

01584: Grundpraktikum Programmierung

durchgeführt vom
Lehrgebiet Softwaretechnik und Theorie der Programmierung
im Wintersemester 2020/21

Dokumentation der Beispiele

(Stand 10.09.2020)



Abbildungsverzeichnis

2.1	110-B1-N01-A00-EineStelleZweiMarken.pnml	8
2.2	111-B1-N01-A00-EineStelleEineTransition.pnml	8
2.3	112-B1-N02-A01-EineStelleEineMarkeEineTransition.pnml	8
2.4	113-B1-N03-A02-EineStelleZweiMarkenEineTransition.pnml	9
2.5	114-B1-N01-A01-StelleInVorUndNachbereich.pnml	9
2.6	115-B1-N02-A02-AlternativeTransitionen.pnml	10
2.7	116-B1-N03-A02-AlternativeStellen.pnml	10
2.8	117-B1-N02-A02-AlternierendeStellen.pnml	11
2.9	118-B1-N03-A04-AlternierendeStellenZweiMarken.pnml	12
2.10	119-B1-N02-A01-ZweiStellenTransitionStelle.pnml	13
2.11	130-B1-N05-A05-Gabelung.pnml	14
2.12	131-B1-N05-A04-BegrenzteSchleife.pnml	15
2.13	132-B1-N10-A12-StelleTransStelleTransStelle.pnml	16
2.14	150-B1-N04-A04-Example01.pnml	17
2.15	151-B1-N05-A07-Example02.pnml	18
2.16	152-B1-N06-A06-Example03v1.pnml	19
2.17	170-B1-N03-A04-MutualExclusion.pnml	20
2.18	171-B1-N05-A05-Kreis3.pnml	21
2.19	172-B1-N15-A16-Kreis-gross.pnml	22
2.20	173-B1-N04-A03-m-auf-anderem-Pfad1.pnml	23
2.21	174-B1-N09-A09-m-auf-anderem-Pfad2.pnml	24
2.22	175-B1-N12-A20-Mailbox-2-Bounded.pnml	25
2.23	176-B1-N16-A28-Mailbox-3-Bounded.pnml	26
2.24	177-B1-N20-A36-Mailbox-4-Bounded.pnml	27
2.25	178-B1-N06-A10-Vending-Machine.pnml	28
2.26	179-B1-N06-A06-Kreis-mit-zwei-Ausgaengen.pnml	29
2.27	180-B1-N15-A25-Kreis-mit-zwei-Eingaengen.pnml	30
2.28	181-B1-N08-A07-Kreis-ohne-Kreis.pnml	31
2.29	210-B0-P01-Qxx-EineTransitionEineStelle.pnml	32
2.30	211-B0-P01-Qxx-TransStelleTrans.pnml	32
2.31	230-B0-P02-Qxx-Counter.pnml	33
2.32	231-B0-P02-Qxx-CounterMitAbfluss.pnml	34
2.33	250-B0-P02-Q03-Example3v2.pnml	35
2.34	251-B0-P01-Qxx-Netz1.pnml	36
2.35	252-B0-P02-Qxx-Netz2.pnml	37
2.36	253-B0-P02-Q03-Netz3.pnml	38
2.37	254-B0-P01-Qxx-Netz4.pnml	39
2.38	255-B0-P01-Qxx-Netz5.pnml	40
2.39	256-B0-P03-Qxx-Netz6.pnml	41
2.40	257-B0-P01-Qxx-Netz7.pnml	42
2.41	258-B0-P02-Qxx-Netz8.pnml	43

2.42	270-B0-P02-Qxx-Keksautomat.pnml	44
2.43	271-B0-P05-Qxx-Kreis3-vor-Omega.pnml	45
2.44	272-B0-P04-Qxx-Kreis3-nach-Omega.pnml	46
2.45	273-B0-P05-Q06-Kreis-nach-Omega-gross.pnml	47
2.46	274-B0-P05-Qxx-Kreis4-vor-und-nach-Omega.pnml	48
2.47	275-B0-P02-Qxx-Mailbox-unbounded.pnml	49
2.48	276-B0-P04-Q10-mehrere-Pfade.pnml	50

1 Hinweise zu den Beispielen

In diesem Dokument sind alle Beispiel-Petrinetze aufgeführt, die wir Ihnen als PNML Dateien zur Verfügung stellen. Die Dateien befinden sich im ProPra-WS20-Basis Projekt im SVN-Repository (siehe dazu die Erläuterungen in der Aufgabenstellung). Innerhalb der ersten sechs Wochen des ProPra werden wir die ursprüngliche Menge an Beispielen noch um einige weitere Beispiele erweitern (und dieses Dokument entsprechend aktualisieren). Sobald neue Beispiele im SVN-Repository zur Verfügung stehen, werden wir Sie darüber in der Betreuungs-Newsgroup informieren. Bitte beachten Sie auch diesbezüglich den Hinweis in der Aufgabenstellung: Ihr Programm muss *alle* Beispiele, die von uns zur Verfügung gestellt werden, korrekt berechnen können.

Noch ein wichtiger Hinweis: Wenn Sie sich (z.B. in der Newsgroup) auf ein konkretes Beispiel beziehen, dann verwenden Sie zur Identifikation des Beispiels bitte immer die *dreistellige Nummer* des Beispiels (und *nicht* die Nummer der Abbildung).

1.1 Dateinamen der Beispiele

Die Dateinamen der PNML Dateien sind nach folgendem Schema aufgebaut:

nnn-**B1**-Nnn-Ann-<Name des Beispiels>.pnml
oder nnn-**B0**-Pnn-Qnn-<Name des Beispiels>.pnml

Die einzelnen Komponenten eines Dateinamens haben folgende Bedeutung:

- nnn eindeutige Nummer des Beispiels
- B1** das Petrinetz ist beschränkt
- Nnn** Anzahl der Knoten (*nodes*) des Erreichbarkeitsgraphen
- Ann** Anzahl der Kanten (*arcs*) des Erreichbarkeitsgraphen
- B0** das Petrinetz ist unbeschränkt
- Pnn** Länge eines Pfades im partiellen Erreichbarkeitsgraphen,
 der das Petrinetz als unbeschränkt identifiziert
- Qnn** falls vorhanden, Länge eines weiteren Pfades (ansonsten **Qxx**)

Dem Dateinamen können Sie also entnehmen, ob das Petrinetz beschränkt ist oder nicht. Für ein beschränktes Petrinetz wird zudem die Anzahl der Knoten und Kanten des endlichen Erreichbarkeitsgraphen mit angegeben. Für ein unbeschränktes Petrinetz wird stattdessen die Länge eines Pfades (vom Anfangsknoten über Knoten m zu m') im partiellen Erreichbarkeitsgraphen mit angegeben. Falls es noch einen alternativen Pfad (mit anderer Länge) gibt, wird dessen Länge ebenfalls mit angegeben. In jedem der (aktuell) vorhandenen Beispiele gibt es höchstens zwei Pfade unterschiedlicher Länge.

Die „sprechenden“ Dateinamen sollen Ihnen lediglich zur einfacheren Orientierung beim Umgang mit den Beispieldateien dienen. Innerhalb Ihres Programms müssen Sie natürlich alle Berechnungen unabhängig vom Dateinamen durchführen. Um eine korrekte

Funktionsweise des Programms sicherzustellen, werden wir Ihr Programm mit den hier aufgeführten, aber zuvor umbenannten Beispielen testen.

1.2 Erreichbarkeitsgraphen und Überdeckungsgraphen

Im folgenden Kapitel wird zu jeder PNML Datei das darin definierte Petrinetz (mit Anfangsmarkierung) dargestellt. Falls das Petrinetz beschränkt ist, wird darunter der dazugehörige endliche Erreichbarkeitsgraph abgebildet. Falls das Petrinetz hingegen unbeschränkt ist, wird darunter ein sogenannter *Überdeckungsgraph* dargestellt. Ein Überdeckungsgraph ermöglicht quasi – sehr grob formuliert – die kompakte Darstellung eines unendlichen Erreichbarkeitsgraphen.

Wir stellen Ihnen die Abbildung eines Überdeckungsgraphen zur Verfügung (quasi im Sinne einer „Lösung“), da Sie daran leicht ablesen können, ob das Ergebnis, welches Ihre Implementierung des Beschränktheits-Algorithmus zurückliefert, tatsächlich ein zulässiges Ergebnis darstellt (d.h. ob die ermittelten Markierungen m und m' sowie der Pfad das Petrinetz tatsächlich als unbeschränkt identifizieren können).

Die genaue Definition eines Überdeckungsgraphen ist hier nicht relevant, Sie müssen im Kontext der Aufgabenstellung nur Folgendes wissen: Der prinzipielle Aufbau eines Überdeckungsgraphen entspricht zunächst dem des Erreichbarkeitsgraphen, mit einem entscheidenden Unterschied: Wenn beim Aufbau ein Knoten m' entdeckt wird, dann wird im Überdeckungsgraph ein ω (anstatt einer konkreten Markenanzahl) in die Beschriftung des Knotens eingetragen (und zwar für diejenigen Stellen, die unbeschränkt sind).

Konkret bedeutet das für Sie, dass Sie an den zur Verfügung gestellten Überdeckungsgraphen sehr einfach alle möglichen Markierungen m und m' sowie den dazugehörigen Pfad von der Anfangsmarkierung ablesen können, die das jeweilige Petrinetz als unbeschränkt identifizieren. Sie müssen dazu nur wie folgt vorgehen: Betrachten Sie im Überdeckungsgraphen (jeweils ausgehend von der Anfangsmarkierung) alle Pfade, die jeweils bis zum *ersten* Auftreten eines ω -Knotens führen. Ein solcher ω -Knoten entspricht dann dem m' Knoten (auf dem jeweiligen Pfad), der das Petrinetz als unbeschränkt identifiziert, wobei ω in der konkreten Markierung m' dann in den meisten Beispielen den Wert 1 hat. Alle weiteren Kanten und Knoten (auch ω -Knoten) im Überdeckungsgraphen, die quasi erst „hinter“ einem m' Knoten liegen, haben für Sie folglich keine Relevanz, da der Beschränktheits-Algorithmus nie so weit laufen würde. Betrachten Sie beispielsweise den Überdeckungsgraphen aus Abb. 2.33; in Abb. 1.1 sind die beiden Pfade (und Markierungen m und m') in diesem Überdeckungsgraphen dargestellt, die jeweils bis zum Erreichen eines ersten ω -Knotens führen. Die in der Abbildung ausgegrauten Kanten und Knoten des Überdeckungsgraphen sind hingegen für keinen dieser Pfade mehr relevant, da sie erst nach Erreichen eines m' Knotens erreicht werden könnten.

Einige der Beispiele haben zwar einen relativ großen Überdeckungsgraphen, allerdings gibt es i.Allg. auch dort nur einige wenige, kurze Pfade bis zum Erreichen eines ersten ω -Knotens (z.B. gibt es in Abb. 2.42 lediglich einen Pfad und der hat nur Länge 2).

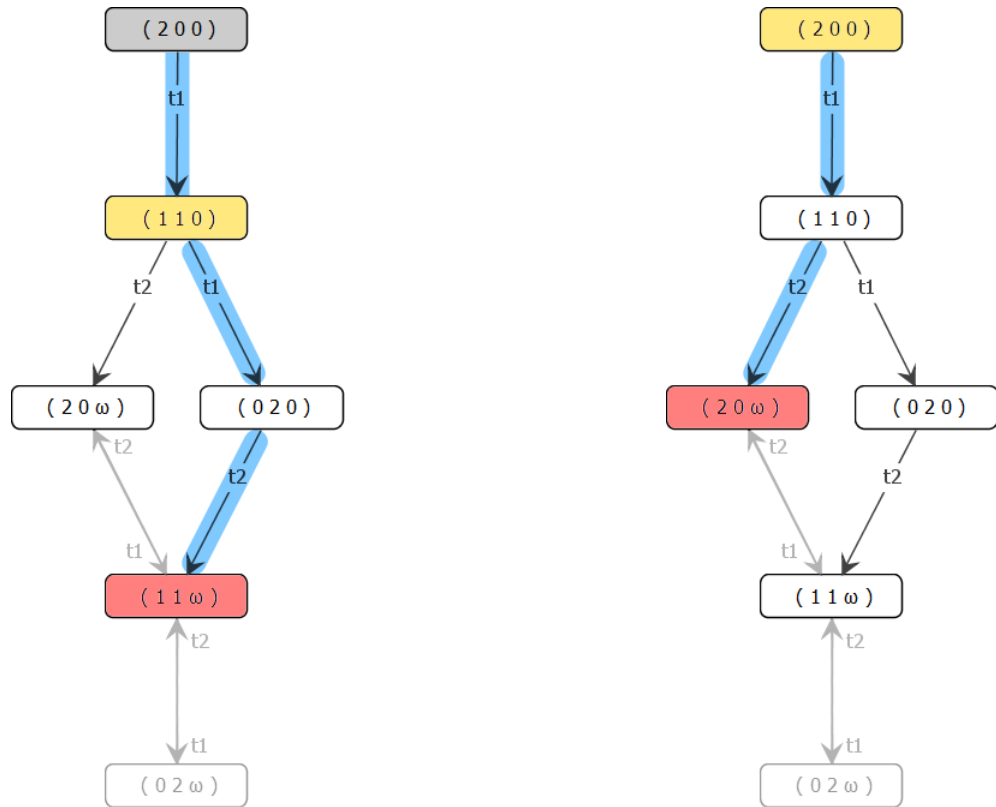


Abbildung 1.1: Die beiden möglichen Pfade im Überdeckungsgraphen aus Abb. 2.33, die jeweils bis zum Erreichen eines ersten ω -Knotens führen. Kanten und Knoten, die auf keinem dieser Pfade liegen, sind ausgegraut dargestellt.

1.3 Darstellung der Beispiele

Die hier verwendete Darstellung der Graphen weicht in den folgenden Punkten leicht von der Darstellung ab, die Sie in Ihrem Programm mittels der GraphStream Bibliothek realisieren sollen:

- Wenn eine Kante mit z.B. $t_1 \vee t_2$ beschriftet ist (wie z.B. in Abb. 2.6), dann steht dies verkürzend für *zwei Kanten* t_1 und t_2 , die beide den selben Anfangs- und Endknoten haben (also quasi zwei parallele Kanten in einem Multigraphen).
- Eine bidirektionale Kante steht abkürzend für *zwei Kanten*, die entgegengesetzt gerichtet sind. Zum Beispiel steht die bidirektionale Kante im Petrinetz in Abb. 2.5 stellvertretend für die zwei Kanten (p_1, t_1) und (t_1, p_1) .
- Die Beschriftung eines Knotens bzw. einer Markierung in einem Erreichbarkeits- oder Überdeckungsgraphen entspricht im Prinzip der in der Aufgabenstellung geforderten Tupel-Form (lediglich das „|“ Zeichen zwischen den Zahlen fehlt).
- In Petrinetzen, in denen Stellen und Transitionen „realistische“ Bezeichnungen ha-

ben (wie z.B. in Abb. 2.17 und 2.42) sind die IDs der Stellen zusätzlich (in blau) dargestellt, da sich aus diesen die Reihenfolge der Stellen in der Tupel-Form ergibt.

1.4 Korrekte Ausgabe der Stapelverarbeitung für alle Beispieldateien

Eine korrekte Ausgabe der Stapelverarbeitung bei Anwendung auf alle Beispieldateien sollte wie im Folgenden dargestellt aussehen.

Beachten Sie jedoch den Hinweis in der Aufgabenstellung, dass es bei manchen Beispielen mit unbeschränkten Petrinetzen mehrere mögliche Pfade geben kann, auf denen dann entsprechende Knoten m und m' liegen. Daher kann es möglich sein, dass sich die Ausgabe Ihres Programms für solche Beispiele von der hier dargestellten Ausgabe unterscheidet.

So könnte für das Beispiel 250-B0-P02-Q03-Example3v2.pnml, in dem es Pfade der Länge zwei und drei gibt, in der Ausgabe anstatt $3:(t_1, t_1, t_2); (1|1|0), (1|1|1)$ auch genauso gut $2:(t_1, t_2); (2|0|0), (2|0|1)$ stehen.

