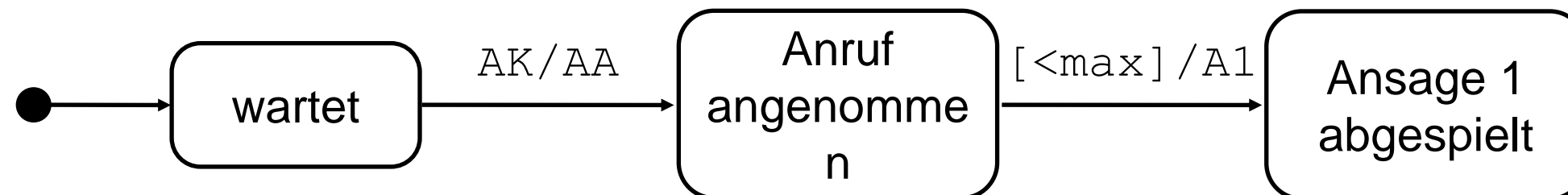


b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (AL) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (VT) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)
--	---	---

b.1) Modellieren Sie das folgende Verhalten des Anrufbeantworters: „Der Anrufbeantworter wartet auf Anrufe. Wenn ein Anruf ankommt, wird der Anruf angenommen. Danach wird entweder *Ansage 1* oder *Ansage 2* abgespielt: Wenn noch nicht die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 1* abgespielt. Wenn die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 2* abgespielt. Nach dem Ende von *Ansage 1* wird die Aufnahme gestartet. Wenn danach der Anrufer auflegt, wird die Verbindung getrennt. Nach dem Ende von *Ansage 2* wird die Verbindung ohne Aufnahme getrennt.“

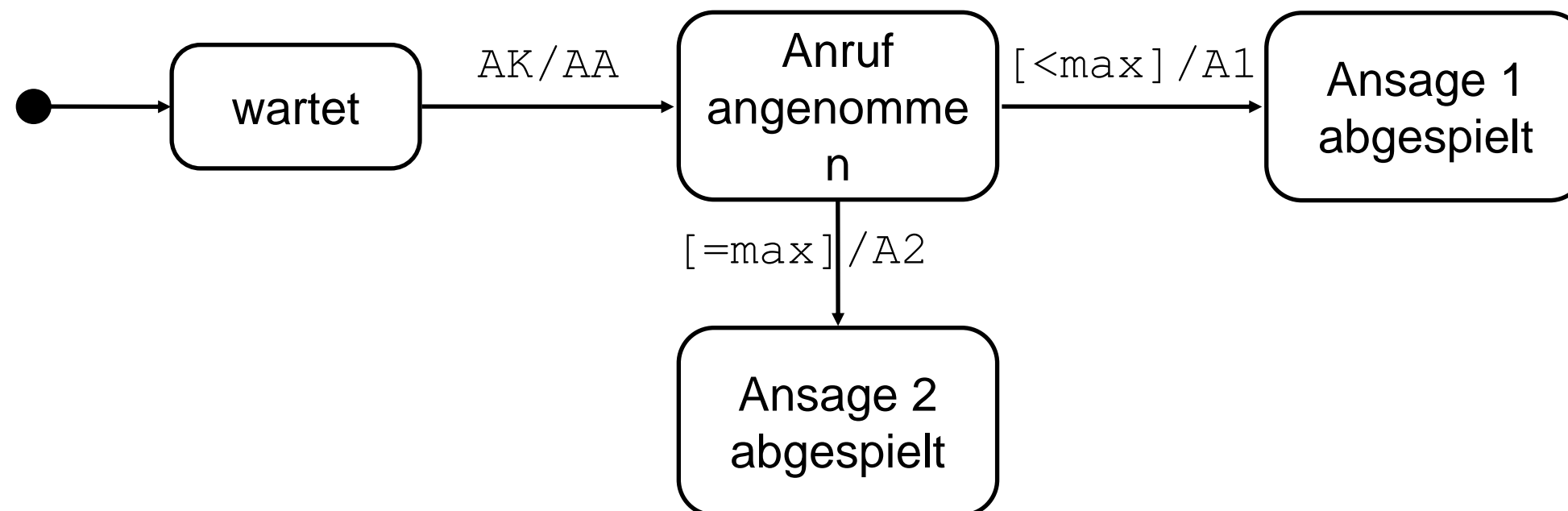


b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (AL) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<p><i>Aktionen:</i></p> <ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (VT) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)
--	---	---

b.1) Modellieren Sie das folgende Verhalten des Anrufbeantworters: „Der Anrufbeantworter wartet auf Anrufe. Wenn ein Anruf ankommt, wird der Anruf angenommen. Danach wird entweder Ansage 1 oder Ansage 2 abgespielt: Wenn noch nicht die maximale Anzahl an Nachrichten gespeichert ist, wird Ansage 1 abgespielt. Wenn die maximale Anzahl an Nachrichten gespeichert ist, wird Ansage 2 abgespielt. Nach dem Ende von Ansage 1 wird die Aufnahme gestartet. Wenn danach der Anrufer auflegt, wird die Verbindung getrennt. Nach dem Ende von Ansage 2 wird die Verbindung ohne Aufnahme getrennt.“

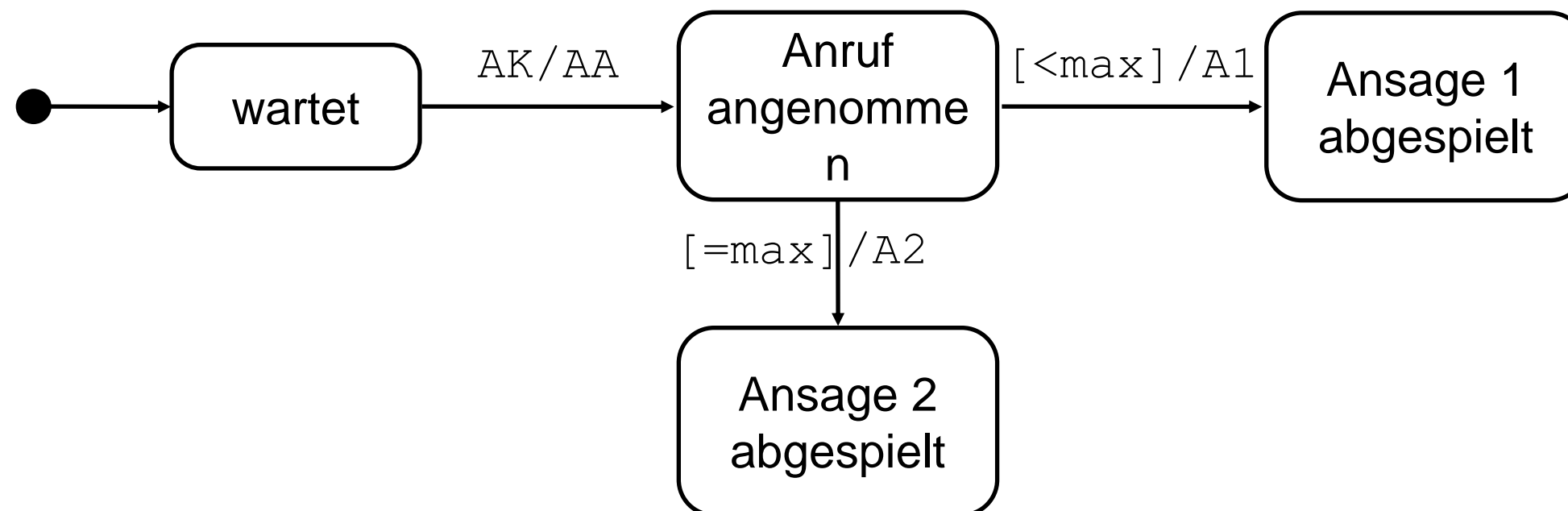


b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (AL) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (VT) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)
--	---	---

b.1) Modellieren Sie das folgende Verhalten des Anrufbeantworters: „Der Anrufbeantworter wartet auf Anrufe. Wenn ein Anruf ankommt, wird der Anruf angenommen. Danach wird entweder *Ansage 1* oder *Ansage 2* abgespielt: Wenn noch nicht die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 1* abgespielt. Wenn die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 2* abgespielt. Nach dem Ende von *Ansage 1* wird die Aufnahme gestartet. Wenn danach der Anrufer auflegt, wird die Verbindung getrennt. Nach dem Ende von *Ansage 2* wird die Verbindung ohne Aufnahme getrennt.“

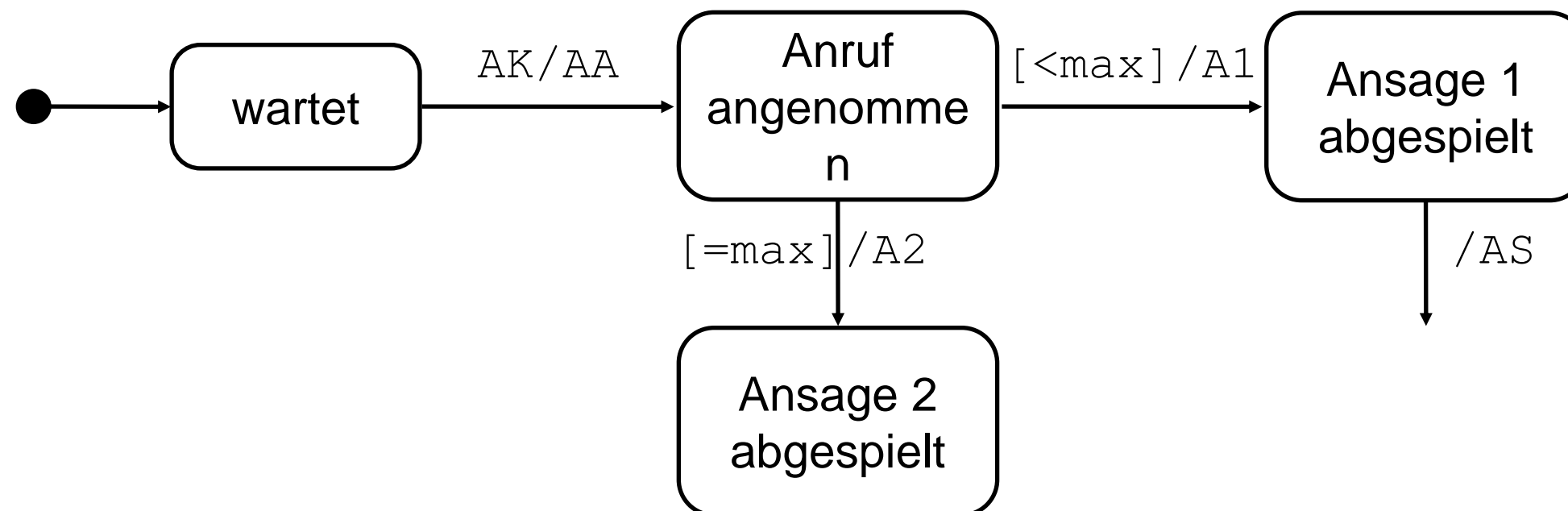


b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (AL) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (VT) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)
--	---	---

b.1) Modellieren Sie das folgende Verhalten des Anrufbeantworters: „Der Anrufbeantworter wartet auf Anrufe. Wenn ein Anruf ankommt, wird der Anruf angenommen. Danach wird entweder *Ansage 1* oder *Ansage 2* abgespielt: Wenn noch nicht die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 1* abgespielt. Wenn die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 2* abgespielt. Nach dem Ende von *Ansage 1* wird die Aufnahme gestartet. Wenn danach der Anrufer auflegt, wird die Verbindung getrennt. Nach dem Ende von *Ansage 2* wird die Verbindung ohne Aufnahme getrennt.“

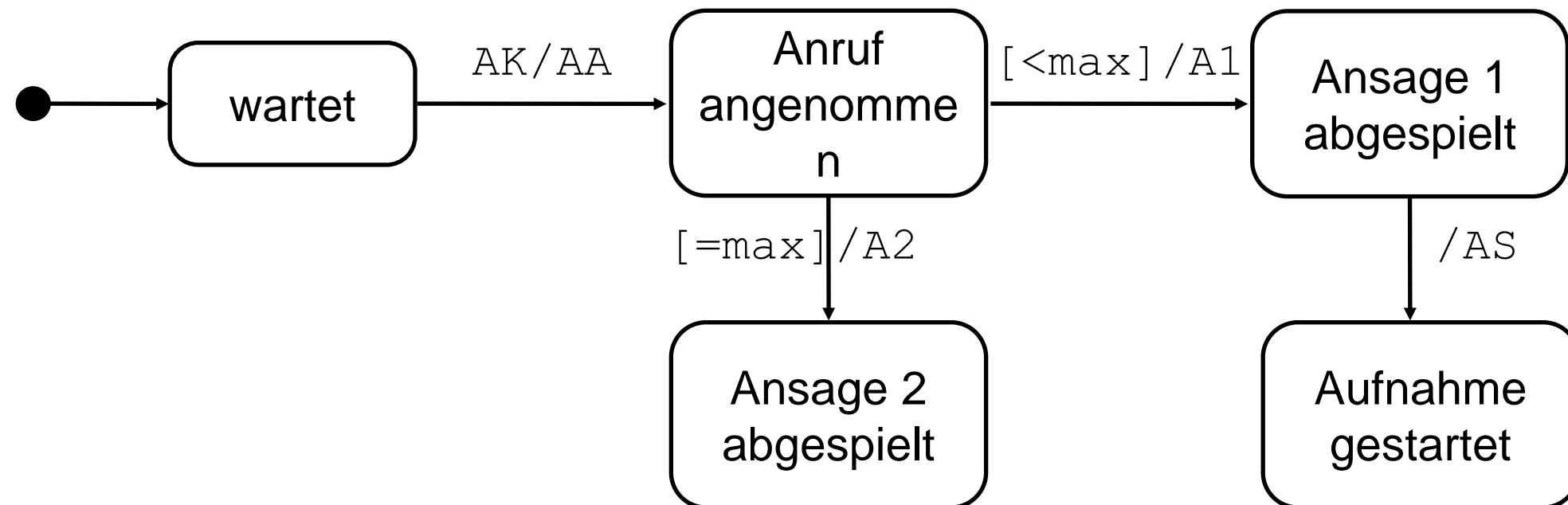


b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (AL) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (VT) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)
--	---	---

b.1) Modellieren Sie das folgende Verhalten des Anrufbeantworters: „Der Anrufbeantworter wartet auf Anrufe. Wenn ein Anruf ankommt, wird der Anruf angenommen. Danach wird entweder *Ansage 1* oder *Ansage 2* abgespielt: Wenn noch nicht die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 1* abgespielt. Wenn die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 2* abgespielt. Nach dem Ende von *Ansage 1* wird die Aufnahme gestartet. Wenn danach der Anrufer auflegt, wird die Verbindung getrennt. Nach dem Ende von *Ansage 2* wird die Verbindung ohne Aufnahme getrennt.“

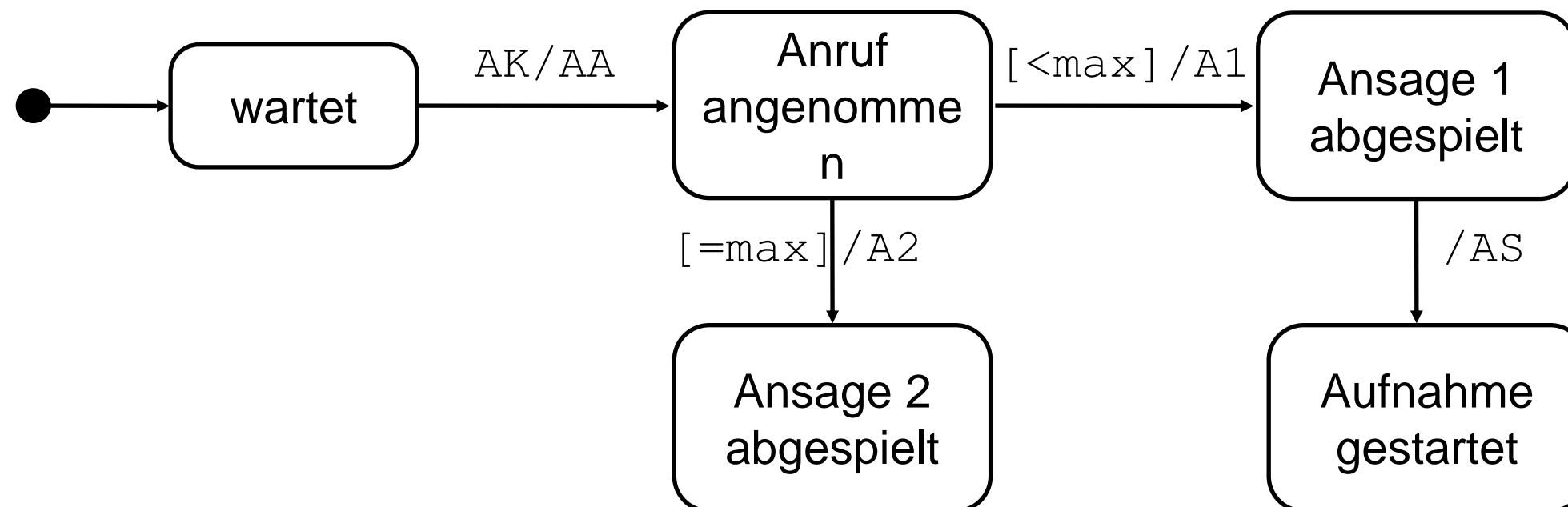


b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (<u>AL</u>) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (<u>VT</u>) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)
---	---	--

b.1) Modellieren Sie das folgende Verhalten des Anrufbeantworters: „Der Anrufbeantworter wartet auf Anrufe. Wenn ein Anruf ankommt, wird der Anruf angenommen. Danach wird entweder *Ansage 1* oder *Ansage 2* abgespielt: Wenn noch nicht die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 1* abgespielt. Wenn die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 2* abgespielt. Nach dem Ende von *Ansage 1* wird die Aufnahme gestartet. Wenn danach der Anrufer auflegt, wird die Verbindung getrennt. Nach dem Ende von *Ansage 2* wird die Verbindung ohne Aufnahme getrennt.“

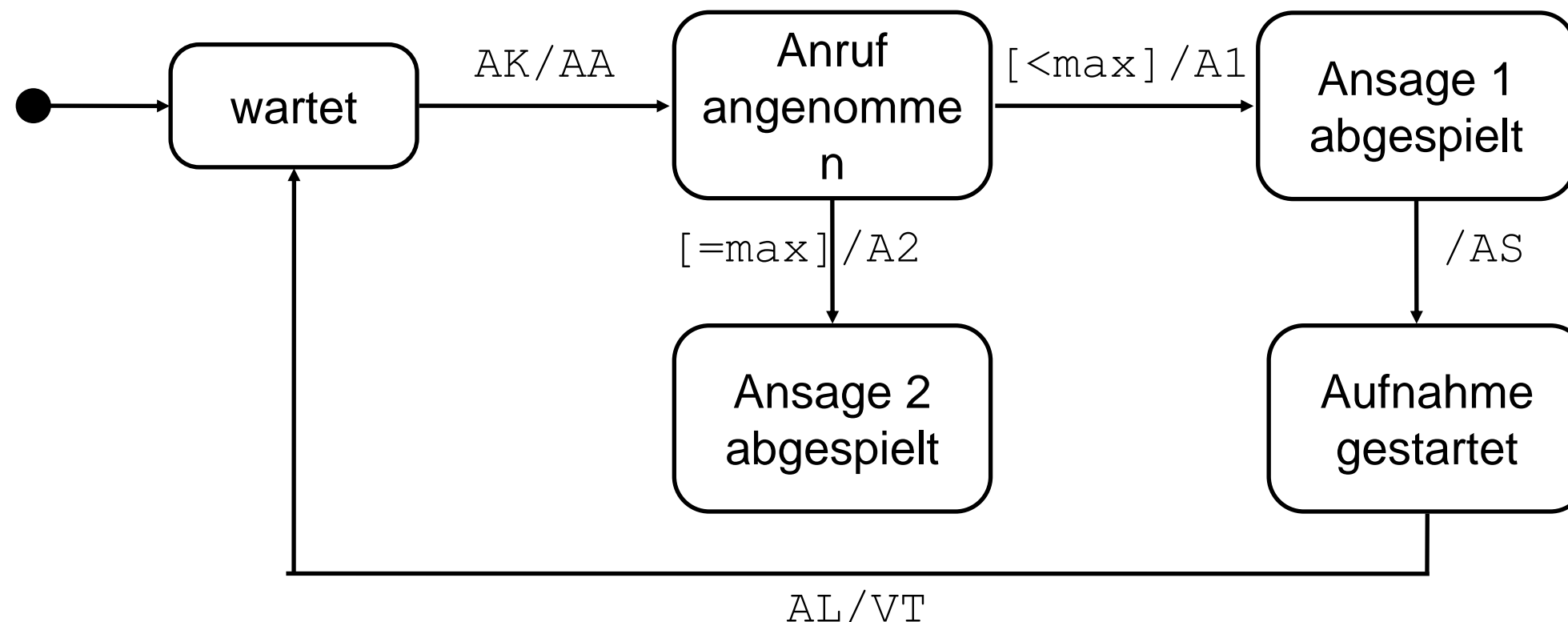


b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (<u>AL</u>) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (<u>VT</u>) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)
---	---	--

b.1) Modellieren Sie das folgende Verhalten des Anrufbeantworters: „Der Anrufbeantworter wartet auf Anrufe. Wenn ein Anruf ankommt, wird der Anruf angenommen. Danach wird entweder *Ansage 1* oder *Ansage 2* abgespielt: Wenn noch nicht die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 1* abgespielt. Wenn die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 2* abgespielt. Nach dem Ende von *Ansage 1* wird die Aufnahme gestartet. Wenn danach der Anrufer auflegt, wird die Verbindung getrennt. Nach dem Ende von *Ansage 2* wird die Verbindung ohne Aufnahme getrennt.“

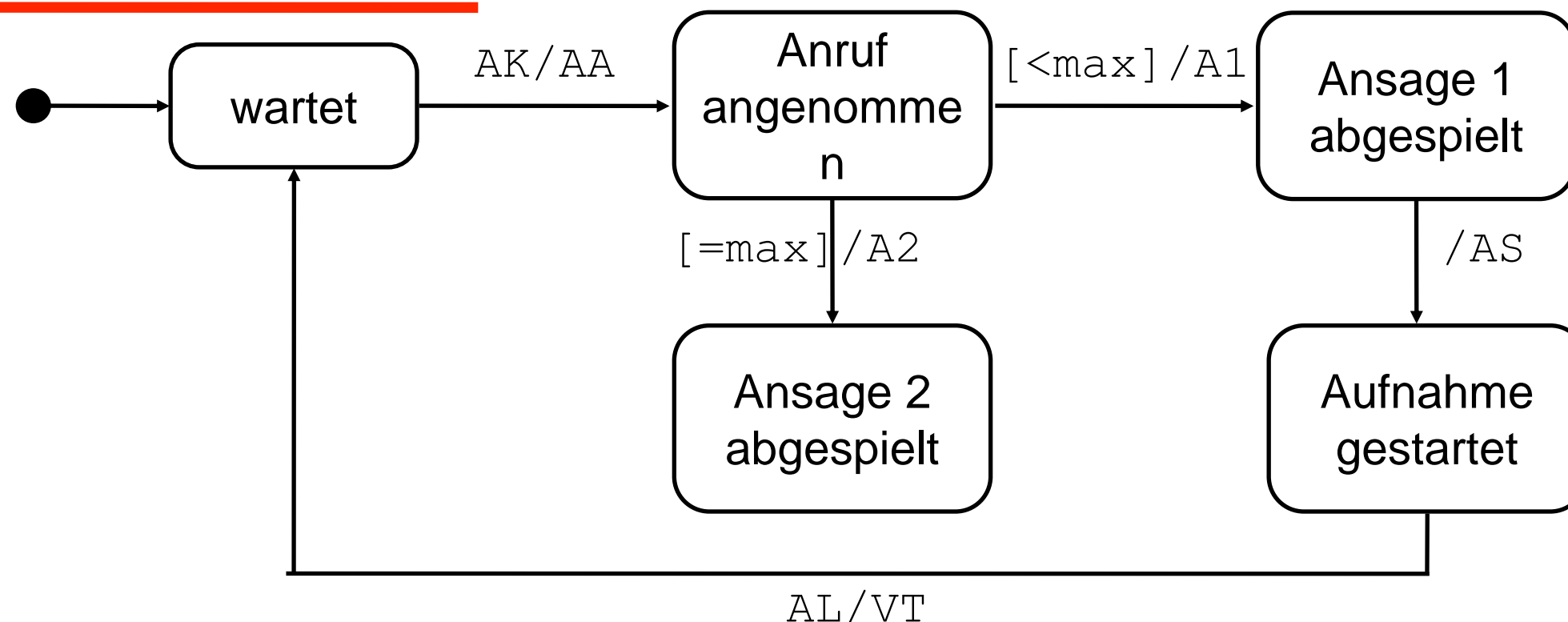


b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (<u>AL</u>) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (<u>VT</u>) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)
---	---	--

b.1) Modellieren Sie das folgende Verhalten des Anrufbeantworters: „Der Anrufbeantworter wartet auf Anrufe. Wenn ein Anruf ankommt, wird der Anruf angenommen. Danach wird entweder *Ansage 1* oder *Ansage 2* abgespielt: Wenn noch nicht die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 1* abgespielt. Wenn die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 2* abgespielt. Nach dem Ende von *Ansage 1* wird die Aufnahme gestartet. Wenn danach der Anrufer auflegt, wird die Verbindung getrennt. Nach dem Ende von *Ansage 2* wird die Verbindung ohne Aufnahme getrennt.“

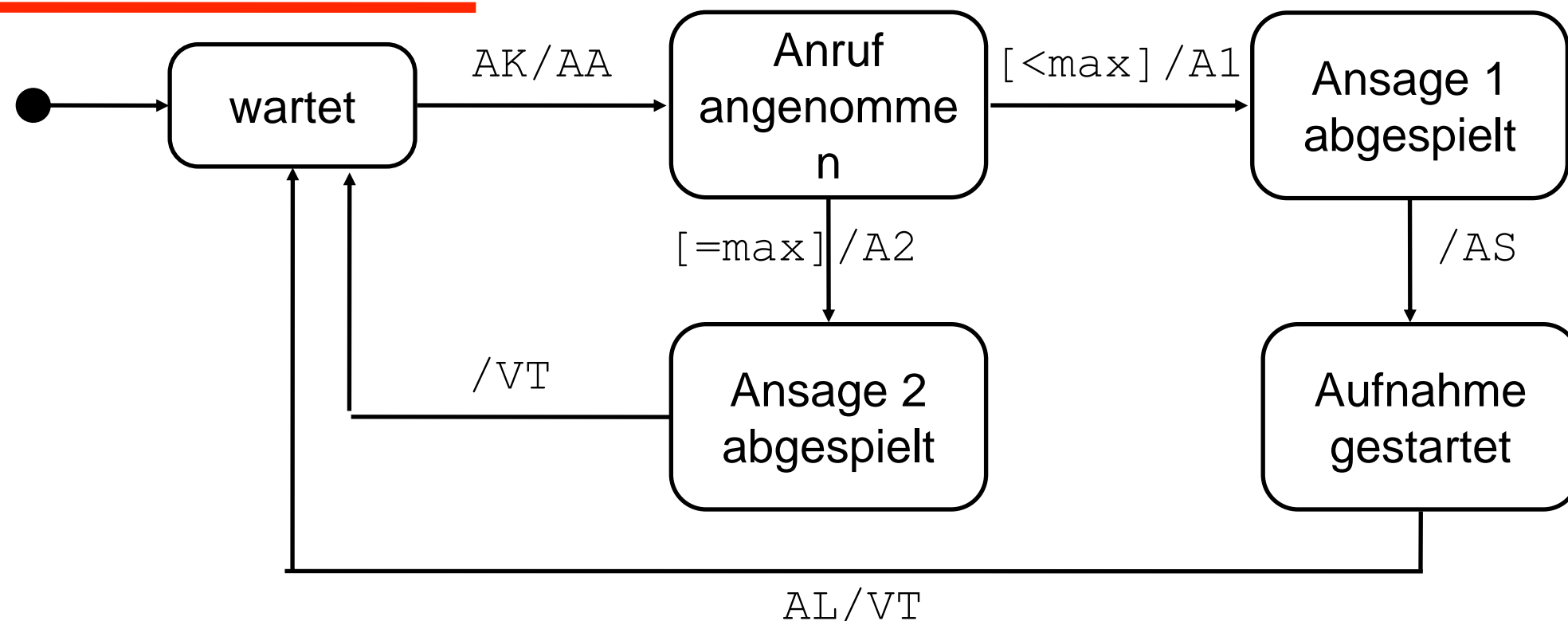


b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (<u>AL</u>) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (<u>VT</u>) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)
---	---	--

b.1) Modellieren Sie das folgende Verhalten des Anrufbeantworters: „Der Anrufbeantworter wartet auf Anrufe. Wenn ein Anruf ankommt, wird der Anruf angenommen. Danach wird entweder *Ansage 1* oder *Ansage 2* abgespielt: Wenn noch nicht die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 1* abgespielt. Wenn die maximale Anzahl an Nachrichten gespeichert ist, wird *Ansage 2* abgespielt. Nach dem Ende von *Ansage 1* wird die Aufnahme gestartet. Wenn danach der Anrufer auflegt, wird die Verbindung getrennt. Nach dem Ende von *Ansage 2* wird die Verbindung ohne Aufnahme getrennt.“



b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

Ereignisse:

