

Seminar

Werkzeuggestützte Modellierung eines Tamagotchi

Tamagotchi in ObjectGEODE



1. Februar 1999

Jürgen Jeitner

Bernd Müller

Alexander Wisspeintner

Lehrstuhl für Informatik

Technische Universität München



Fraunhofer
Einrichtung
Experimentelles
Software Engineering



ObjectGEODE

- Kommerzielles CASE-Tool von der Firma Verilog France
- Beschreibungstechniken
 - OMT - Object Modelling Technique von Rumbaugh
 - SDL92 - Specification and Description Language
 - MSC92 - Message Sequence Charts
- Hierarchische Projektverwaltung
 - Diagramme der Beschreibungstechniken
 - Dokumente aus anderen Programmen
 - Mehrere Benutzer können an einem Projekt arbeiten
 - Versionskontrolle
- Graphische Editoren

ObjectGEODE

- Abbildung von OMT Klassendiagrammen und Zustandsmaschinen nach SDL
- Simulation von SDL Spezifikationen
- Automatische Generierung von MSCs
- Code Generierung für verschiedene Zielplattformen

SDL - Specification and Description Language

- Graphische Beschreibungssprache
- Ursprünglich zur Beschreibung von Telekommunikationssystemen
- Spezifikation von Kommunikationsprotokollen
- Erste Version von SDL 1970
- Standardisiert durch ITU (International Telecommunication Union) 1976
- Folgestandards SDL80, SDL84, SDL88, SDL92, SDL96
- Beschreibung von reaktiven verteilten Systemen

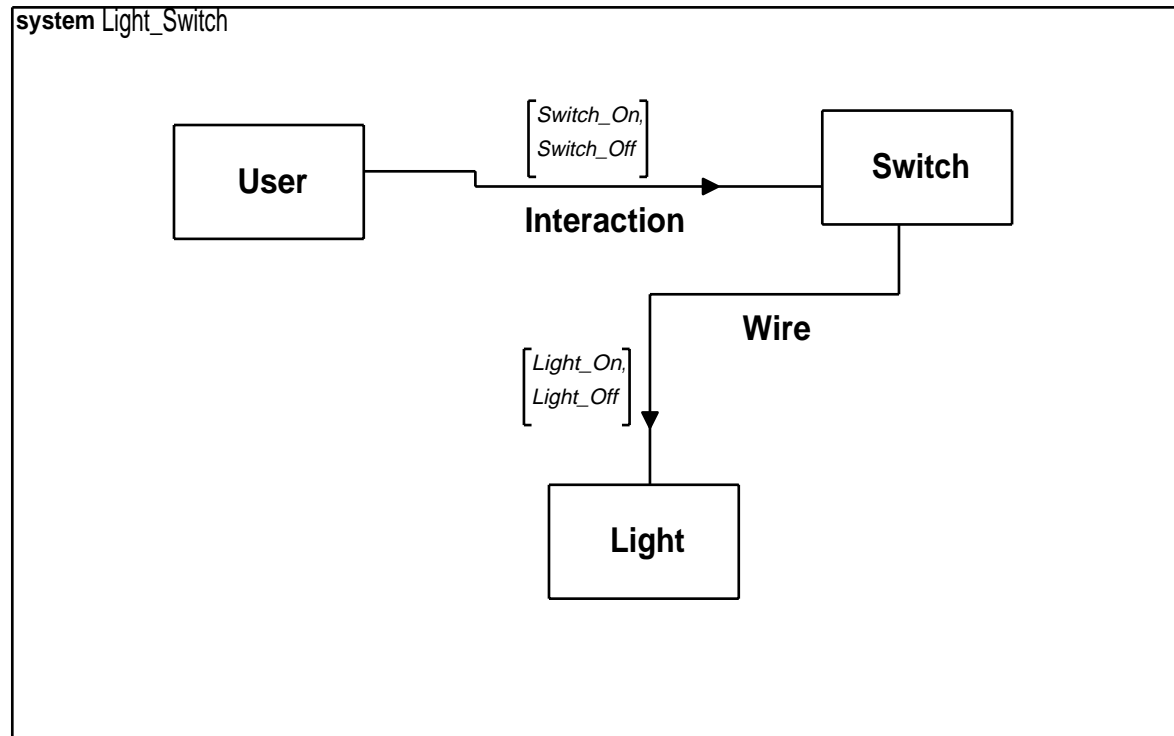
SDL96 Features

- Pufferung der Kommunikationssignale in Priority-Queues
- Mehrere Signale können über einen Kanal versendet werden
- Zusammenfassen von mehreren Signalen in Signallisten
- Abstrakte Datentypen (ADT)
- Unterstützung interner Timer
- Objektorientierter Systementwurf
- Dynamische Instanzerzeugung von Prozessen zur Laufzeit