Dateiname	beschränkt	Knoten / Kanten bzw. Pfadlänge:Pfad; m, m'
110-B1-N01-A00-EineStelleZweiMarken.pnml	ja	1 / 0
111-B1-N01-A00-EineStelleEineTransition.pnml	ja	1 / 0
112-B1-N02-A01-EineStelleEineMarkeEineTransition.pnml	ja	2 / 1
113-B1-N03-A02-EineStelleZweiMarkenEineTransition.pnml	ja	3 / 2
114-B1-N01-A01-StelleInVorUndNachbereich.pnml	ja	1 / 1
115-B1-N02-A02-AlternativeTransitionen.pnml	ja	2 / 2
116-B1-N03-A02-AlternativeStellen.pnml	ja	3 / 2
117-B1-N02-A02-AlternierendeStellen.pnml	ja	2 / 2
118-B1-N03-A04-AlternierendeStellenZweiMarken.pnml	ja	3 / 4
119-B1-N02-A01-ZweiStellenTransitionStelle.pnml	ja	2 / 1
130-B1-N05-A05-Gabelung.pnml	ja	5 / 5
131-B1-N05-A04-BegrenzteSchleife.pnml	ja	5 / 4
132-B1-N10-A12-StelleTransStelleTransStelle.pnml	ja	10 / 12
150-B1-N04-A04-Example01.pnml	ja	4 / 4
151-B1-N05-A07-Example02.pnml	ja	5 / 7
152-B1-N06-A06-Example03v1.pnml	ja	6 / 6
170-B1-N03-A04-MutualExclusion.pnml	ja	3 / 4
171-B1-N05-A05-Kreis3.pnml	ja	5 / 5
172-B1-N15-A16-Kreis-gross.pnml	ja	15 / 16
173-B1-N04-A03-m-auf-anderem-Pfad1.pnml	ja	4 / 3
174-B1-N09-A09-m-auf-anderem-Pfad2.pnml	ja	9 / 9
175-B1-N12-A20-Mailbox-2-Bounded.pnml	ja	12 / 20
176-B1-N16-A28-Mailbox-3-Bounded.pnml	ja	16 / 28
177-B1-N20-A36-Mailbox-4-Bounded.pnml	ja	20 / 36
178-B1-N06-A10-Vending-Machine.pnml	ja	6 / 10
179-B1-N06-A06-Kreis-mit-zwei-Ausgaengen.pnml	ja	6 / 6
180-B1-N15-A25-Kreis-mit-zwei-Eingaengen.pnml	ja	15 / 25
181-B1-N08-A07-Kreis-ohne-Kreis.pnml	ja	8 / 7
210-B0-P01-Qxx-EineTransitionEineStelle.pnml	nein	1: (t1); (0), (1)
211-B0-P01-Qxx-TransStelleTrans.pnml	nein	1: (t1); (0), (1)
230-B0-P02-Qxx-Counter.pnml	nein	2: (t1, t2); (1 0 0), (1 0 1)
231-B0-P02-Qxx-CounterMitAbfluss.pnml	nein	2: (t1, t2); (1 0 0), (1 0 1)
250-B0-P02-Q03-Example3v2.pnml	nein	3: (t1, t1, t2); (1 1 0), (1 1 1)
251-B0-P01-Qxx-Netz1.pnml	nein	1: (t1); (0 0), (1 0)
252-B0-P02-Qxx-Netz2.pnml	nein	2: (t4, t5); (1 0 0), (1 0 1)
253-B0-P02-Q03-Netz3.pnml	nein	2: (t1, t4); (0 1 0), (0 1 1)
254-B0-P01-Qxx-Netz4.pnml	nein	1: (t1); (1 0), (1 1)
255-B0-P01-Qxx-Netz5.pnml	nein	1: (t3); (1 0 0), (1 0 1)
256-B0-P03-Qxx-Netz6.pnml	nein	3: (t1, t3, t2); (1 0 0 0 0), (1 1 0 0 0)
257-B0-P01-Qxx-Netz7.pnml	nein	1: (t1); (1 0 0), (1 0 1)
258-B0-P02-Qxx-Netz8.pnml	nein	2: (t1, t2); (0 1), (1 1)
270-B0-P02-Qxx-Keksautomat.pnml	nein	2: (t1, t4); (0 1 0 0 5 0 0), (0 1 1 1 5 0 0)
271-B0-P05-Qxx-Kreis3-vor-Omega.pnml	nein	5: (t1, t2, t4, t7, t6); (0 0 0 1 0 0 0), (0 0 0 1 0 1 0)
272-B0-P04-Qxx-Kreis3-nach-Omega.pnml	nein	4: (t1, t2, t3, t5); (0 1 0 0 0), (0 1 0 1 0)
273-B0-P05-Q06-Kreis-nach-Omega-gross.pnml	nein	6: (t1, t2, t3, t5, t6, t7); (0 1 0 0 0 0 0), (0 1 0 0 0 1 0)
274-B0-P05-Qxx-Kreis4-vor-und-nach-Omega.pnml	nein	5: (t1, t2, t4, t6, t7); (0 1 0 0 0 0 0), (0 1 0 0 0 1 0)
275-B0-P02-Qxx-Mailbox-unbounded.pnml	nein	2: (t0, t1); (1 0 0 0 1), (1 0 1 0 1)
276-B0-P04-Q10-mehrere-Pfade.pnml	nein	10: (t1, t2, t3, t4, t9, t9, t2, t2, t3, t9); (0 1 1 0 0 0), (0 1 1 1 0 0)

2 Petrinetze mit dazugehörigen Erreichbarkeits- bzw. Überdeckungsgraphen

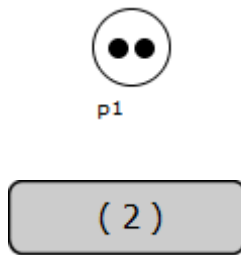


Abbildung 2.1: 110-B1-N01-A00-EineStelleZweiMarken.pnml

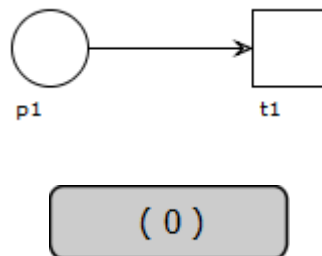


Abbildung 2.2: 111-B1-N01-A00-EineStelleEineTransition.pnml

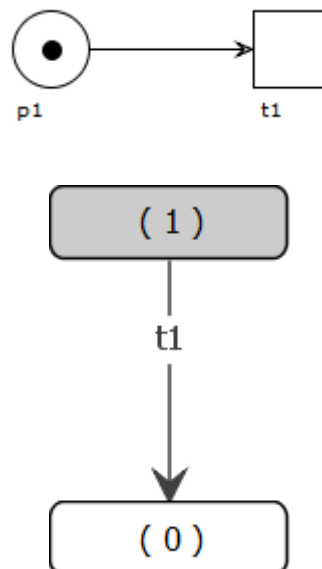


Abbildung 2.3: 112-B1-N02-A01-EineStelleEineMarkeEineTransition.pnml

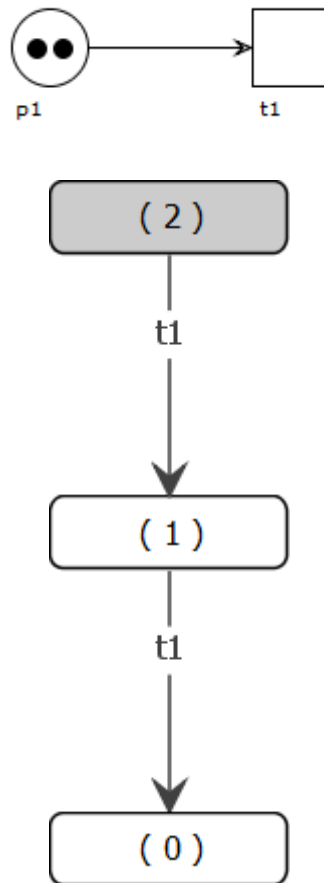


Abbildung 2.4: 113-B1-N03-A02-EineStelleZweiMarkenEineTransition.pnml

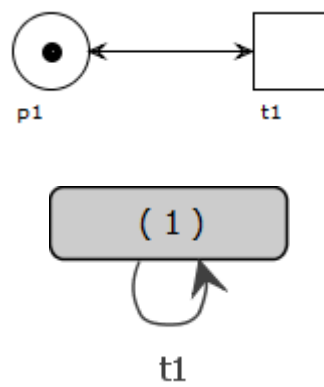


Abbildung 2.5: 114-B1-N01-A01-StelleInVorUndNachbereich.pnml

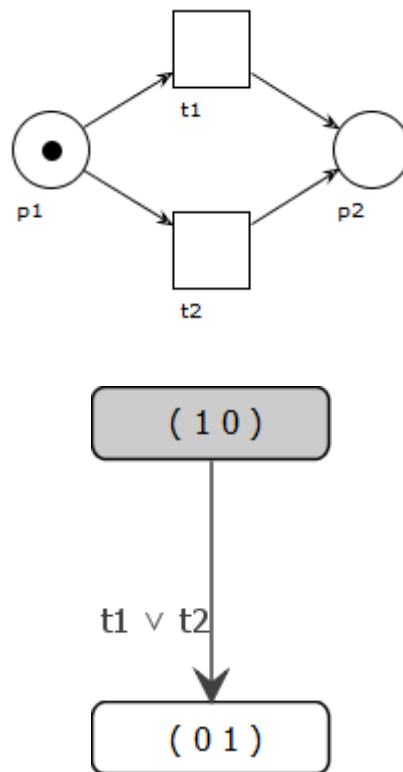


Abbildung 2.6: 115-B1-N02-A02-AlternativeTransitionen.pnml

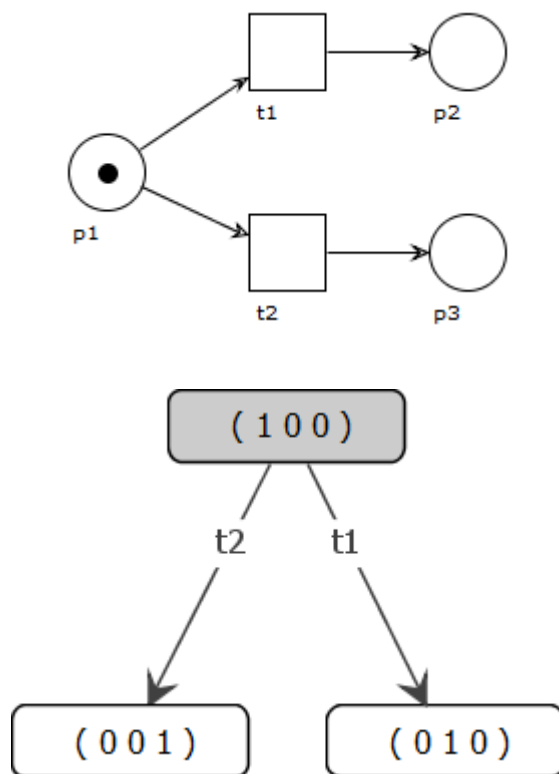


Abbildung 2.7: 116-B1-N03-A02-AlternativeStellen.pnml

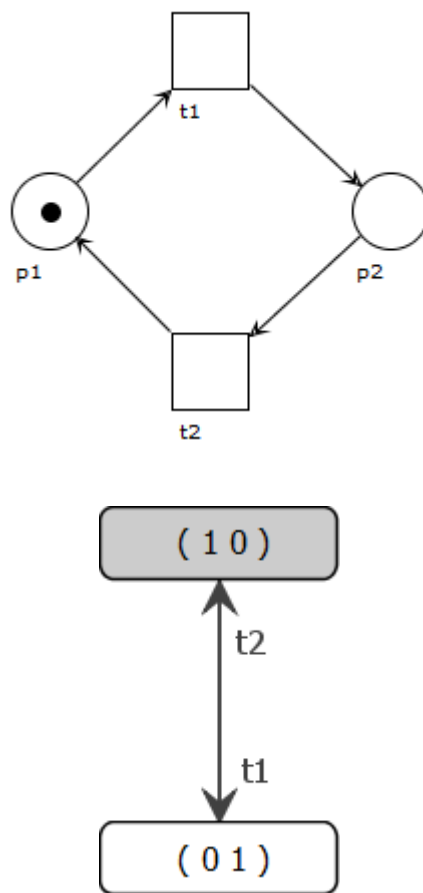


Abbildung 2.8: 117-B1-N02-A02-AlternierendeStellen.pnml

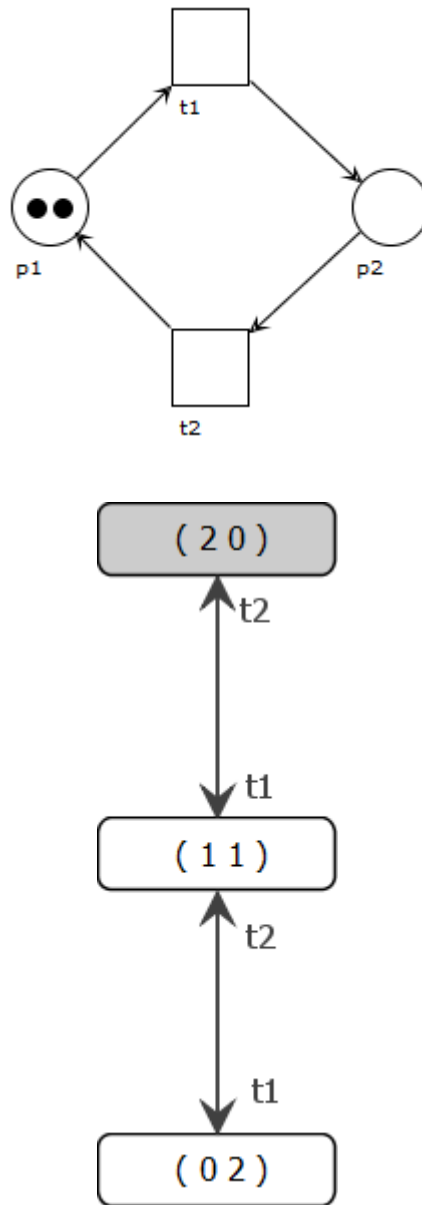


Abbildung 2.9: 118-B1-N03-A04-AlternierendeStellenZweiMarken.pnml

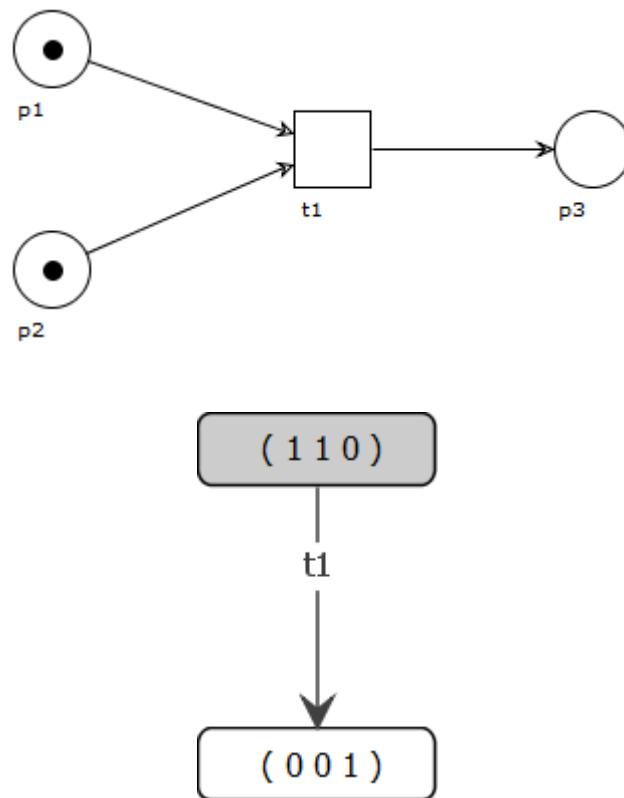


Abbildung 2.10: 119-B1-N02-A01-ZweiStellenTransitionStelle.pnml

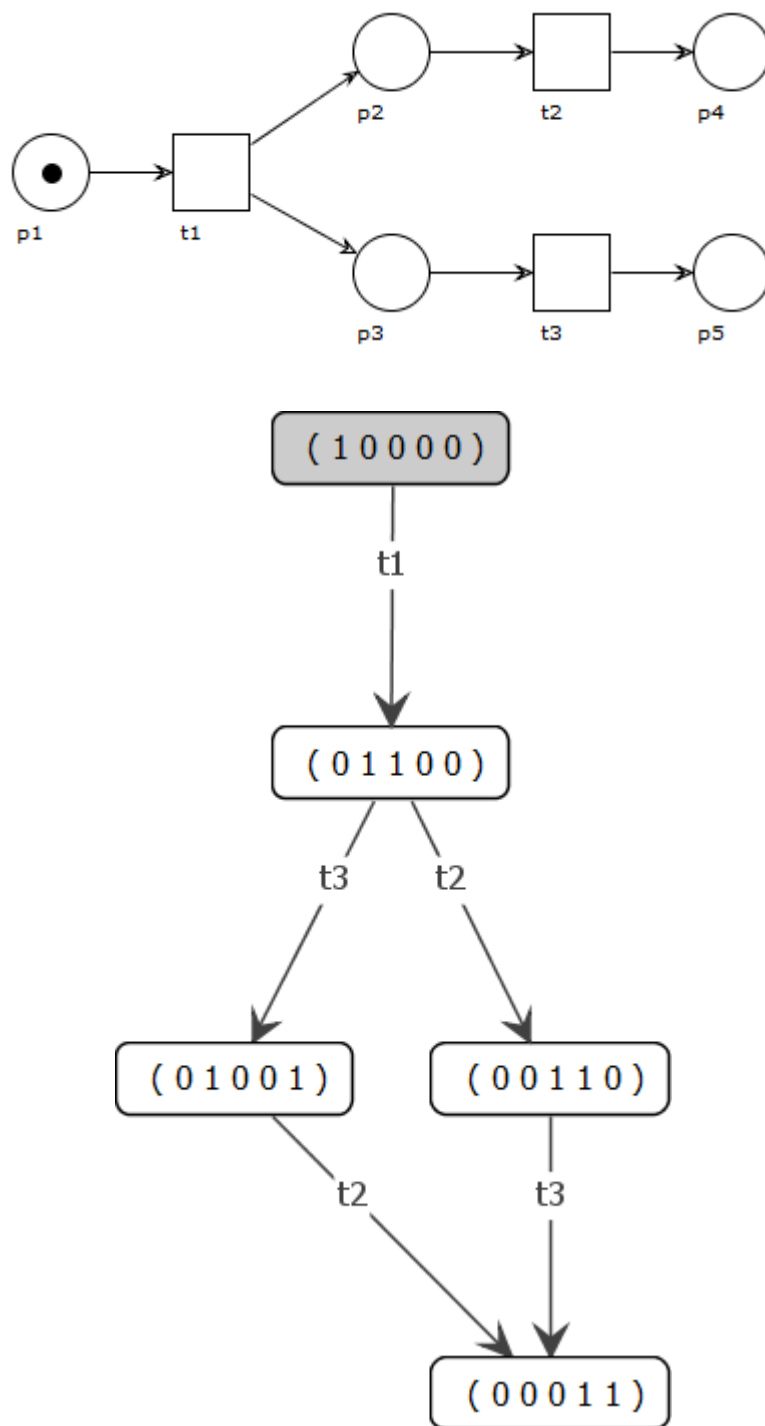


Abbildung 2.11: 130-B1-N05-A05-Gabelung.pnm1

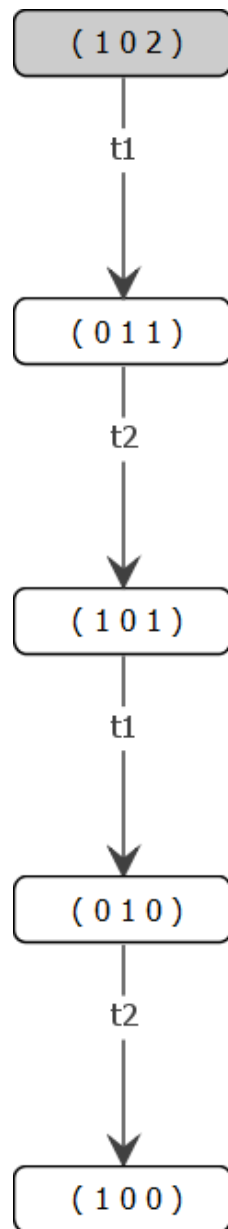
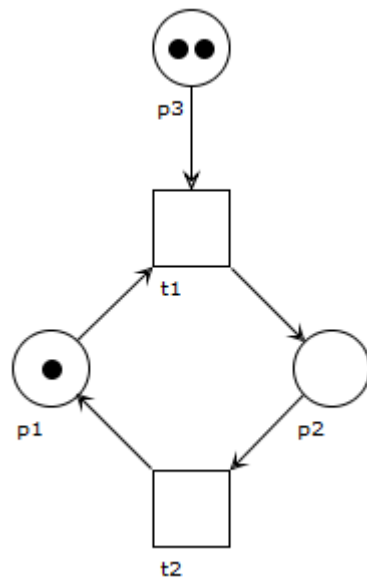