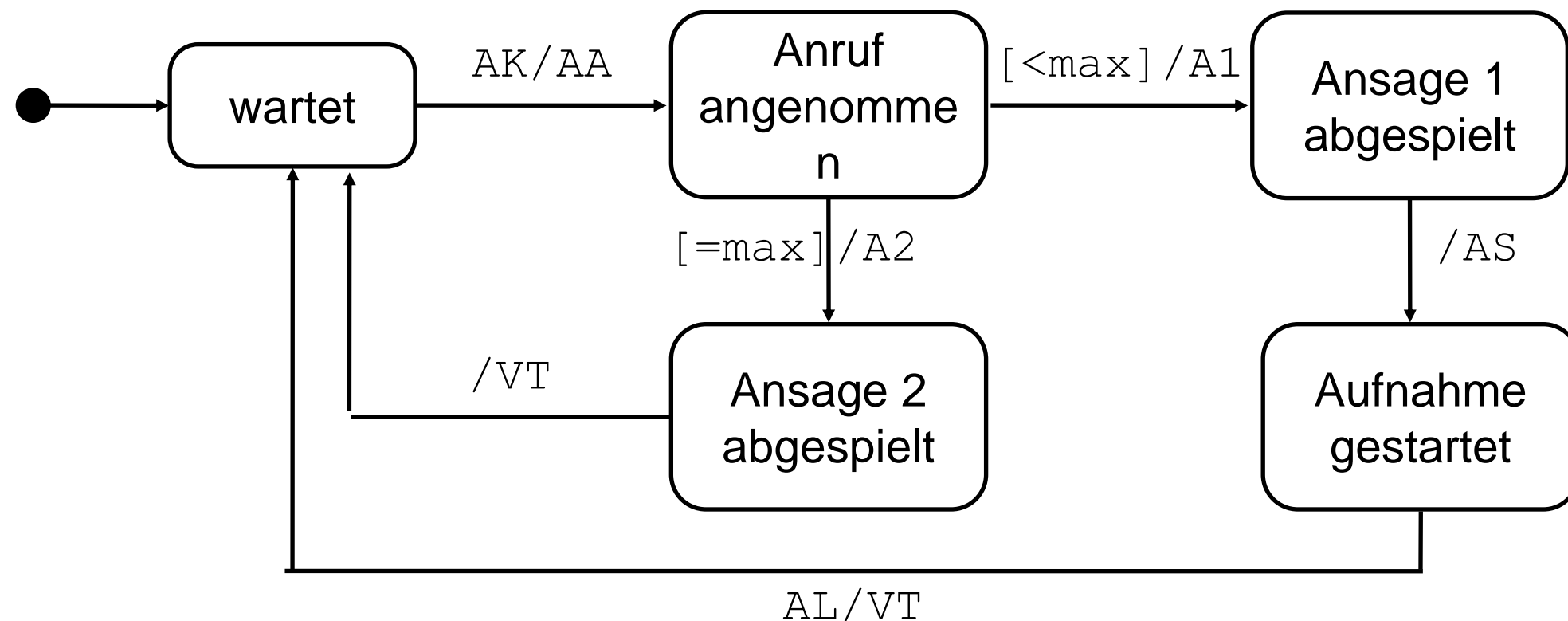
- Anruf kommt an (AK)
- Anrufer legt auf (AL)
- Knopf zur Wiedergabe drücken (WD)
- Stopp- Knopf drücken (SD)

Bedingungen:

- gespeicherte Nachricht < max (<max)
- gespeicherte Nachricht = max (=max)
- mindestens eine Nachricht gespeichert (EN)
- keine Nachricht gespeichert (KN)
- alle Nachrichten abgespielt (AN)

- Anruf annehmen (AA)
- Ansage 1 abspielen (A1)
- Ansage 2 abspielen (A2)
- Aufnahme starten (AS)
- Verbindung trennen (VT)
- gespeicherte Nachrichten abspielen (NA)
- Wiedergabe stoppen (WS)

Do you see anything problematic here?



b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

Ereignisse:

- Anruf kommt an (AK)
- Anrufer legt auf (AL)
- Knopf zur Wiedergabe drücken (WD)
- Stopp- Knopf drücken (SD)

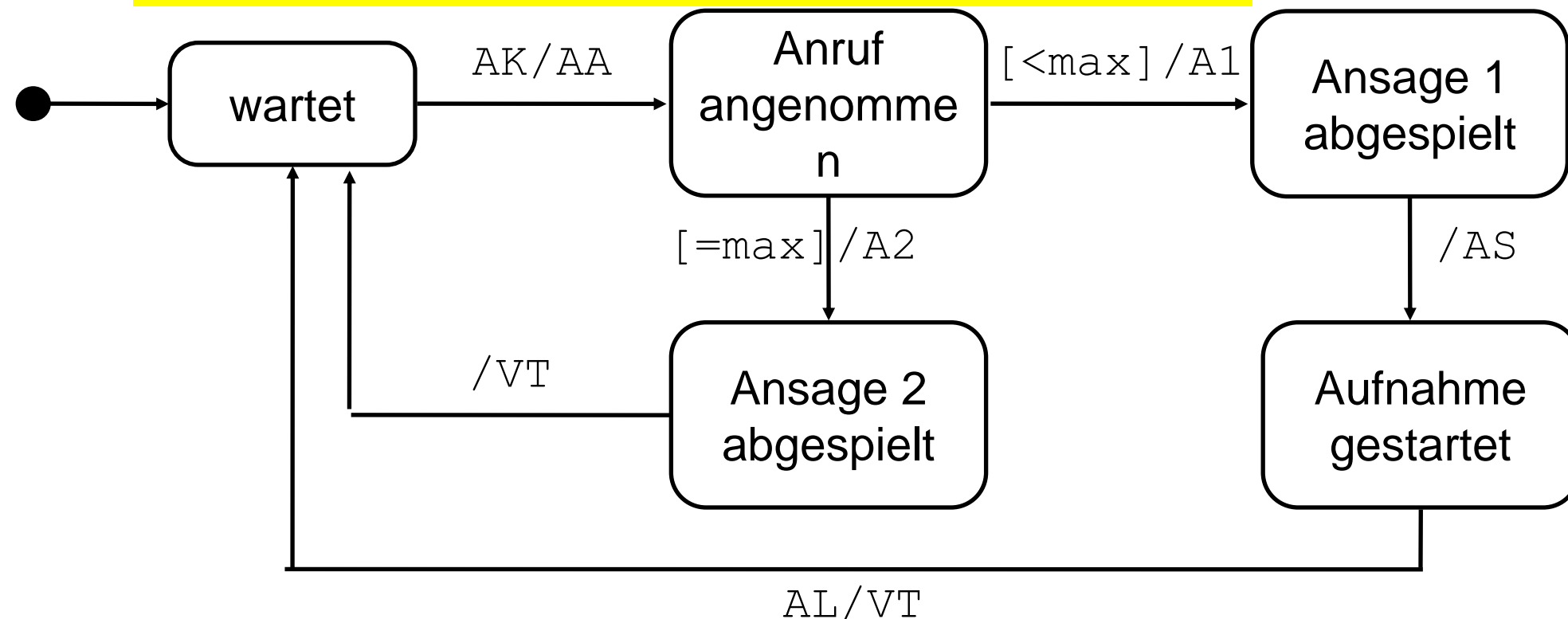
Bedingungen:

- gespeicherte Nachricht < max (<max)
- gespeicherte Nachricht = max (=max)
- mindestens eine Nachricht gespeichert (EN)
- keine Nachricht gespeichert (KN)
- alle Nachrichten abgespielt (AN)

- Anruf annehmen (AA)
- Ansage 1 abspielen (A1)
- Ansage 2 abspielen (A2)
- Aufnahme starten (AS)
- Verbindung trennen (VT)
- gespeicherte Nachrichten abspielen (NA)
- Wiedergabe stoppen (WS)

Do you see anything problematic here?

— maximum time of a message?

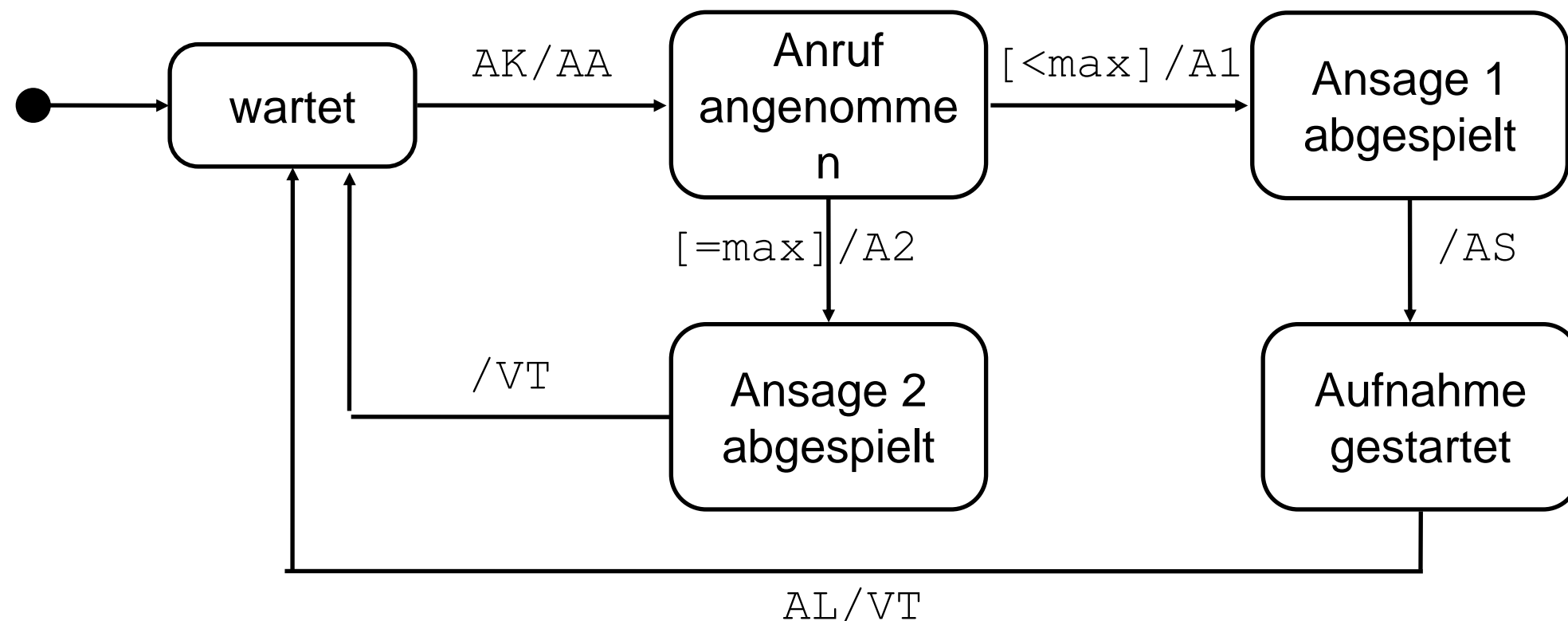


b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (<u>AL</u>) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (<u>VT</u>) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)
---	---	--

b.2) Modellieren Sie möglichst elegant, dass der Anrufer jederzeit, nachdem der Anrufbeantworter den Anruf angenommen hat, auflegen kann und die Verbindung getrennt wird.

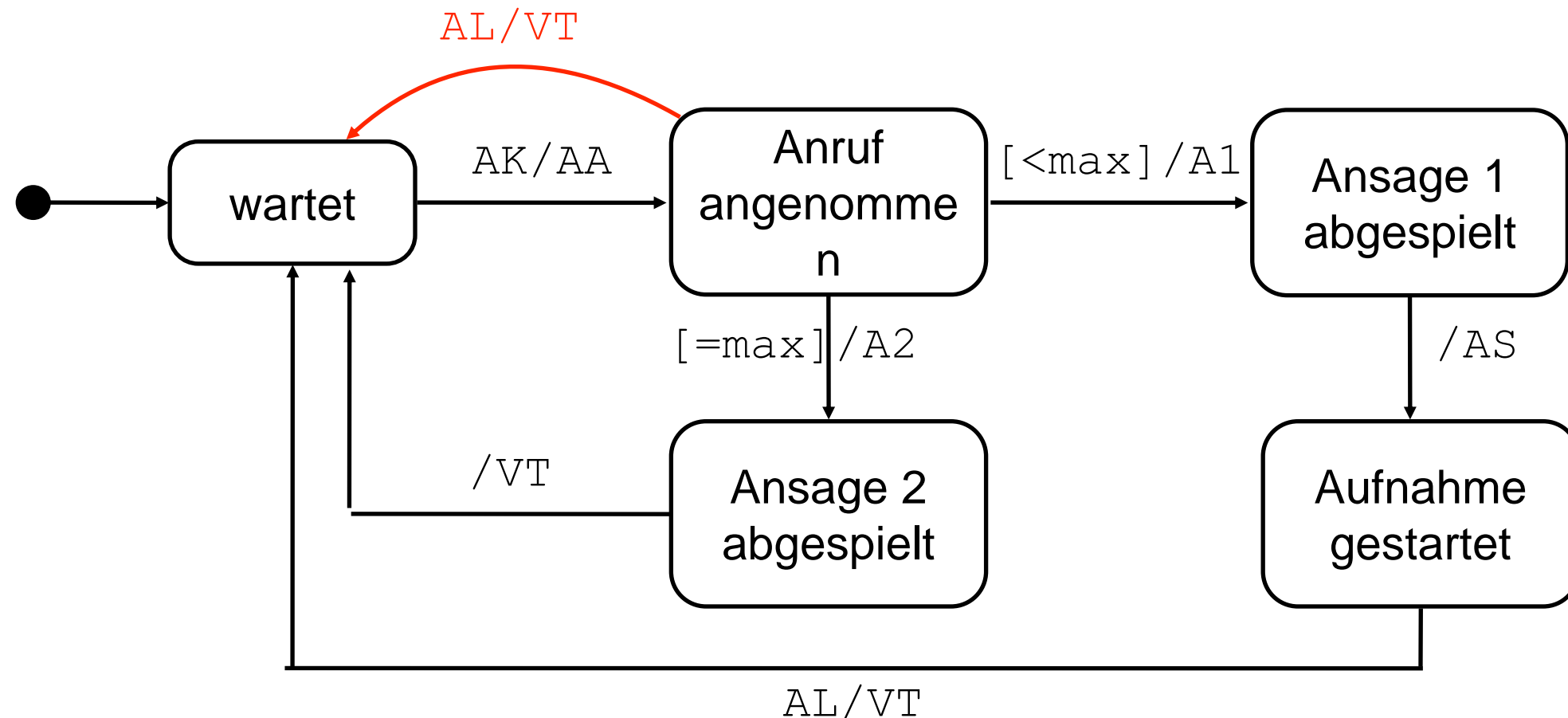


b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (<u>AL</u>) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (<u>VT</u>) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)
---	---	--

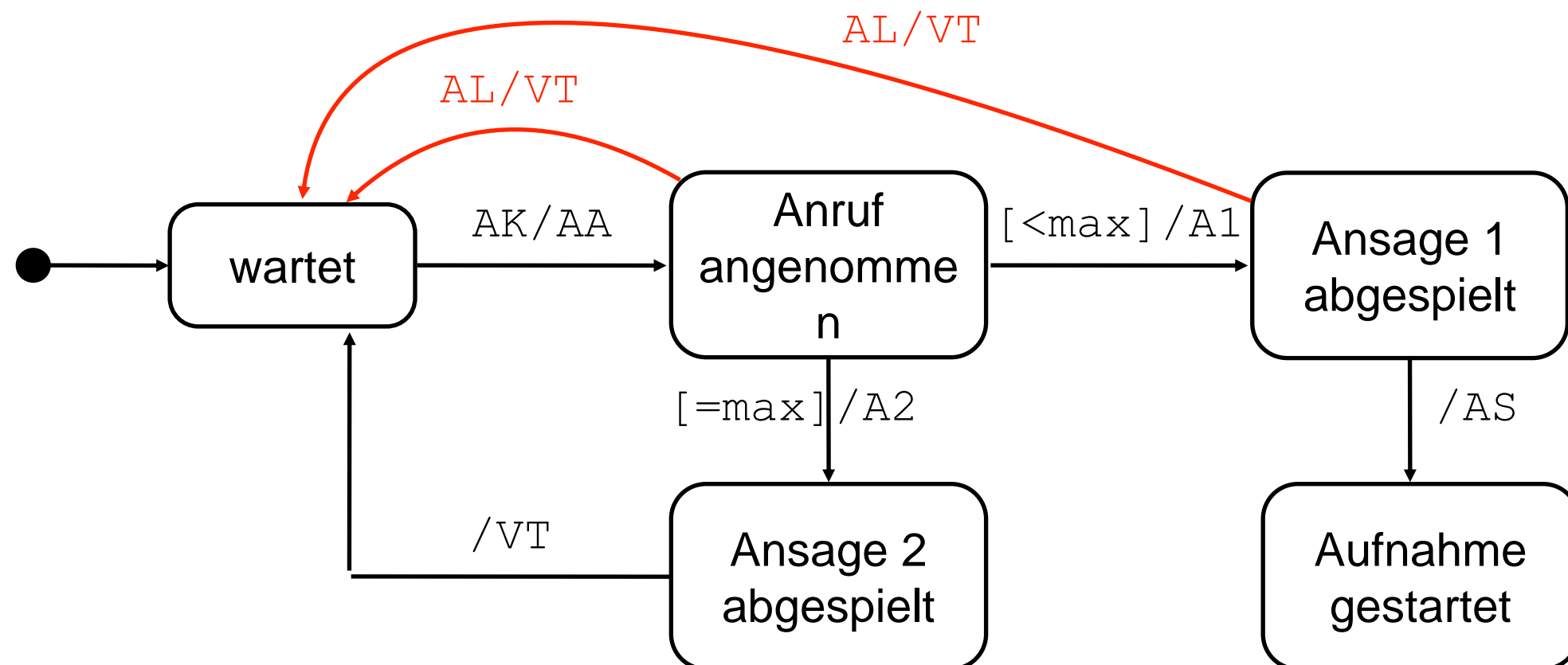
b.2) Modellieren Sie möglichst elegant, dass der Anrufer jederzeit, nachdem der Anrufbeantworter den Anruf angenommen hat, auflegen kann und die Verbindung getrennt wird.



b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

<i>Ereignisse:</i>	<i>Bedingungen:</i>	<i>Aktionen:</i>
<ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (<u>AL</u>) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (<u>VT</u>) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)

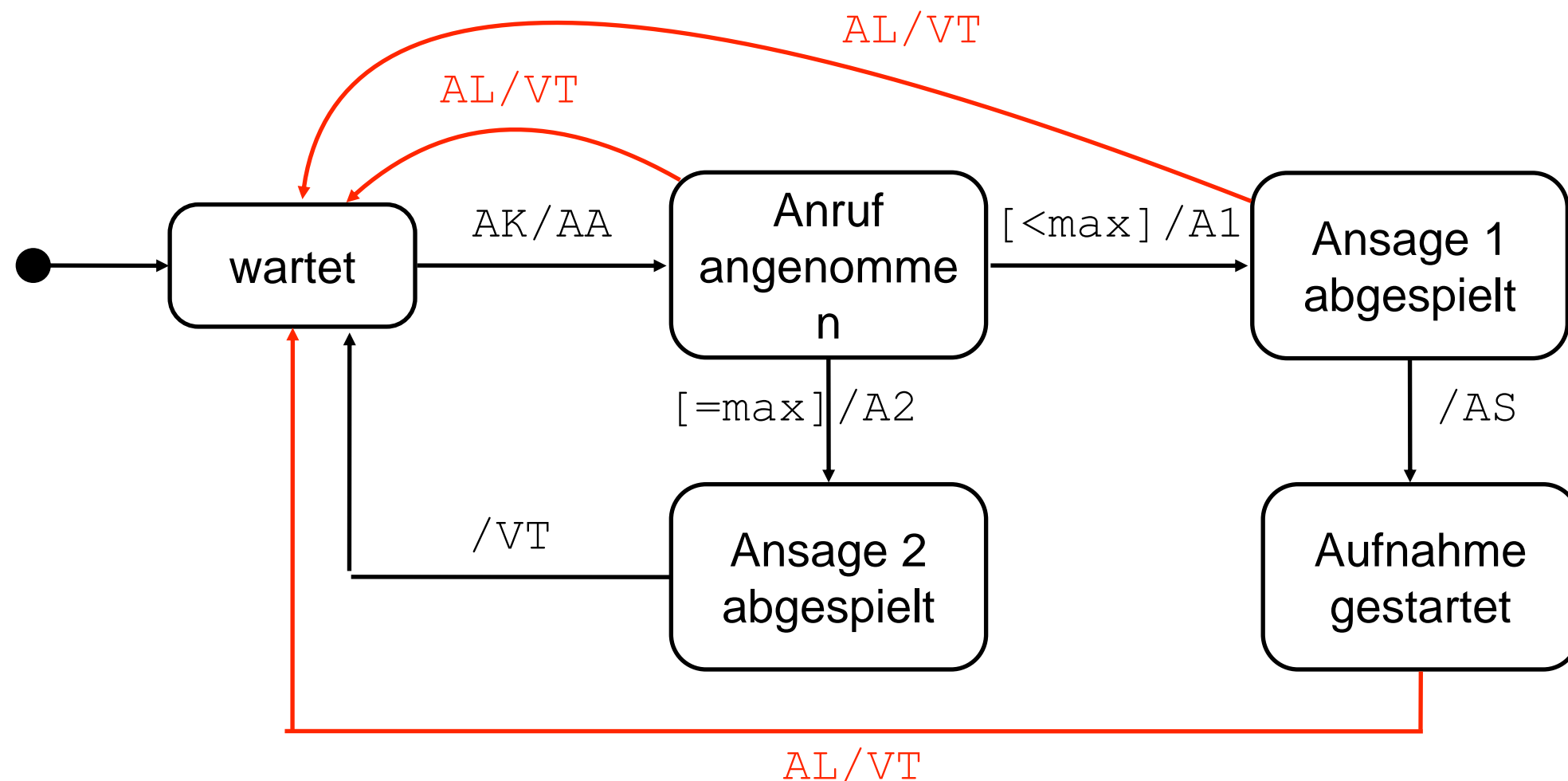
b.2) Modellieren Sie möglichst elegant, dass der Anrufer jederzeit, nachdem der Anrufbeantworter den Anruf angenommen hat, auflegen kann und die Verbindung getrennt wird.



b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Ereignisse:	Bedingungen:	Aktionen:
<ul style="list-style-type: none"> Anruf kommt an (AK) Anrufer legt auf (<u>AL</u>) Knopf zur Wiedergabe drücken (WD) Stopp- Knopf drücken (SD) 	<ul style="list-style-type: none"> gespeicherte Nachricht < max (<max) gespeicherte Nachricht = max (=max) mindestens eine Nachricht gespeichert (EN) keine Nachricht gespeichert (KN) alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> Anruf annehmen (AA) Ansage 1 abspielen (A1) Ansage 2 abspielen (A2) Aufnahme starten (AS) Verbindung trennen (<u>VT</u>) gespeicherte Nachrichten abspielen (NA) Wiedergabe stoppen (WS)

b.2) Modellieren Sie möglichst elegant, dass der Anrufer jederzeit, nachdem der Anrufbeantworter den Anruf angenommen hat, auflegen kann und die Verbindung getrennt wird.