Konzept

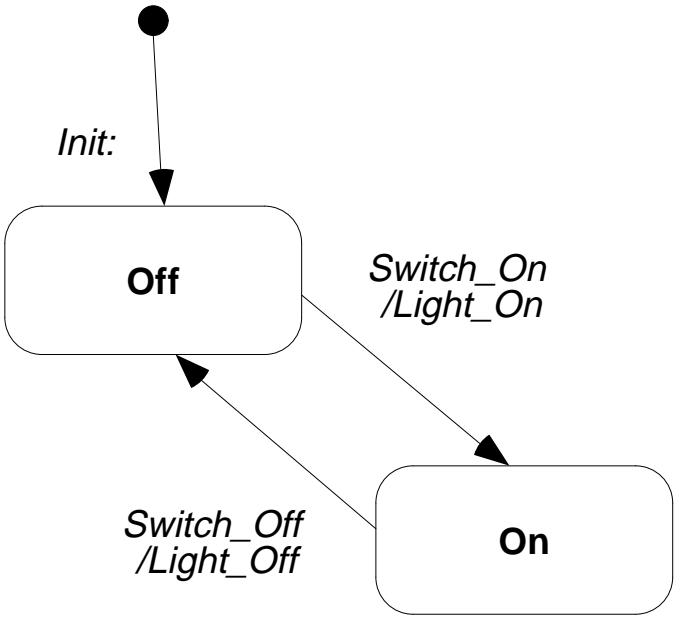
- Sichten
 - Struktur
 - Funktionalität
- Struktur
 - Kommunikation der Komponenten mit Signalen über Kanäle
 - Komponente kann aus weiteren Unterkomponenten bestehen
 - Durch hierarchische Anordnung beliebig viele Abstraktionsebenen
- Funktionalität
 - Verhalten einer Komponente wird durch eine Zustandsmaschine beschrieben
 - Abstraktion durch Unterfunktionen
 - Unterautomaten werden nicht unterstützt

SDL Strukturdiagramm

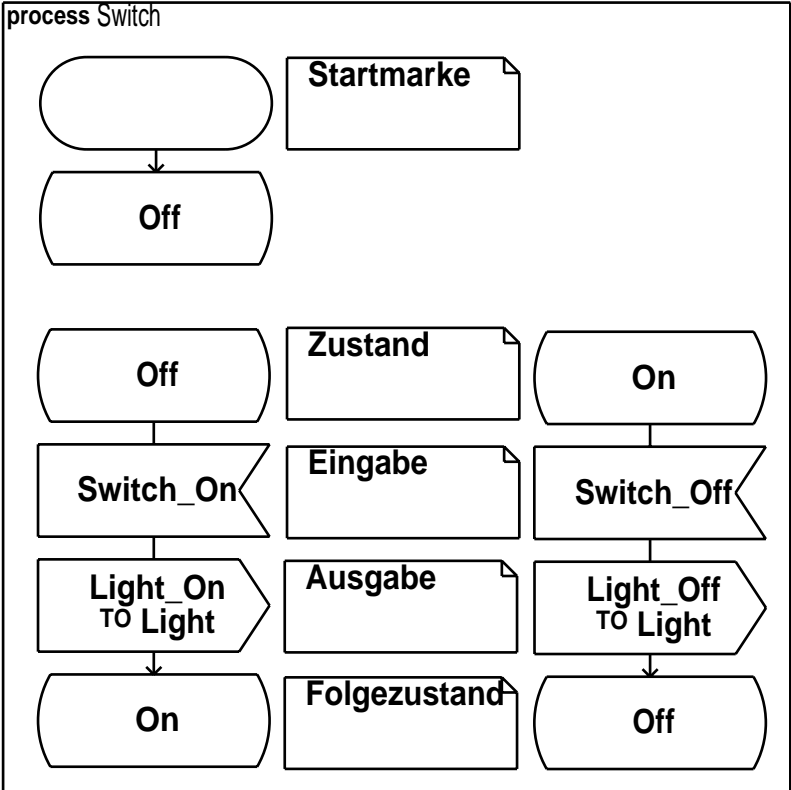


Endliche Zustandsmaschine (FSM)