Abbildung 2.12: 131-B1-N05-A04-BegrenzteSchleife.pnml

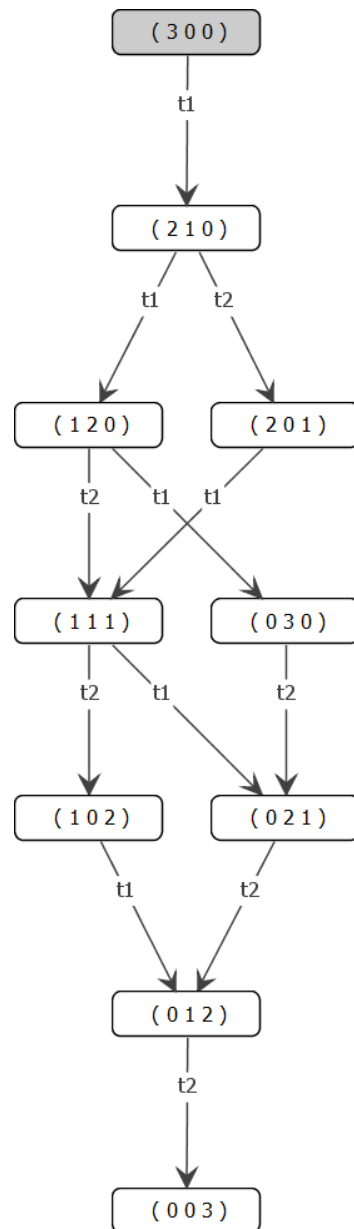
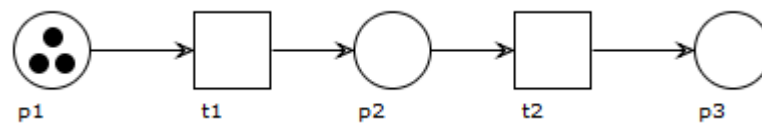


Abbildung 2.13: 132-B1-N10-A12-StelleTransStelleTransStelle.pnml

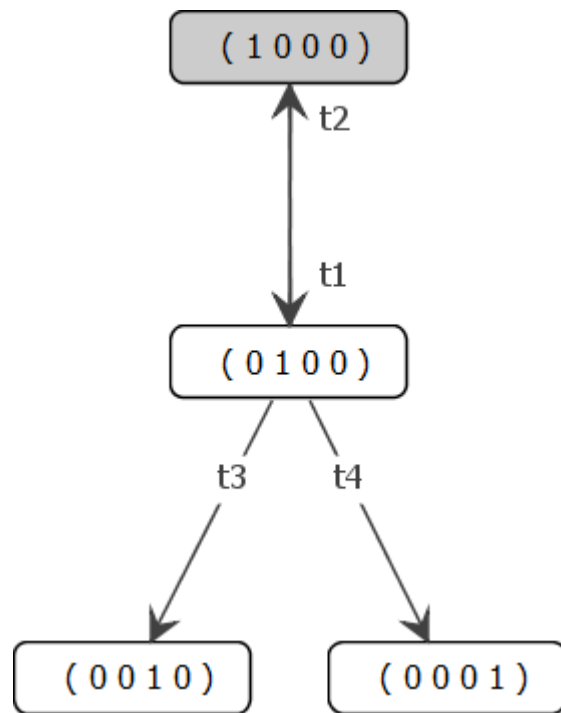
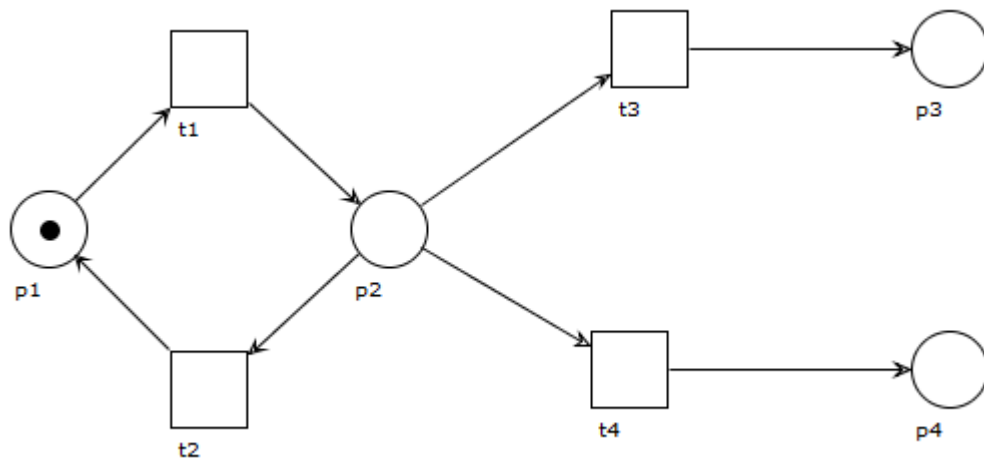


Abbildung 2.14: 150-B1-N04-A04-Example01.pnml

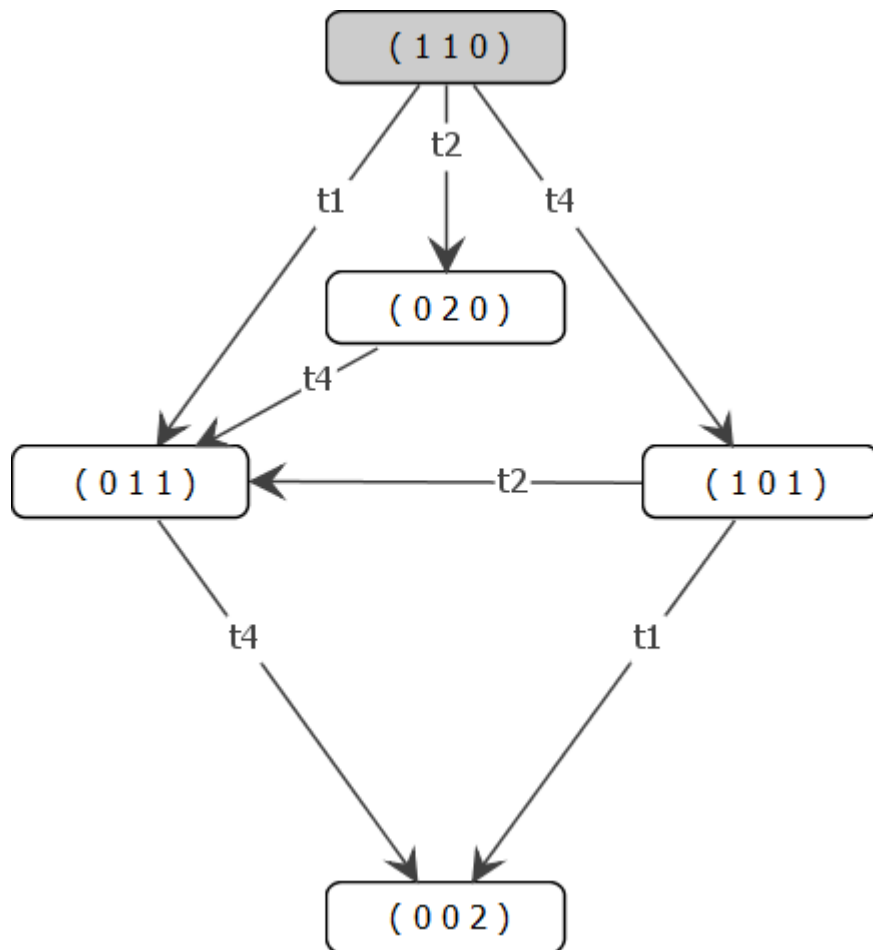
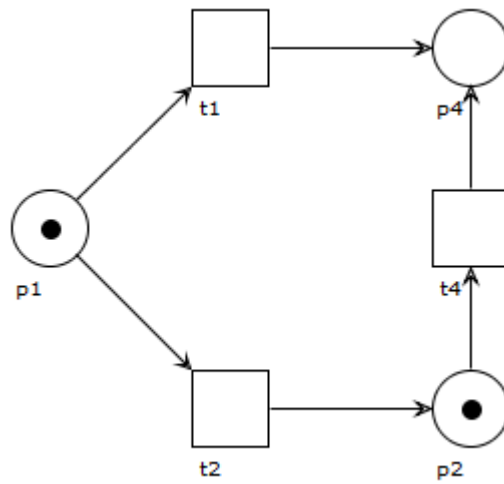


Abbildung 2.15: 151-B1-N05-A07-Example02.pnml

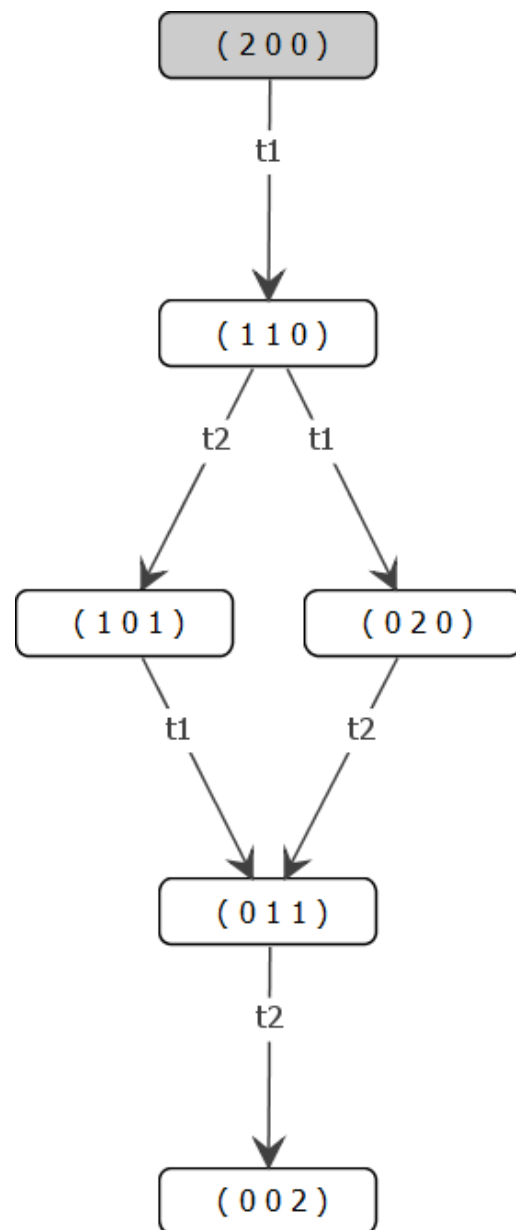
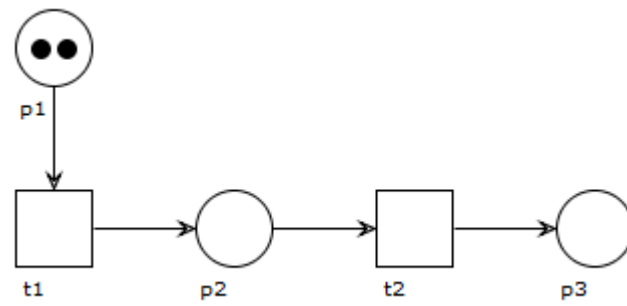


Abbildung 2.16: 152-B1-N06-A06-Example03v1.pnml

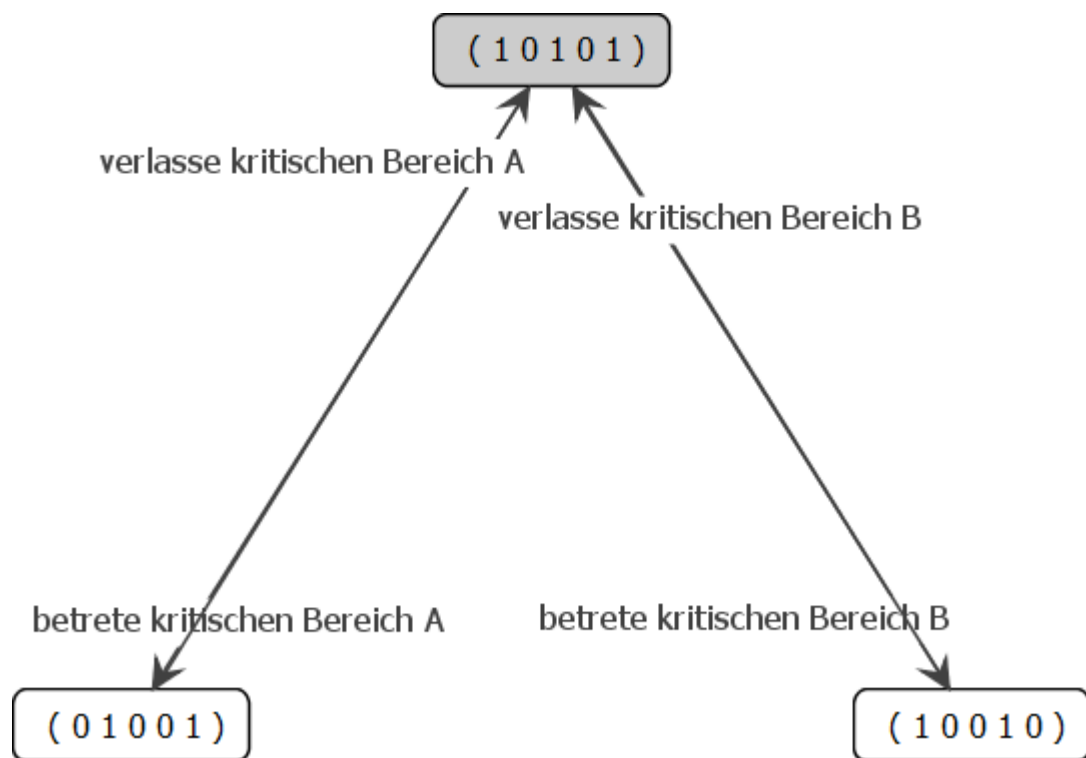
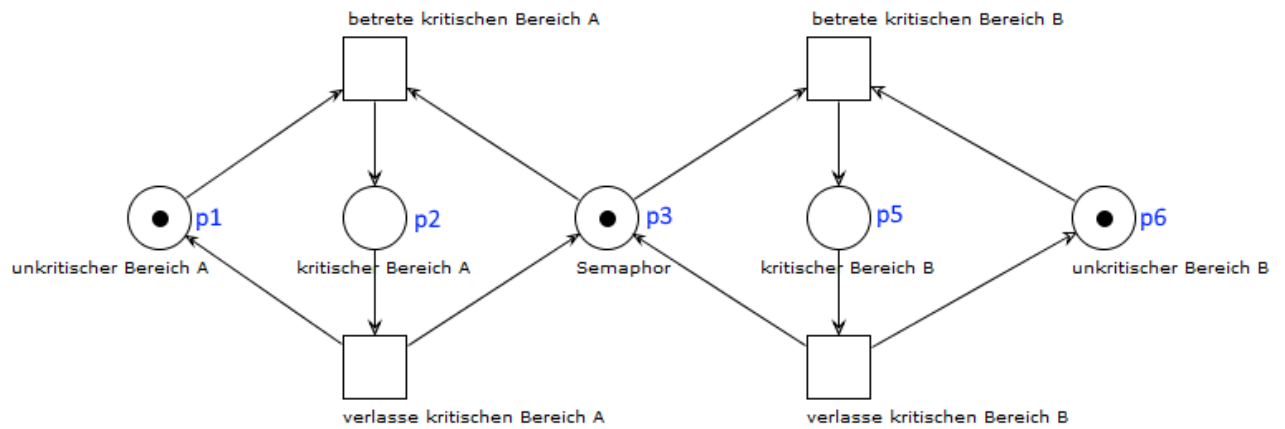