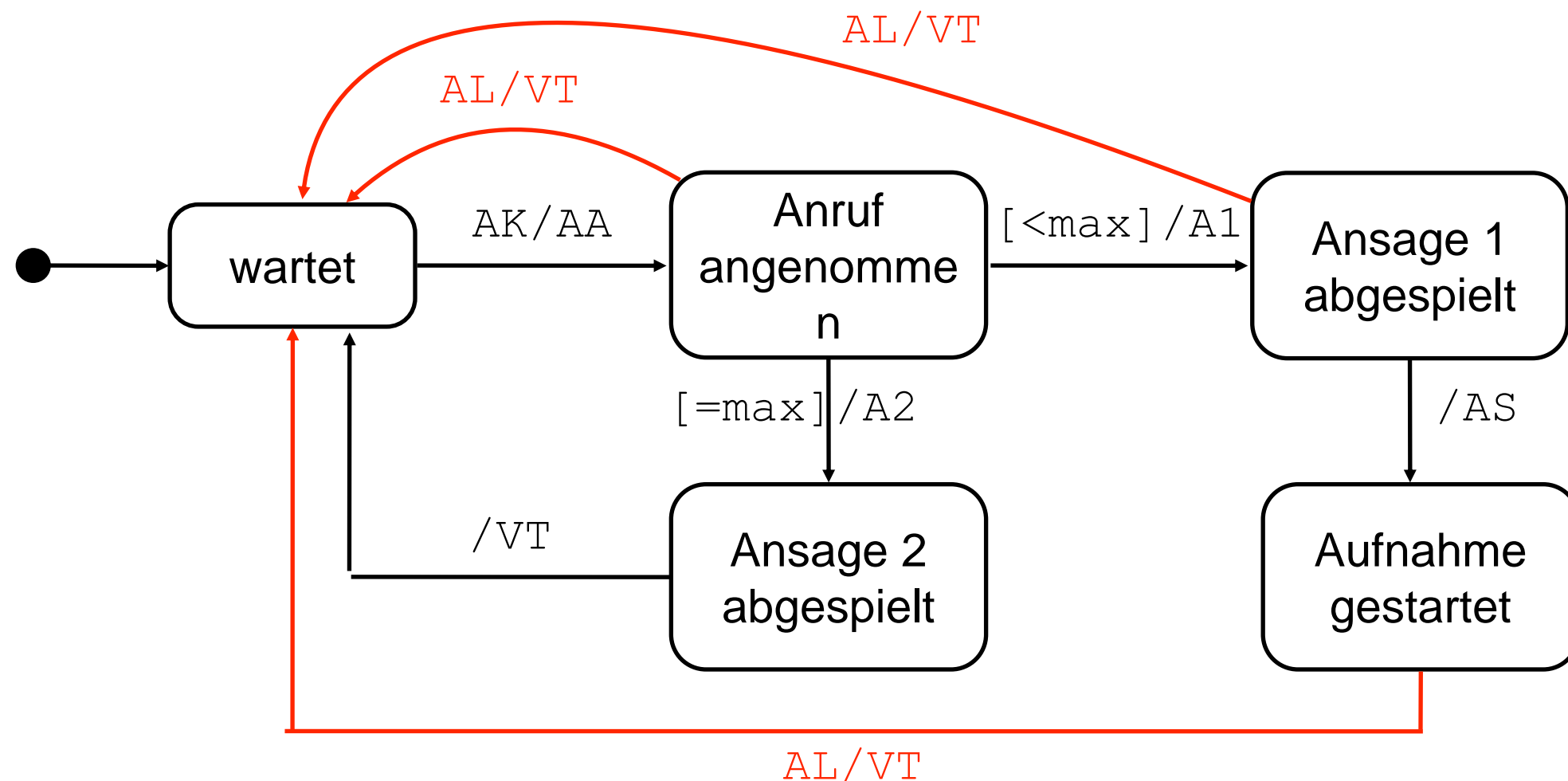


b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Ereignisse:	Bedingungen:	Aktionen:
<ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (<u>AL</u>) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht • alle Nachrichten 	<ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (VT) • Nachricht abspielen (NA)

How can we “refactor” the three (AL/VT)-transitions?

b.2) Modellieren Sie möglichst elegant, dass der Anrufbeantworter den Anruf angenommen hat, auflegen kann und die Verbindung ge-

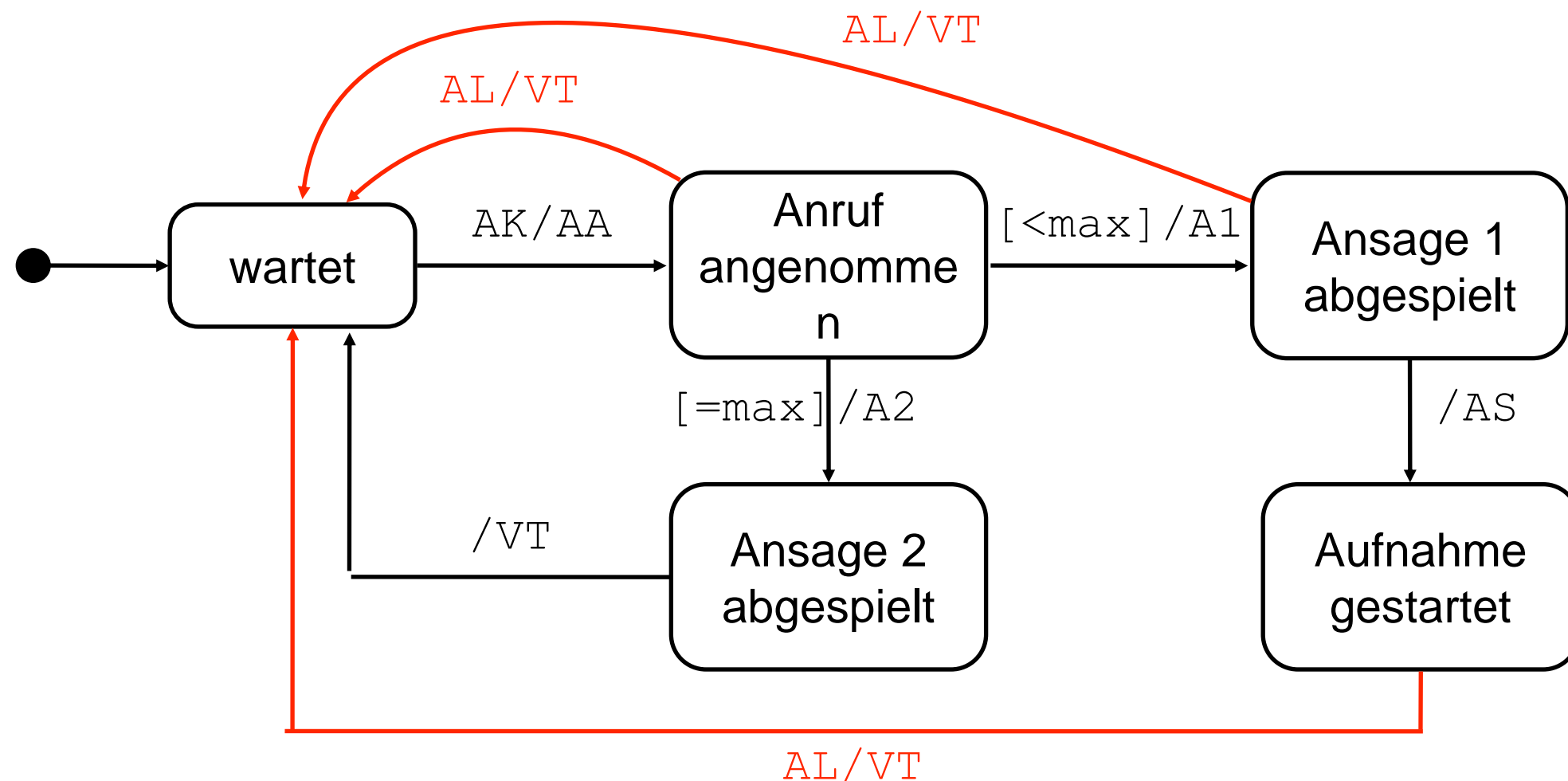


b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Ereignisse: <ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (<u>AL</u>) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	Bedingungen: <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht • alle Nachrichten 	Aktionen: <ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • ...
--	---	--

Use a nested (sub) automaton!

b.2) Modellieren Sie möglichst elegant, dass der Anrufer jederzeit, nachdem der Anrufbeantworter den Anruf angenommen hat, auflegen kann und die Verbindung getrennt wird.



b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

Ereignisse:

- Anruf kommt an (AK)
- Anrufer legt auf (AL)
- Knopf zur Wiedergabe drücken (WD)
- Stopp- Knopf drücken (SD)

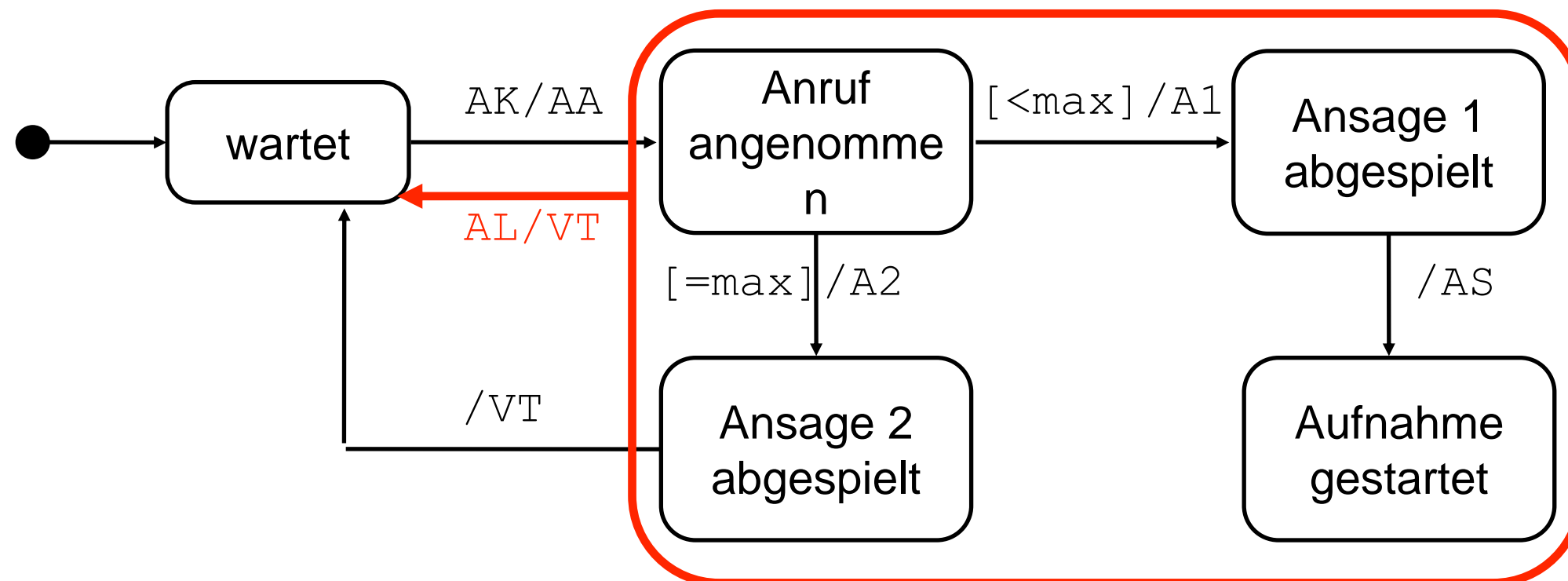
Bedingungen:

- gespeicherte Nachricht < max (<max)
- gespeicherte Nachricht = max (=max)
- mindestens eine Nachricht gespeichert (EN)
- keine Nachricht gespeichert (NA)
- alle Nachrichten abgespielt (AT)

- Anruf annehmen (AA)
- Ansage 1 abspielen (A1)
- Ansage 2 abspielen (A2)
- Aufnahme starten (AS)
- Wiedergabe stoppen (VT)
- Wiedergabe fortsetzen (VF)
- Wiedergabe beenden (VB)
- Wiedergabe neu starten (VN)
- Wiedergabe neu starten (NA)

Use a nested (sub) automaton!

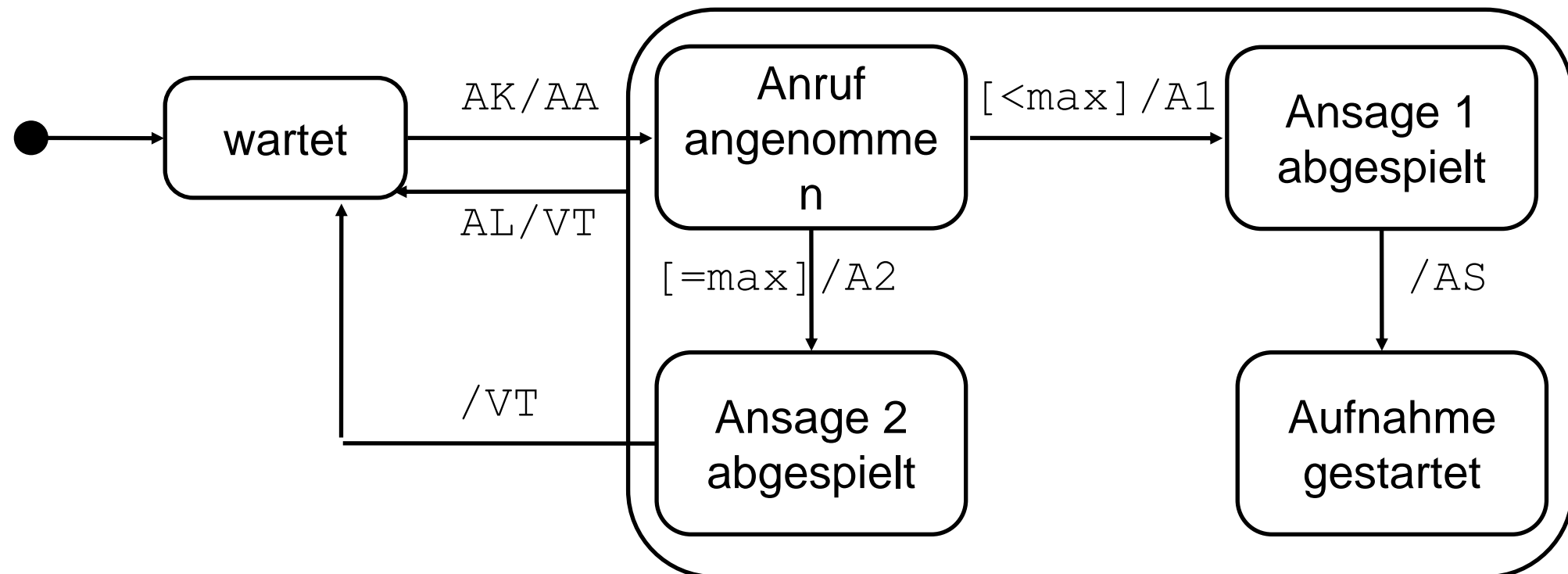
b.2) Modellieren Sie möglichst elegant, dass der Anrufer jederzeit, nachdem der Anrufbeantworter den Anruf angenommen hat, auflegen kann und die Verbindung getrennt wird.



b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Ereignisse:	Bedingungen:	Aktionen:
<ul style="list-style-type: none"> Anruf kommt an (AK) Anrufer legt auf (AL) Knopf zur Wiedergabe drücken (<u>WD</u>) Stopp- Knopf drücken (SD) 	<ul style="list-style-type: none"> gespeicherte Nachricht < max (<max) gespeicherte Nachricht = max (=max) mindestens eine Nachricht gespeichert (<u>EN</u>) keine Nachricht gespeichert (KN) alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> Anruf annehmen (AA) Ansage 1 abspielen (A1) Ansage 2 abspielen (A2) Aufnahme starten (AS) Verbindung trennen (VT) gespeicherte Nachrichten abspielen (<u>NA</u>) Wiedergabe stoppen (WS)

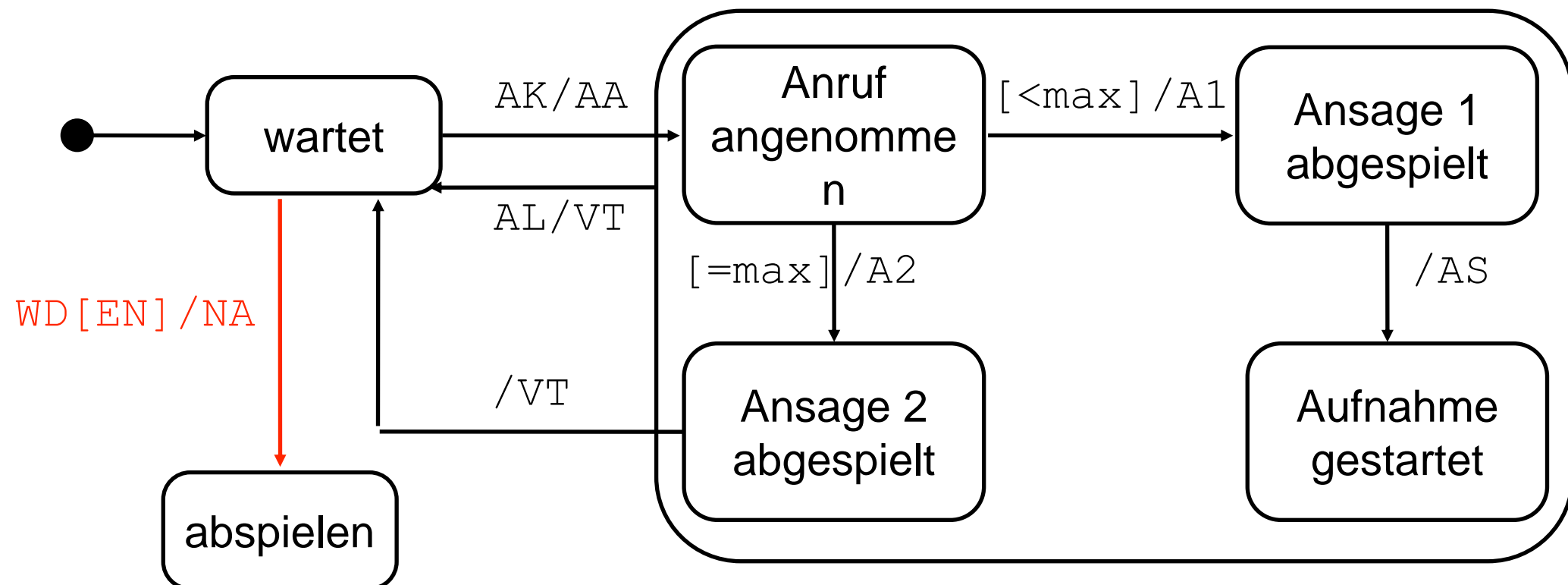
b.3) Modellieren Sie zusätzlich das folgende Verhalten: „Wenn am Anrufbeantworter der Knopf zur Wiedergabe gedrückt wird und mindestens eine Nachricht gespeichert ist, werden die gespeicherten Nachrichten abgespielt. Wenn der Knopf gedrückt wird und keine Nachricht gespeichert ist, führt der Anrufbeantworter keine Aktion aus. Während der Wiedergabe der gespeicherten Nachrichten kann der Stopp-Knopf gedrückt werden und die Wiedergabe wird gestoppt. Wenn alle Nachrichten abgespielt worden sind, wird die Wiedergabe ebenfalls gestoppt.“



b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Ereignisse:	Bedingungen:	Aktionen:
<ul style="list-style-type: none"> Anruf kommt an (AK) Anrufer legt auf (AL) Knopf zur Wiedergabe drücken (<u>WD</u>) Stopp- Knopf drücken (SD) 	<ul style="list-style-type: none"> gespeicherte Nachricht < max (<max) gespeicherte Nachricht = max (=max) mindestens eine Nachricht gespeichert (<u>EN</u>) keine Nachricht gespeichert (KN) alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> Anruf annehmen (AA) Ansage 1 abspielen (A1) Ansage 2 abspielen (A2) Aufnahme starten (AS) Verbindung trennen (VT) gespeicherte Nachrichten abspielen (<u>NA</u>) Wiedergabe stoppen (WS)

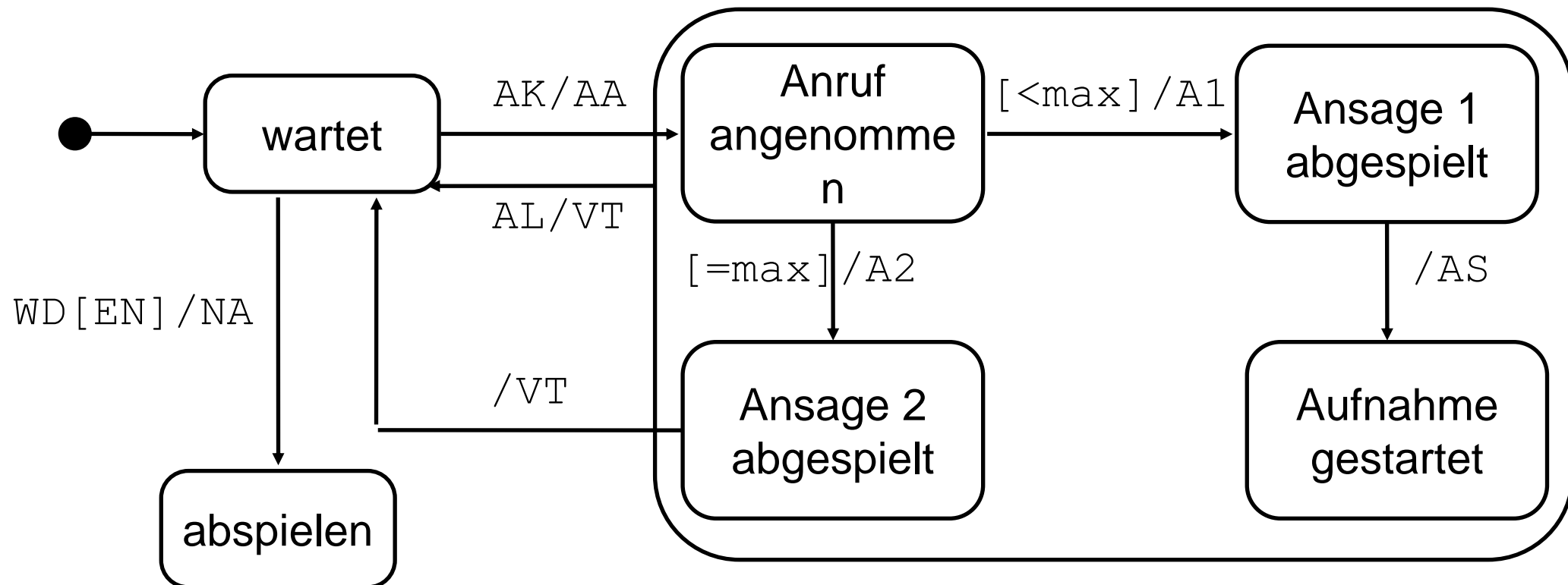
b.3) Modellieren Sie zusätzlich das folgende Verhalten: „Wenn am Anrufbeantworter der Knopf zur Wiedergabe gedrückt wird und mindestens eine Nachricht gespeichert ist, werden die gespeicherten Nachrichten abgespielt. Wenn der Knopf gedrückt wird und keine Nachricht gespeichert ist, führt der Anrufbeantworter keine Aktion aus. Während der Wiedergabe der gespeicherten Nachrichten kann der Stopp-Knopf gedrückt werden und die Wiedergabe wird gestoppt. Wenn alle Nachrichten abgespielt worden sind, wird die Wiedergabe ebenfalls gestoppt.“



b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Ereignisse:	Bedingungen:	Aktionen:
<ul style="list-style-type: none"> Anruf kommt an (AK) Anrufer legt auf (AL) Knopf zur Wiedergabe drücken (<u>WD</u>) Stopp- Knopf drücken (SD) 	<ul style="list-style-type: none"> gespeicherte Nachricht < max (<max) gespeicherte Nachricht = max (=max) mindestens eine Nachricht gespeichert (EN) keine Nachricht gespeichert (<u>KN</u>) alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> Anruf annehmen (AA) Ansage 1 abspielen (A1) Ansage 2 abspielen (A2) Aufnahme starten (AS) Verbindung trennen (VT) gespeicherte Nachrichten abspielen (NA) Wiedergabe stoppen (WS)

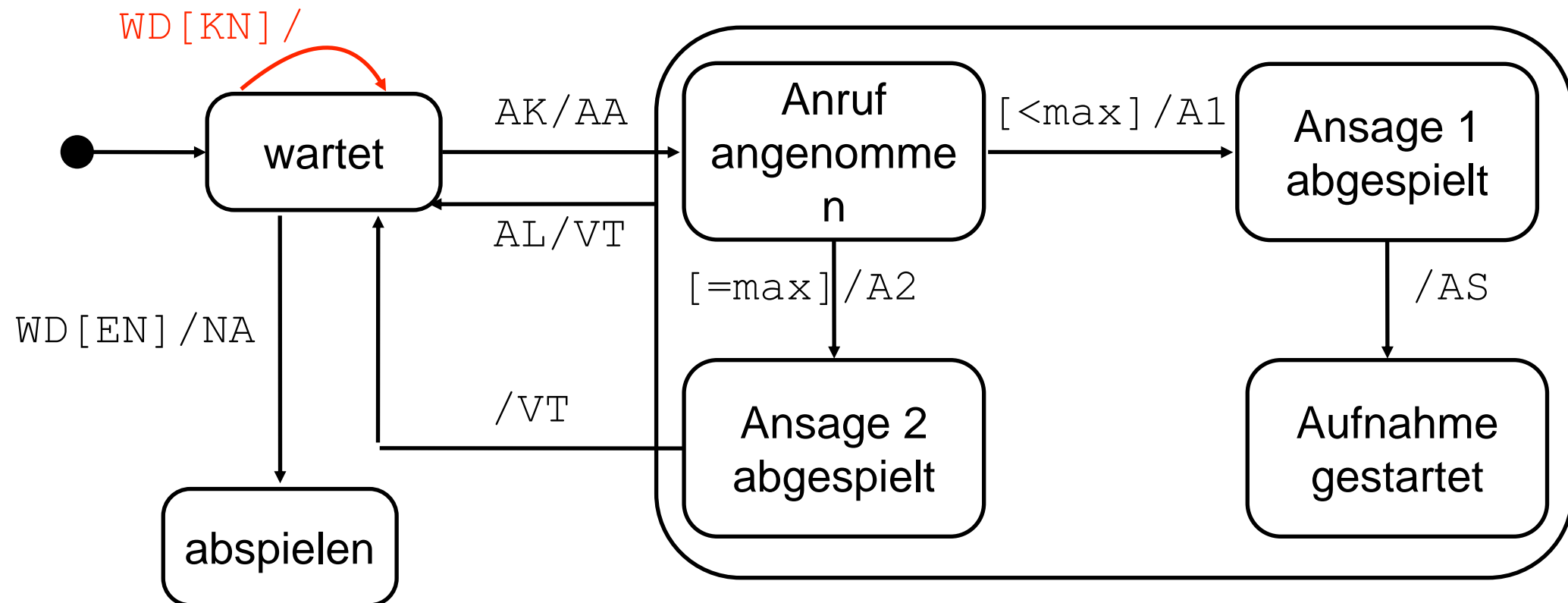
b.3) Modellieren Sie zusätzlich das folgende Verhalten: „ Wenn am Anrufbeantworter der Knopf zur Wiedergabe gedrückt wird und mindestens eine Nachricht gespeichert ist, werden die gespeicherten Nachrichten abgespielt. Wenn der Knopf gedrückt wird und keine Nachricht gespeichert ist, führt der Anrufbeantworter keine Aktion aus. Während der Wiedergabe der gespeicherten Nachrichten kann der Stopp-Knopf gedrückt werden und die Wiedergabe wird gestoppt. Wenn alle Nachrichten abgespielt worden sind, wird die Wiedergabe ebenfalls gestoppt.“



b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Ereignisse:	Bedingungen:	Aktionen:
<ul style="list-style-type: none"> Anruf kommt an (AK) Anrufer legt auf (AL) Knopf zur Wiedergabe drücken (<u>WD</u>) Stopp- Knopf drücken (SD) 	<ul style="list-style-type: none"> gespeicherte Nachricht < max (<max) gespeicherte Nachricht = max (=max) mindestens eine Nachricht gespeichert (EN) keine Nachricht gespeichert (<u>KN</u>) alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> Anruf annehmen (AA) Ansage 1 abspielen (A1) Ansage 2 abspielen (A2) Aufnahme starten (AS) Verbindung trennen (VT) gespeicherte Nachrichten abspielen (NA) Wiedergabe stoppen (WS)

b.3) Modellieren Sie zusätzlich das folgende Verhalten: „ Wenn am Anrufbeantworter der Knopf zur Wiedergabe gedrückt wird und mindestens eine Nachricht gespeichert ist, werden die gespeicherten Nachrichten abgespielt. Wenn der Knopf gedrückt wird und keine Nachricht gespeichert ist, führt der Anrufbeantworter keine Aktion aus. Während der Wiedergabe der gespeicherten Nachrichten kann der Stopp-Knopf gedrückt werden und die Wiedergabe wird gestoppt. Wenn alle Nachrichten abgespielt worden sind, wird die Wiedergabe ebenfalls gestoppt.“



b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die

Frage:

Was passiert beim event "WD" wenn diese Transition *nicht* vorhanden ist?