OMT State Machine



SDL State Machine

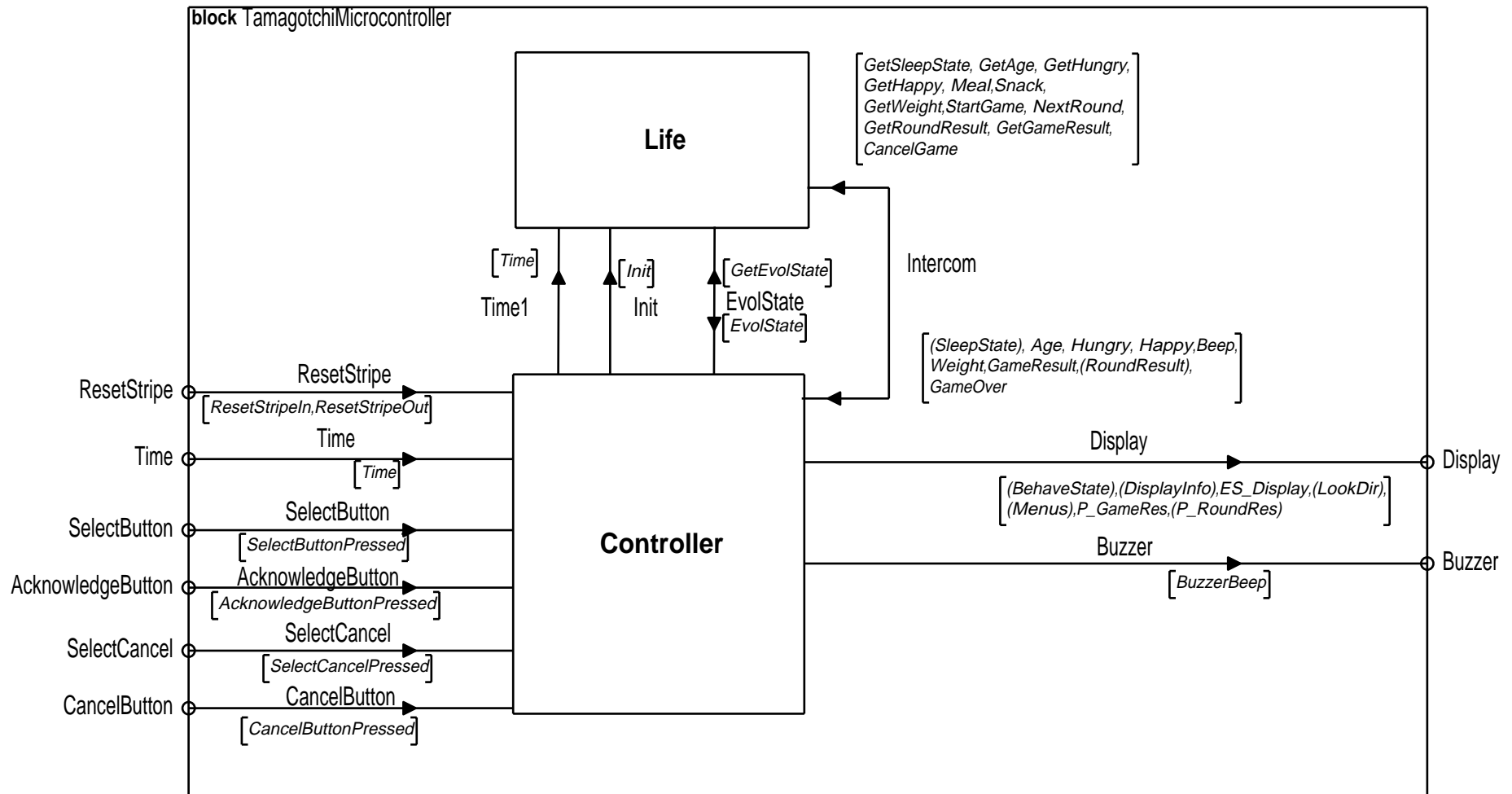


Modellierung

- Vorgehensweise
 - Schnittstelle zur Umwelt
 - Struktur Top-Down festgelegt
 - Verhalten der einzelnen Komponenten auf unterster Ebene beschrieben
- Alle Anforderungen, soweit eindeutig, modelliert
- Spezifikation getestet bis zum Entwicklungsstadium “Teenager”
- Simulation während der Modellierung nicht möglich
- Aufteilung der Arbeiten im Team nicht möglich - nur eine Lizenz vorhanden
- Umfang der Spezifikation
 - Zeitaufwand Einarbeitung ca. 25 Stunden
 - Zeitaufwand Tamagotchi ca. 70 Stunden
 - Tamagotchi SDL Diagramme benötigen 136 Din A4 Seiten