Abbildung 2.17: 170-B1-N03-A04-MutualExclusion.pnml

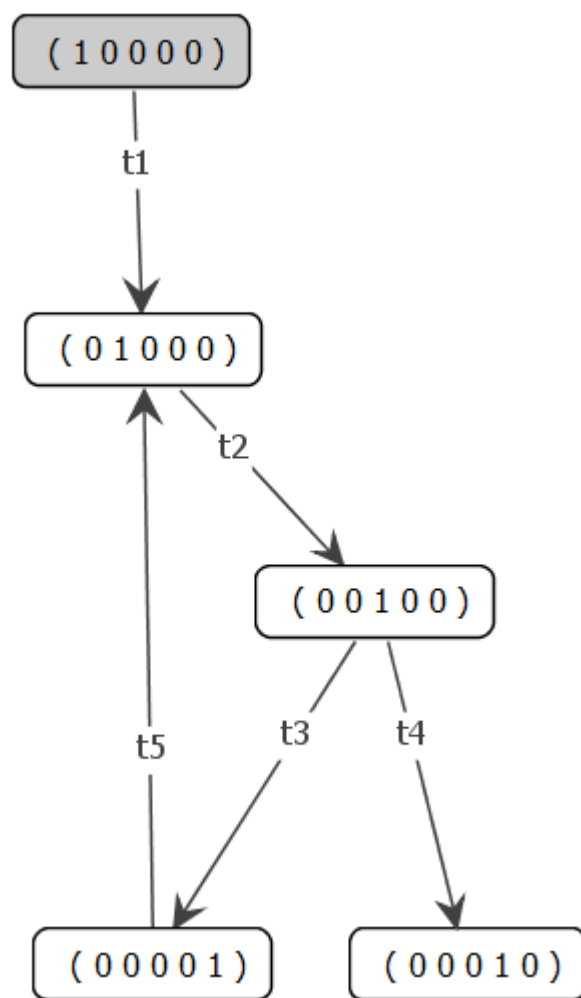
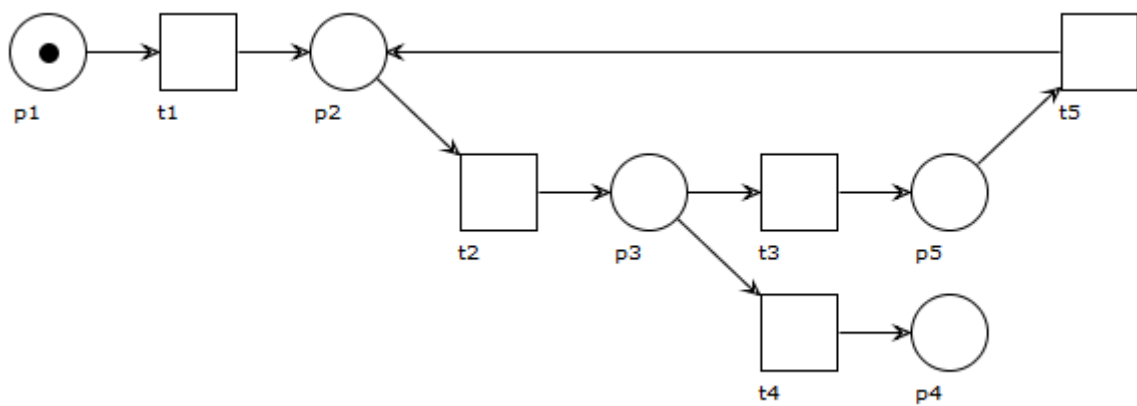


Abbildung 2.18: 171-B1-N05-A05-Kreis3.pnml

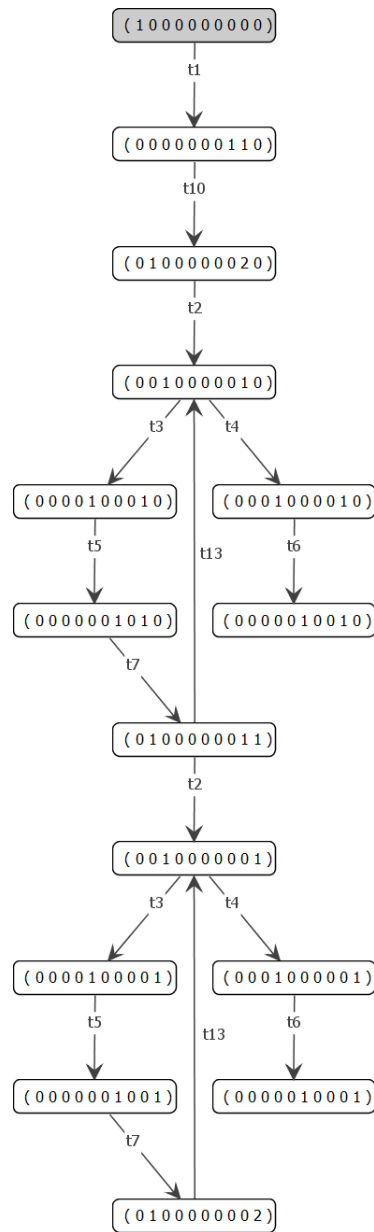
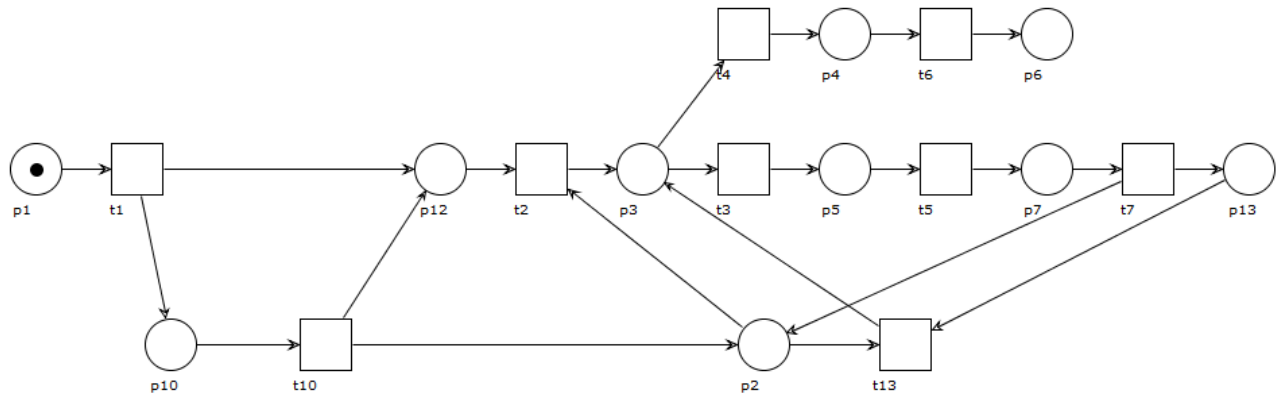


Abbildung 2.19: 172-B1-N15-A16-Kreis-gross.pnml

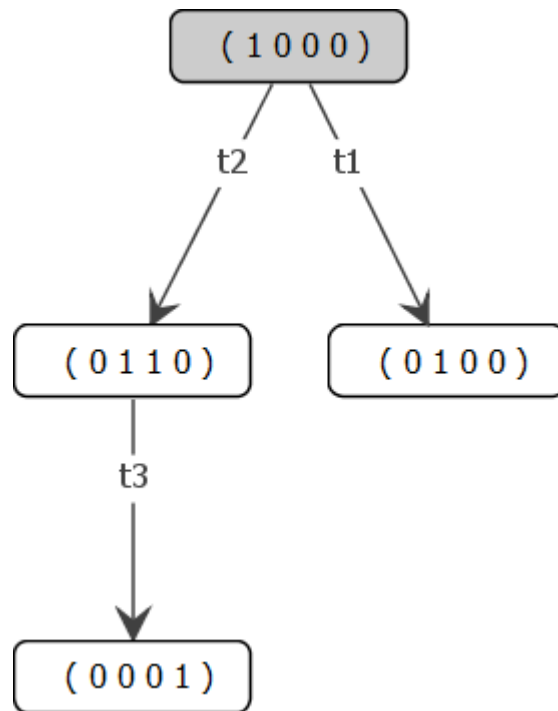
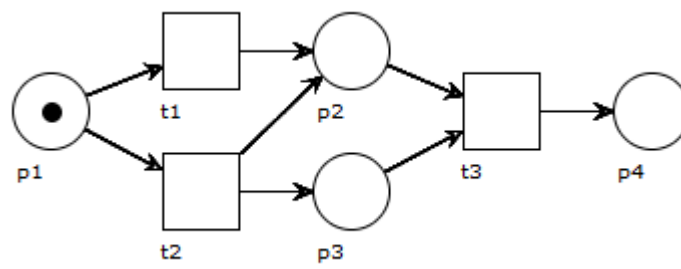


Abbildung 2.20: 173-B1-N04-A03-m-auf-anderem-Pfad1.pnml

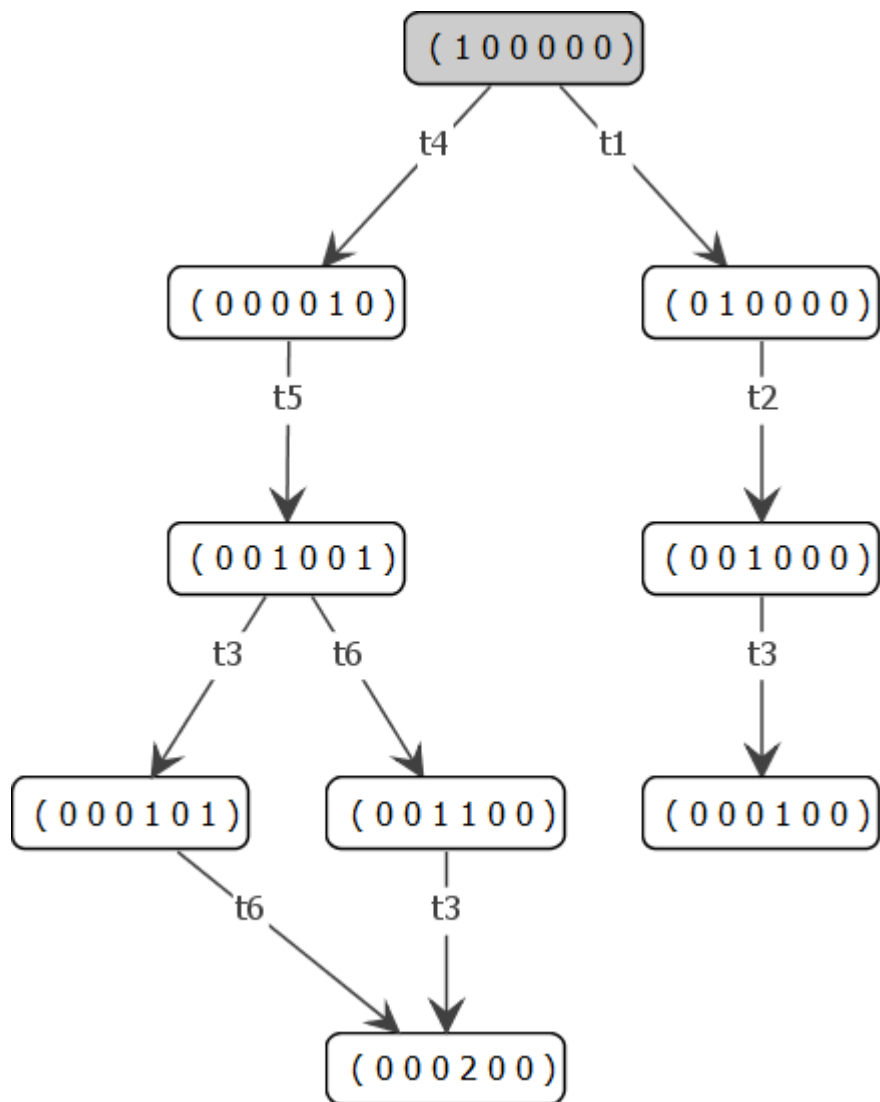
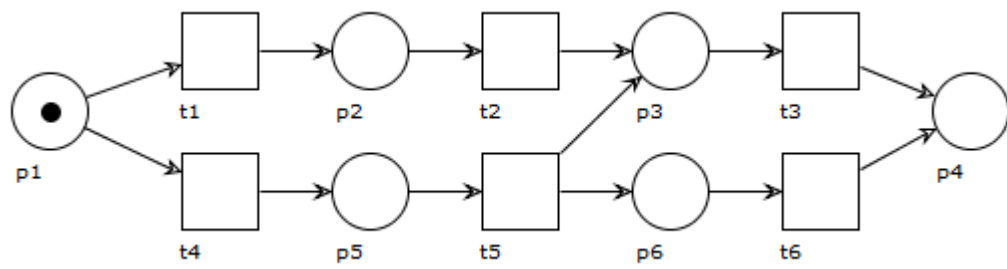


Abbildung 2.21: 174-B1-N09-A09-m-auf-anderem-Pfad2.pnml

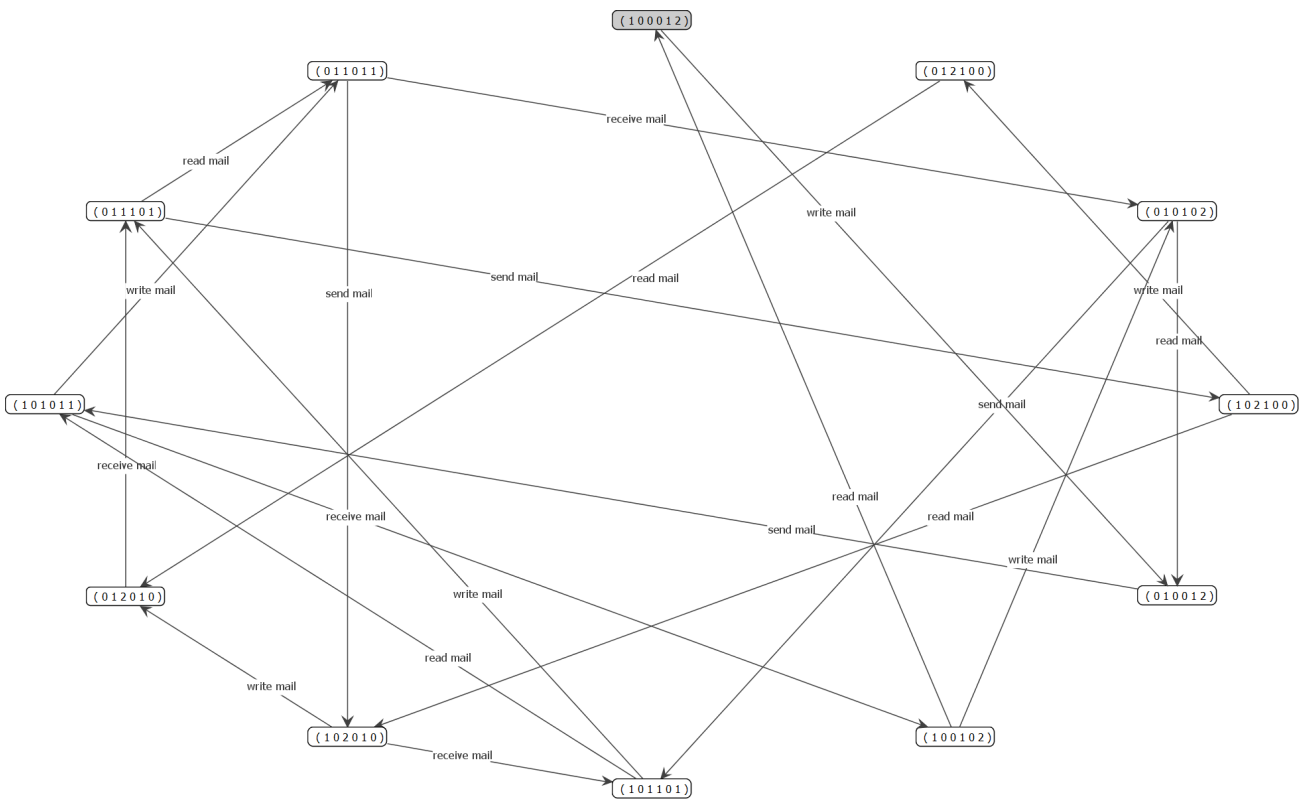
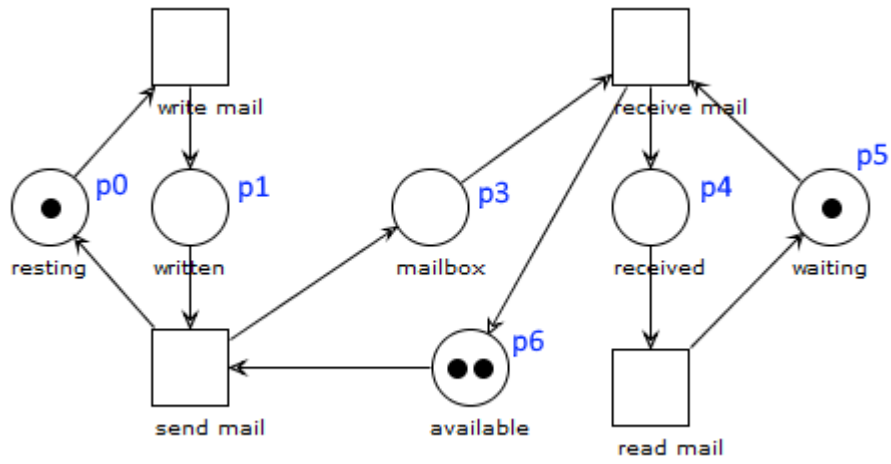


