Echtzeitsysteme 2

Eisenbahn

Daniel Brinkers, Markus Becher, Henry Schäfer

FAU Erlangen

2007-05-09 / Komposition

Gliederung

Gliederung

Probleme

- Keine Beachtung im Entwurf
- Jedes potentielle Byte muss gelesen werden
- Daher polling beim Senden unmöglich
- Sendevorgang laenger als Zeit für Task
- Nachrichten können in beliebiger Reihenfolge ankommen

Probleme

- Keine Beachtung im Entwurf
- Jedes potentielle Byte muss gelesen werden
- Daher polling beim Senden unmöglich
- Sendevorgang laenger als Zeit für Task
- Nachrichten können in beliebiger Reihenfolge ankommen

Probleme

- Keine Beachtung im Entwurf
- Jedes potentielle Byte muss gelesen werden
- Daher polling beim Senden unmöglich
- Sendevorgang laenger als Zeit für Task
- Nachrichten können in beliebiger Reihenfolge ankommen

Probleme

- Keine Beachtung im Entwurf
- Jedes potentielle Byte muss gelesen werden
- Daher polling beim Senden unmöglich
- Sendevorgang laenger als Zeit für Task
- Nachrichten können in beliebiger Reihenfolge ankommen

Probleme

- Keine Beachtung im Entwurf
- Jedes potentielle Byte muss gelesen werden
- Daher polling beim Senden unmöglich
- Sendevorgang laenger als Zeit für Task
- Nachrichten können in beliebiger Reihenfolge ankommen

Redesign der Kommunikationsklasse

- Senden und Empfangen in eigenem Task
- Send-Funktion kehrt sofort zurück
- Gesendet wird erst später
- Dafuer Platz in Schedule-Table erforderlich
- Empfangene Bytes werden gespeichert
- In anderen Tasks verwendet

Redesign der Kommunikationsklasse

- Senden und Empfangen in eigenem Task
- Send-Funktion kehrt sofort zurück
- Gesendet wird erst später
- Dafuer Platz in Schedule-Table erforderlich
- Empfangene Bytes werden gespeichert
- In anderen Tasks verwendet

Redesign der Kommunikationsklasse

- Senden und Empfangen in eigenem Task
- Send-Funktion kehrt sofort zurück
- Gesendet wird erst später
- Dafuer Platz in Schedule-Table erforderlich
- Empfangene Bytes werden gespeichert
- In anderen Tasks verwendet

Redesign der Kommunikationsklasse

- Senden und Empfangen in eigenem Task
- Send-Funktion kehrt sofort zurück
- Gesendet wird erst später
- Dafuer Platz in Schedule-Table erforderlich
- Empfangene Bytes werden gespeichert
- In anderen Tasks verwendet

Redesign der Kommunikationsklasse

- Senden und Empfangen in eigenem Task
- Send-Funktion kehrt sofort zurück
- Gesendet wird erst später
- Dafuer Platz in Schedule-Table erforderlich
- Empfangene Bytes werden gespeichert
- In anderen Tasks verwendet

Redesign der Kommunikationsklasse

- Senden und Empfangen in eigenem Task
- Send-Funktion kehrt sofort zurück
- Gesendet wird erst später
- Dafuer Platz in Schedule-Table erforderlich
- Empfangene Bytes werden gespeichert
- In anderen Tasks verwendet

Weitere Folgen

- Auswirkungen auf alle Module
- Zugklasse will variable Anzahl von Weichen gleichzeitig stellen
- Stellen einer Weiche erfordert Empfangen
- Eigener Task zum Stellen der Weichen

Weitere Folgen

- Auswirkungen auf alle Module
- Zugklasse will variable Anzahl von Weichen gleichzeitig stellen
- Stellen einer Weiche erfordert Empfangen
- Eigener Task zum Stellen der Weichen

