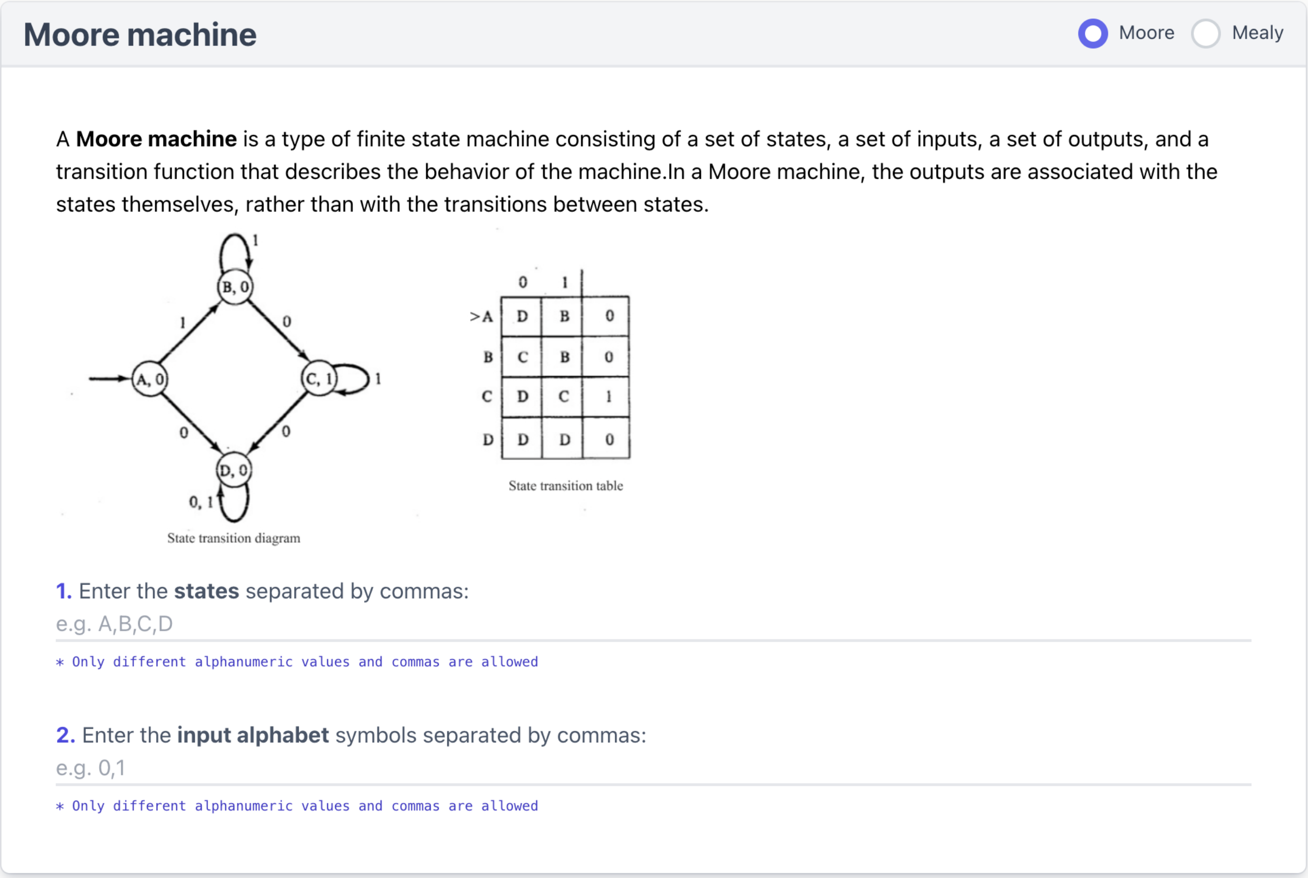
Minimal Connected Automaton Generator
User's Manual

Created by: Nicolás Penagos

As an initial state, the user will find an empty template to build the machine. Note that the type of the machine can be changed (between Moore and Mealy) using the toggle in the upper right corner. The user will also see a short definition of the selected machine, as well as a corresponding example in both state transition and diagram representation.