Abbildung 2.22: 175-B1-N12-A20-Mailbox-2-Bounded.pnml

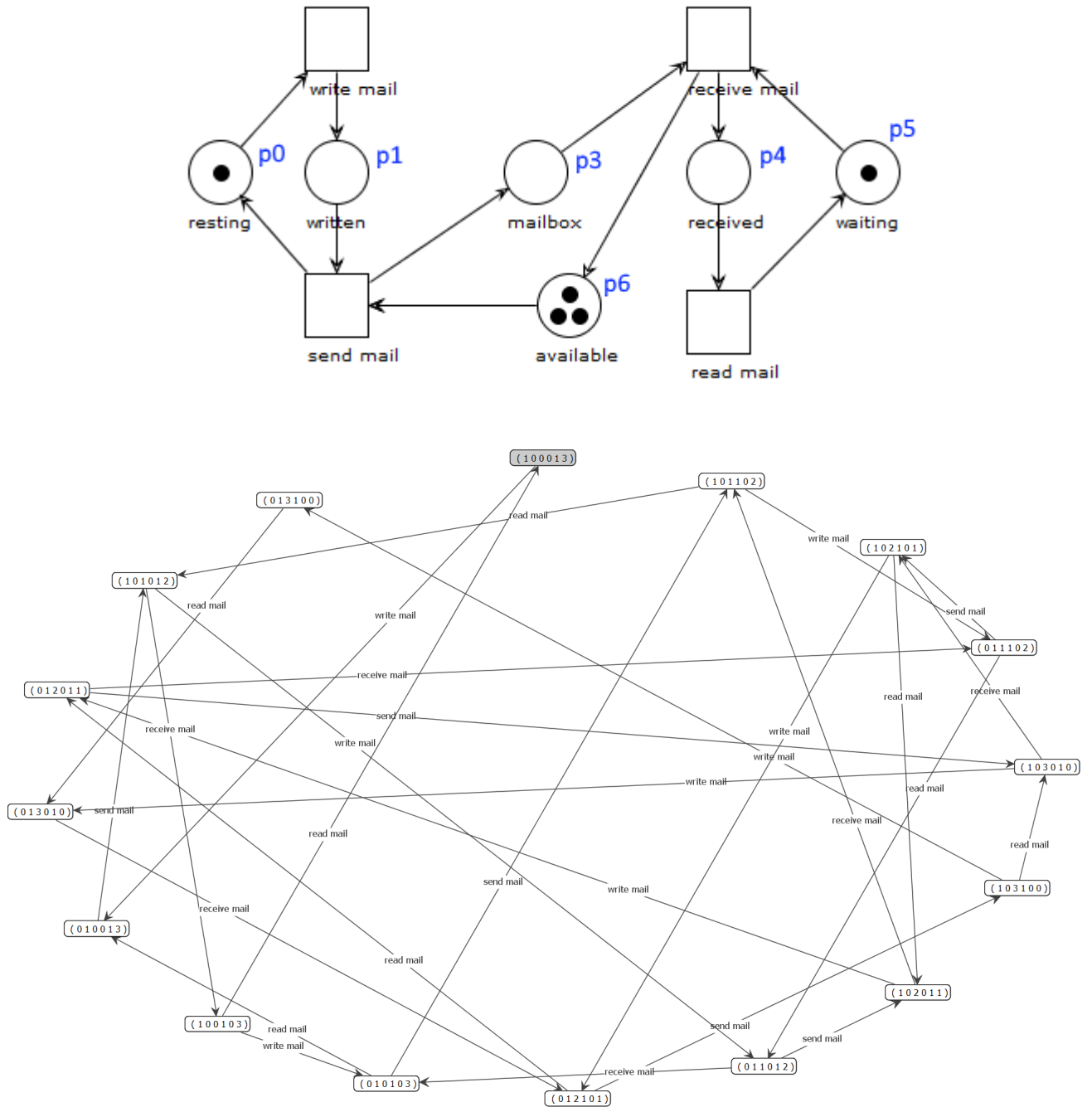


Abbildung 2.23: 176-B1-N16-A28-Mailbox-3-Bounded.pnml

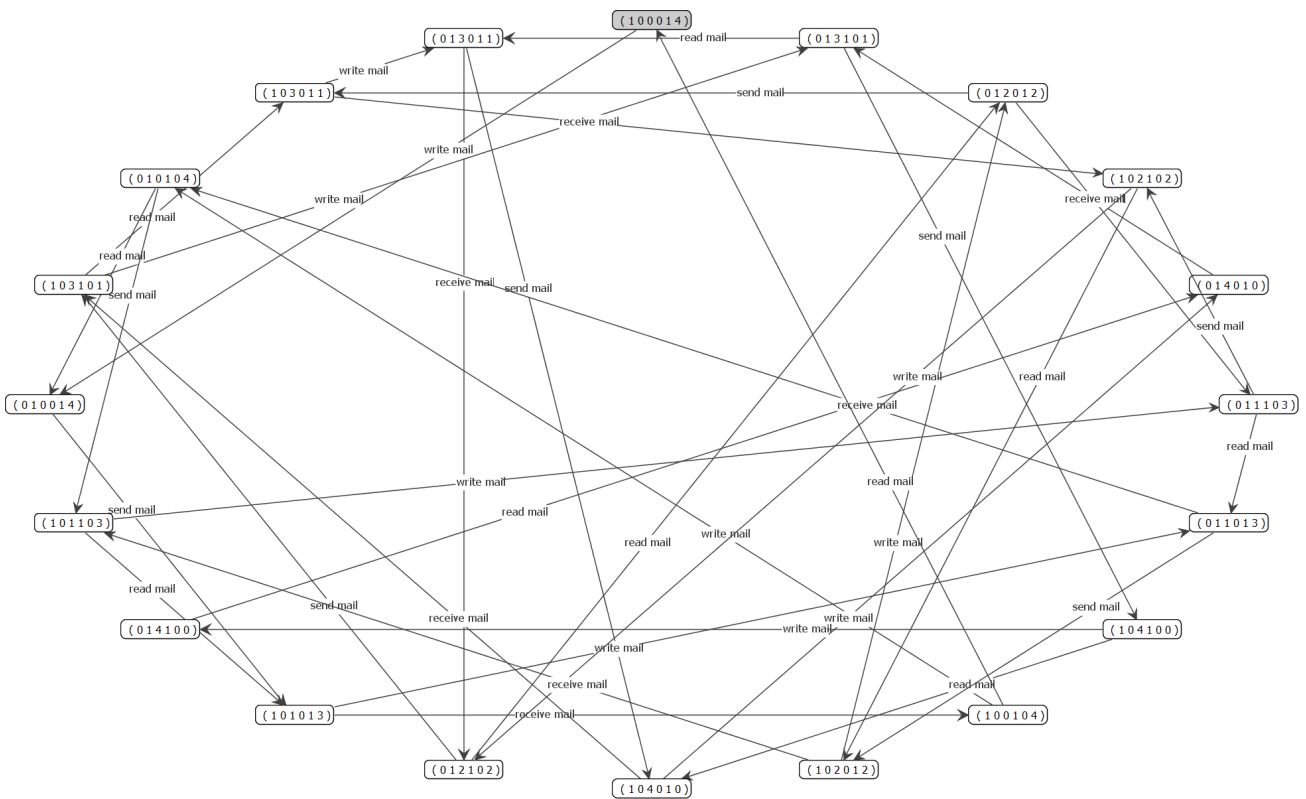
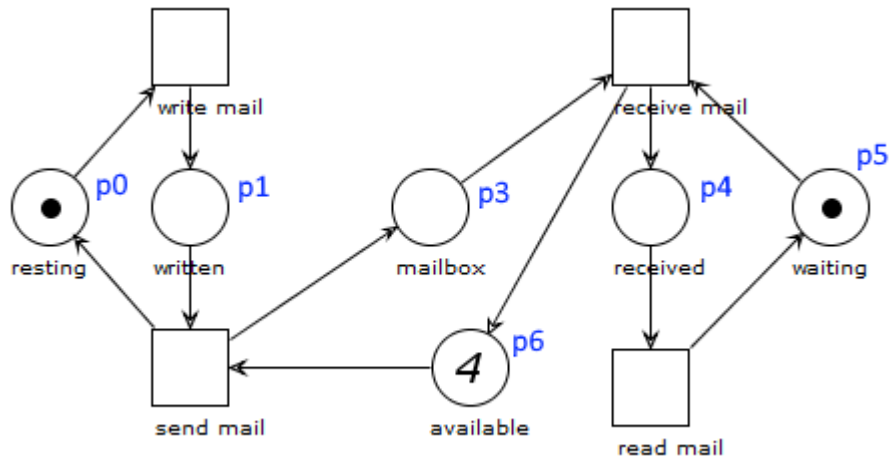


Abbildung 2.24: 177-B1-N20-A36-Mailbox-4-Bounded.pnml

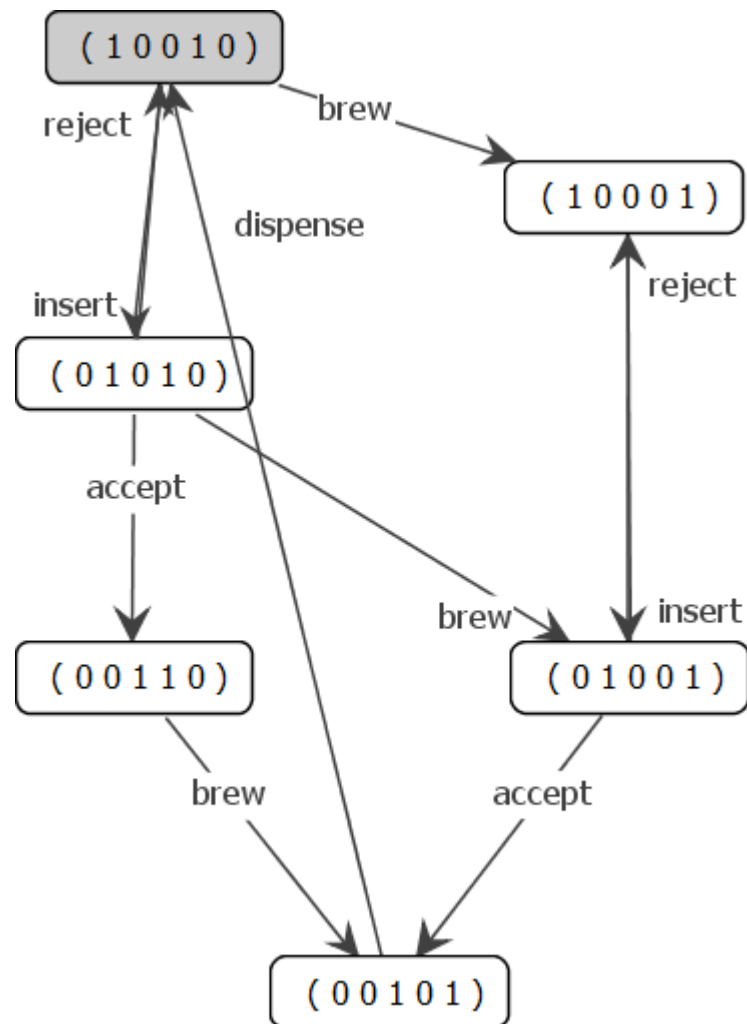
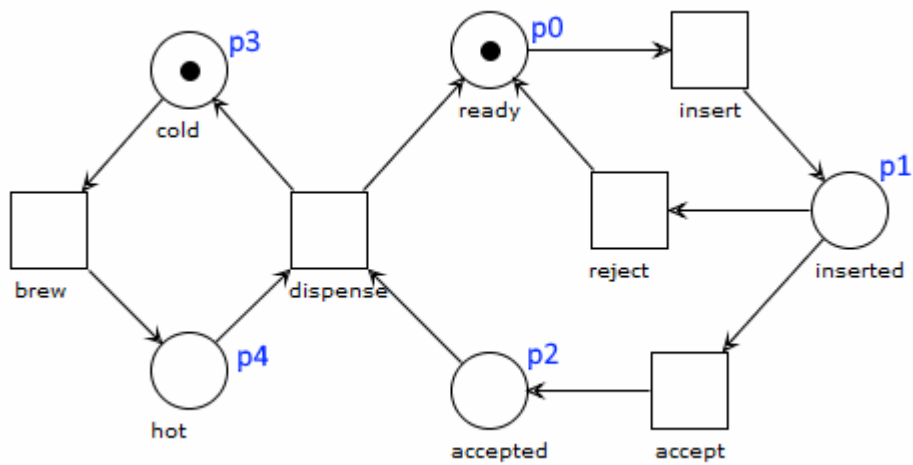


Abbildung 2.25: 178-B1-N06-A10-Vending-Machine.pnml

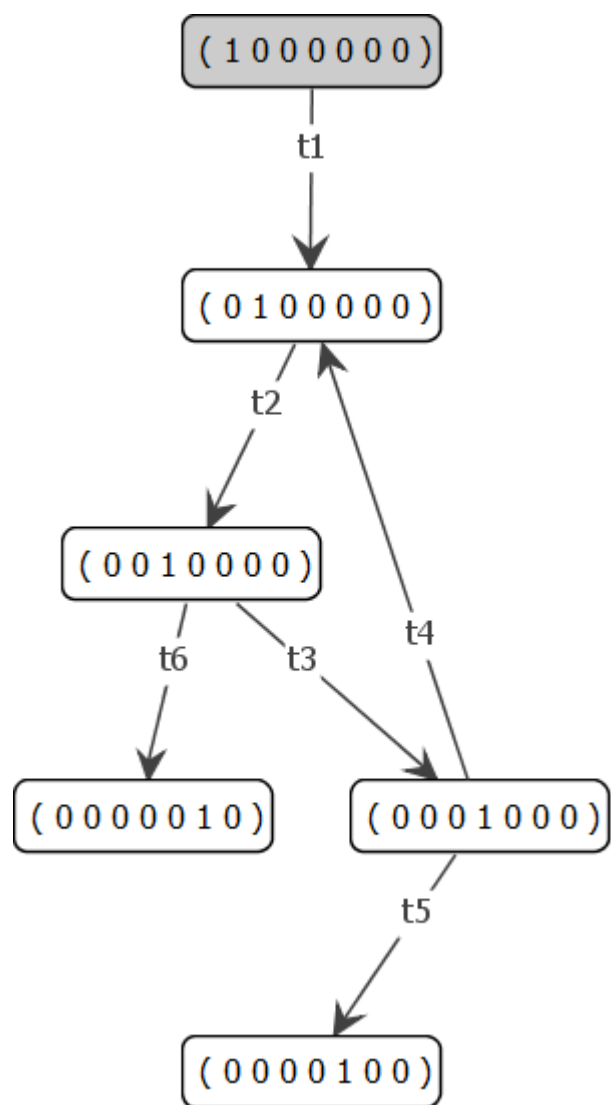
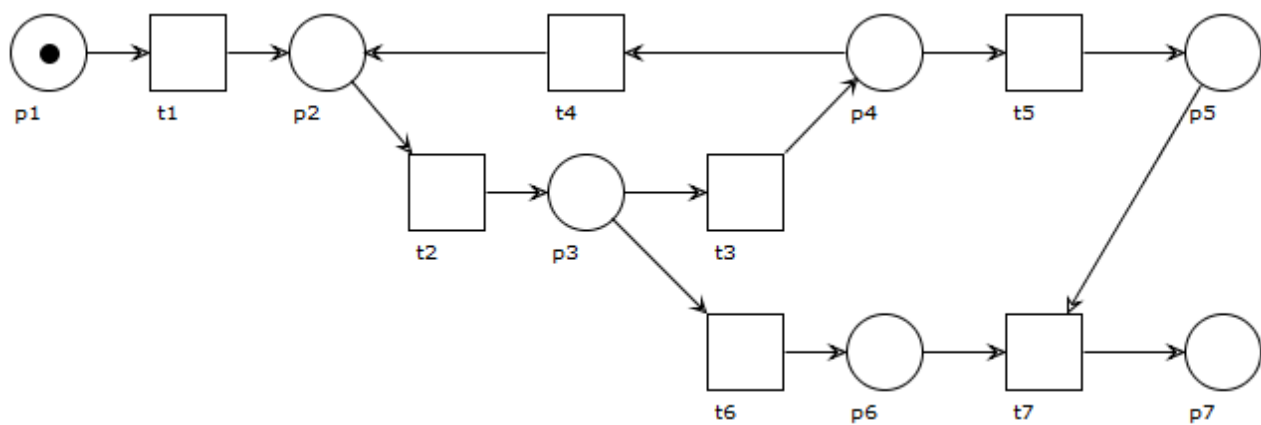


Abbildung 2.26: 179-B1-N06-A06-Kreis-mit-zwei-Ausgaengen.pnml

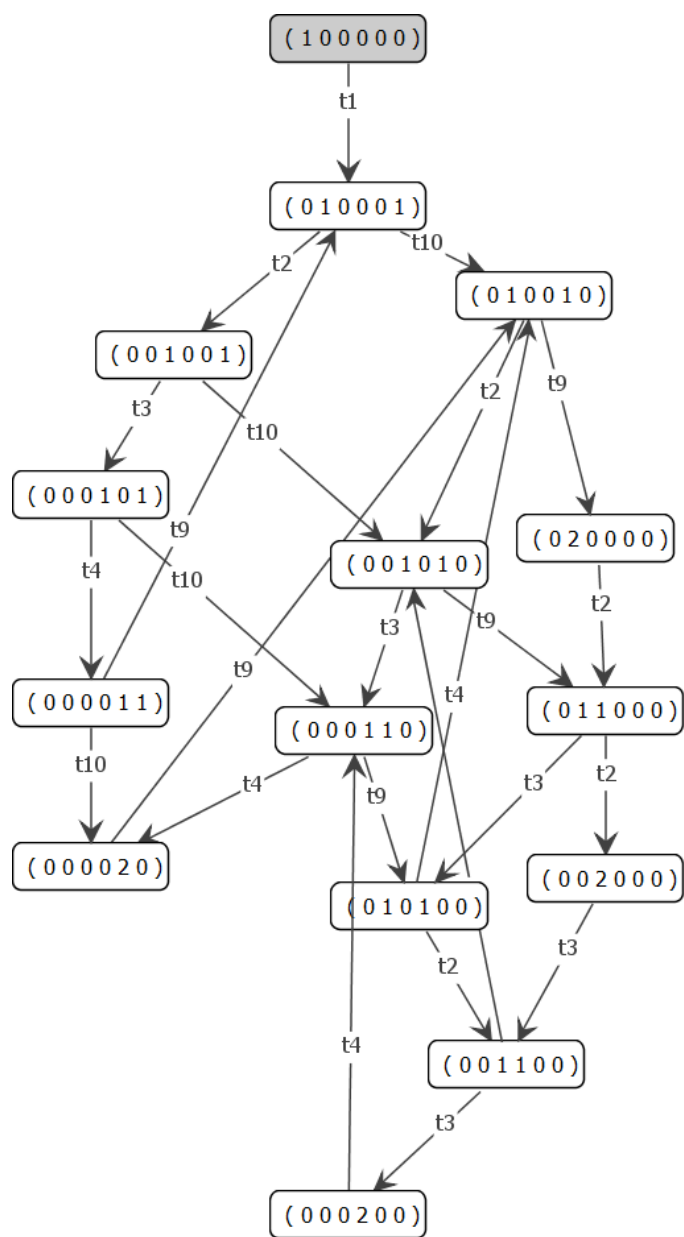
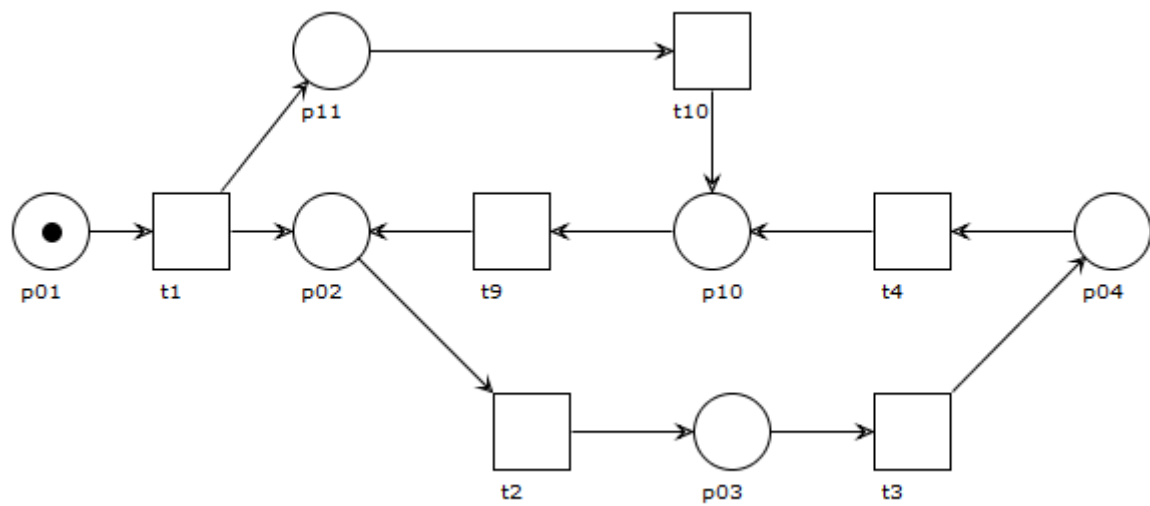