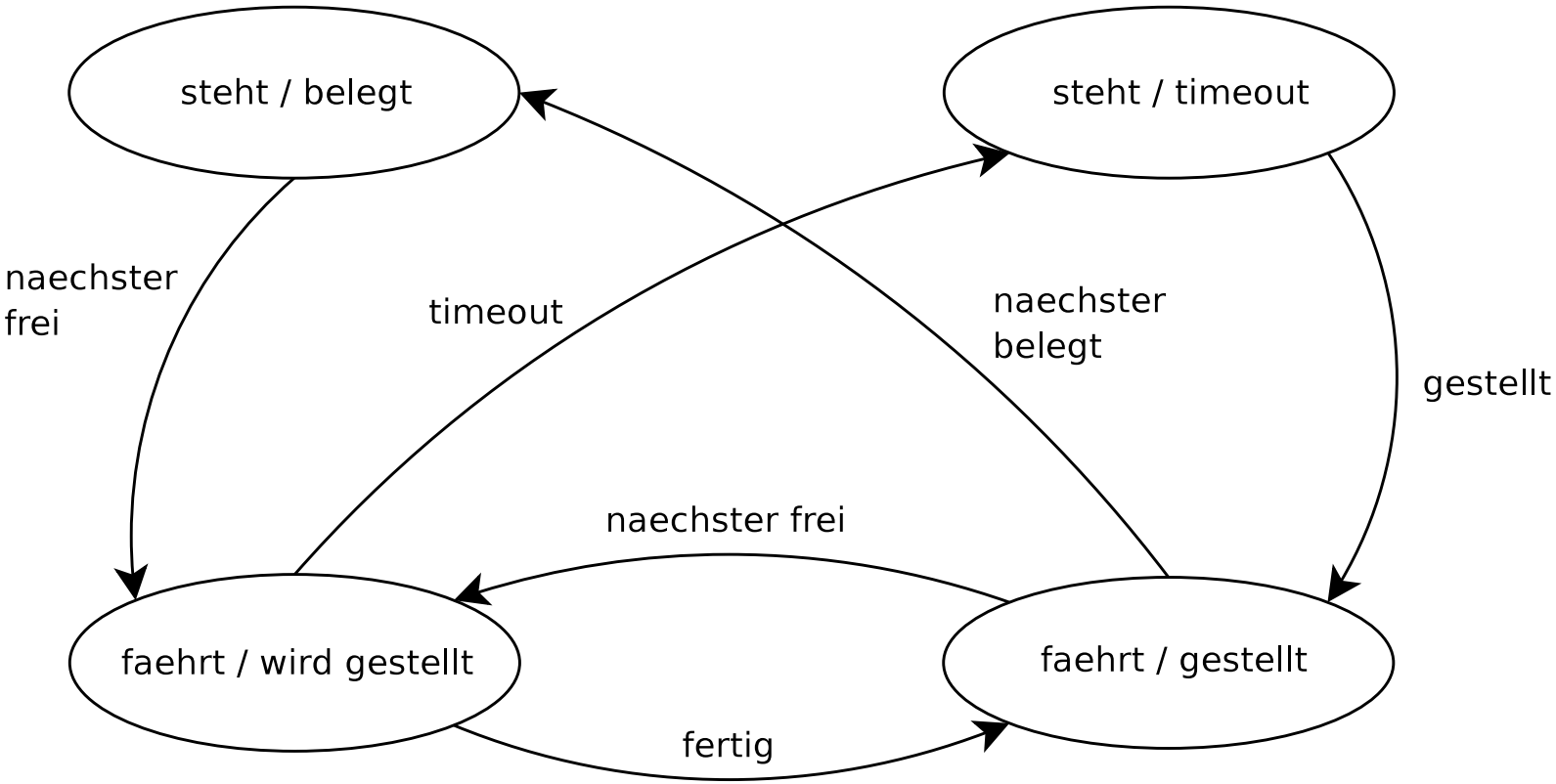
Weitere Folgen

- Auswirkungen auf alle Module
- Zugklasse will variable Anzahl von Weichen gleichzeitig stellen
- Stellen einer Weiche erfordert Empfangen
- Eigener Task zum Stellen der Weichen

Weitere Folgen

- Auswirkungen auf alle Module
- Zugklasse will variable Anzahl von Weichen gleichzeitig stellen
- Stellen einer Weiche erfordert Empfangen
- Eigener Task zum Stellen der Weichen

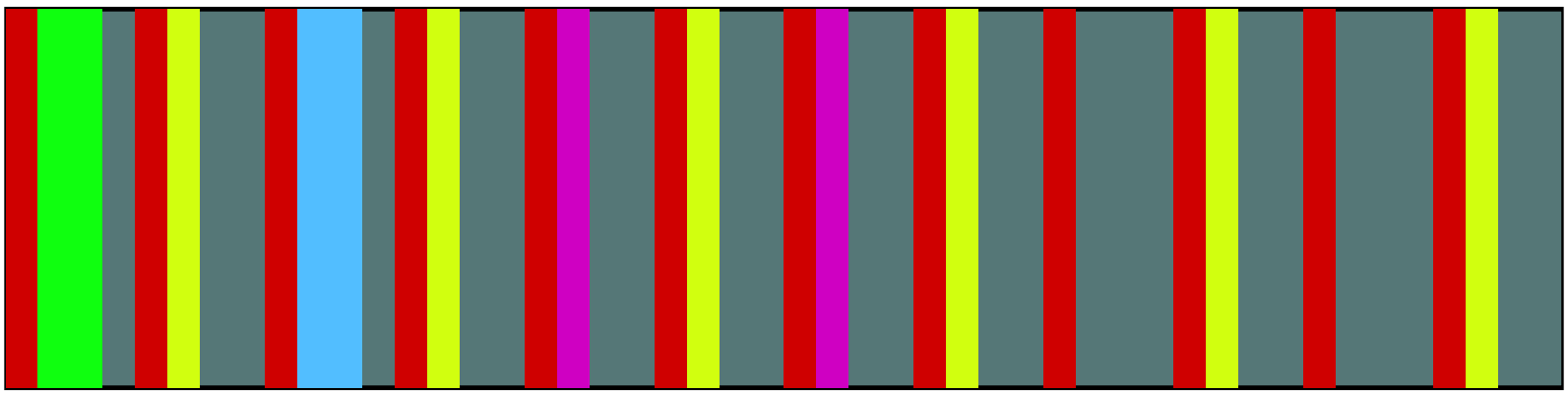
Zustandsdiagramm Train



Gliederung

Timetable

- Aufgrund der Vorüberlegung und Ihrer Probleme verworfen
- Nun Top-Down statt Buttom-Up :-)



Timetable

- Aufgrund der Vorüberlegung und Ihrer Probleme verworfen
- Nun Top-Down statt Buttom-Up :-)