Moore machine

● Moore ○ Mealy

A **Moore machine** is a type of finite state machine consisting of a set of states, a set of inputs, a set of outputs, and a transition function that describes the behavior of the machine.In a Moore machine, the outputs are associated with the states themselves, rather than with the transitions between states.

State transition diagram

	0	1	
>A	D	B	0
B	C	B	0
C	D	C	1
D	D	D	0

State transition table

1. Enter the **states** separated by commas:

e.g. A,B,C,D

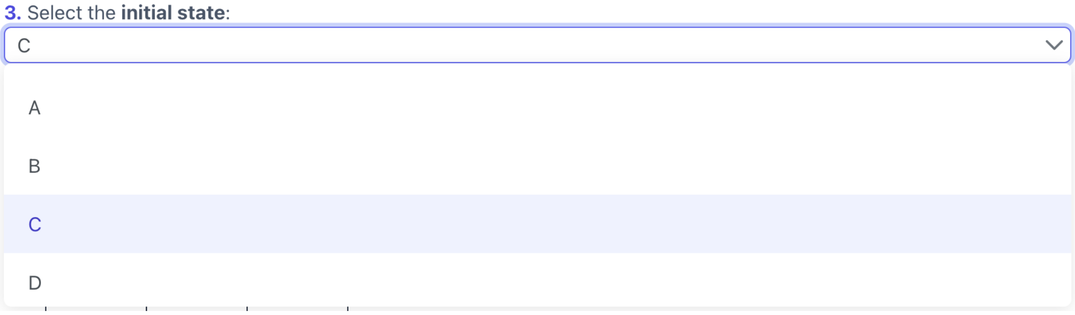
* Only different alphanumeric values and commas are allowed

2. Enter the **input alphabet** symbols separated by commas:

e.g. 0,1

* Only different alphanumeric values and commas are allowed

The user can input as many **states (1)** as they want, but only alphanumeric characters are allowed. Additionally, the software is coded to prevent states with the same name from being repeated. Therefore, the user should keep in mind that when entering a new state, any string that is a prefix of a previously entered state's name will not be accepted, the same applies when entering the **input alphabet (2)**. The **initial state (3)** will be selected from a drop-down list made up of the previously inserted states.

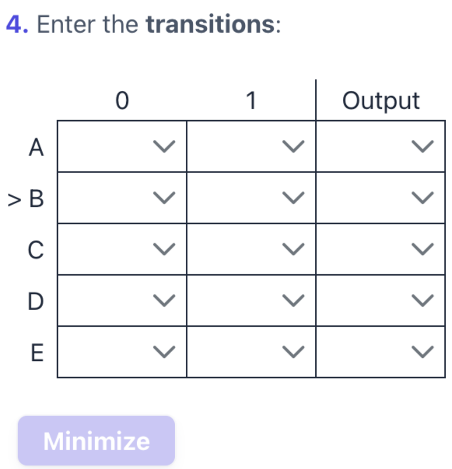


3. Select the **initial state**:

C

Once the user has inserted the states, input alphabet, and initial state, one the following table will be drawn depending on the type of machine that has been selected

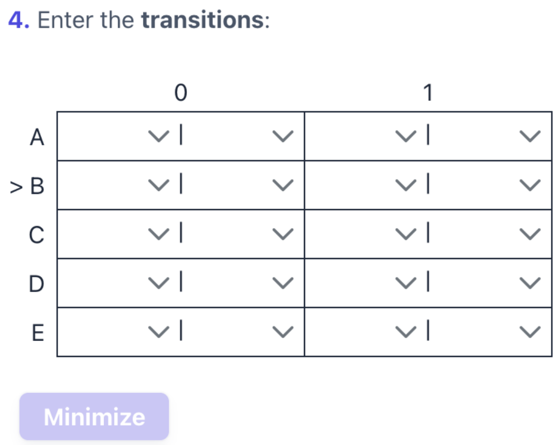
For Moore Machine:



4. Enter the **transitions**:

	0	1	Output
A			
> B			
C			
D			
E			

For Mealy Machine:



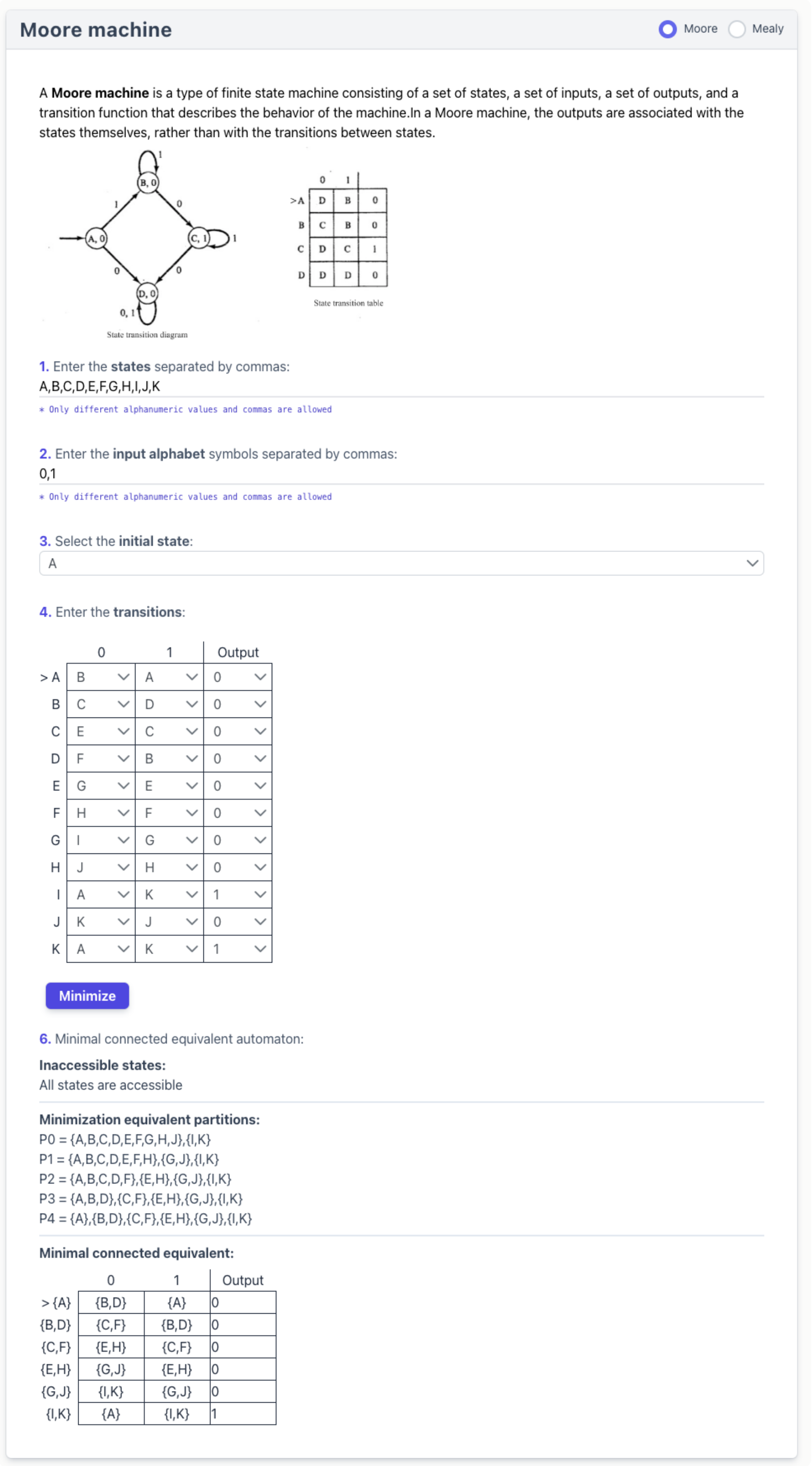
4. Enter the **transitions**:

	0	1
A		
> B		
C		
D		
E		

In both cases, the rows represent the machine states, and the columns the input alphabet. According to the definition, for the case of the **Moore machine**, each cell of the table on the left allows entering the corresponding transition state for the input entered. In the last column of the table, it is possible to select the output alphabet (in this case 0 and 1) of the state. For the case of the **Mealy machine**, the table on the right allows entering each transition state including the output associated with that transition.