Aktionen:

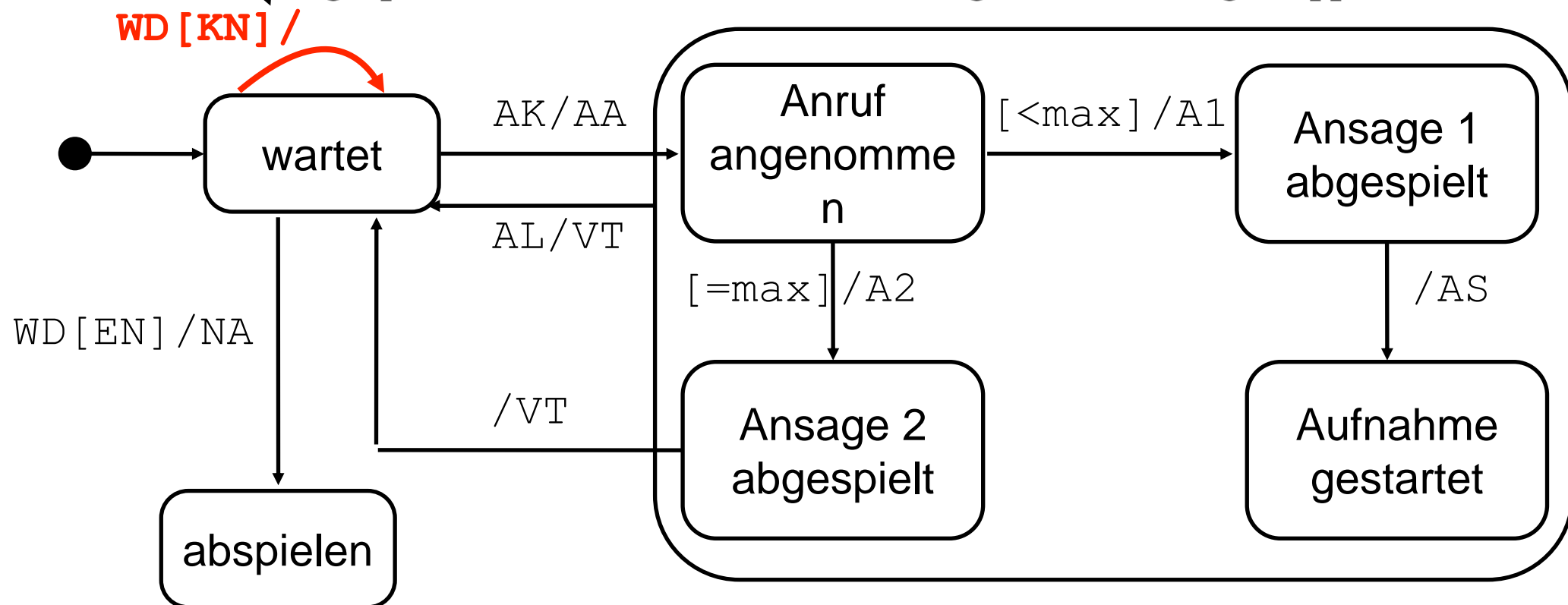
- Anruf annehmen (AA)
- Ansage 1 abspielen (A1)
- Ansage 2 abspielen (A2)
- Aufnahme starten (AS)
- Verbindung trennen (VT)
- gespeicherte Nachrichten abspielen (NA)
- Wiedergabe stoppen (WS)

Ereignisse:

- Anruf k
- Anrufe
- Knopf zur Wiedergabe drücken (WD)
- Stopp- Knopf drücken (SD)

- mindestens eine Nachricht gespeichert (EN)
- keine Nachricht gespeichert (KN)
- alle Nachrichten abgespielt (AN)

b.3) Modellieren Sie zusätzlich das folgende Verhalten: „ Wenn am Anrufbeantworter der Knopf zur Wiedergabe gedrückt wird und mindestens eine Nachricht gespeichert ist, werden die gespeicherten Nachrichten abgespielt. Wenn der Knopf gedrückt wird und keine Nachricht gespeichert ist, führt der Anrufbeantworter keine Aktion aus. Während der Wiedergabe der gespeicherten Nachrichten kann der Stopp-Knopf gedrückt werden und die Wiedergabe wird gestoppt. Wenn alle Nachrichten abgespielt worden sind, wird die Wiedergabe ebenfalls gestoppt.“

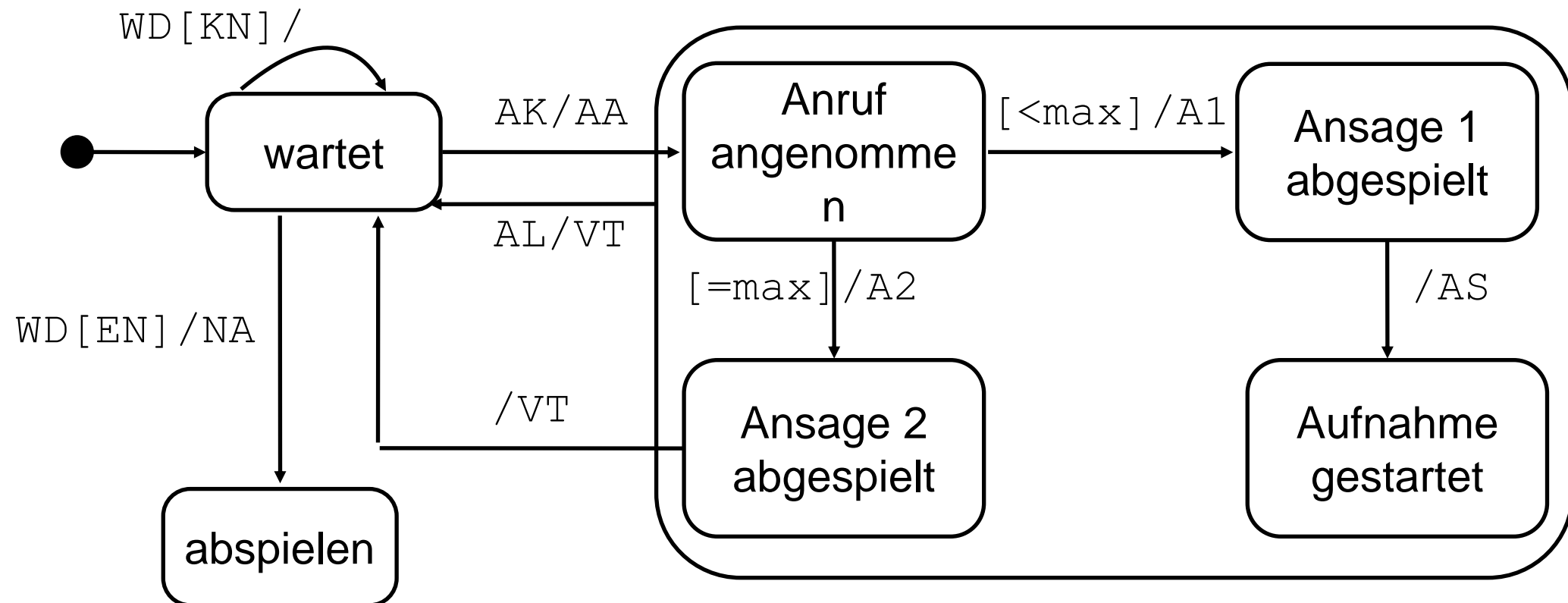


b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (AL) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken <u>(SD)</u> 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (VT) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen <u>(WS)</u>
---	---	--

b.3) Modellieren Sie zusätzlich das folgende Verhalten: „ Wenn am Anrufbeantworter der Knopf zur Wiedergabe gedrückt wird und mindestens eine Nachricht gespeichert ist, werden die gespeicherten Nachrichten abgespielt. Wenn der Knopf gedrückt wird und keine Nachricht gespeichert ist, führt der Anrufbeantworter keine Aktion aus. Während der Wiedergabe der gespeicherten Nachrichten kann der Stopp-Knopf gedrückt werden und die Wiedergabe wird gestoppt. Wenn alle Nachrichten abgespielt worden sind, wird die Wiedergabe ebenfalls gestoppt.“

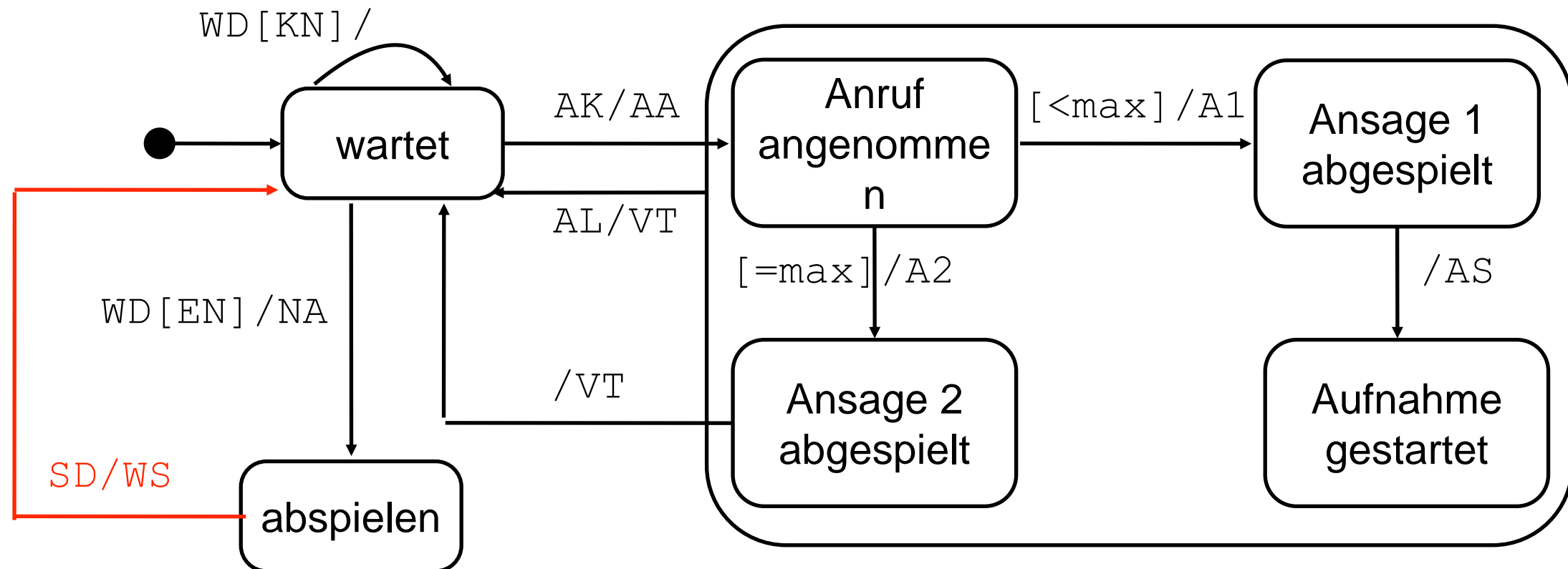


b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (AL) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken <u>(SD)</u> 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (VT) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen <u>(WS)</u>
---	---	--

b.3) Modellieren Sie zusätzlich das folgende Verhalten: „ Wenn am Anrufbeantworter der Knopf zur Wiedergabe gedrückt wird und mindestens eine Nachricht gespeichert ist, werden die gespeicherten Nachrichten abgespielt. Wenn der Knopf gedrückt wird und keine Nachricht gespeichert ist, führt der Anrufbeantworter keine Aktion aus. Während der Wiedergabe der gespeicherten Nachrichten kann der Stopp-Knopf gedrückt werden und die Wiedergabe wird gestoppt. Wenn alle Nachrichten abgespielt worden sind, wird die Wiedergabe ebenfalls gestoppt.“

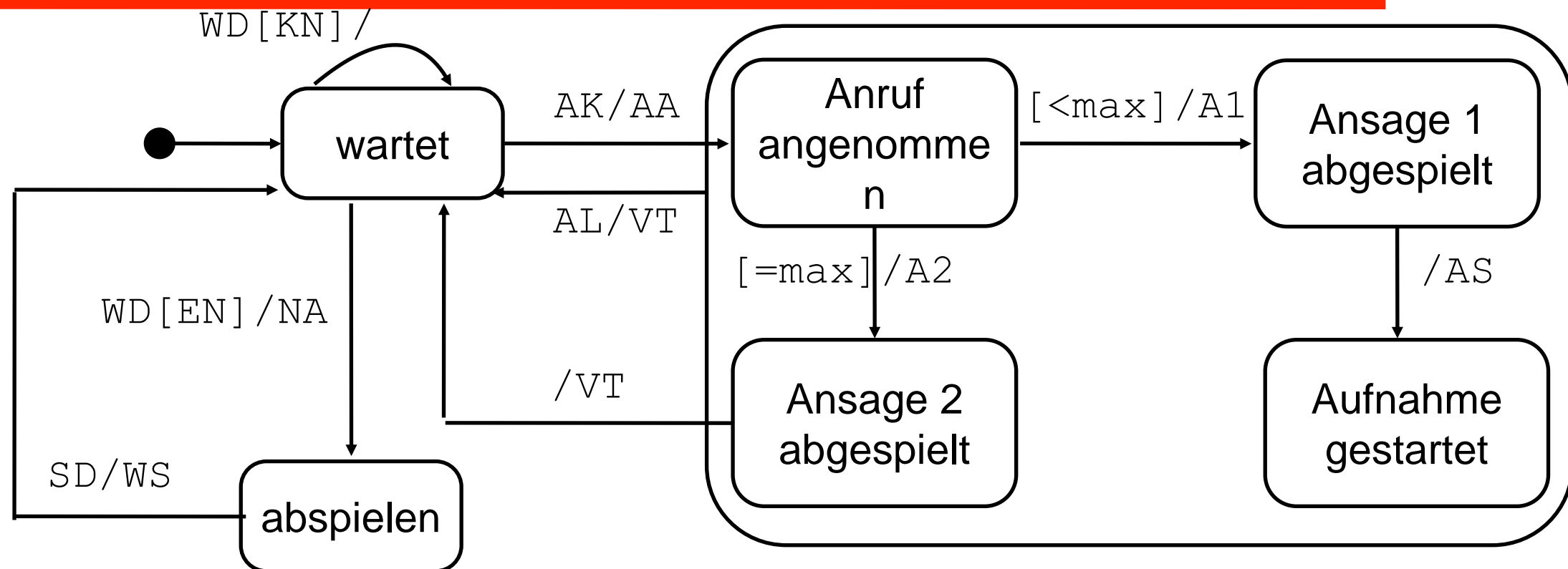


b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (AL) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (VT) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)
--	---	---

b.3) Modellieren Sie zusätzlich das folgende Verhalten: „ Wenn am Anrufbeantworter der Knopf zur Wiedergabe gedrückt wird und mindestens eine Nachricht gespeichert ist, werden die gespeicherten Nachrichten abgespielt. Wenn der Knopf gedrückt wird und keine Nachricht gespeichert ist, führt der Anrufbeantworter keine Aktion aus. Während der Wiedergabe der gespeicherten Nachrichten kann der Stopp-Knopf gedrückt werden und die Wiedergabe wird gestoppt. Wenn alle Nachrichten abgespielt worden sind, wird die Wiedergabe ebenfalls gestoppt.“

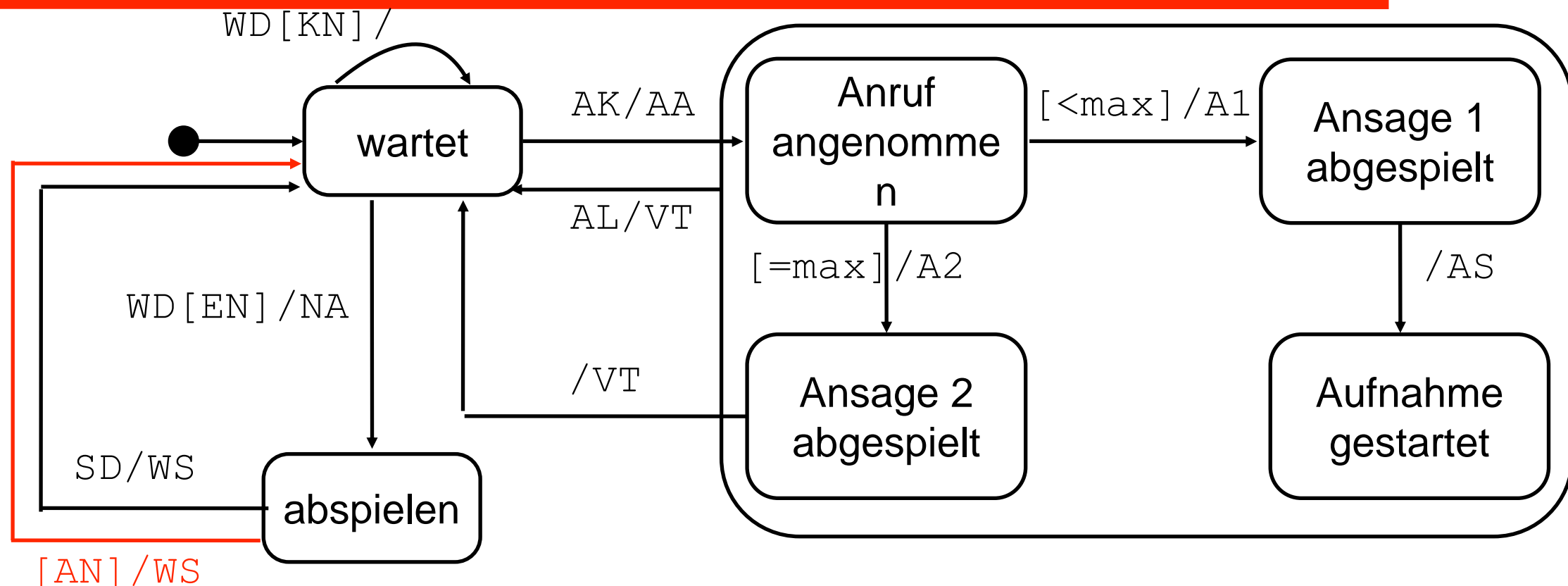


b) Beschreiben Sie das Verhalten eines Anrufbeantworters mit Hilfe eines UML-Statecharts. Bitte benutzen sie in Ihre Lösung die folgenden Abkürzungen:

Aktionen:

<p><i>Ereignisse:</i></p> <ul style="list-style-type: none"> • Anruf kommt an (AK) • Anrufer legt auf (AL) • Knopf zur Wiedergabe drücken (WD) • Stopp- Knopf drücken (SD) 	<p><i>Bedingungen:</i></p> <ul style="list-style-type: none"> • gespeicherte Nachricht < max (<max) • gespeicherte Nachricht = max (=max) • mindestens eine Nachricht gespeichert (EN) • keine Nachricht gespeichert (KN) • alle Nachrichten abgespielt (AN) 	<ul style="list-style-type: none"> • Anruf annehmen (AA) • Ansage 1 abspielen (A1) • Ansage 2 abspielen (A2) • Aufnahme starten (AS) • Verbindung trennen (VT) • gespeicherte Nachrichten abspielen (NA) • Wiedergabe stoppen (WS)
--	---	---

b.3) Modellieren Sie zusätzlich das folgende Verhalten: „ Wenn am Anrufbeantworter der Knopf zur Wiedergabe gedrückt wird und mindestens eine Nachricht gespeichert ist, werden die gespeicherten Nachrichten abgespielt. Wenn der Knopf gedrückt wird und keine Nachricht gespeichert ist, führt der Anrufbeantworter keine Aktion aus. Während der Wiedergabe der gespeicherten Nachrichten kann der Stopp-Knopf gedrückt werden und die Wiedergabe wird gestoppt. Wenn alle Nachrichten abgespielt worden sind, wird die Wiedergabe ebenfalls gestoppt.“

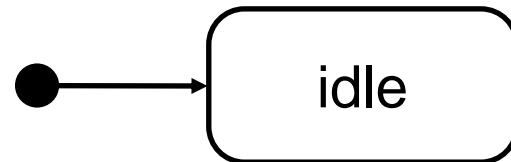


Fuel pump

- Model the behavior of a fuel pump controller.
- User can buy fuel after inserting a credit card, which is read and validated by the controller.
- Then the user takes the hose out of the holster, and pushes the nozzle trigger, to fuel his car.
- When the nozzle is off, the fuel flow is stopped and the price is charged on the credit card
- If invalid card or timeout the system returns to the initial waiting state.

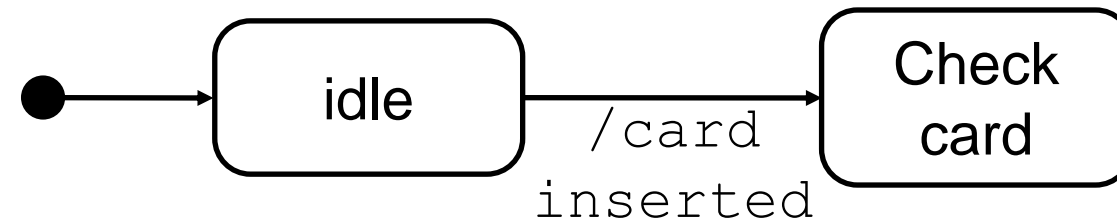
Fuel pump

- Model the behavior of a fuel pump controller.
- User can buy fuel after inserting a credit card, which is read and validated by the controller.
- Then the user takes the hose out of the holster, and pushes the nozzle trigger, to fuel his car.
- When the nozzle is off, the fuel flow is stopped and the price is charged on the credit card
- If invalid card or timeout the system returns to the initial waiting state.



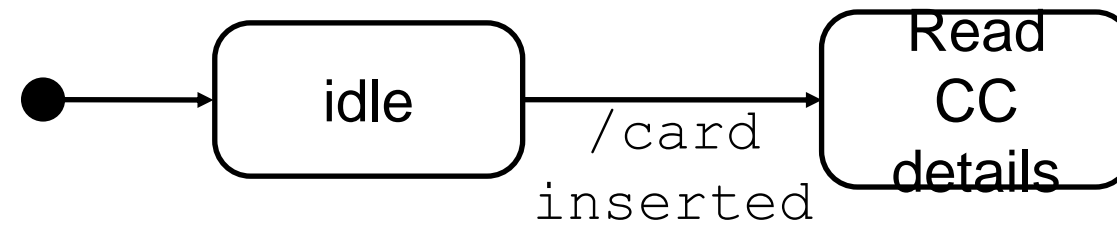
Fuel pump

- Model the behavior of a fuel pump controller.
- User can buy fuel after inserting a credit card, which is read and validated by the controller.
- Then the user takes the hose out of the holster, and pushes the nozzle trigger, to fuel his car.
- When the nozzle is off, the fuel flow is stopped and the price is charged on the credit card
- If invalid card or timeout the system returns to the initial waiting state.



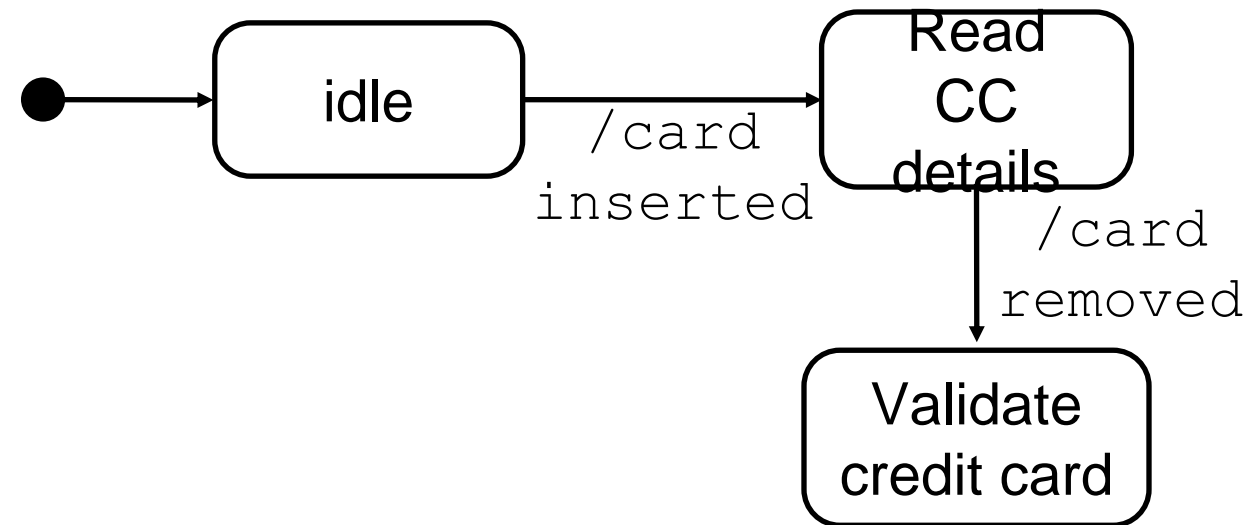
Fuel pump

- Model the behavior of a fuel pump controller.
- User can buy fuel after inserting a credit card, which is read and validated by the controller.
- Then the user takes the hose out of the holster, and pushes the nozzle trigger, to fuel his car.
- When the nozzle is off, the fuel flow is stopped and the price is charged on the credit card
- If invalid card or timeout the system returns to the initial waiting state.



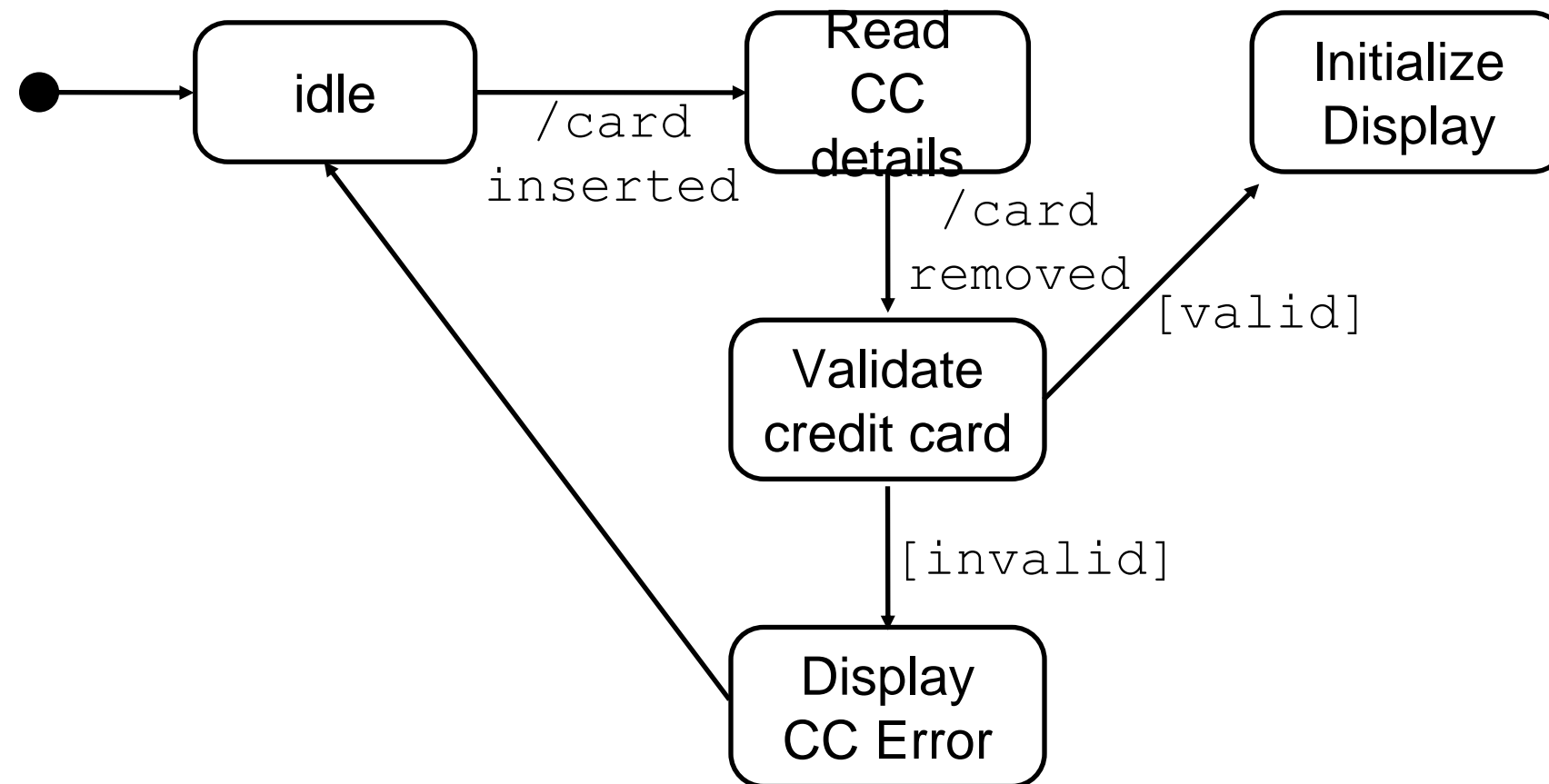
Fuel pump

- Model the behavior of a fuel pump controller.
- User can buy fuel after inserting a credit card, which is read and validated by the controller.
- Then the user takes the hose out of the holster, and pushes the nozzle trigger, to fuel his car.
- When the nozzle is off, the fuel flow is stopped and the price is charged on the credit card
- If invalid card or timeout the system returns to the initial waiting state.