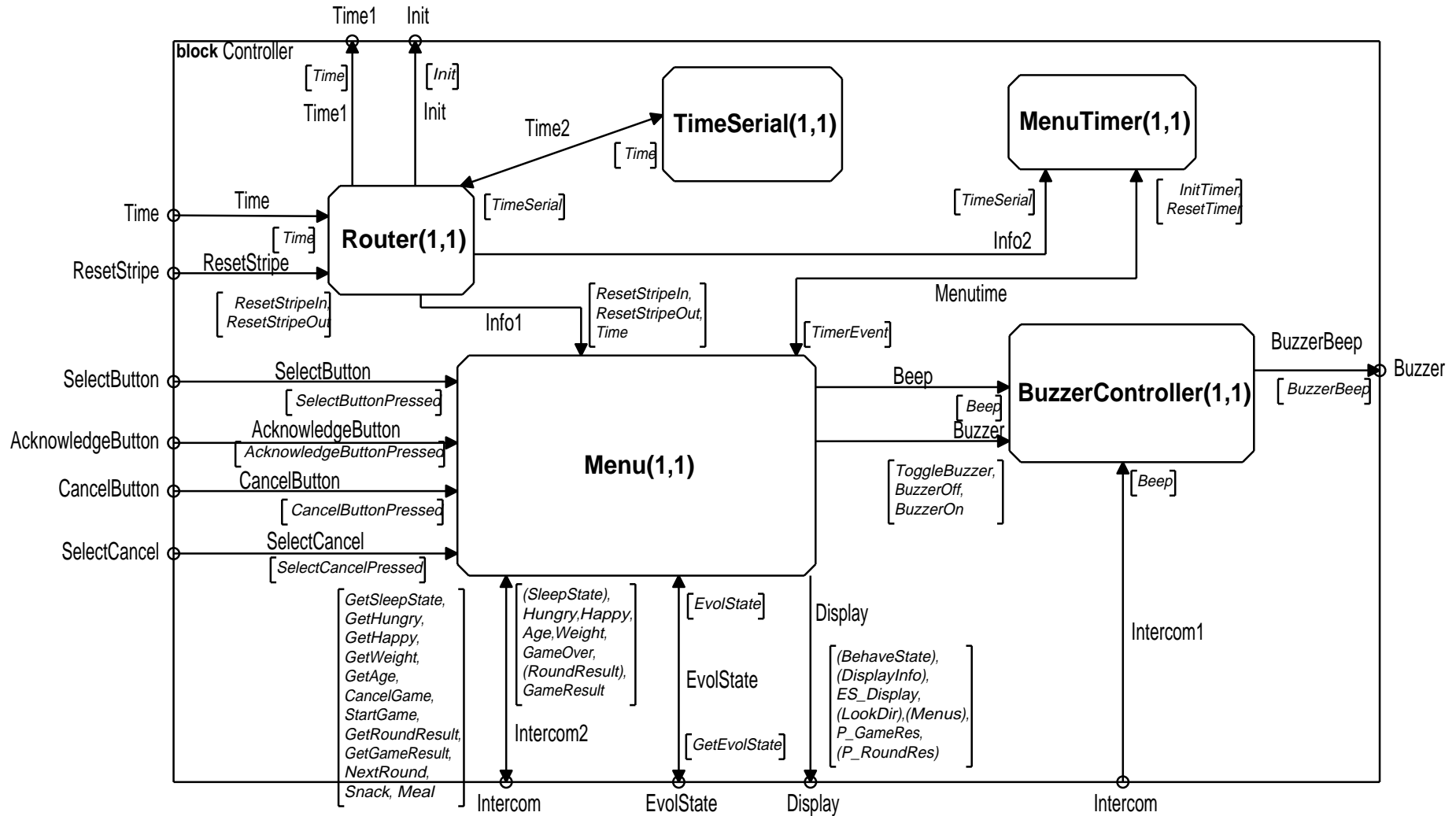
4. Spezifikation

Strukturdiagramm der Komponente "Tamagotchi"