Abbildung 2.27: 180-B1-N15-A25-Kreis-mit-zwei-Eingaengen.pnml

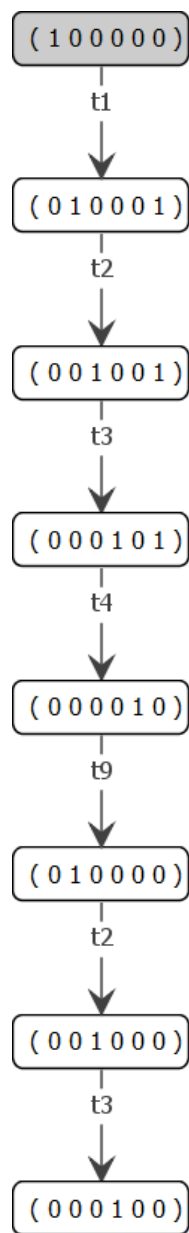
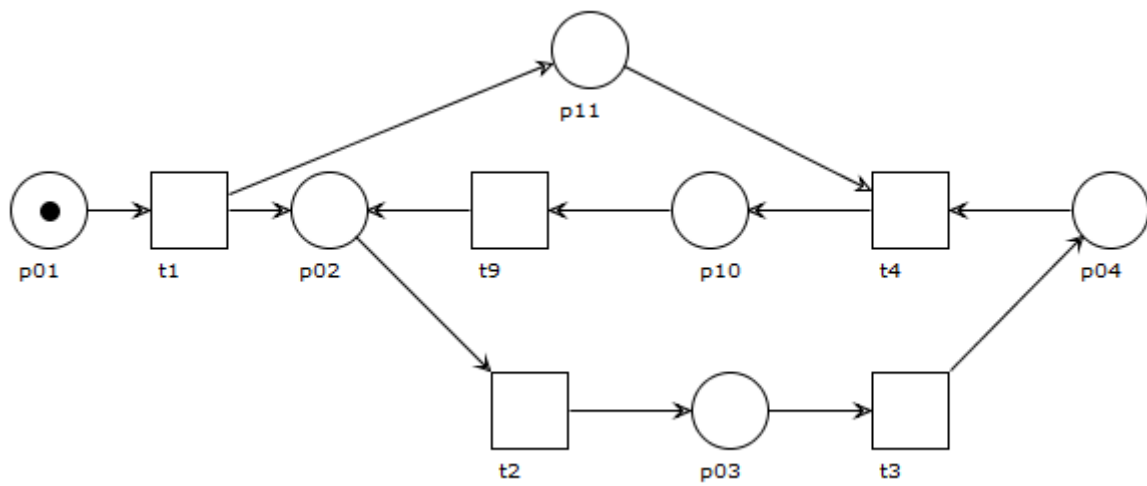


Abbildung 2.28: 181-B1-N08-A07-Kreis-ohne-Kreis.pnml

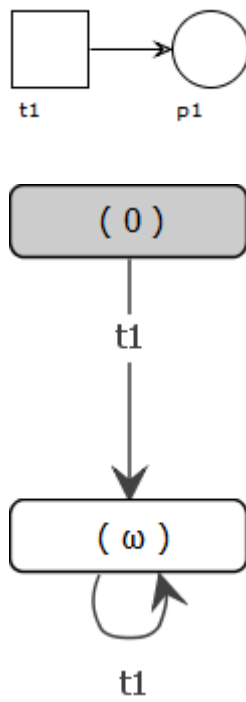


Abbildung 2.29: 210-B0-P01-Qxx-EineTransitionEineStelle.pnml

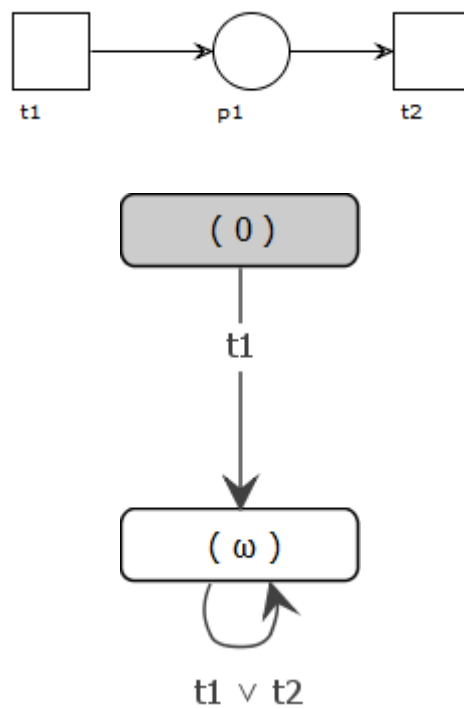


Abbildung 2.30: 211-B0-P01-Qxx-TransStelleTrans.pnml

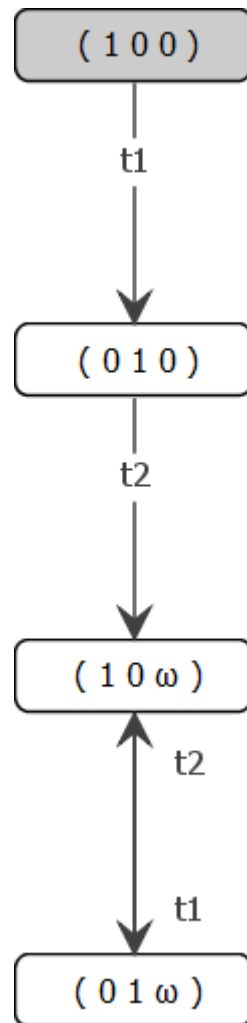
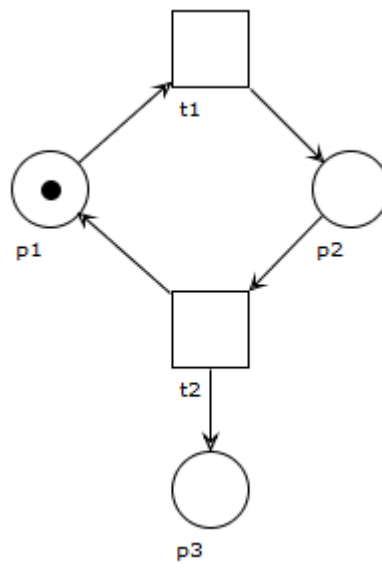


Abbildung 2.31: 230-B0-P02-Qxx-Counter.pnml

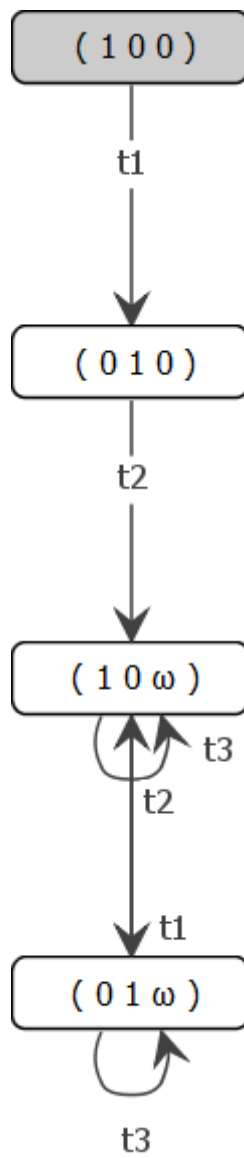
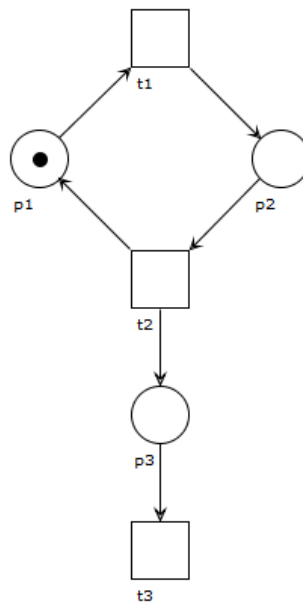


Abbildung 2.32: 231-B0-P02-Qxx-CounterMitAbfluss.pnml

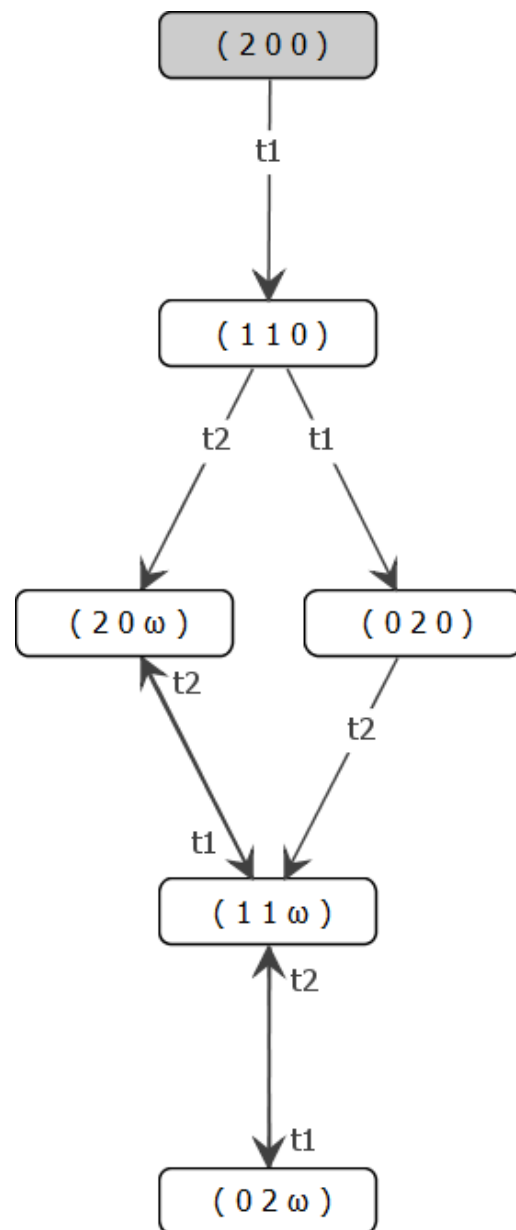
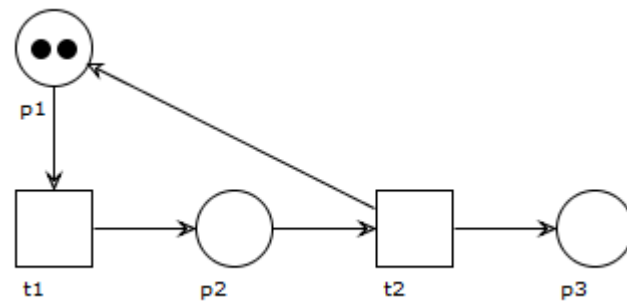


Abbildung 2.33: 250-B0-P02-Q03-Example3v2.pnml

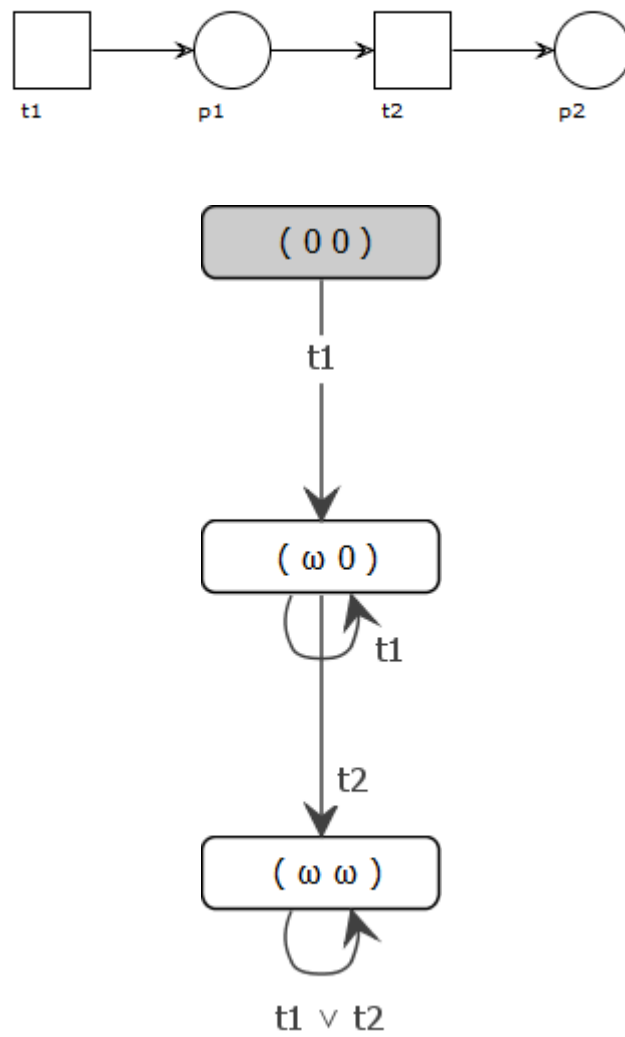


Abbildung 2.34: 251-B0-P01-Qxx-Netz1.pnml

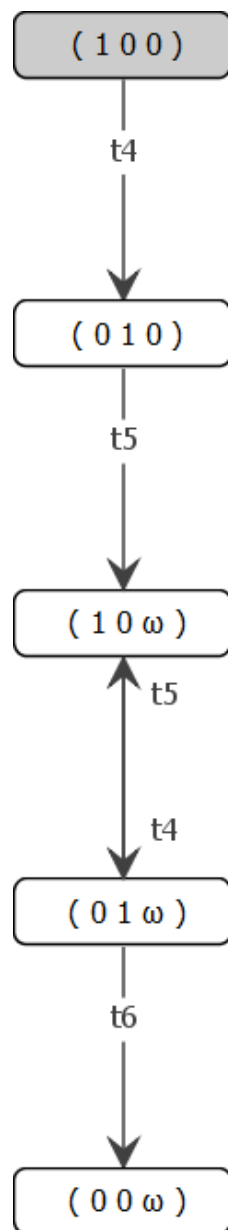
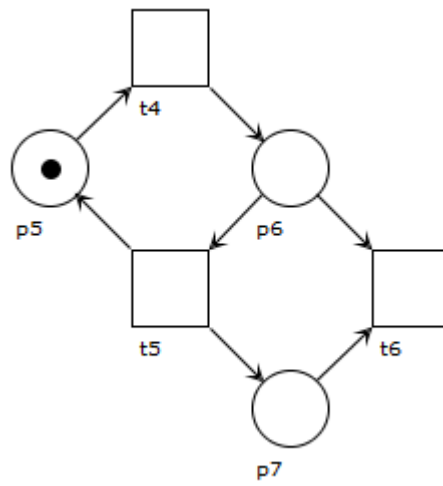


Abbildung 2.35: 252-B0-P02-Qxx-Netz2.pnml

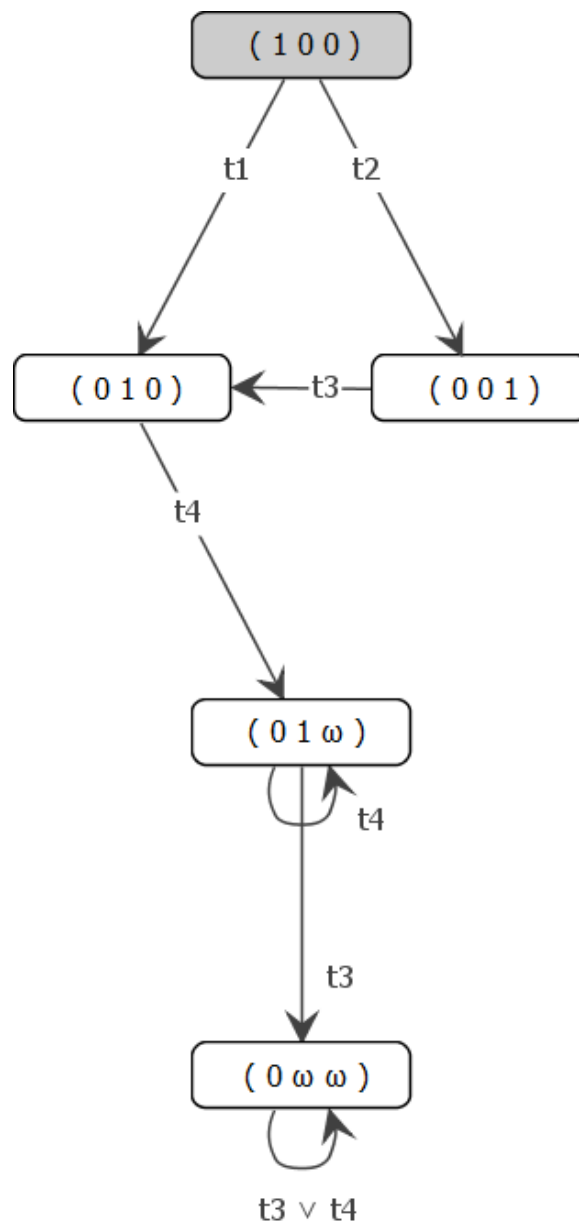
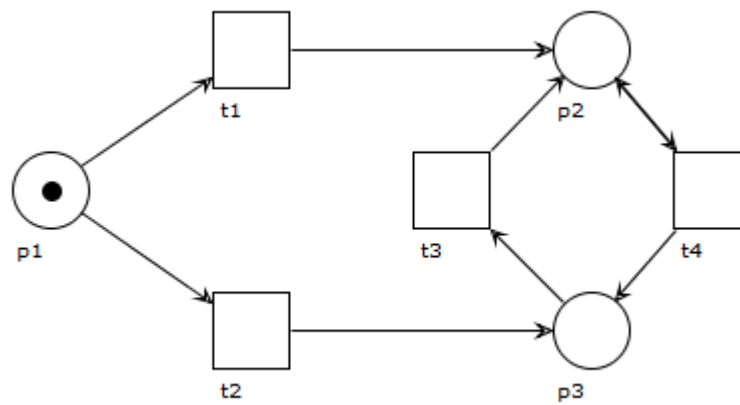