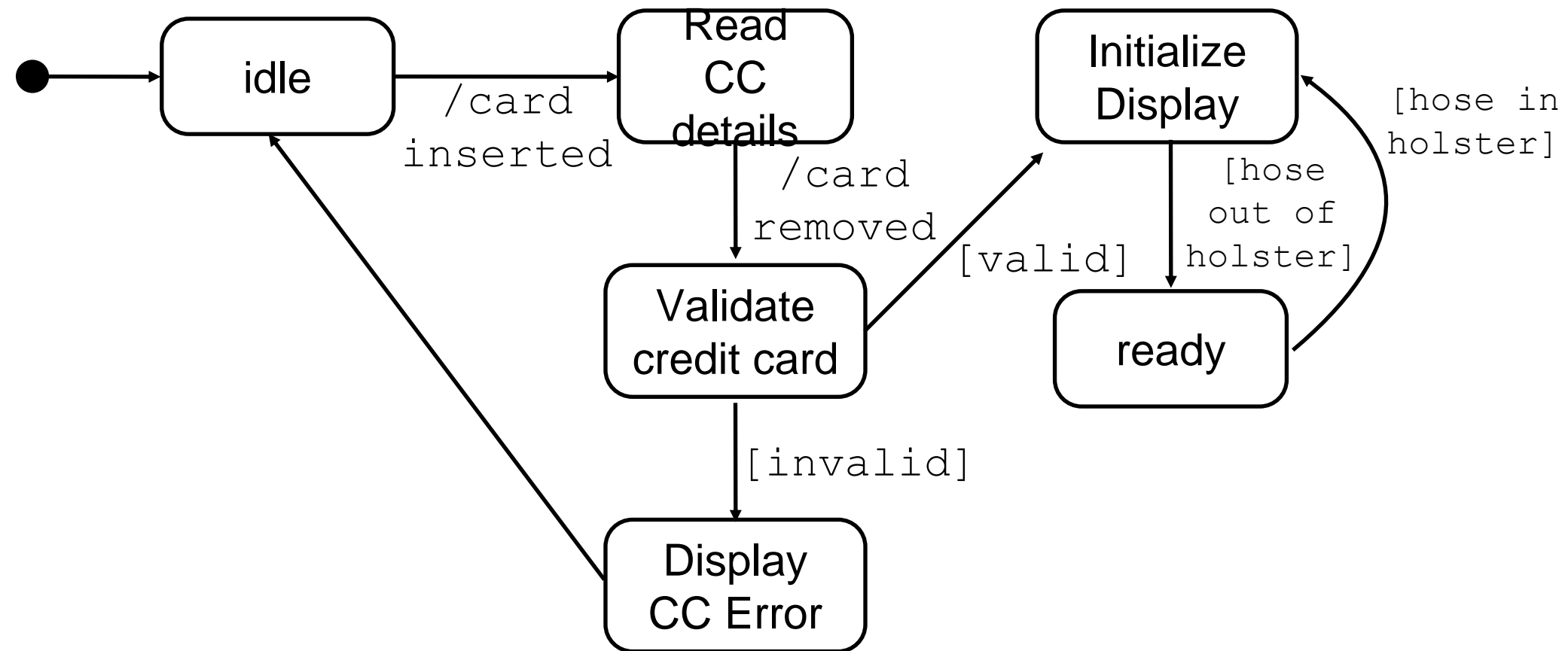
Fuel pump

- Model the behavior of a fuel pump controller.
- User can buy fuel after inserting a credit card, which is read and validated by the controller.
- Then the user takes the hose out of the holster, and pushes the nozzle trigger, to fuel his car.
- When the nozzle is off, the fuel flow is stopped and the price is charged on the credit card
- If invalid card or timeout the system returns to the initial waiting state.



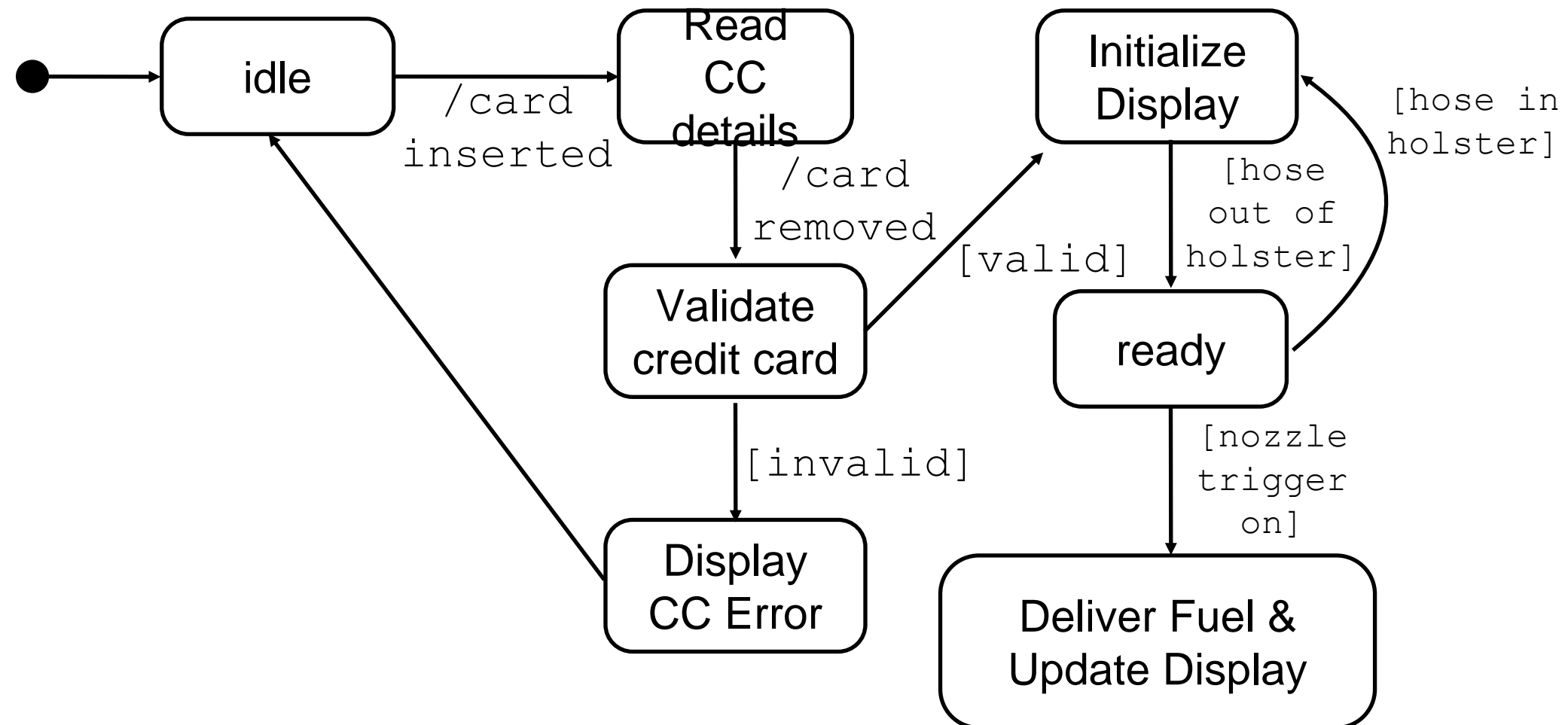
Fuel pump

- Model the behavior of a fuel pump controller.
 - User can buy fuel after inserting a credit card, which is read and validated by the controller.
 - Then the user takes the hose out of the holster, and pushes the nozzle trigger, to fuel his car.
- When the nozzle is off, the fuel flow is stopped and the price is charged on the credit card
 - If invalid card or timeout the system returns to the initial waiting state.



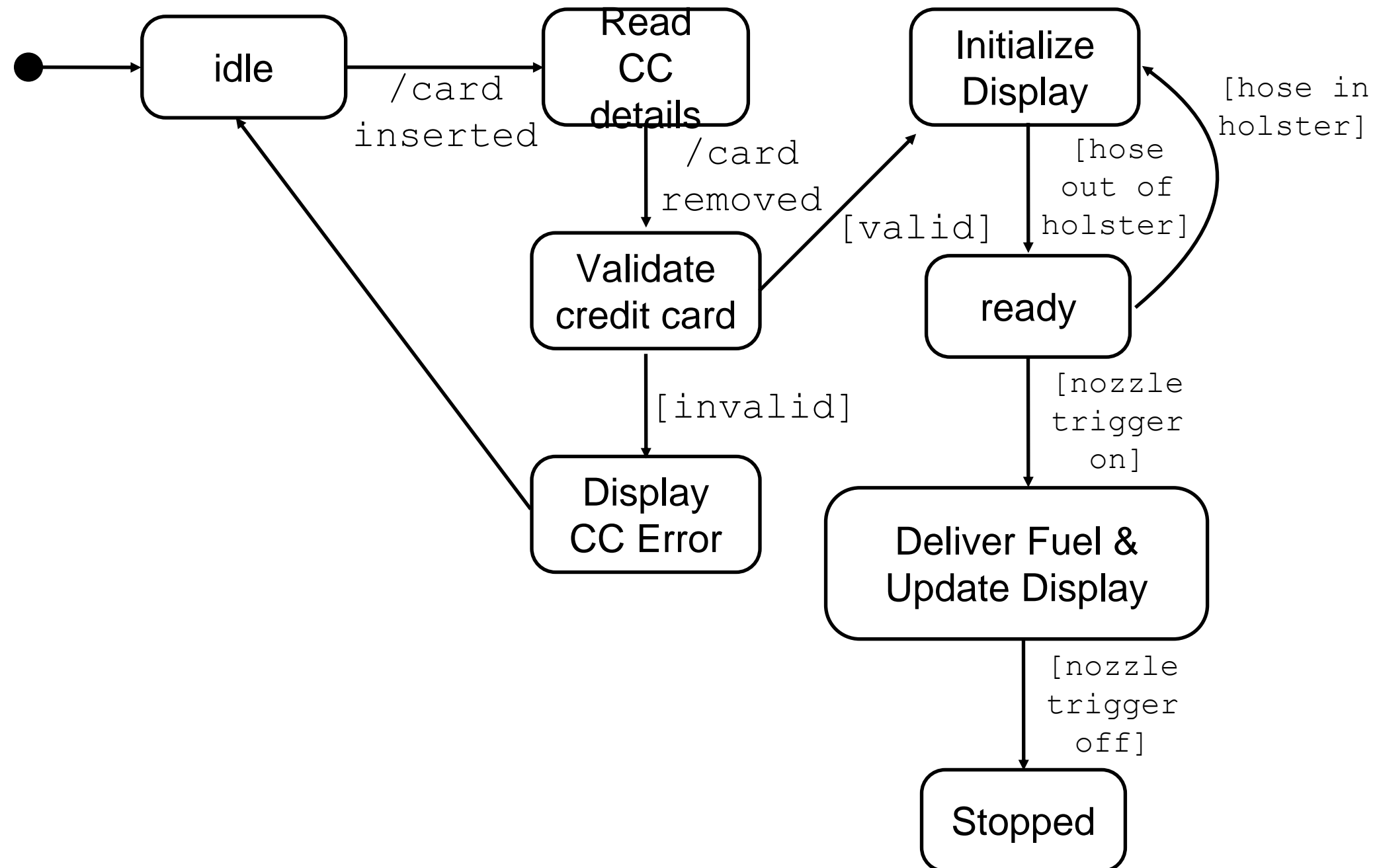
Fuel pump

- Model the behavior of a fuel pump controller.
- User can buy fuel after inserting a credit card, which is read and validated by the controller.
- Then the user takes the hose out of the holster, and pushes the nozzle trigger, to fuel his car.
- When the nozzle is off, the fuel flow is stopped and the price is charged on the credit card
- If invalid card or timeout the system returns to the initial waiting state.



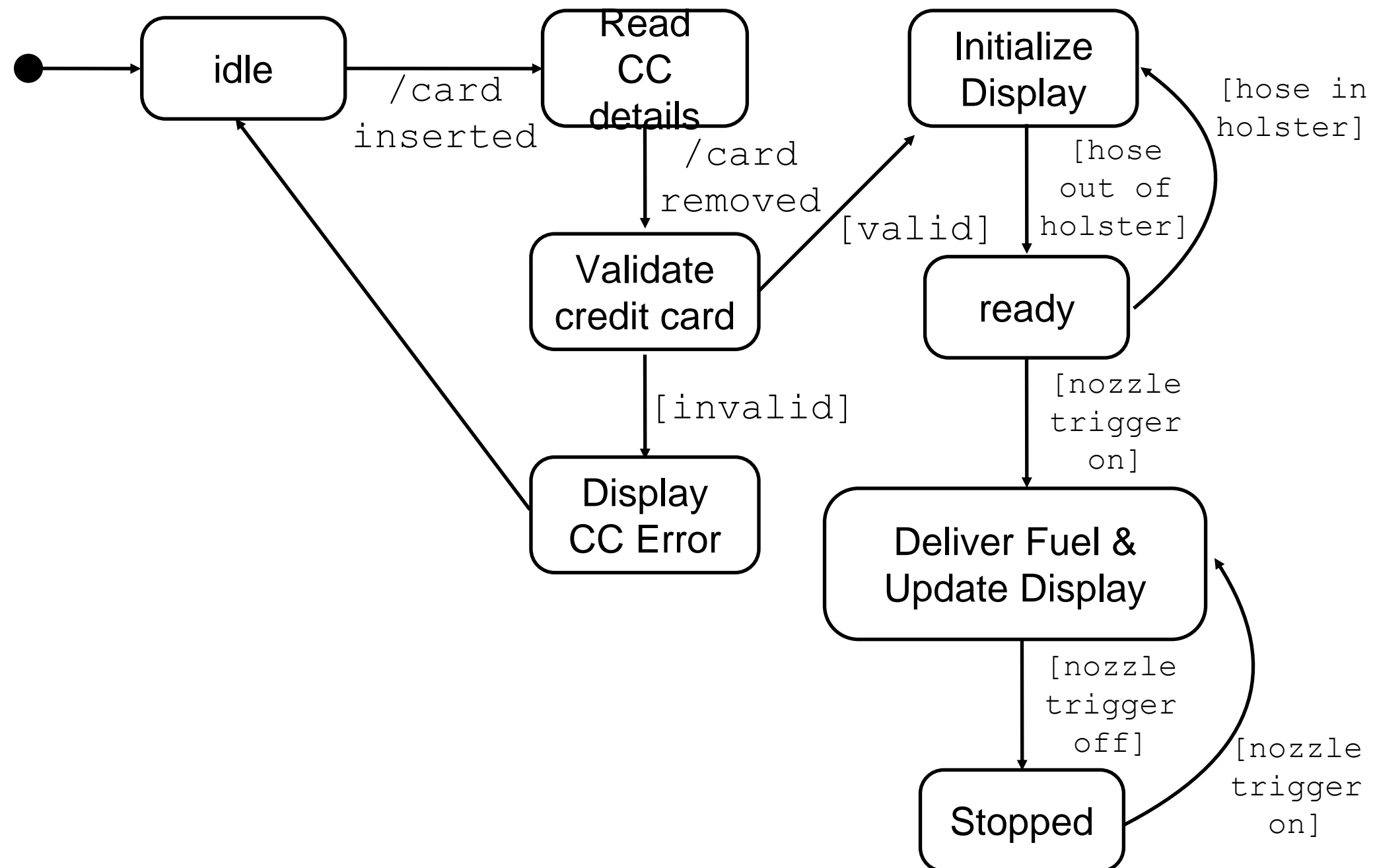
Fuel pump

- Model the behavior of a fuel pump controller.
- User can buy fuel after inserting a credit card, which is read and validated by the controller.
- Then the user takes the hose out of the holster, and pushes the nozzle trigger, to fuel his car.
- When the nozzle is off, the fuel flow is stopped and the price is charged on the credit card
- If invalid card or timeout the system returns to the initial waiting state.



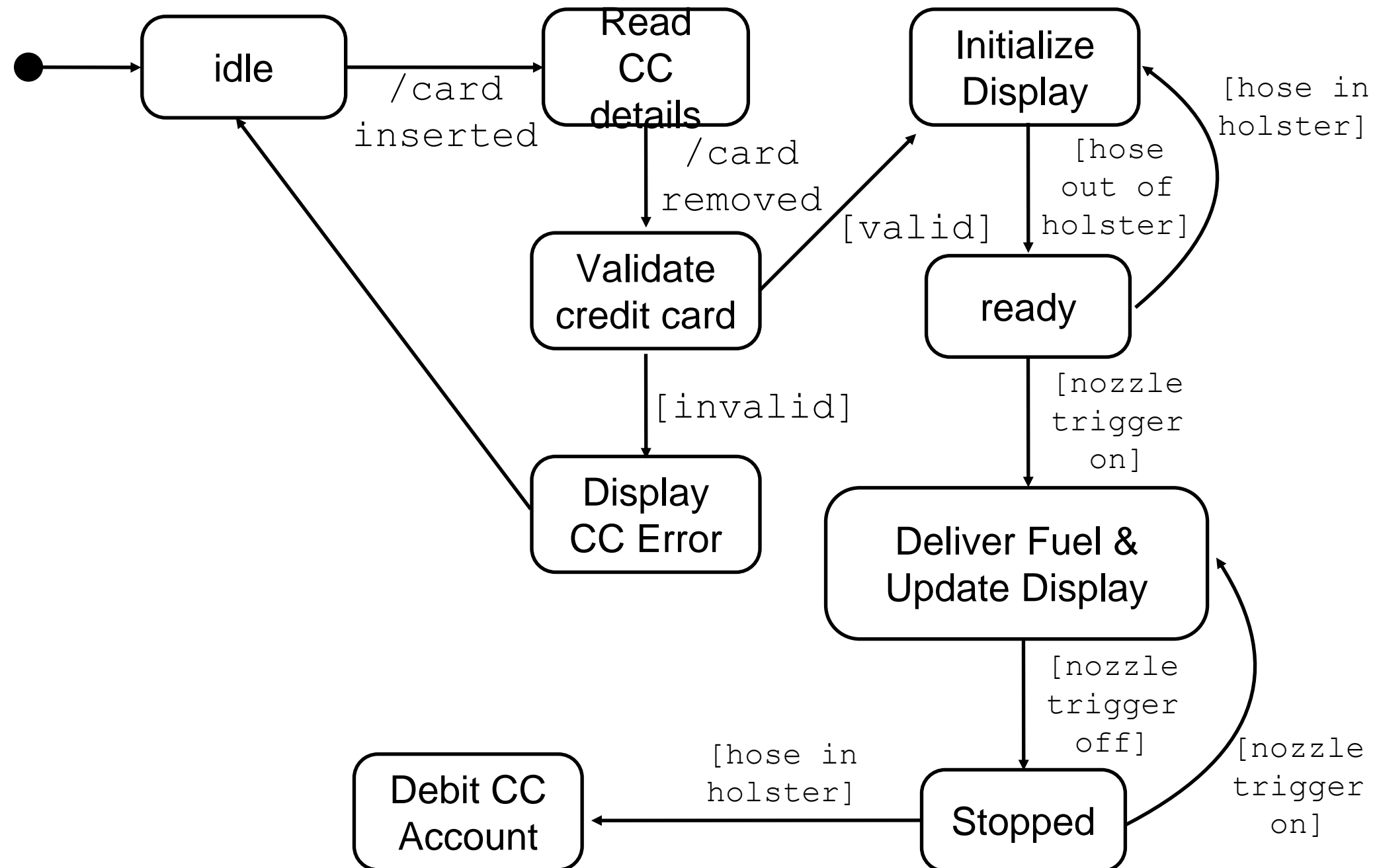
Fuel pump

- Model the behavior of a fuel pump controller.
- User can buy fuel after inserting a credit card, which is read and validated by the controller.
- Then the user takes the hose out of the holster, and pushes the nozzle trigger, to fuel his car.
- When the nozzle is off, the fuel flow is stopped and the price is charged on the credit card
- If invalid card or timeout the system returns to the initial waiting state.



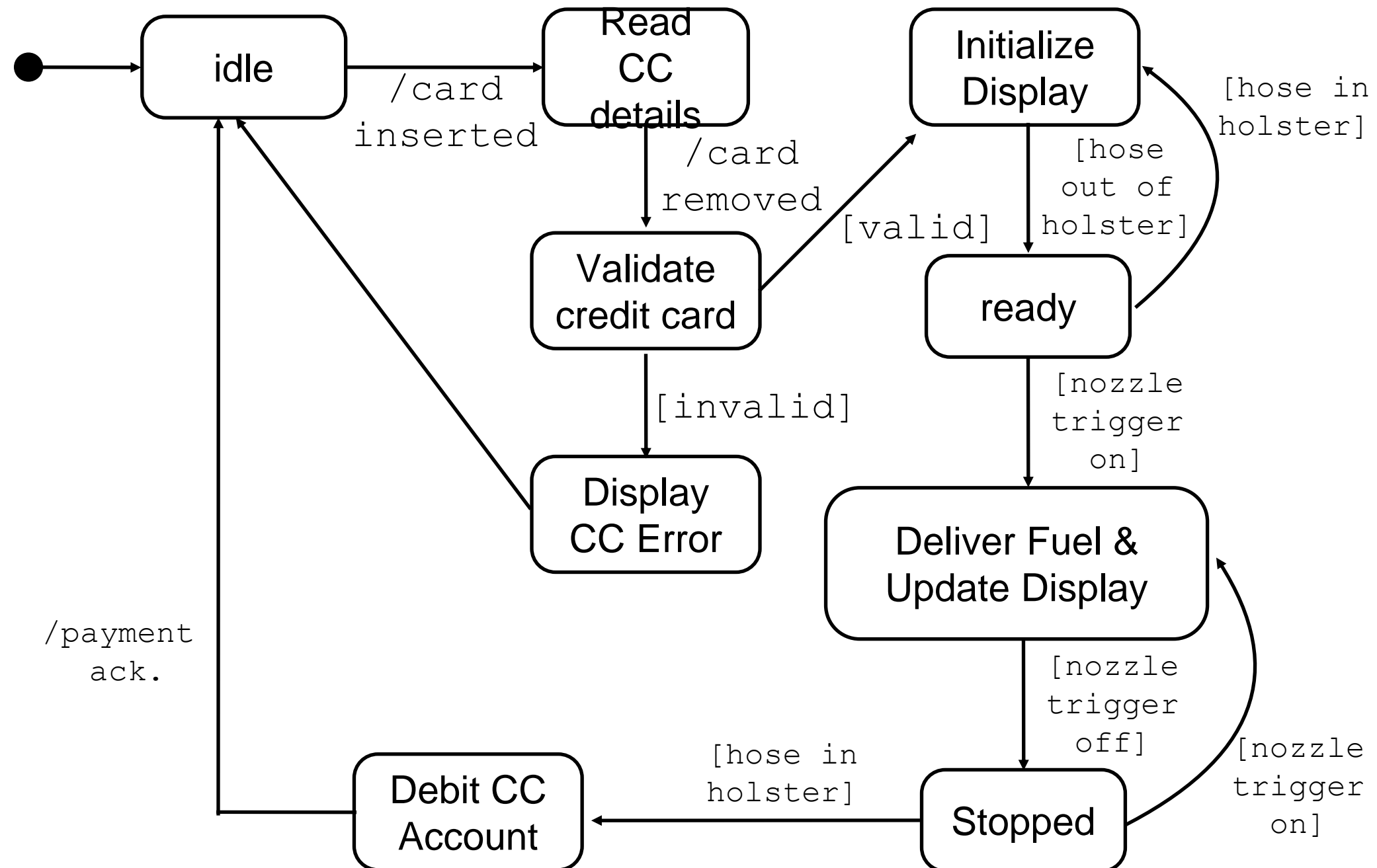
Fuel pump

- Model the behavior of a fuel pump controller.
- User can buy fuel after inserting a credit card, which is read and validated by the controller.
- Then the user takes the hose out of the holster, and pushes the nozzle trigger, to fuel his car.
- When the nozzle is off, the fuel flow is stopped and the price is charged on the credit card
- If invalid card or timeout the system returns to the initial waiting state.



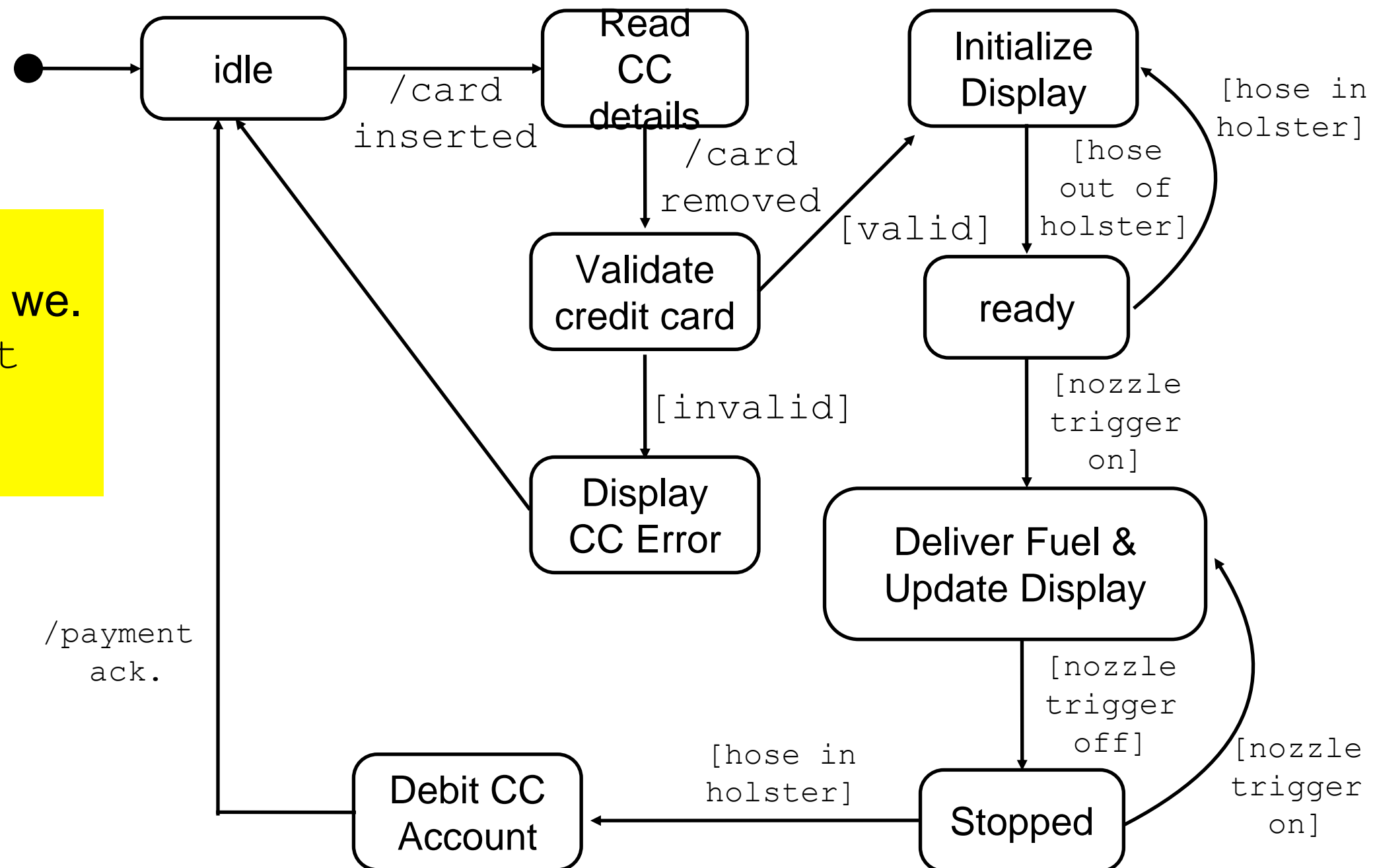
Fuel pump

- Model the behavior of a fuel pump controller.
- User can buy fuel after inserting a credit card, which is read and validated by the controller.
- Then the user takes the hose out of the holster, and pushes the nozzle trigger, to fuel his car.
- When the nozzle is off, the fuel flow is stopped and the price is charged on the credit card
- If invalid card or timeout the system returns to the initial waiting state.