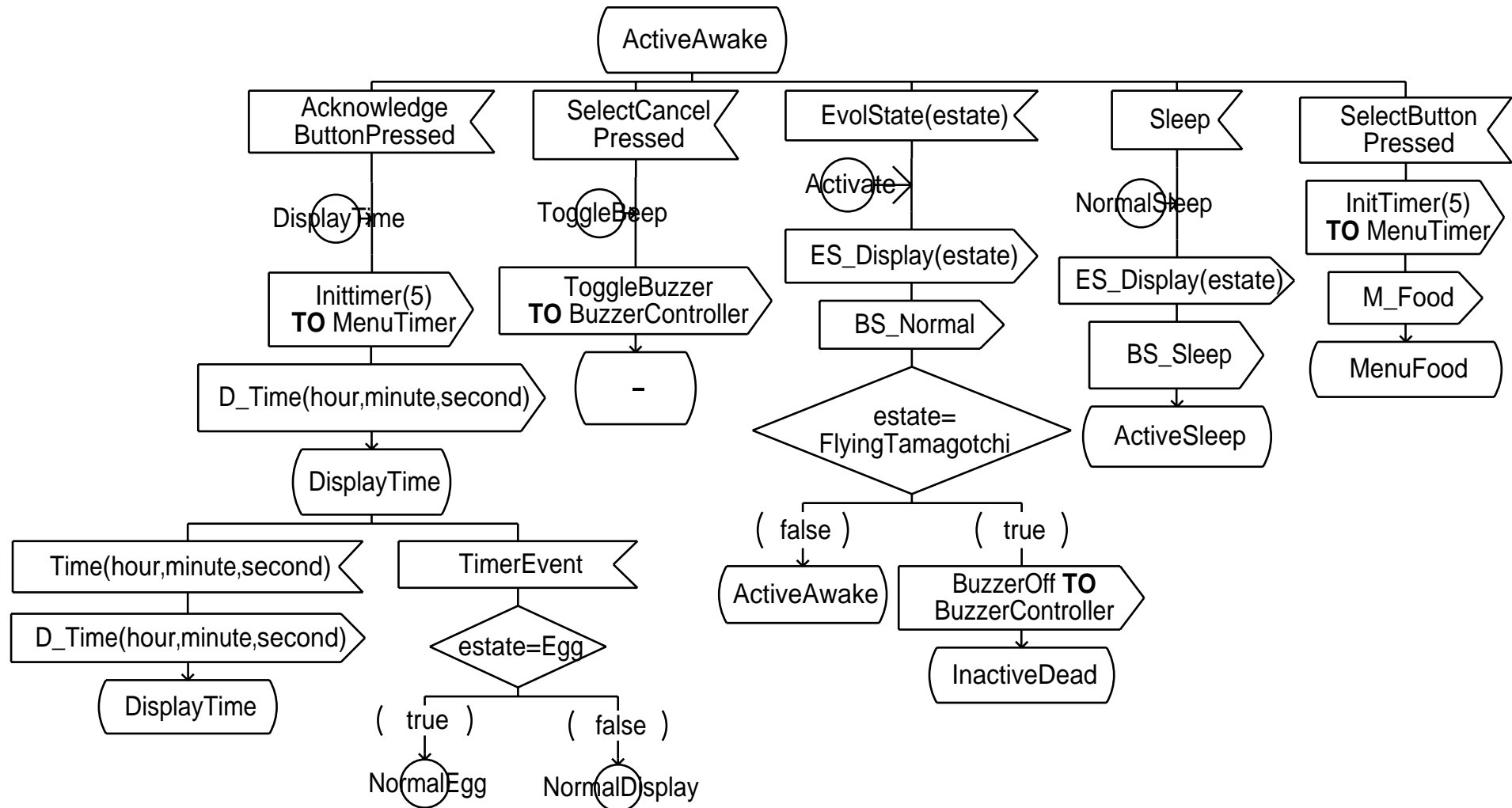
4. Spezifikation

Strukturdiagramm der Komponente "Controller"