It is worth mentioning that only when all cells are filled out, the minimize button will be enabled. The output of the program will be the transition table representation of the new **minimal connected machine**, the **eliminated non-connected states** (if any), and the **equivalence partitions** obtained during the minimization process.

Finally, two examples (one for Mealy and one for Moore) that illustrate the use of this program will be attached in the next pages.



Moore machine

● Moore ○ Mealy

A **Moore machine** is a type of finite state machine consisting of a set of states, a set of inputs, a set of outputs, and a transition function that describes the behavior of the machine.In a Moore machine, the outputs are associated with the states themselves, rather than with the transitions between states.

State transition diagram

	0	1	
>A	D	B	0
B	C	B	0
C	D	C	1
D	D	D	0

State transition table

1. Enter the **states** separated by commas:

A,B,C,D,E,F,G,H,I,J,K

`* Only different alphanumeric values and commas are allowed`

2. Enter the **input alphabet** symbols separated by commas:

0,1

`* Only different alphanumeric values and commas are allowed`

3. Select the **initial state**:

A

4. Enter the **transitions**:

	0	1	Output
> A	B	A	0
B	C	D	0
C	E	C	0
D	F	B	0
E	G	E	0
F	H	F	0
G	I	G	0
H	J	H	0
I	A	K	1
J	K	J	0
K	A	K	1

6. Minimal connected equivalent automaton:

Inaccessible states:

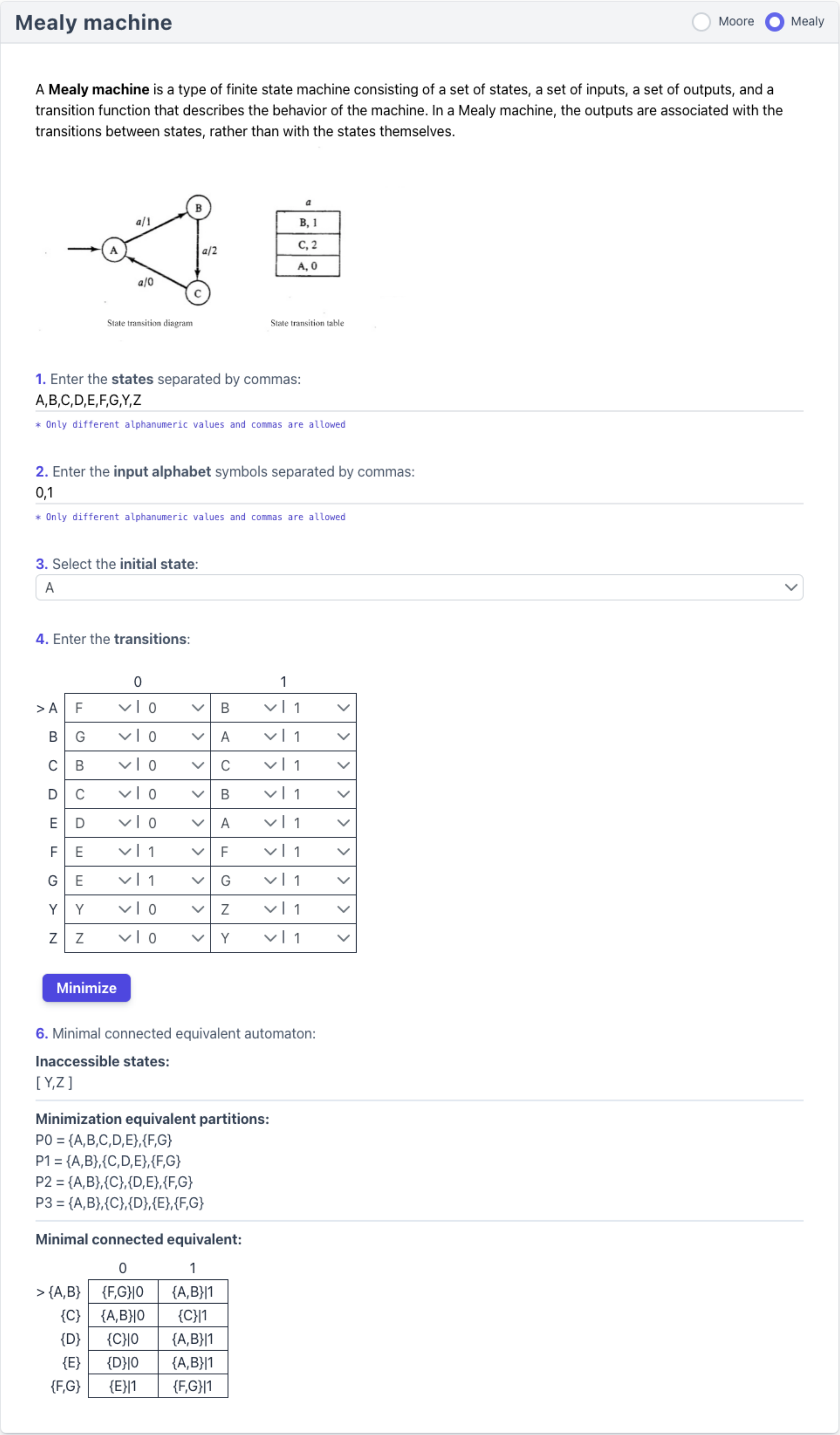
All states are accessible

Minimization equivalent partitions:

P0 = {A,B,C,D,E,F,G,H,J},{I,K}
P1 = {A,B,C,D,E,F,H},{G,J},{I,K}
P2 = {A,B,C,D,F},{E,H},{G,J},{I,K}
P3 = {A,B,D},{C,F},{E,H},{G,J},{I,K}
P4 = {A},{B,D},{C,F},{E,H},{G,J},{I,K}

Minimal connected equivalent:

	0	1	Output
> {A}	{B,D}	{A}	0
{B,D}	{C,F}	{B,D}	0
{C,F}	{E,H}	{C,F}	0
{E,H}	{G,J}	{E,H}	0
{G,J}	{I,K}	{G,J}	0
{I,K}	{A}	{I,K}	1



Mealy machine

○ Moore ● Mealy

A **Mealy machine** is a type of finite state machine consisting of a set of states, a set of inputs, a set of outputs, and a transition function that describes the behavior of the machine. In a Mealy machine, the outputs are associated with the transitions between states, rather than with the states themselves.

State transition diagram

State transition table

1. Enter the **states** separated by commas:

A,B,C,D,E,F,G,Y,Z

* Only different alphanumeric values and commas are allowed

2. Enter the **input alphabet** symbols separated by commas:

0,1

* Only different alphanumeric values and commas are allowed

3. Select the **initial state**:

A

4. Enter the **transitions**:

	0		1	
> A	F	0	B	1
B	G	0	A	1
C	B	0	C	1
D	C	0	B	1
E	D	0	A	1
F	E	1	F	1
G	E	1	G	1
Y	Y	0	Z	1
Z	Z	0	Y	1

Minimize

6. Minimal connected equivalent automaton:

Inaccessible states:

[Y,Z]

Minimization equivalent partitions:

P0 = {A,B,C,D,E},{F,G}
P1 = {A,B},{C,D,E},{F,G}
P2 = {A,B},{C},{D,E},{F,G}
P3 = {A,B},{C},{D},{E},{F,G}

Minimal connected equivalent:

	0	1
> {A,B}	{F,G}|0	{A,B}|1
{C}	{A,B}|0	{C}|1
{D}	{C}|0	{A,B}|1
{E}	{D}|0	{A,B}|1
{F,G}	{E}|1	{F,G}|1