Fuel pump

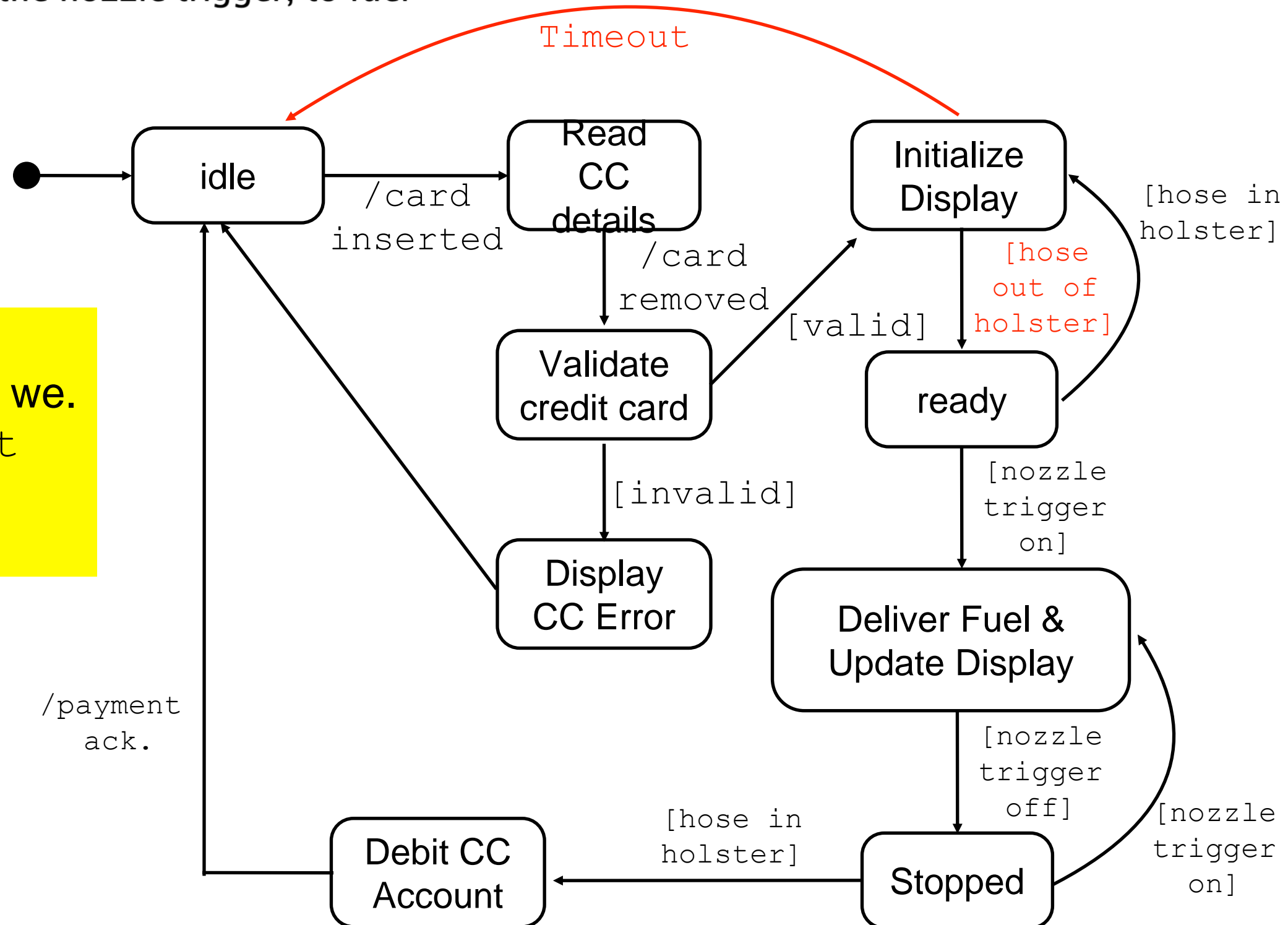
- Model the behavior of a fuel pump controller.
- User can buy fuel after inserting a credit card, which is read and validated by the controller.
- Then the user takes the hose out of the holster, and pushes the nozzle trigger, to fuel his car.
- When the nozzle is off, the fuel flow is stopped and the price is charged on the credit card
- If invalid card or timeout the system returns to the initial waiting state.



Where should we have Timeout events?

Fuel pump

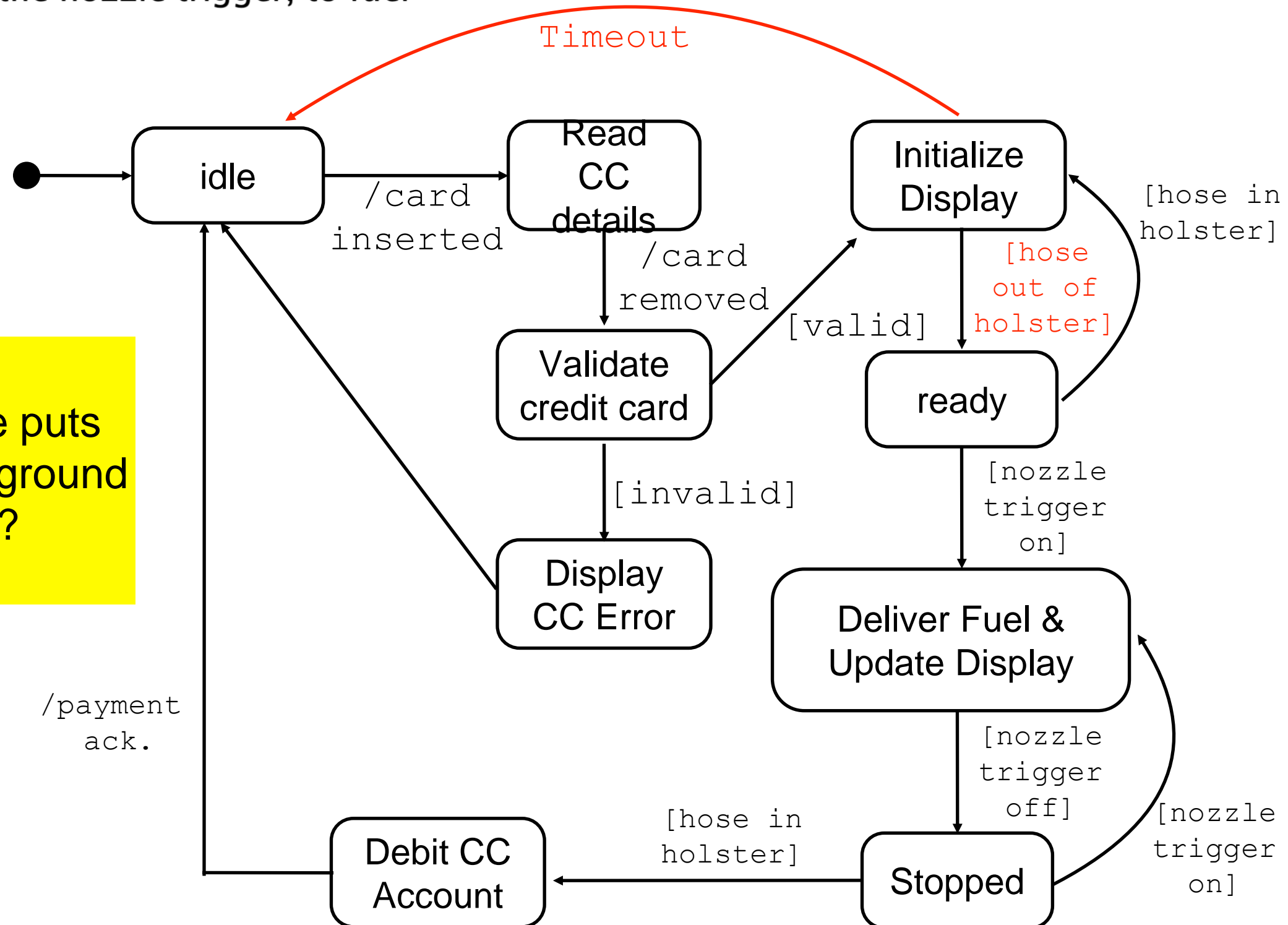
- Model the behavior of a fuel pump controller.
- User can buy fuel after inserting a credit card, which is read and validated by the controller.
- Then the user takes the hose out of the holster, and pushes the nozzle trigger, to fuel his car.
- When the nozzle is off, the fuel flow is stopped and the price is charged on the credit card
- If invalid card or timeout the system returns to the initial waiting state.



Where should we have Timeout events?

Fuel pump

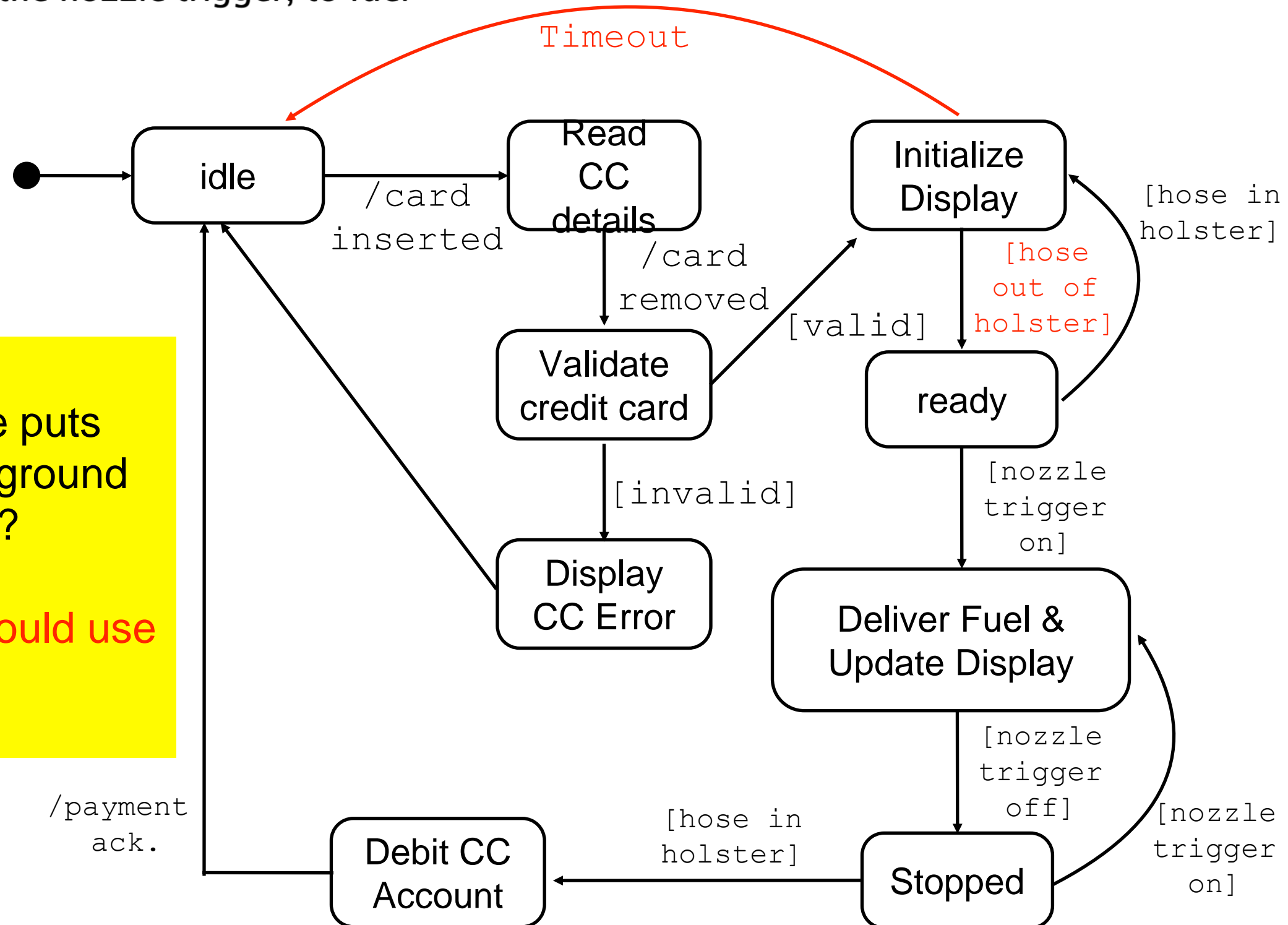
- Model the behavior of a fuel pump controller.
- User can buy fuel after inserting a credit card, which is read and validated by the controller.
- Then the user takes the hose out of the holster, and pushes the nozzle trigger, to fuel his car.
- When the nozzle is off, the fuel flow is stopped and the price is charged on the credit card
- If invalid card or timeout the system returns to the initial waiting state.



What if someone puts the hose on the ground and drives away?

Fuel pump

- Model the behavior of a fuel pump controller.
- User can buy fuel after inserting a credit card, which is read and validated by the controller.
- Then the user takes the hose out of the holster, and pushes the nozzle trigger, to fuel his car.
- When the nozzle is off, the fuel flow is stopped and the price is charged on the credit card
- If invalid card or timeout the system returns to the initial waiting state.

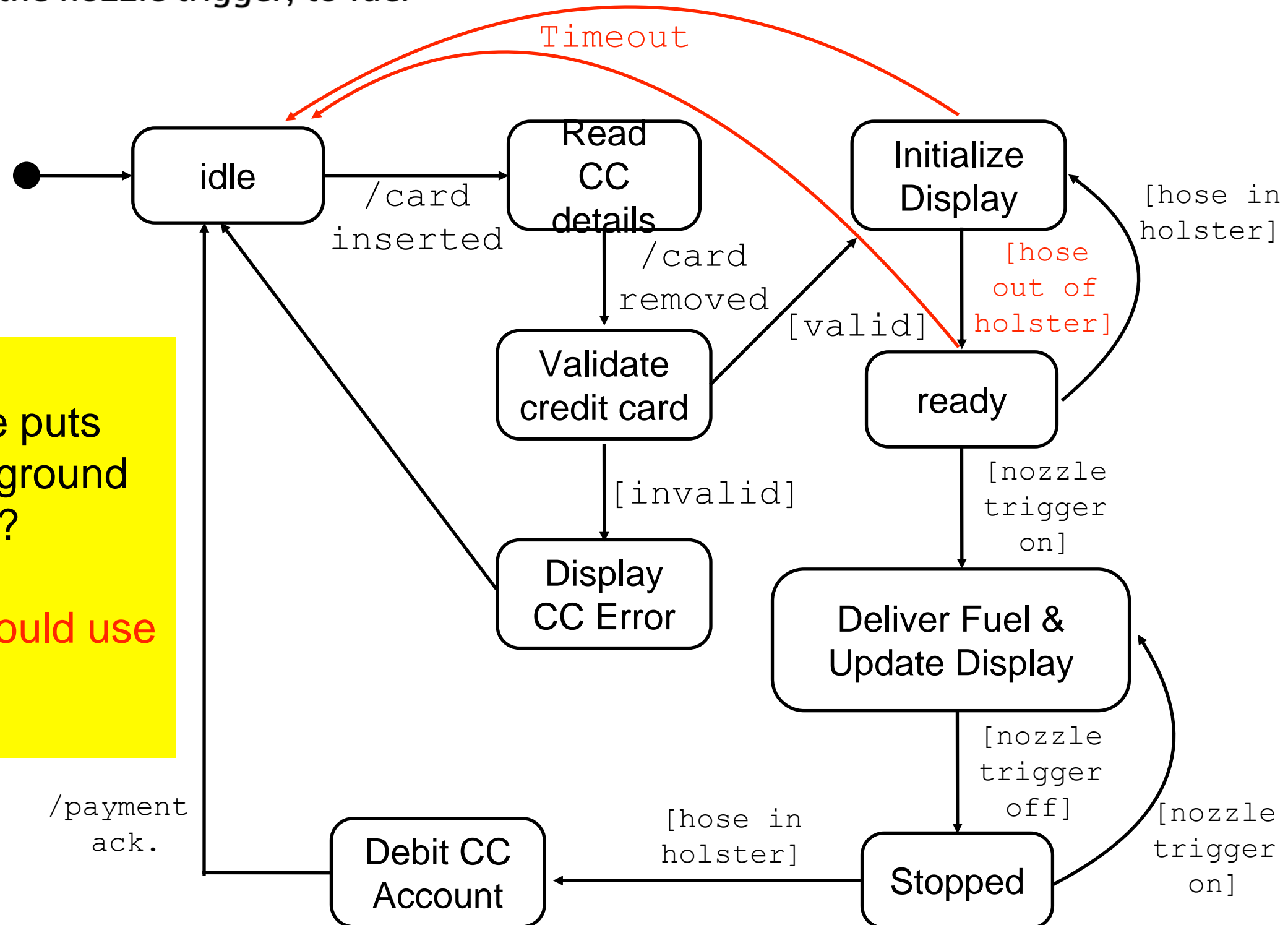


What if someone puts the hose on the ground and drives away?

Someone else could use this CC account!

Fuel pump

- Model the behavior of a fuel pump controller.
- User can buy fuel after inserting a credit card, which is read and validated by the controller.
- Then the user takes the hose out of the holster, and pushes the nozzle trigger, to fuel his car.
- When the nozzle is off, the fuel flow is stopped and the price is charged on the credit card
- If invalid card or timeout the system returns to the initial waiting state.

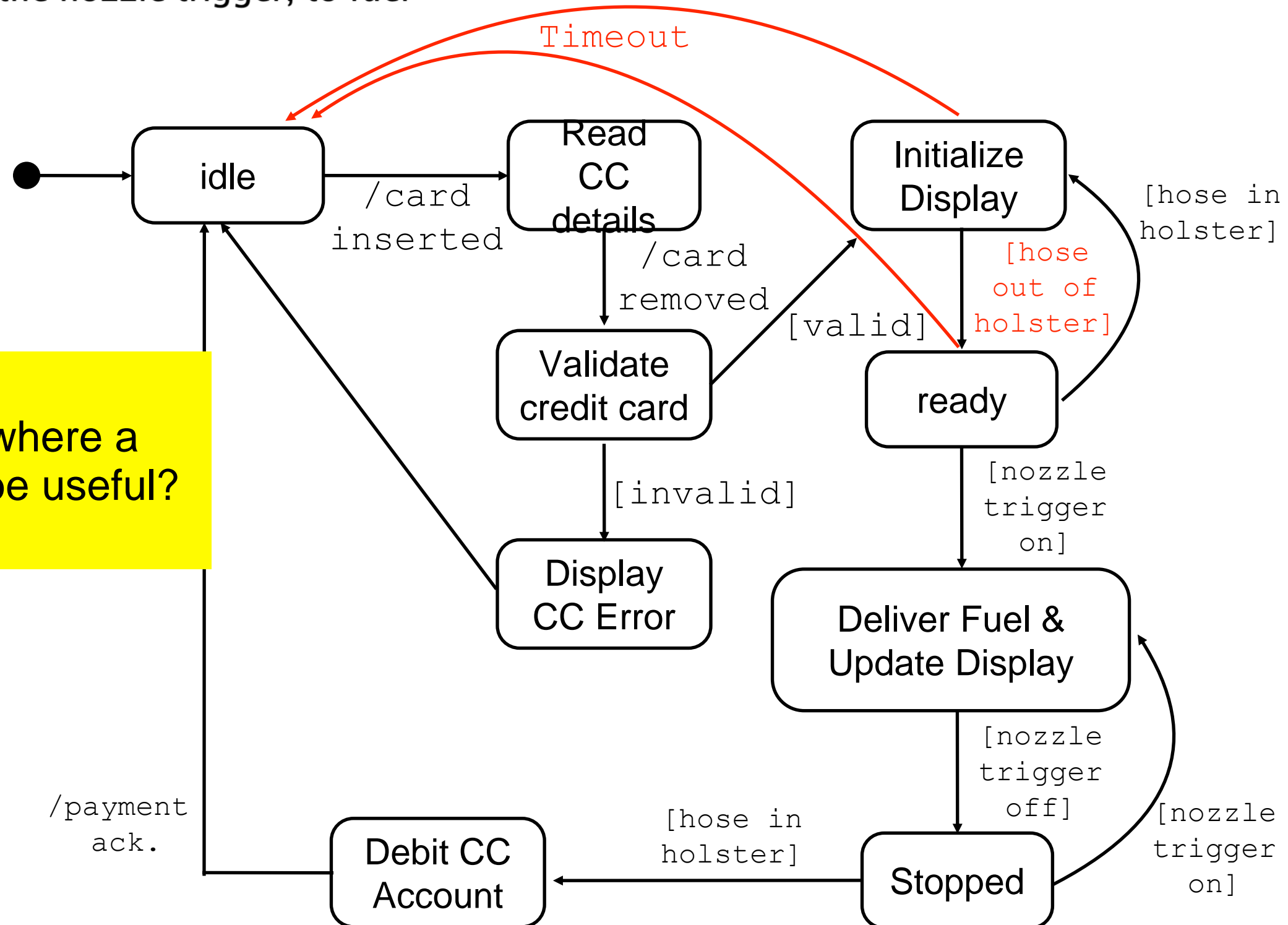


What if someone puts the hose on the ground and drives away?

Someone else could use this CC account!

Fuel pump

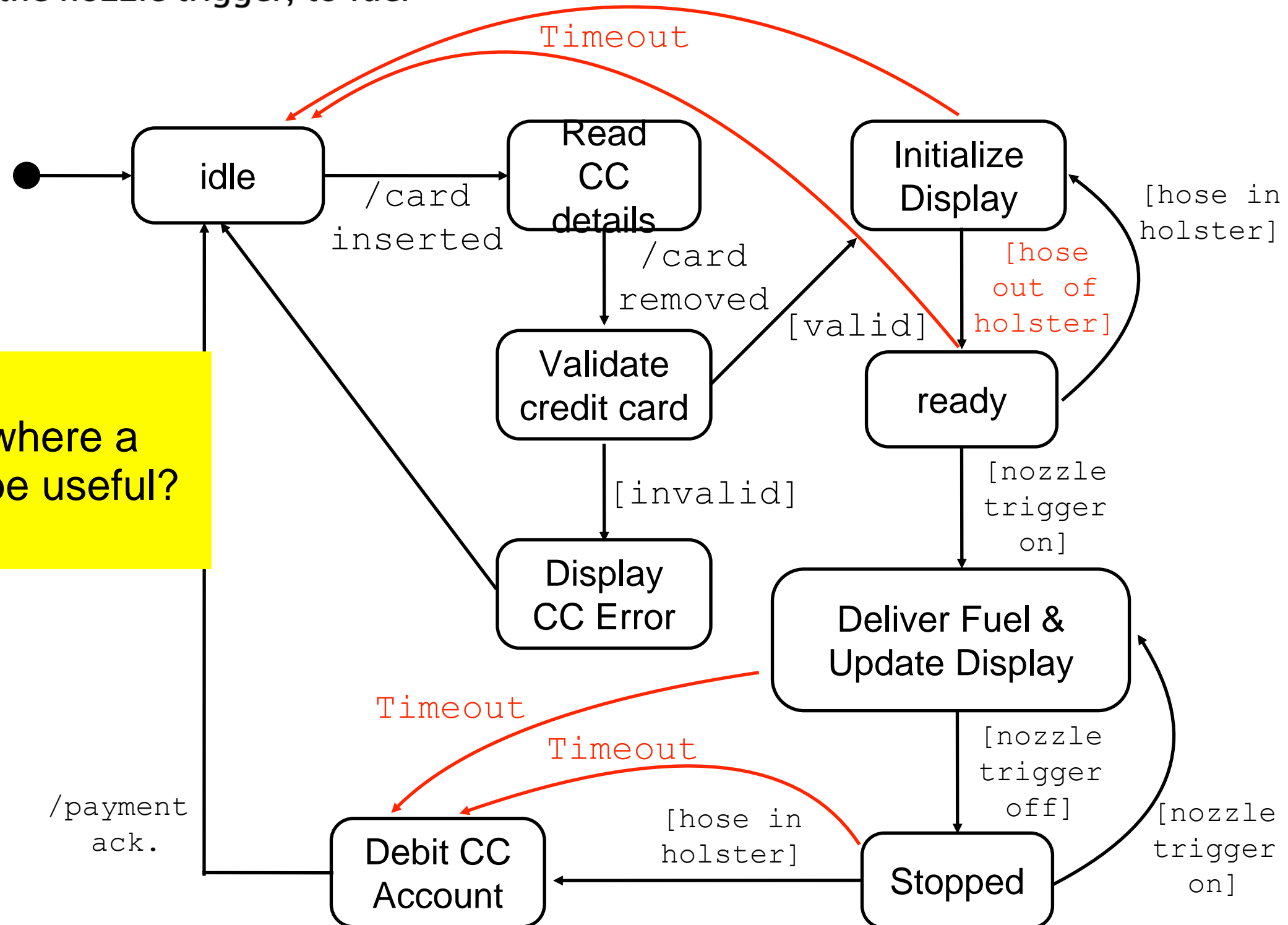
- Model the behavior of a fuel pump controller.
- User can buy fuel after inserting a credit card, which is read and validated by the controller.
- Then the user takes the hose out of the holster, and pushes the nozzle trigger, to fuel his car.
- When the nozzle is off, the fuel flow is stopped and the price is charged on the credit card
- If invalid card or timeout the system returns to the initial waiting state.



Any other place where a Timeout could be useful?

Fuel pump

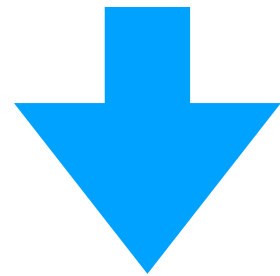
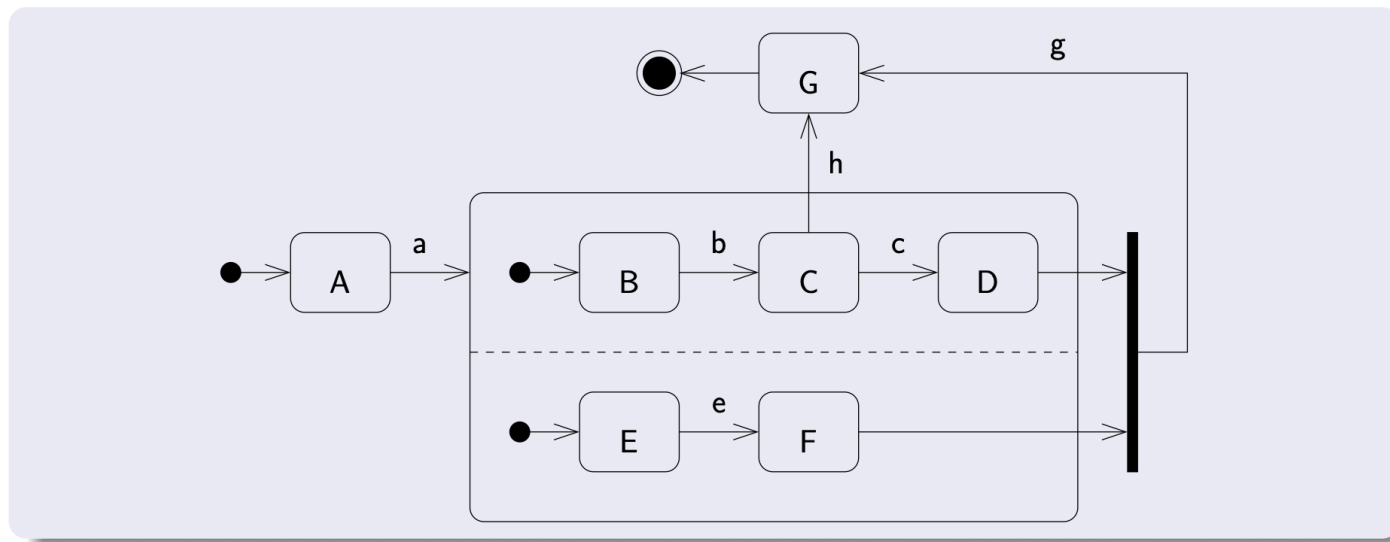
- Model the behavior of a fuel pump controller.
- User can buy fuel after inserting a credit card, which is read and validated by the controller.
- Then the user takes the hose out of the holster, and pushes the nozzle trigger, to fuel his car.
- When the nozzle is off, the fuel flow is stopped and the price is charged on the credit card
- If invalid card or timeout the system returns to the initial waiting state.