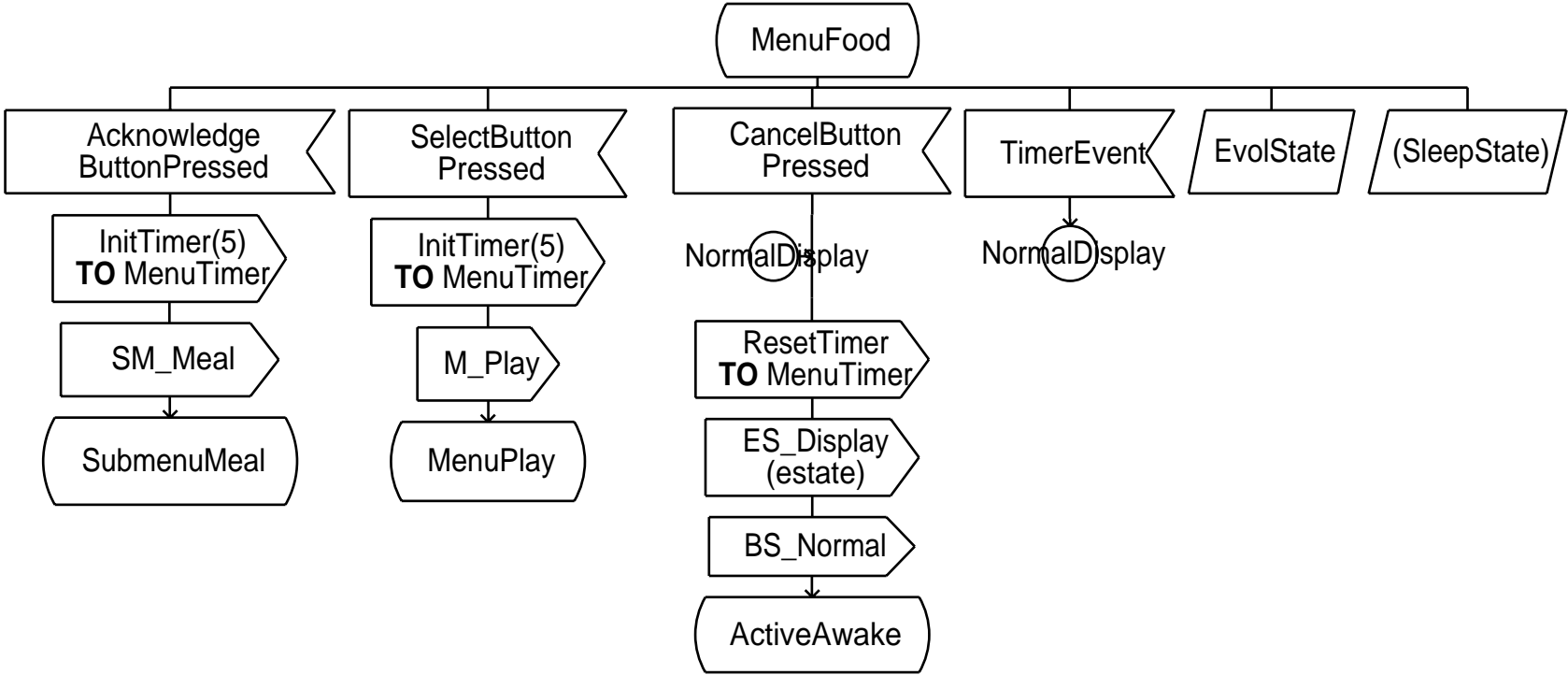
4. Spezifikation

FSM der Komponente "Menu" - Zustand "Active Awake"