Abbildung 2.36: 253-B0-P02-Q03-Netz3.pnml

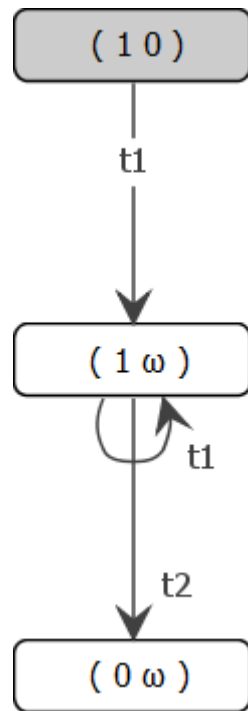
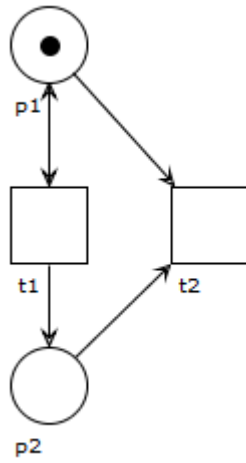


Abbildung 2.37: 254-B0-P01-Qxx-Netz4.pnml

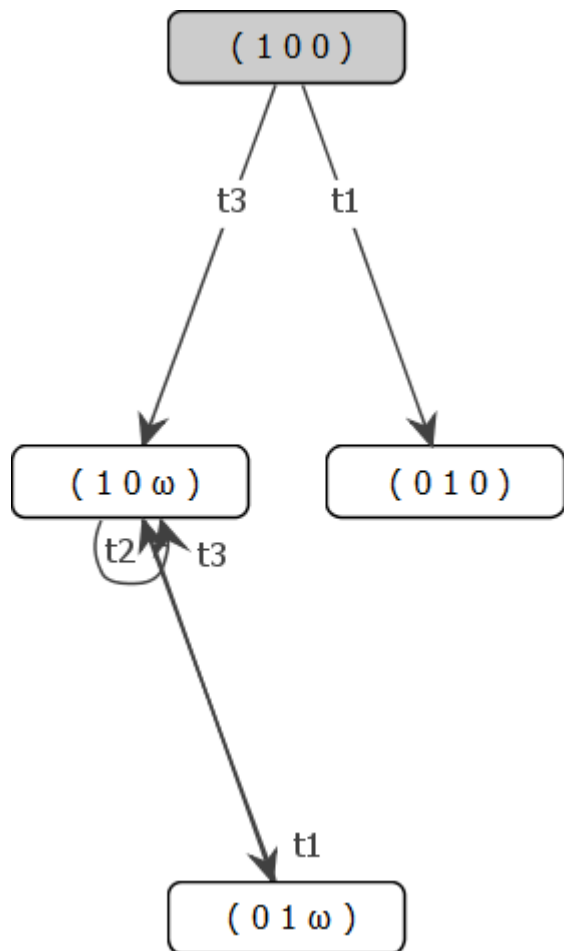
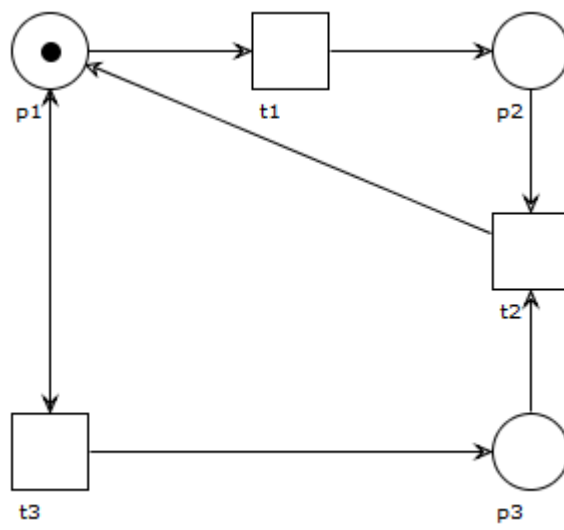


Abbildung 2.38: 255-B0-P01-Qxx-Netz5.pnml

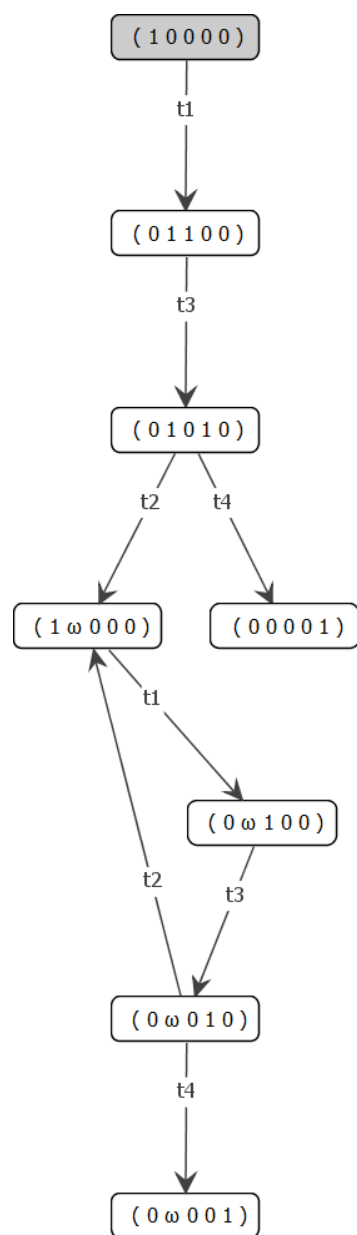
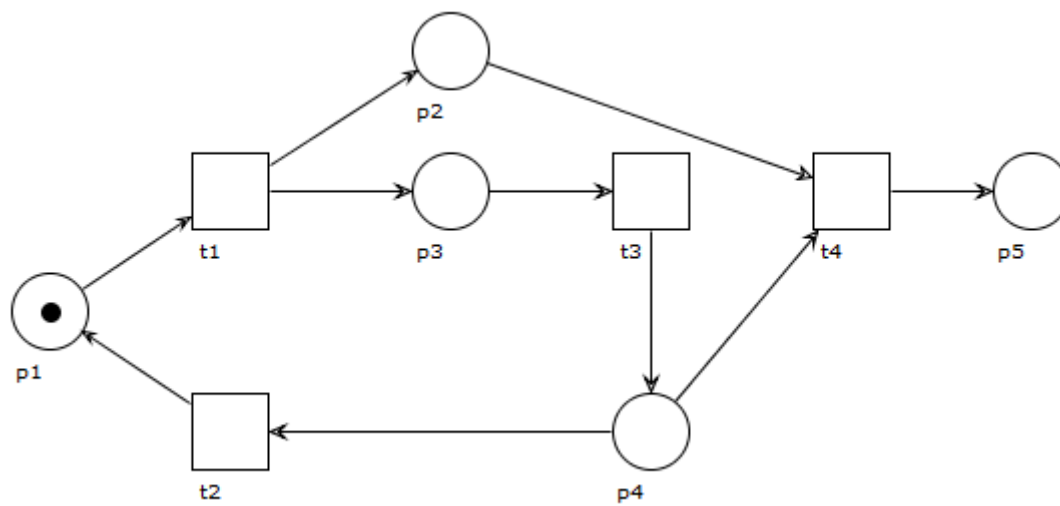


Abbildung 2.39: 256-B0-P03-Qxx-Netz6.pnml

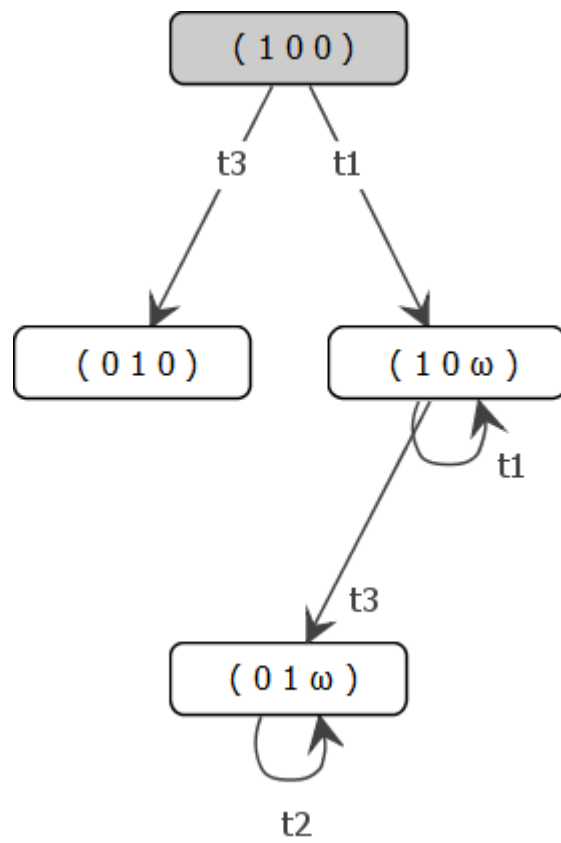
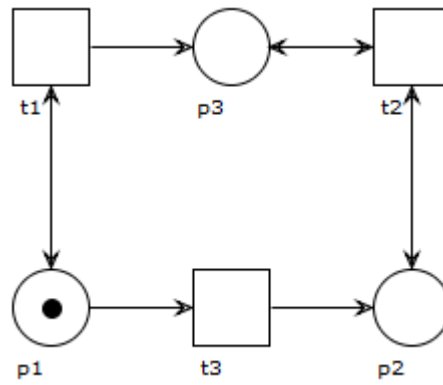


Abbildung 2.40: 257-B0-P01-Qxx-Netz7.pnml

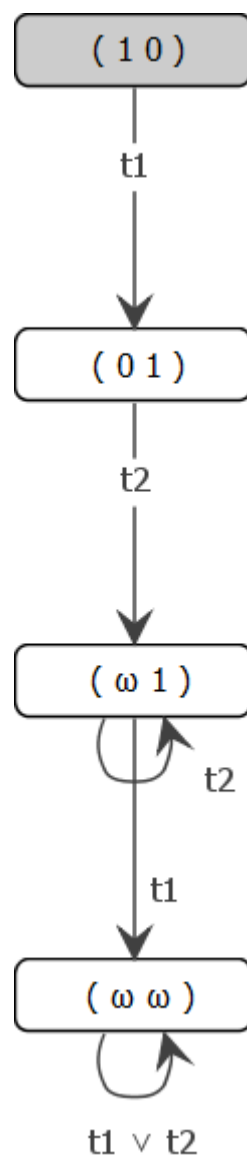
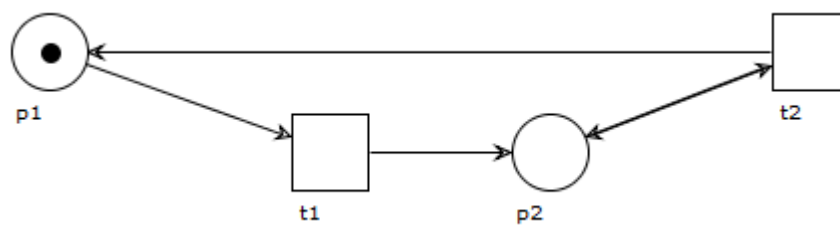


Abbildung 2.41: 258-B0-P02-Qxx-Netz8.pnml

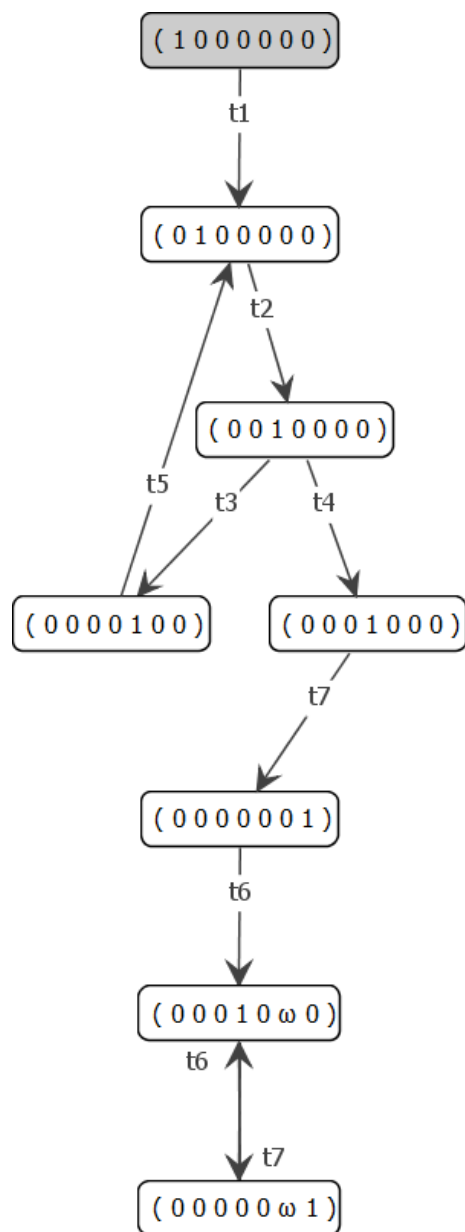
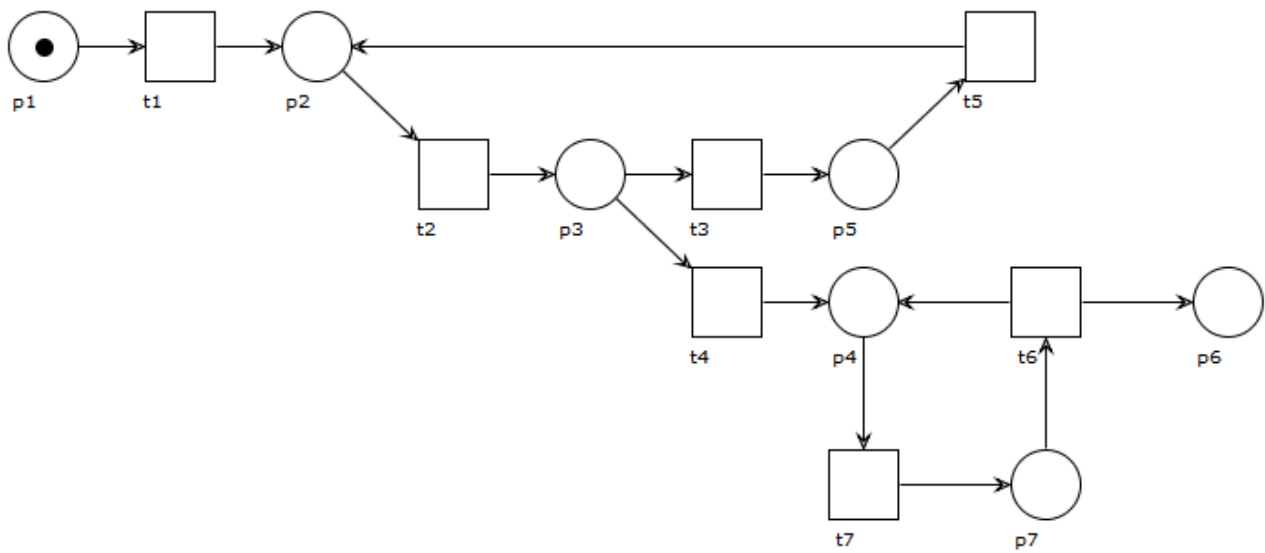


Abbildung 2.43: 271-B0-P05-Qxx-Kreis3-vor-Omega.pnml

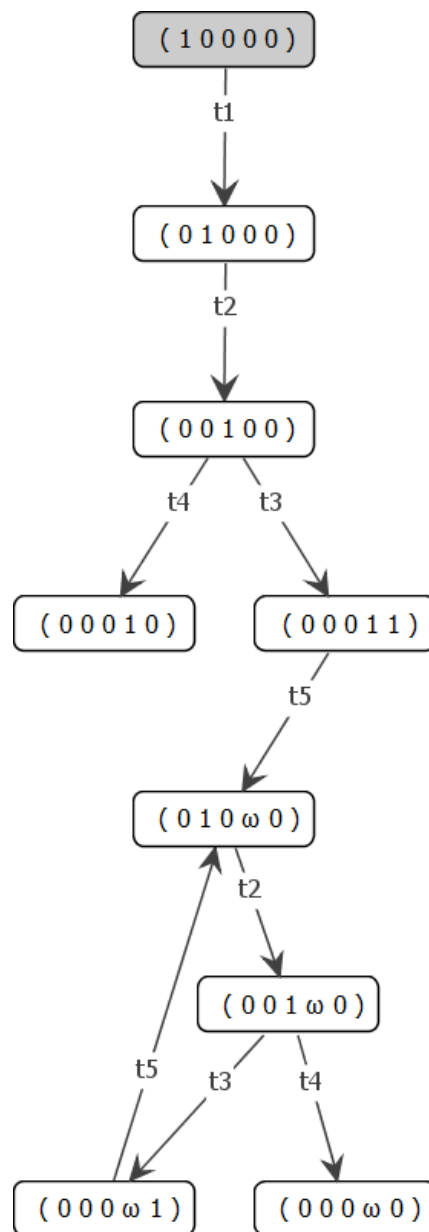
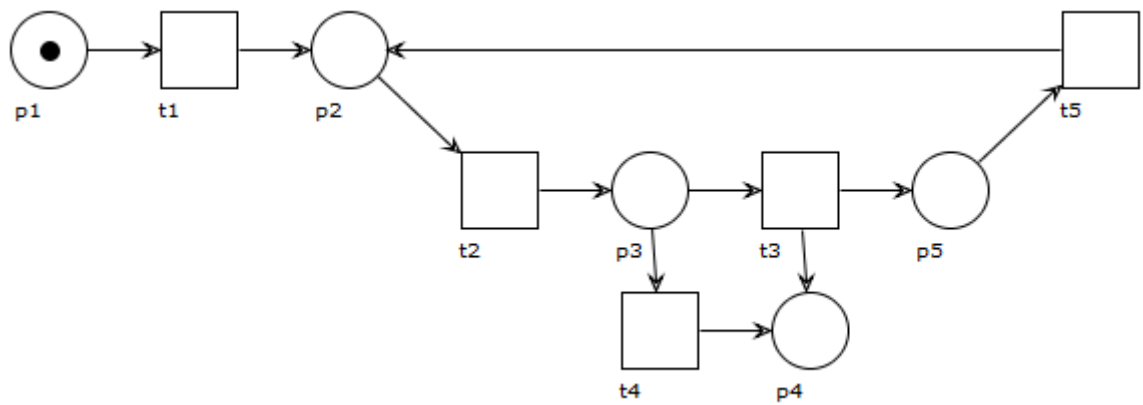