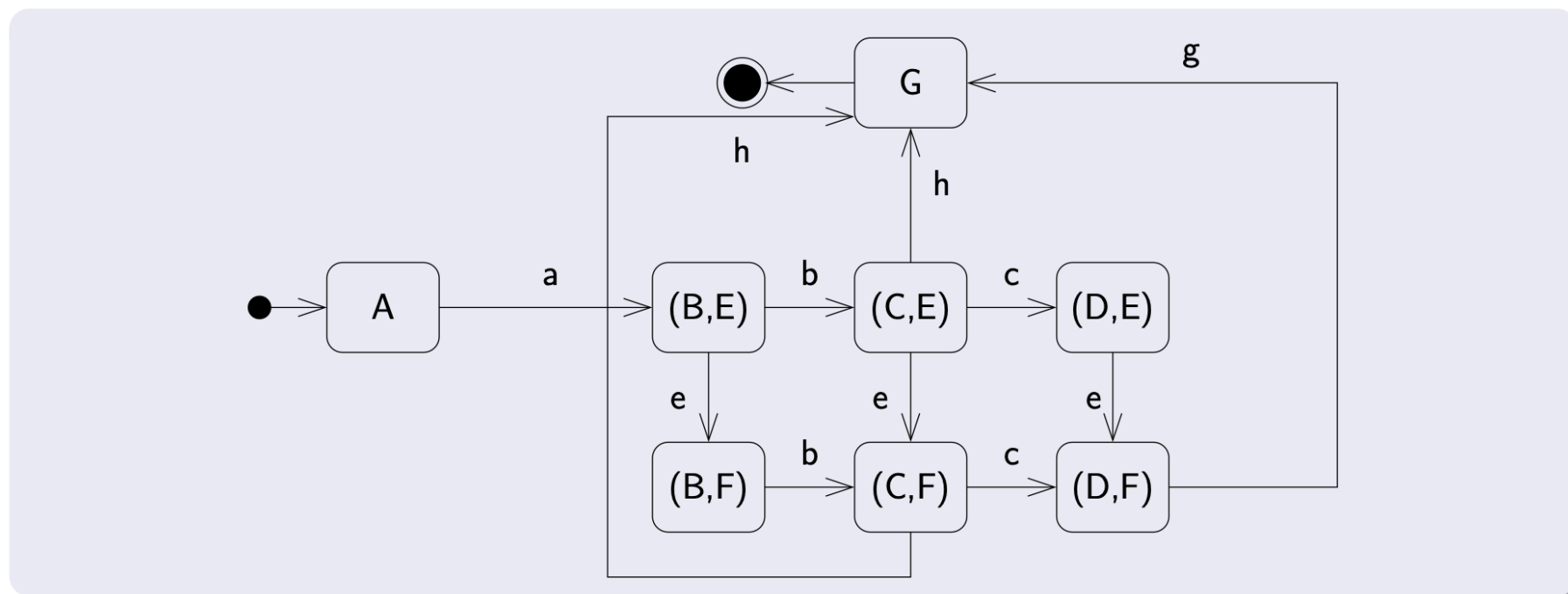
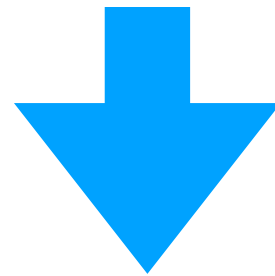
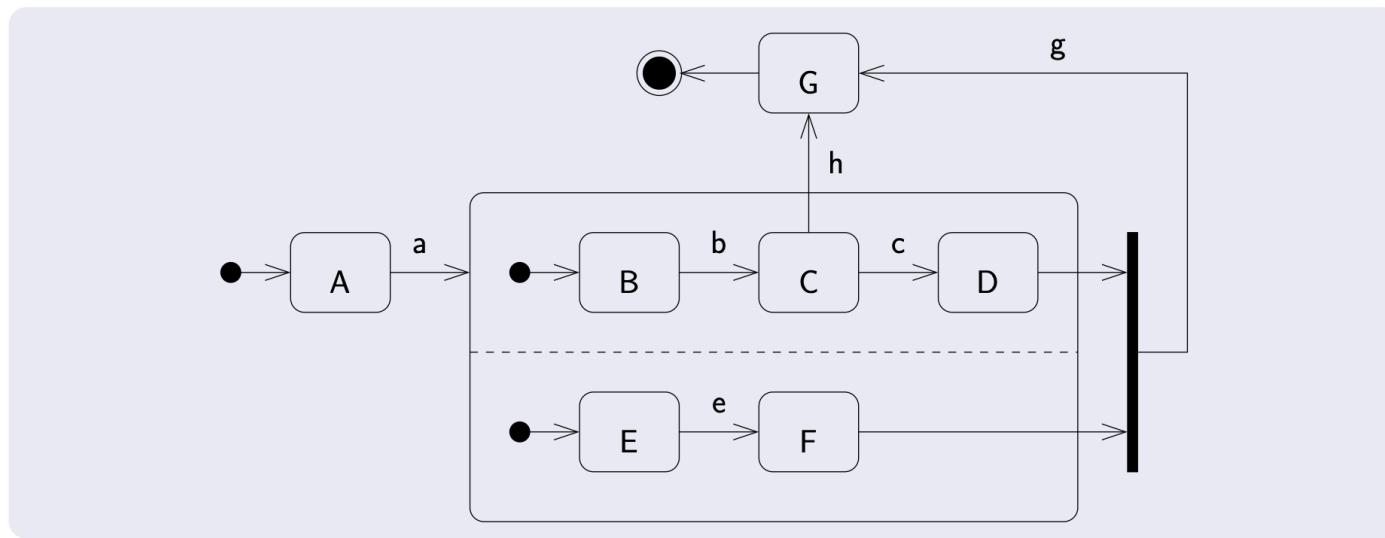
Any other place where a Timeout could be useful?

Conversion: from State Chart to Finite “Combined” Automaton

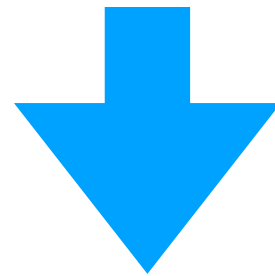
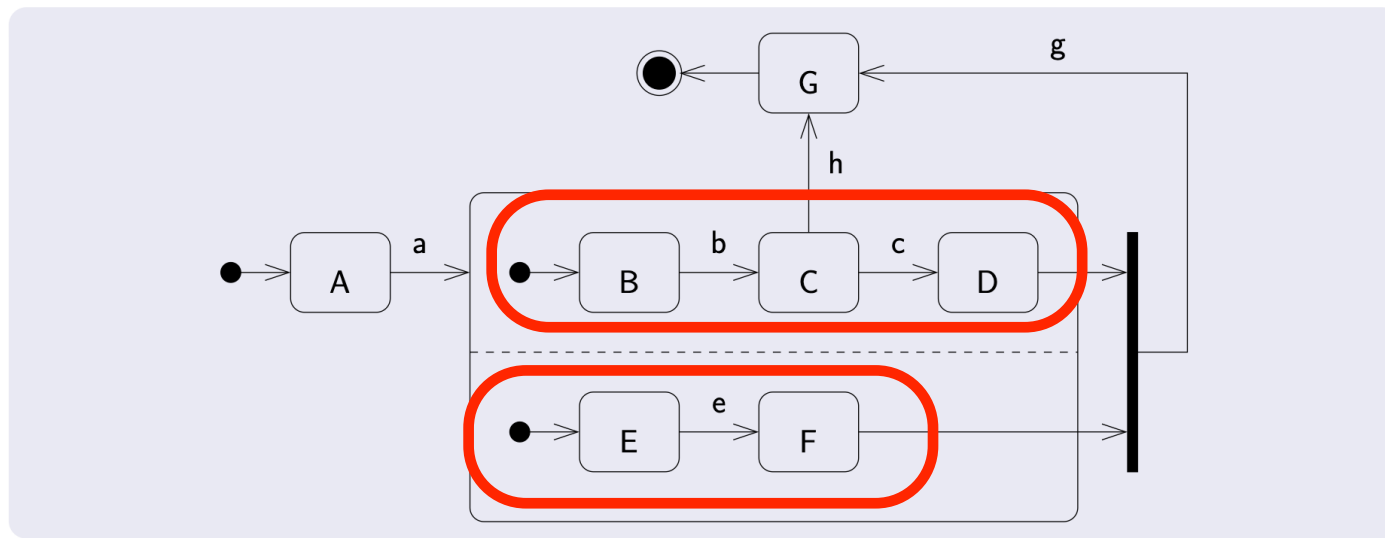
Conversion: from State Chart to Finite “Combined” Automaton



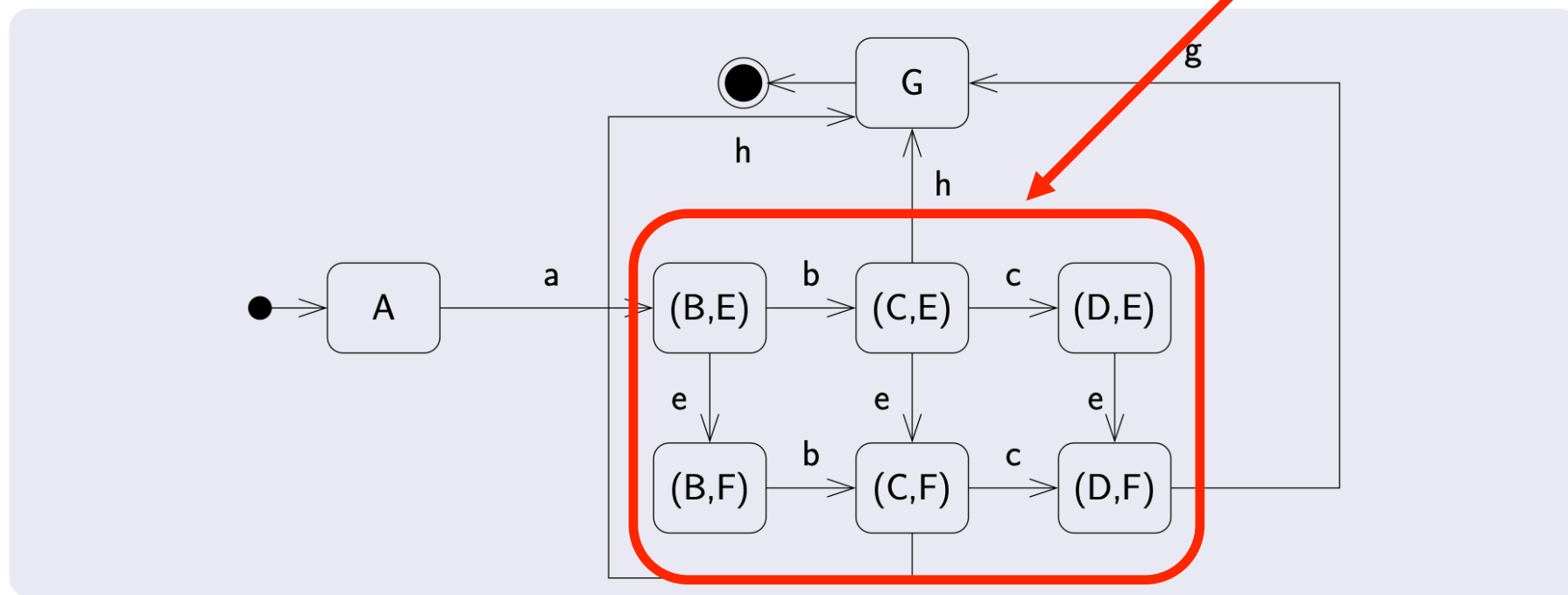
Conversion: from State Chart to Finite “Combined” Automaton

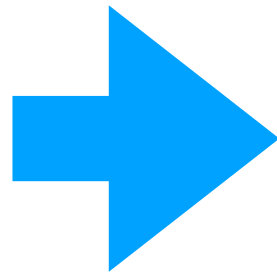
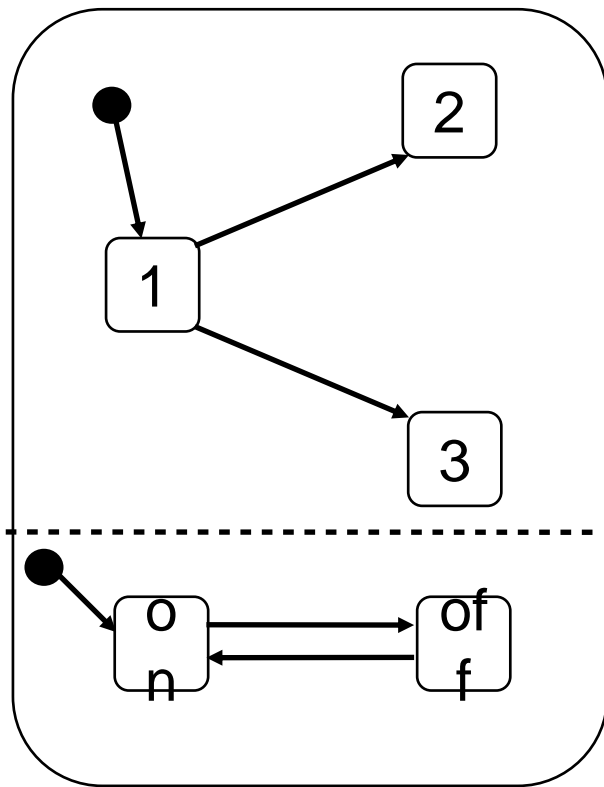


Conversion: from State Chart to Finite “Combined” Automaton



product automaton of
the two region automata

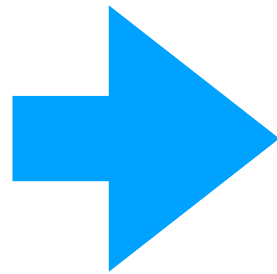
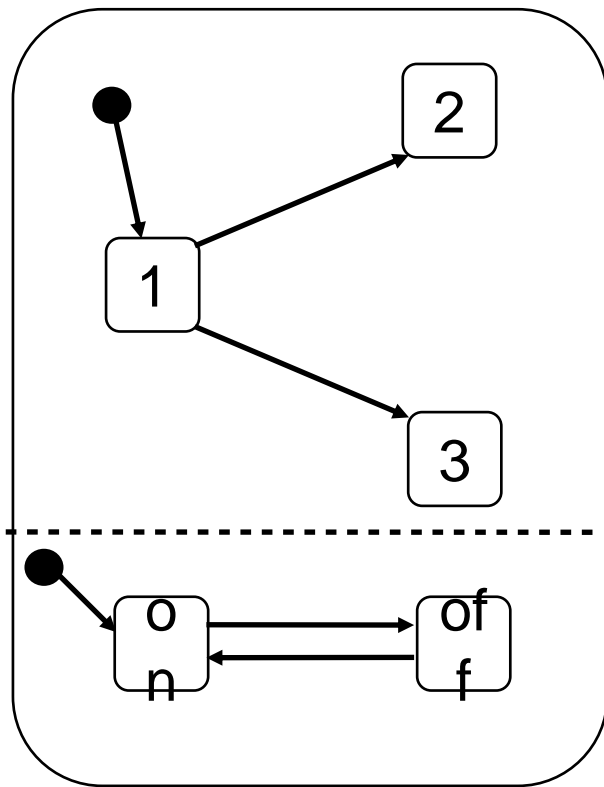




Combined Automaton

states?

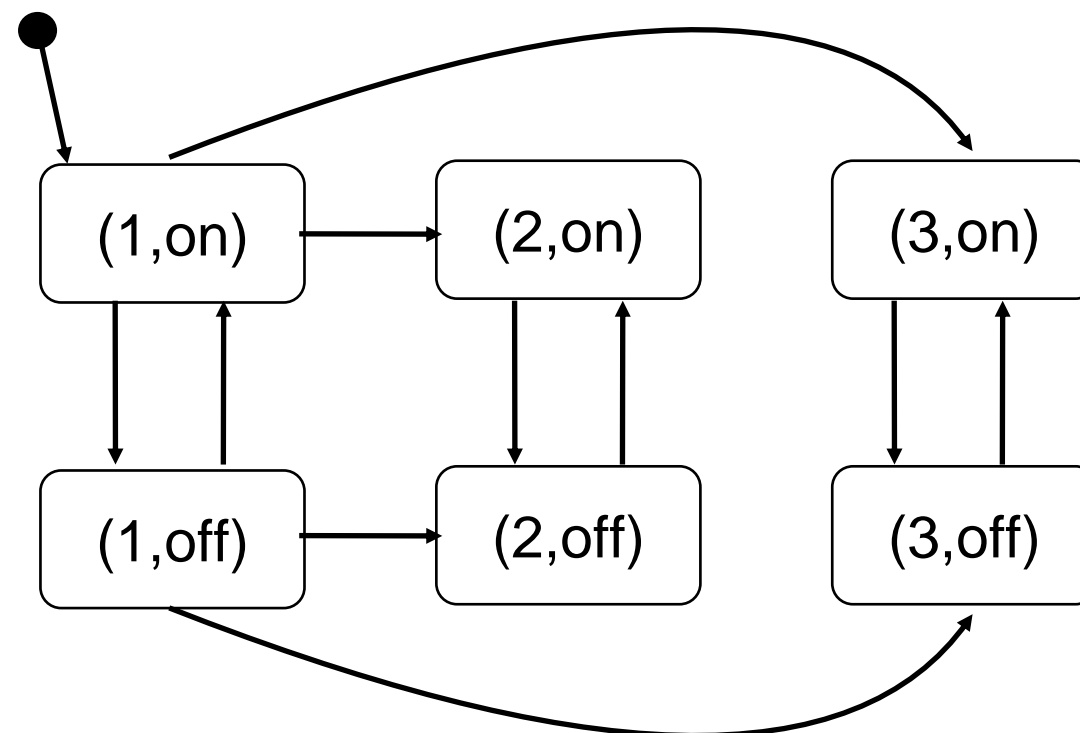
transitions?

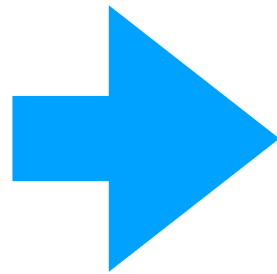
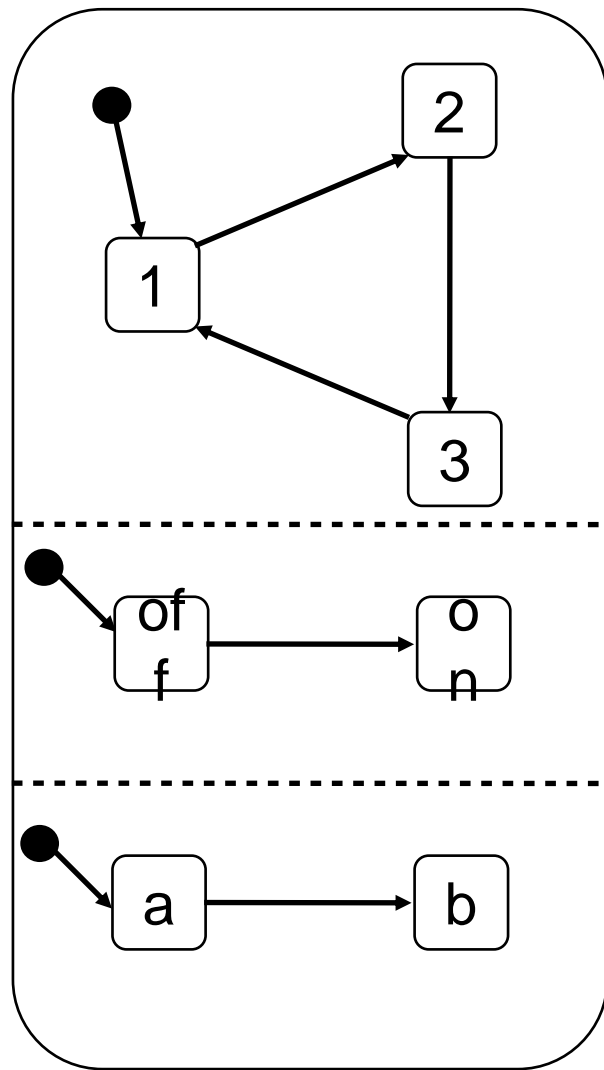


Combined Automaton

states?

transitions?

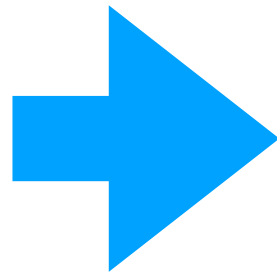
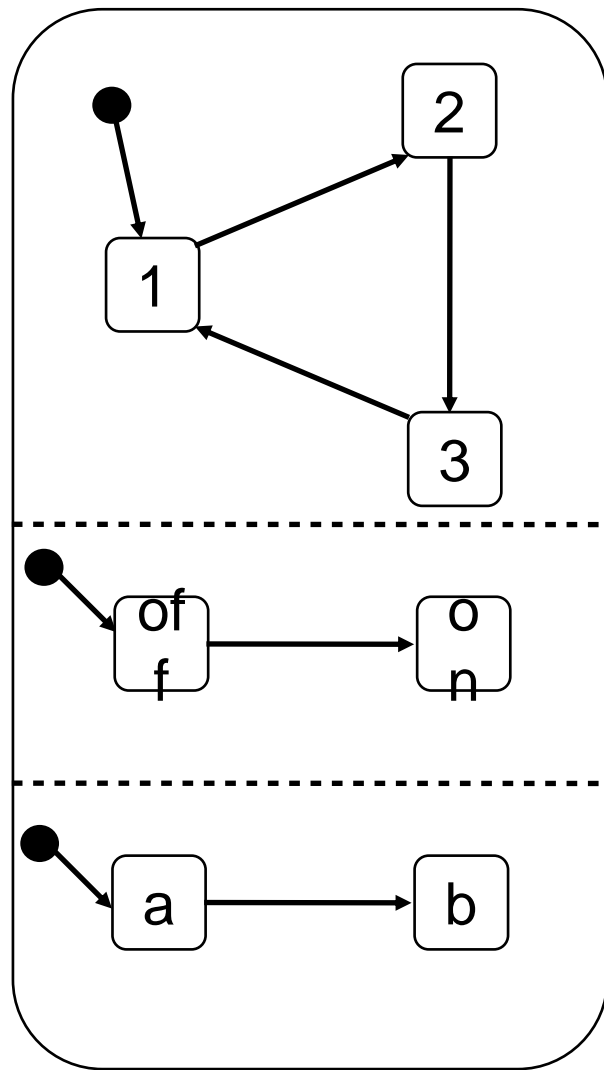




Combined Automaton

states?

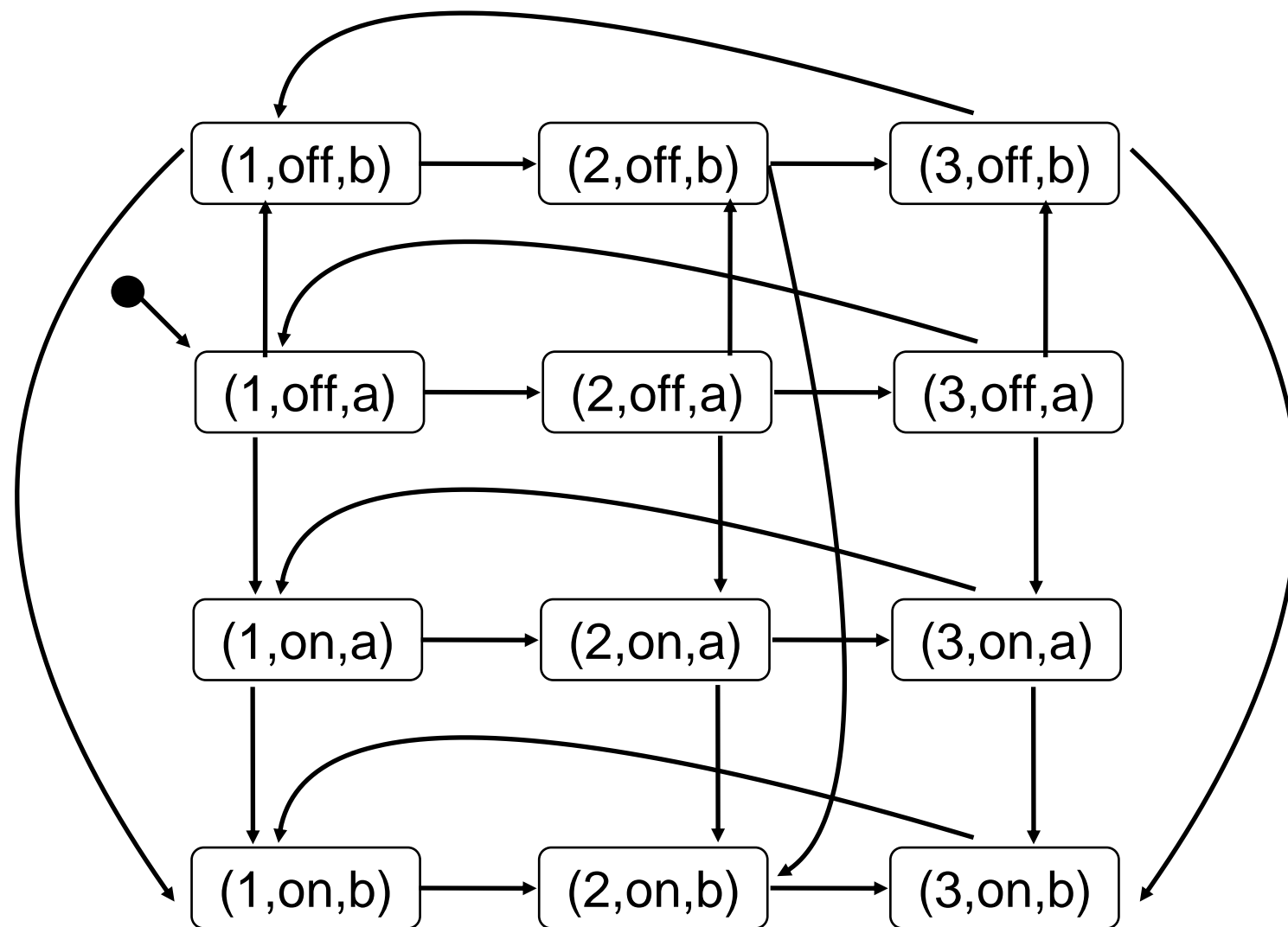
transitions?



Combined Automaton

states?

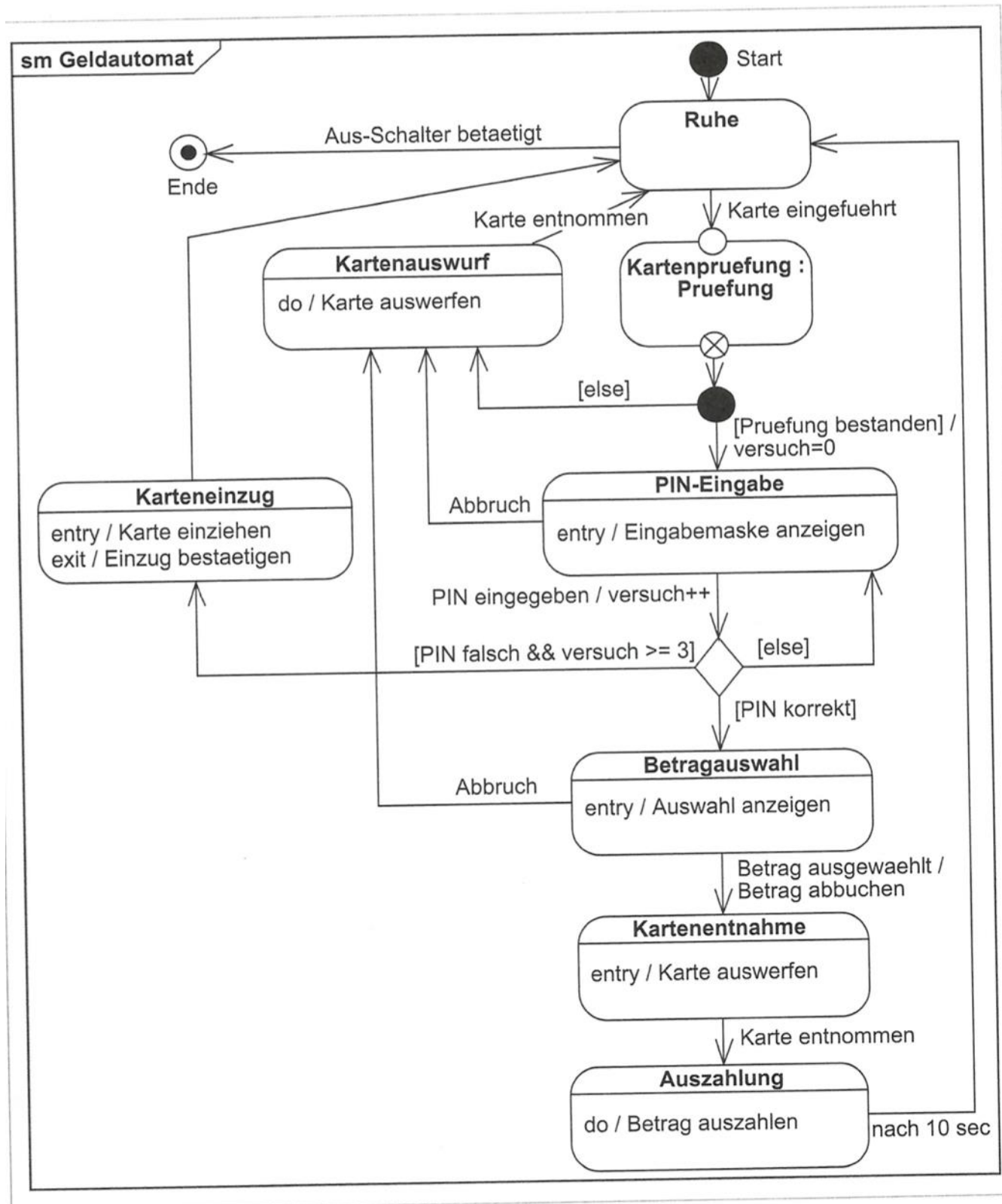
transitions?