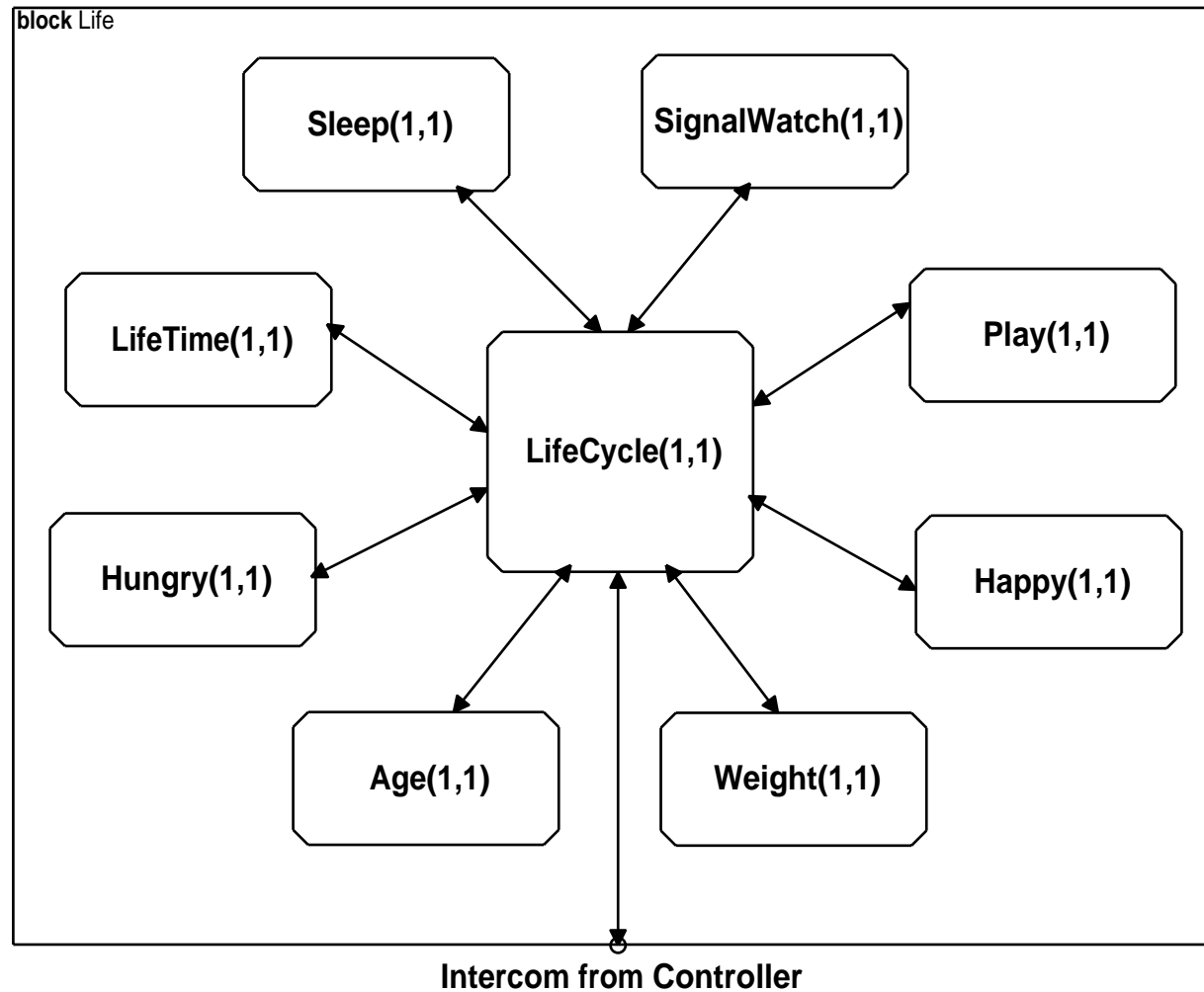


4. Spezifikation

FSM der Komponente "Menu" - Zustand "Menu Food"

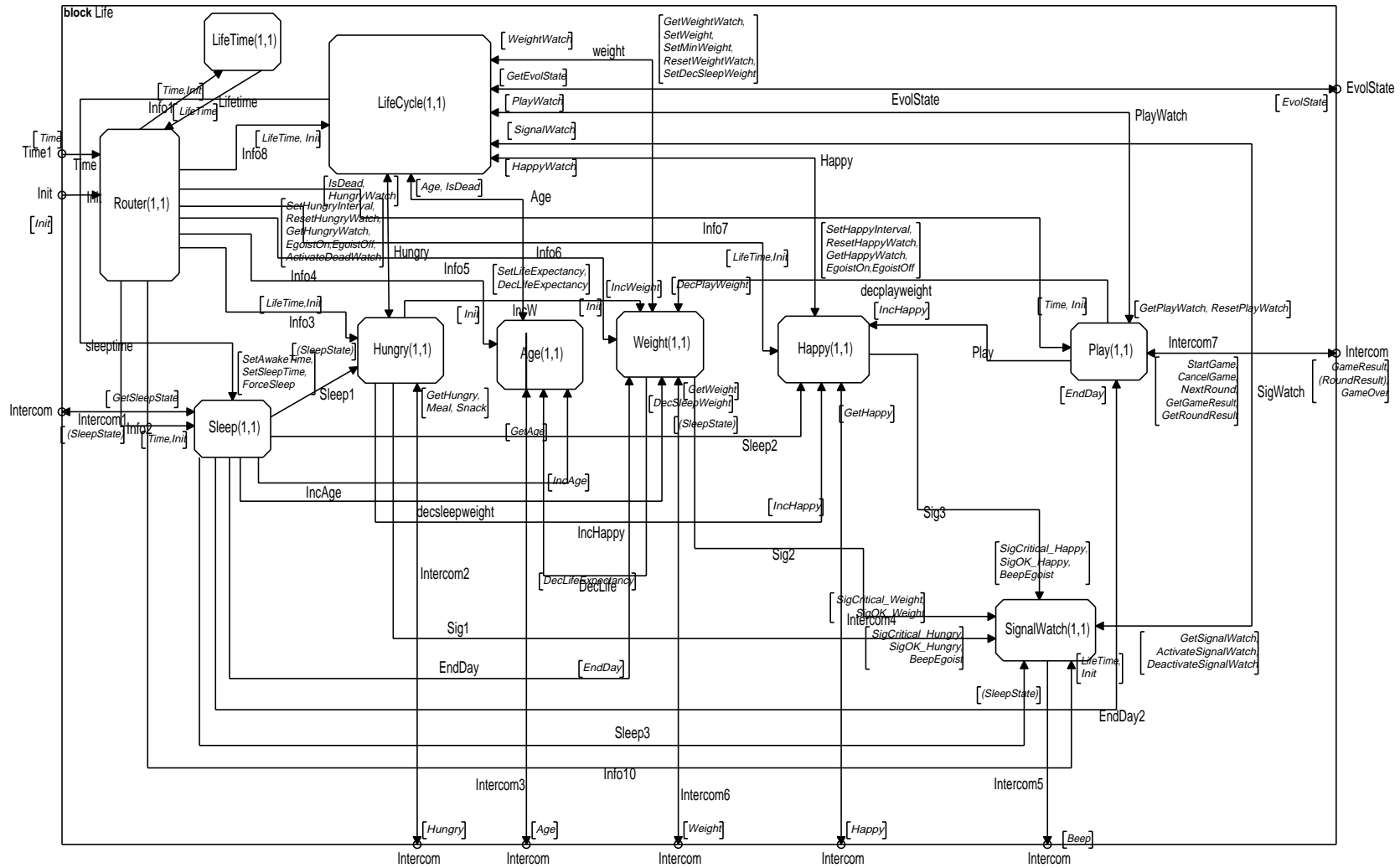


Vereinfachtes Strukturdiagramm von "Life"