Abbildung 2.44: 272-B0-P04-Qxx-Kreis3-nach-Omega.pnm1

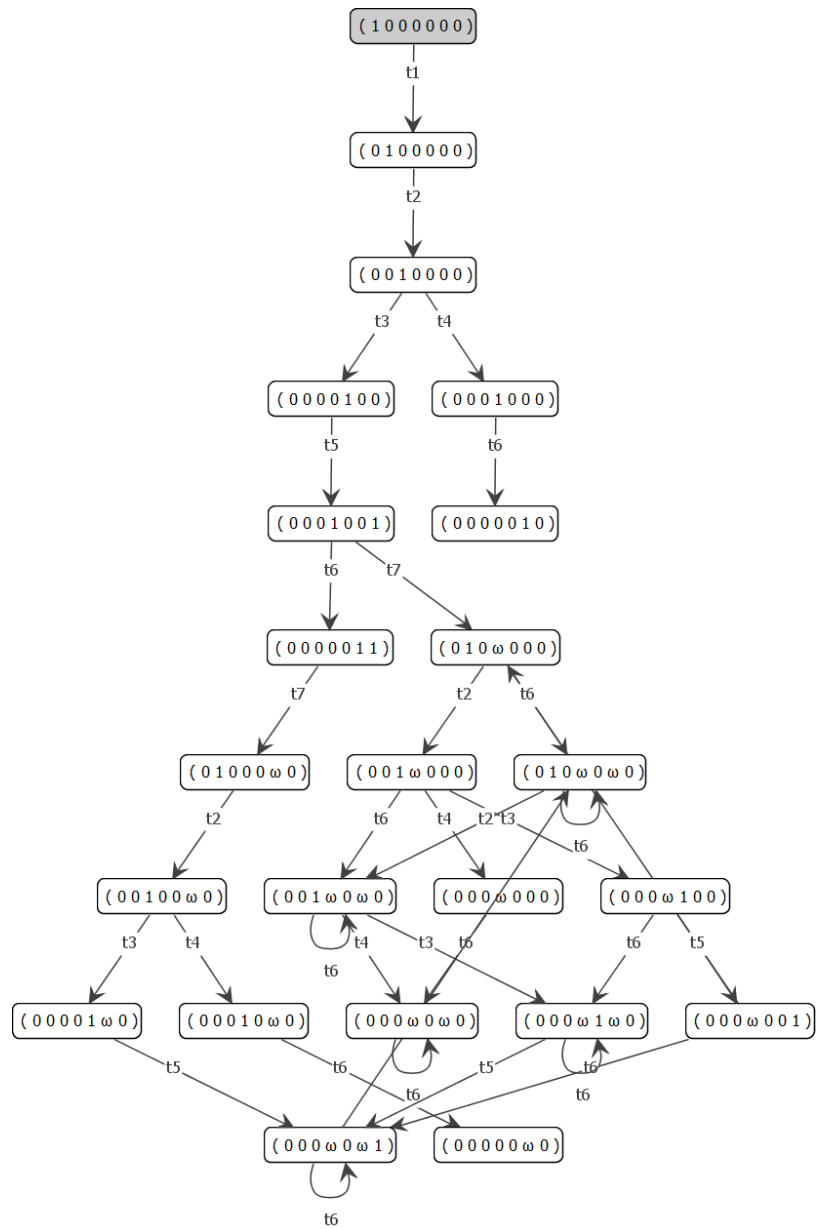


Abbildung 2.45: 273-B0-P05-Q06-Kreis-nach-0omega-gross.pnml

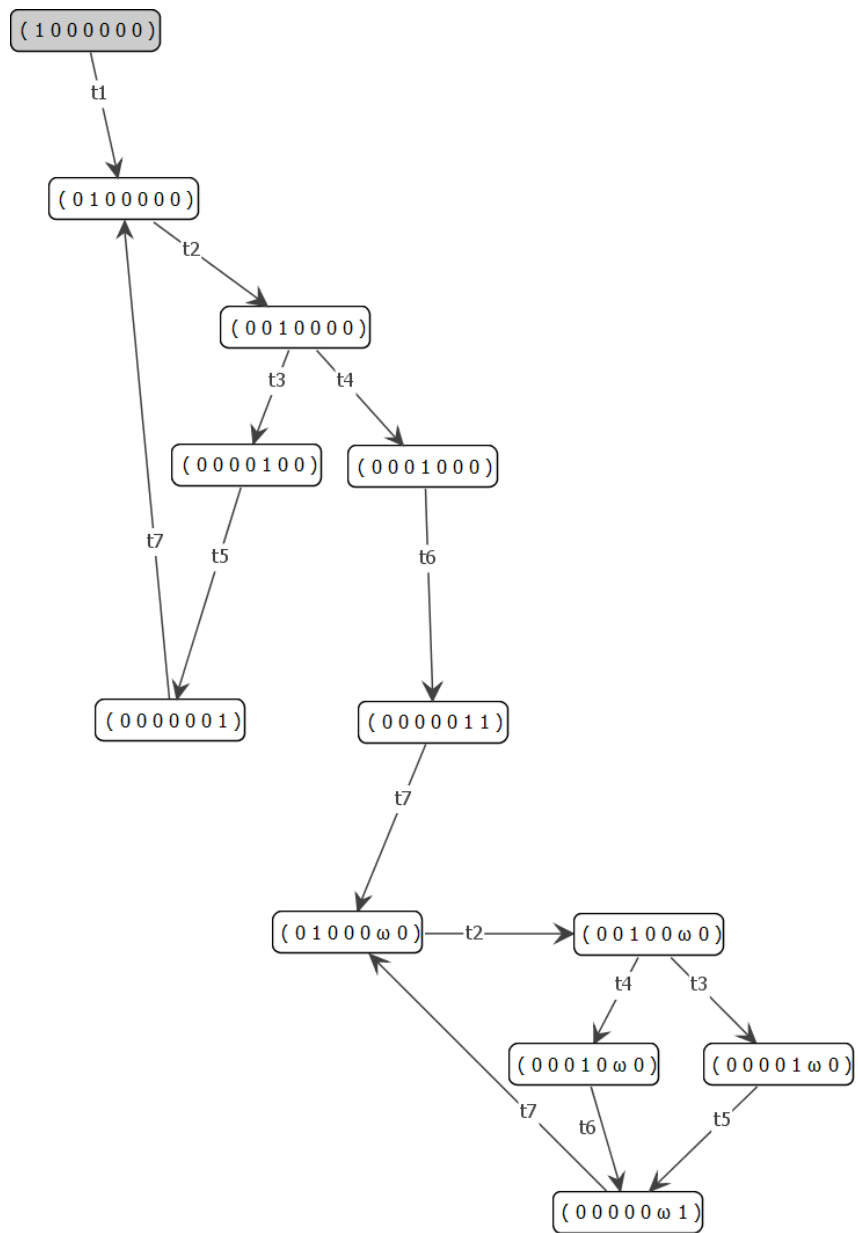
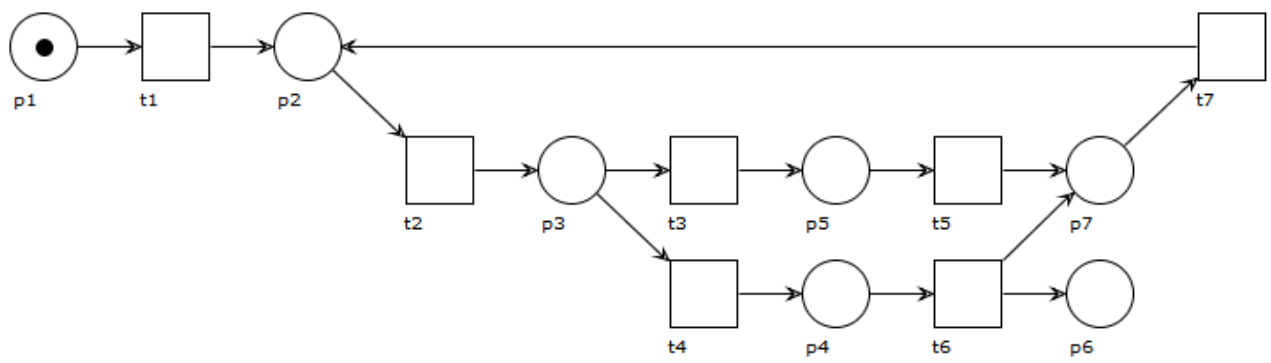


Abbildung 2.46: 274-B0-P05-Qxx-Kreis4-vor-und-nach-Omega.pnml

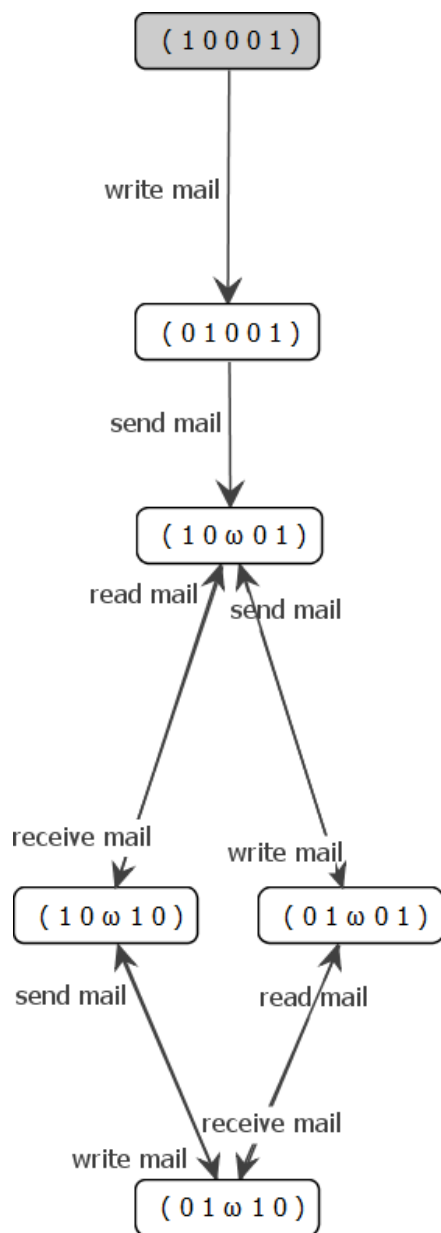
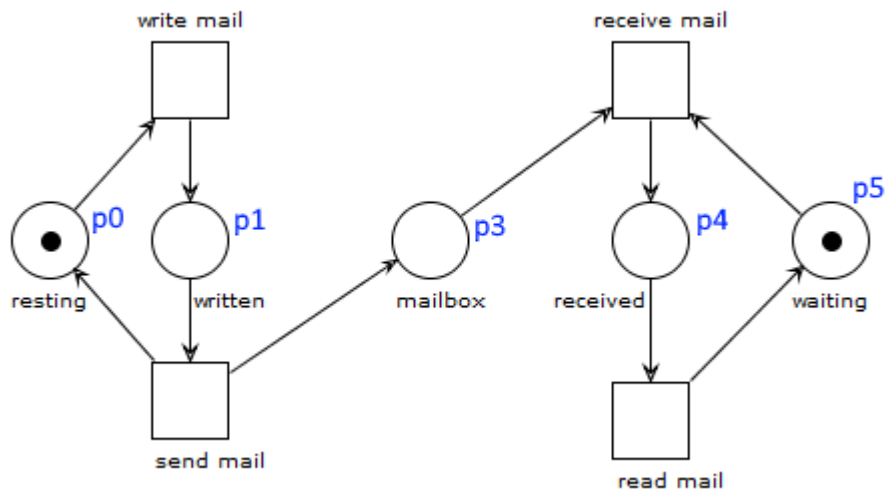


Abbildung 2.47: 275-B0-P02-Qxx-Mailbox-unbounded.pnml

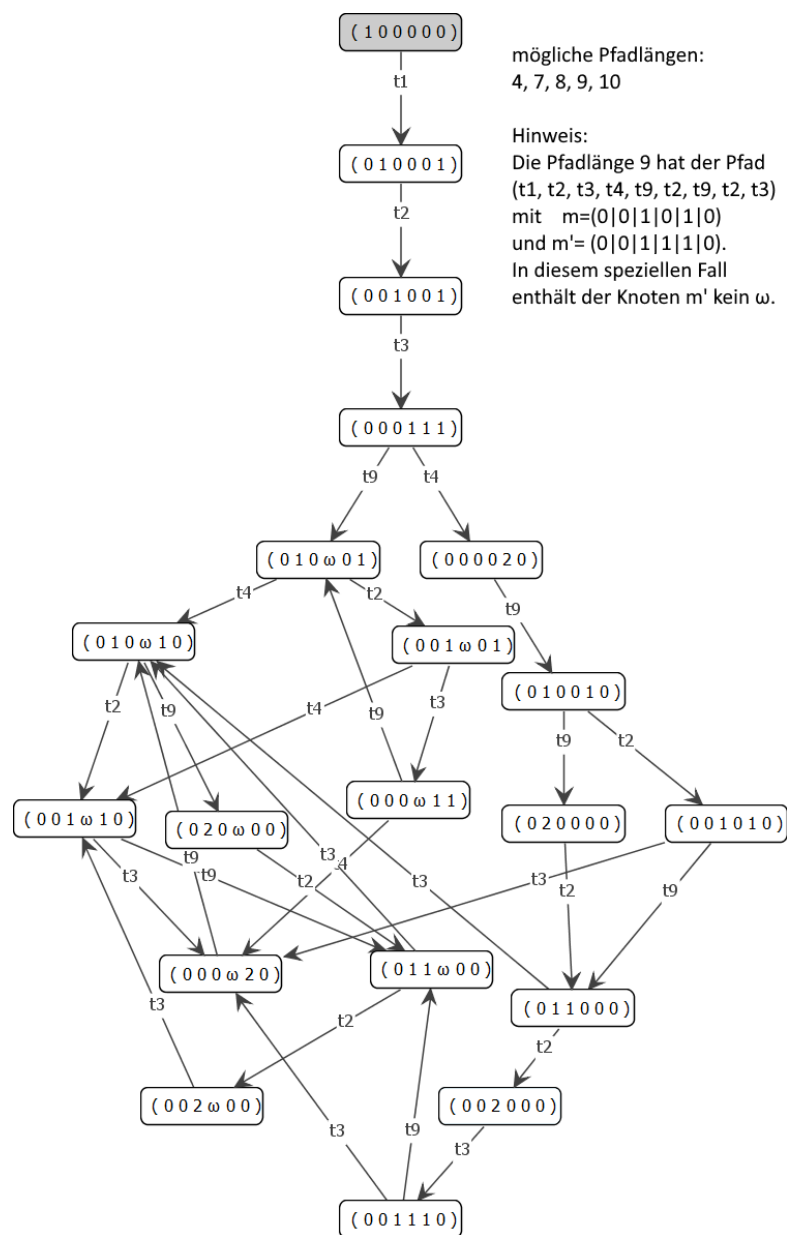
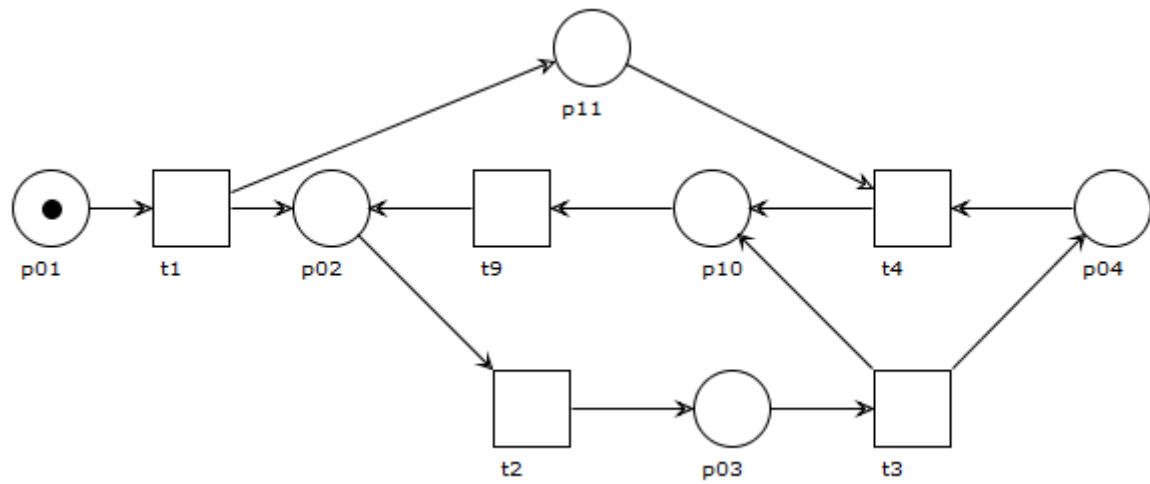


Abbildung 2.48: 276-B0-P04-Q10-mehrere-Pfade.pnml