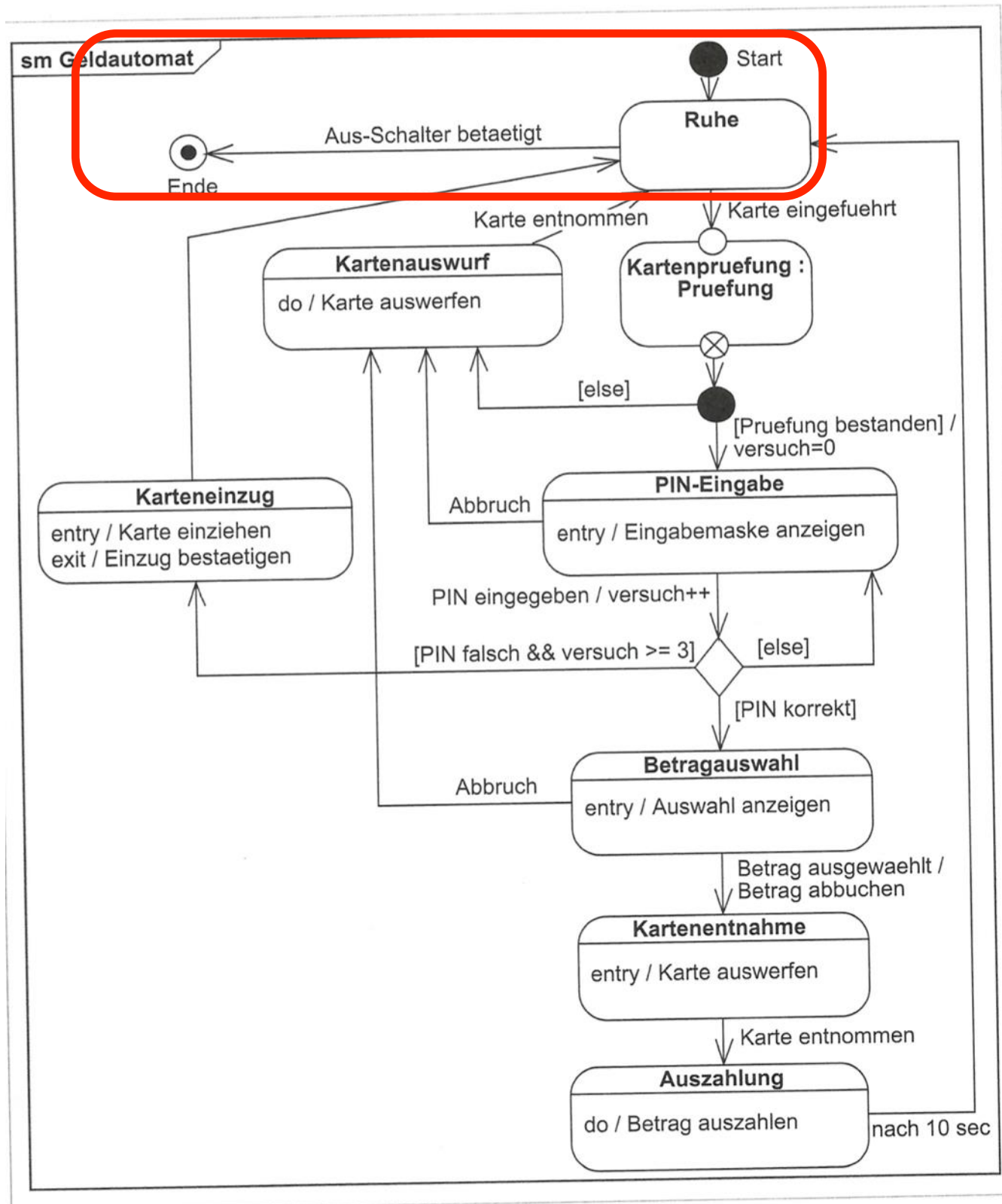
How to Read a Statechart

How to Read a Statechart

Make a narrative that explains the behaviour of this Statechart.



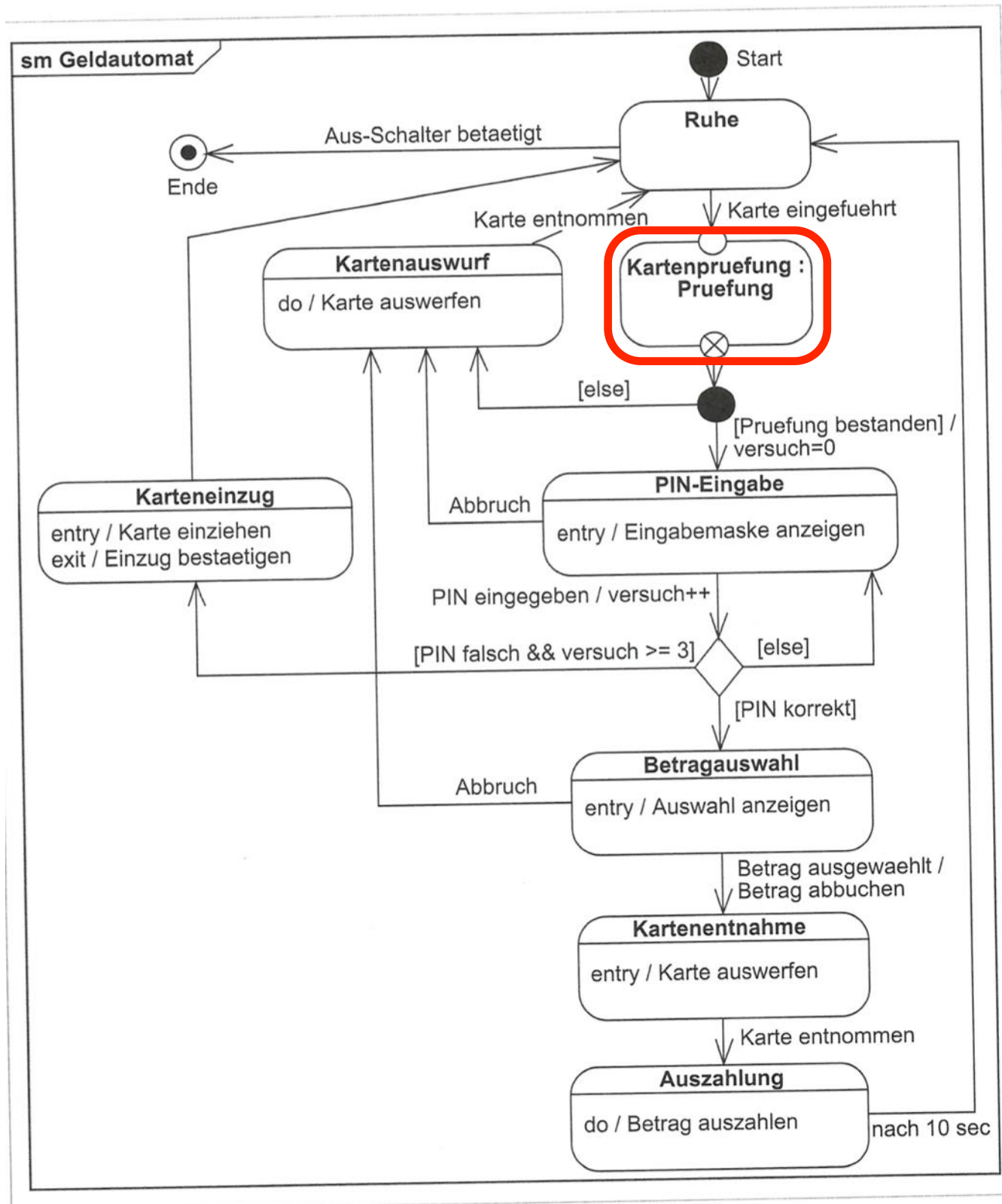
How to Read a Statechart



Make a narrative that explains the behaviour of this Statechart.

E.g.
Der Geldautomat kann per Schalter ausgeschaltet werden.

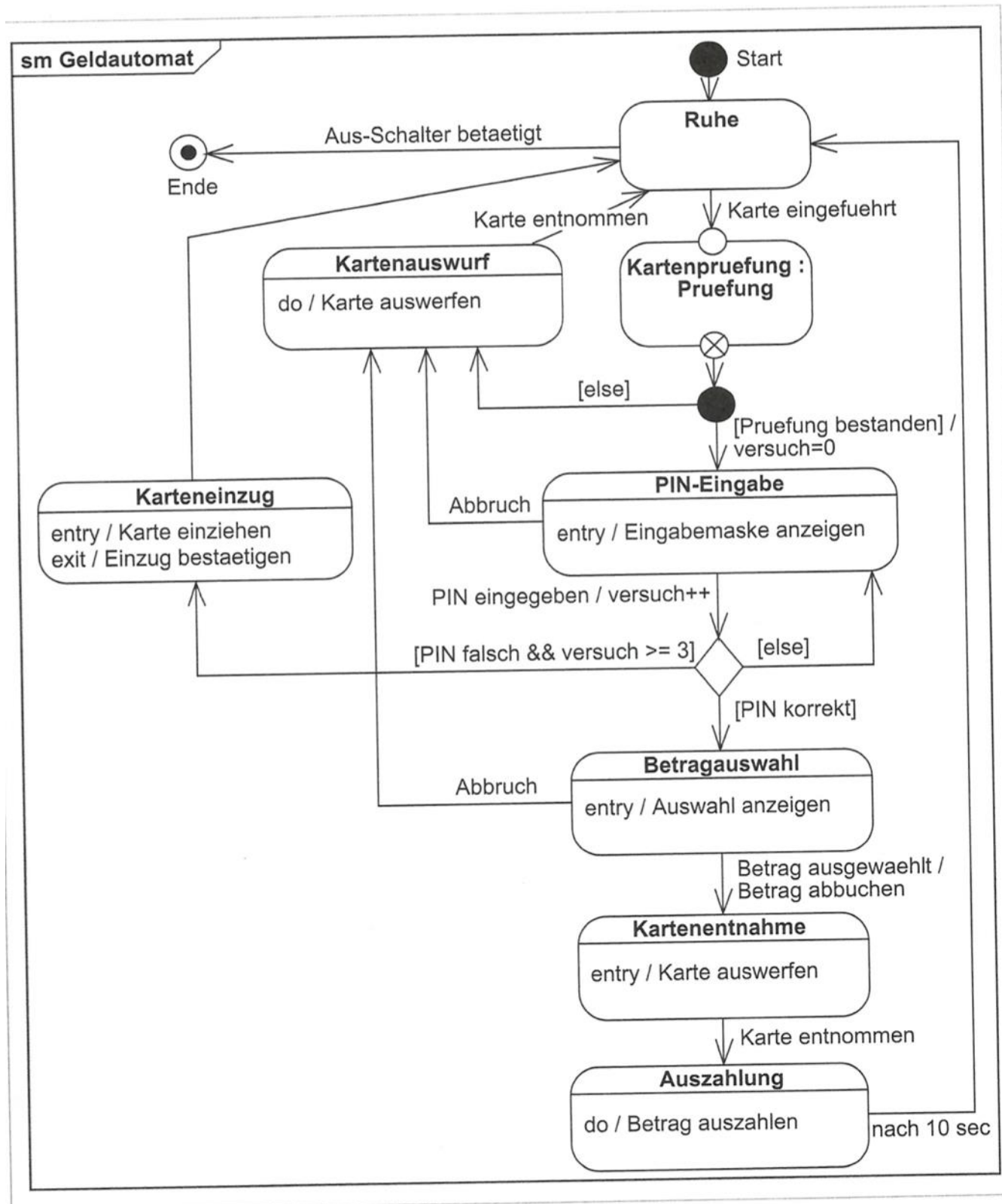
How to Read a Statechart



Make a narrative that explains the behaviour of this Statechart.

E.g.
Der Geldautomat kann per Schalter ausgeschaltet werden.
Wenn eine Karte eingeführt wird, wird diese zunächst geprüft.

How to Read a Statechart



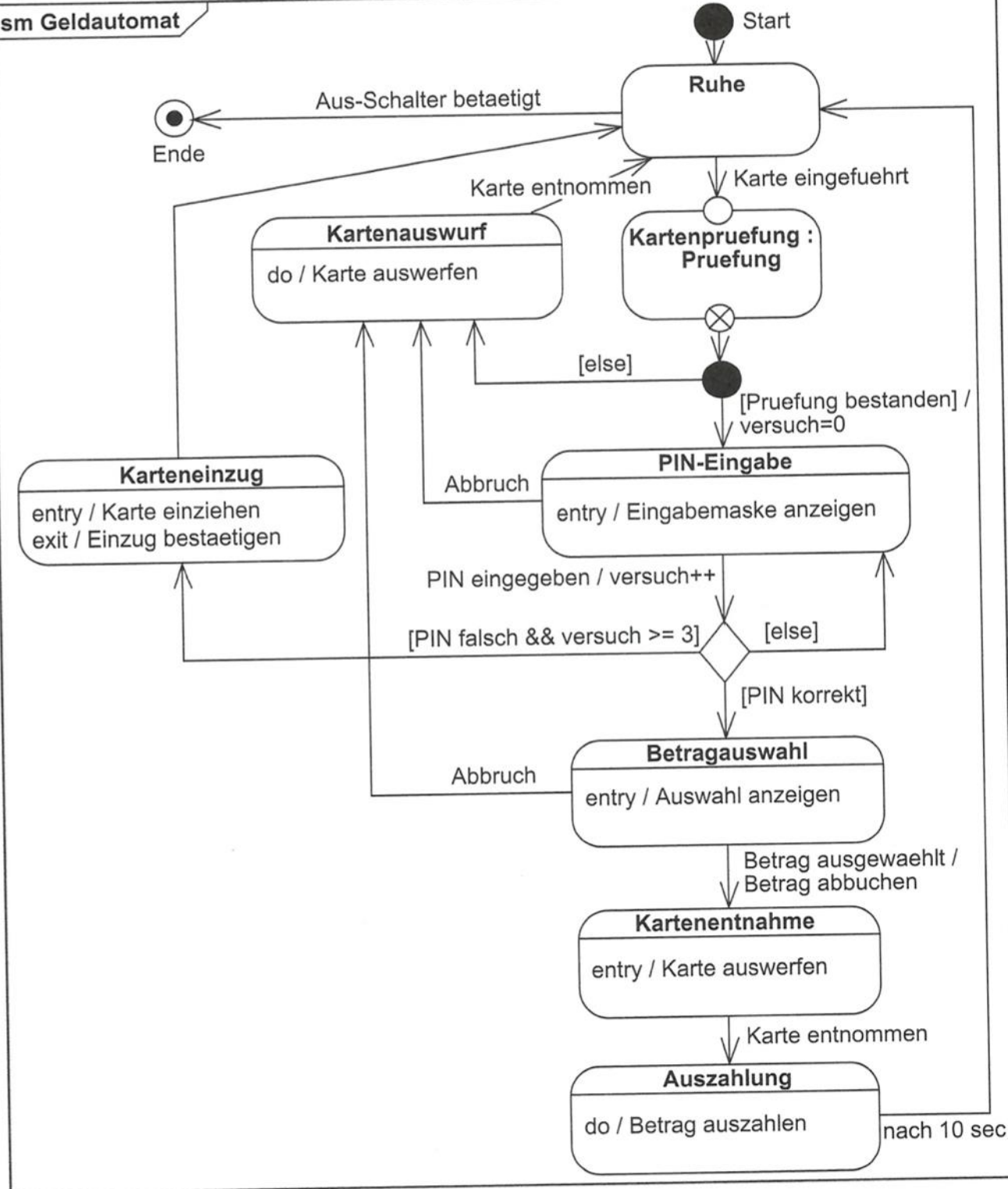
Make a narrative that explains the behaviour of this Statechart.

E.g.

Der Geldautomat kann per Schalter ausge-
schaltet werden.

Wenn eine Karte eingeführt wird, wird diese zunächst geprüft. Ist die Prüfung negativ, wird die Karte ausgeworfen, so dass der Benutzer sie wieder entnehmen kann.

sm Geldautomat ● Sta



Make a narrative that explains the behaviour of this Statechart.

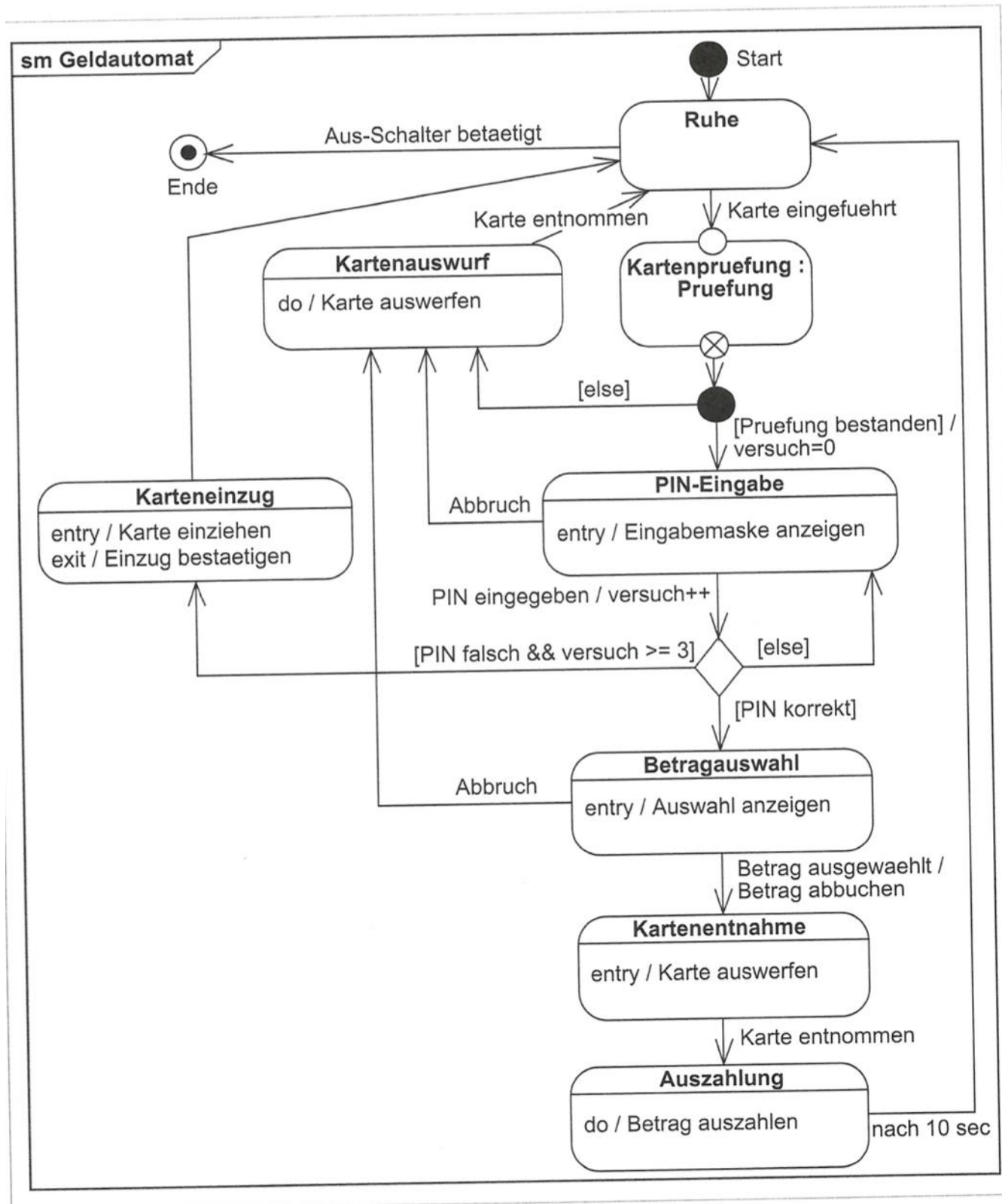
E.g.

Der Geldautomat kann per Schalter ausgeschaltet werden.

Wenn eine Karte eingeführt wird, wird diese zunächst geprüft. Ist die Prüfung negativ, wird die Karte ausgeworfen, so dass der Benutzer sie wieder entnehmen kann.

Der Benutzer wird aufgefordert die PIN einzugeben.

How to Read a Statechart



Make a narrative that explains the behaviour of this Statechart.

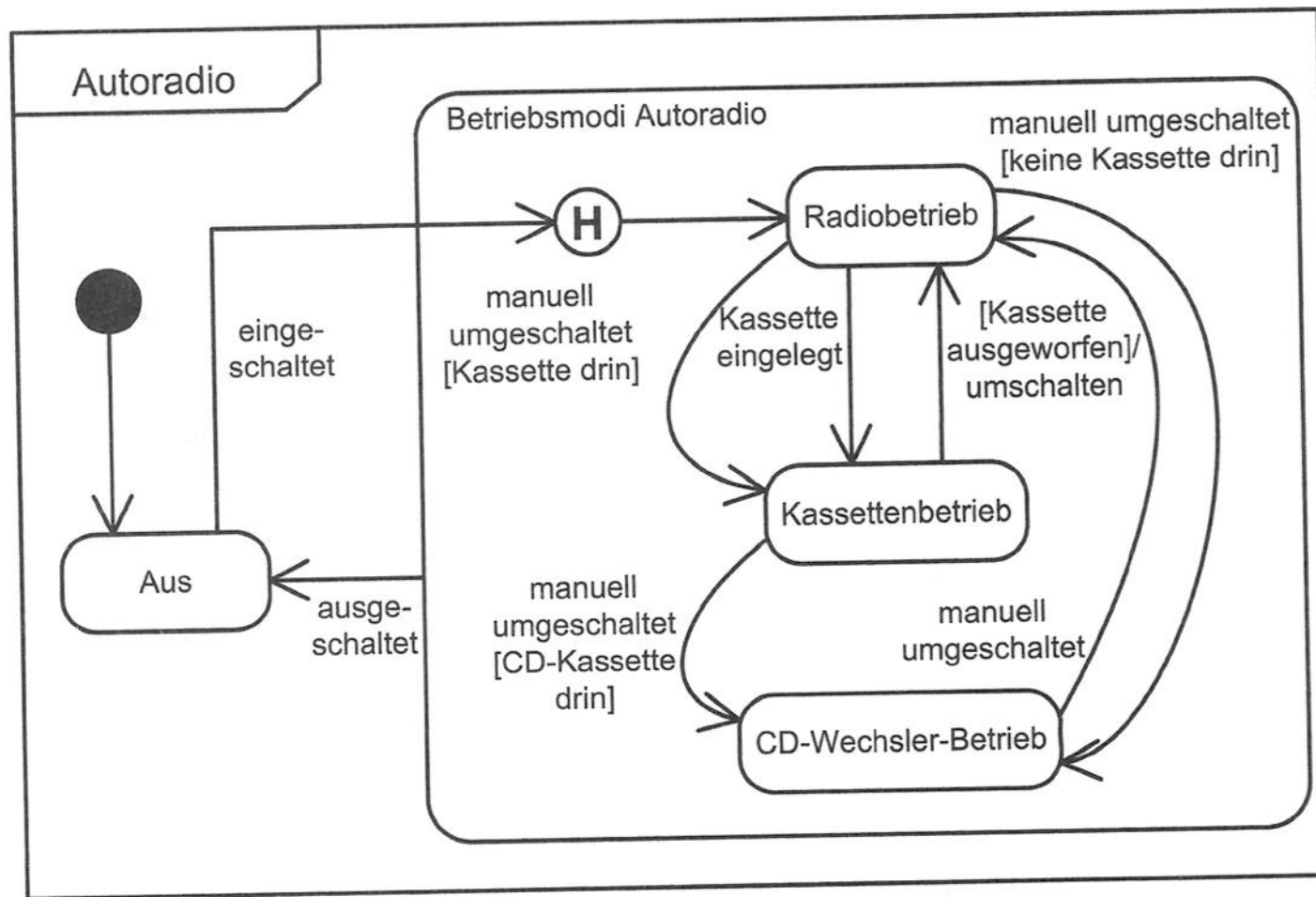
E.g.

Der Geldautomat kann per Schalter ausgeschaltet werden.

Wenn eine Karte eingeführt wird, wird diese zunächst geprüft. Ist die Prüfung negativ, wird die Karte ausgeworfen, so dass der Benutzer sie wieder entnehmen kann.

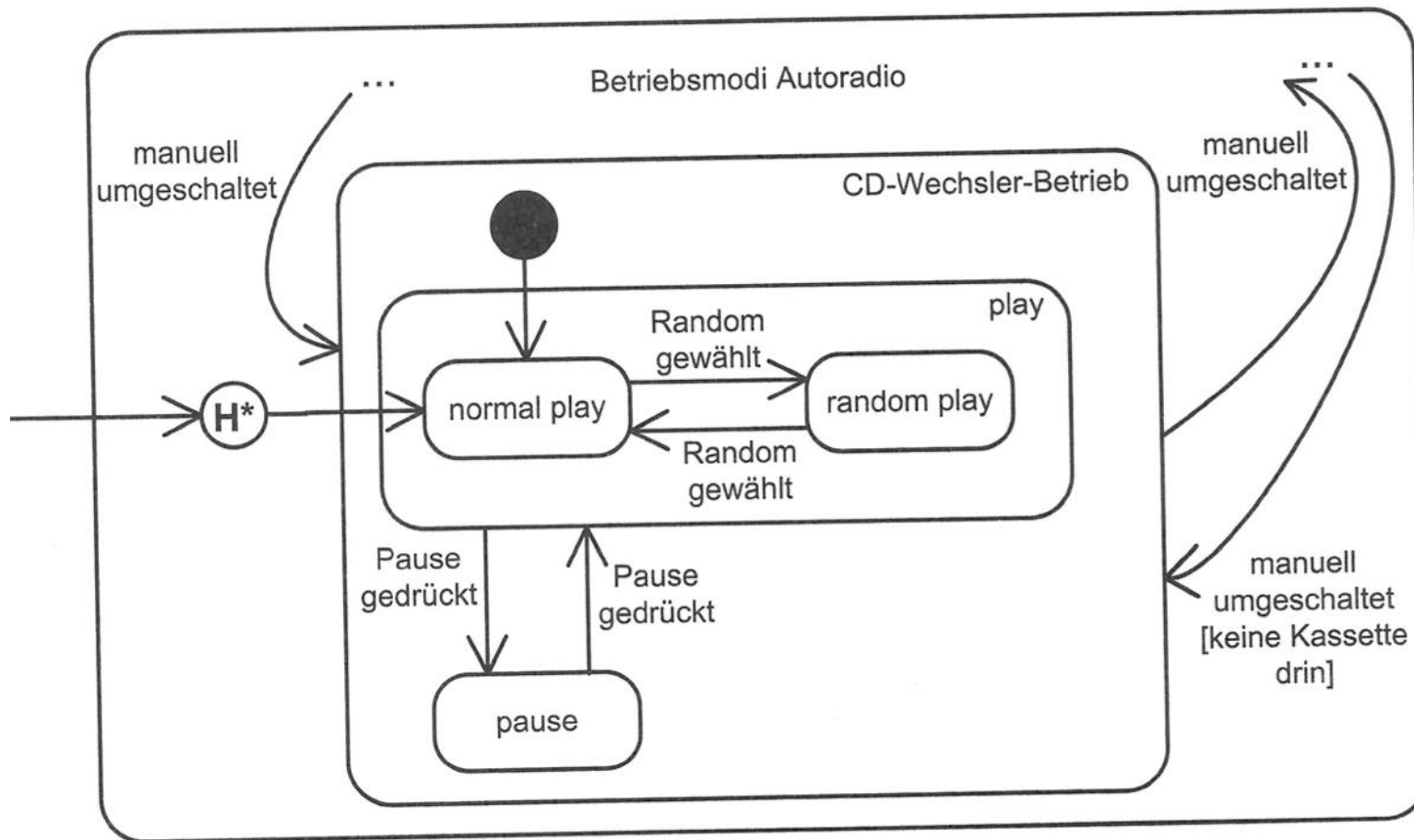
Der Benutzer wird aufgefordert die PIN einzugeben. Der Benutzer hat drei Versuche, um die richtige PIN einzugeben. Wenn dreimal die falsche PIN eingegeben wurde, wird die Karte eingezogen.

How to Read a Statechart



Make a narrative that explains the behaviour of this Statechart.

How to Read a Statechart



Make a narrative that explains the behaviour of this Statechart.

Find the Mistakes and Explain them

[illegible]