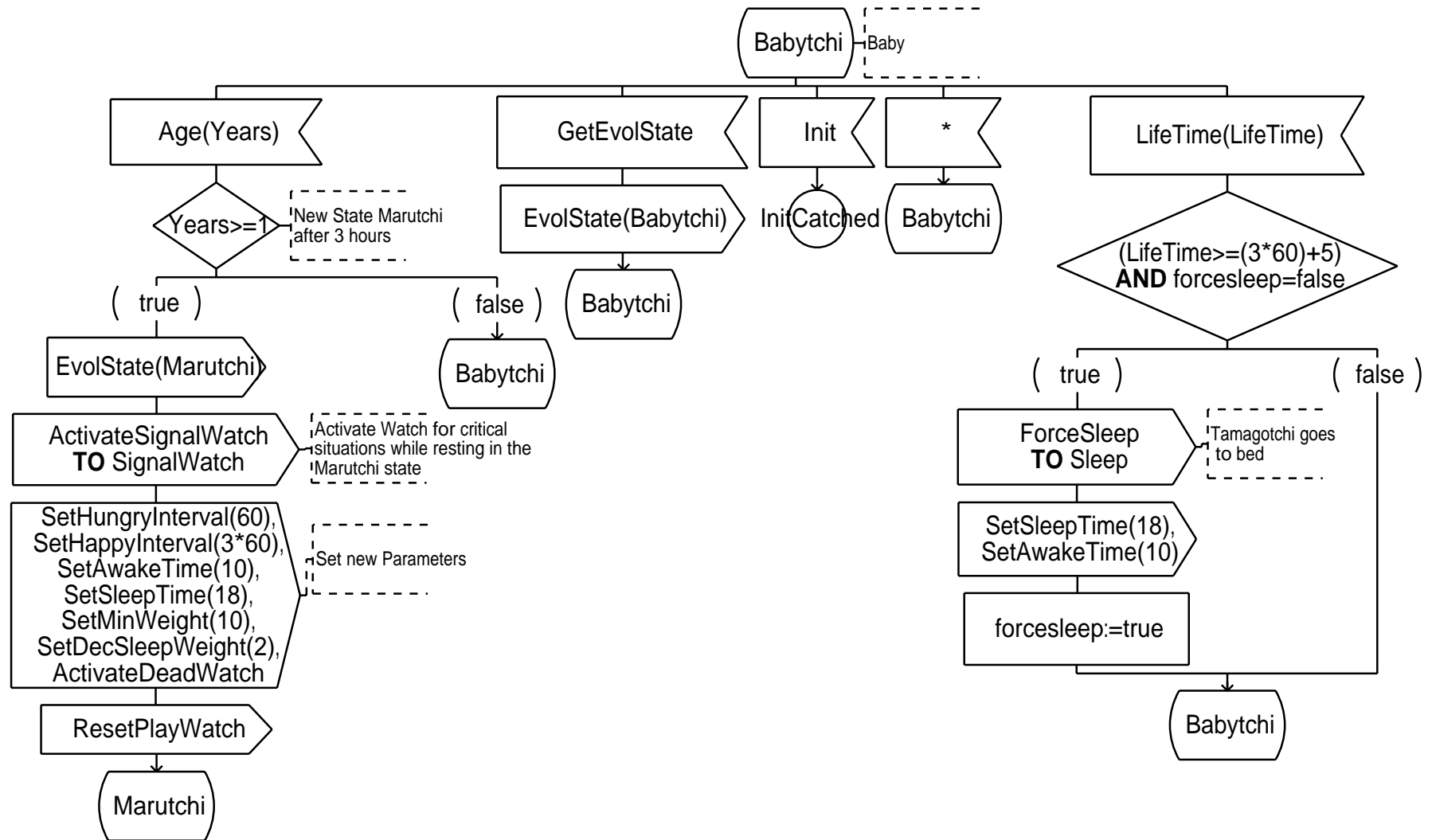
4. Spezifikation

Strukturdiagramm der Komponente "Life"

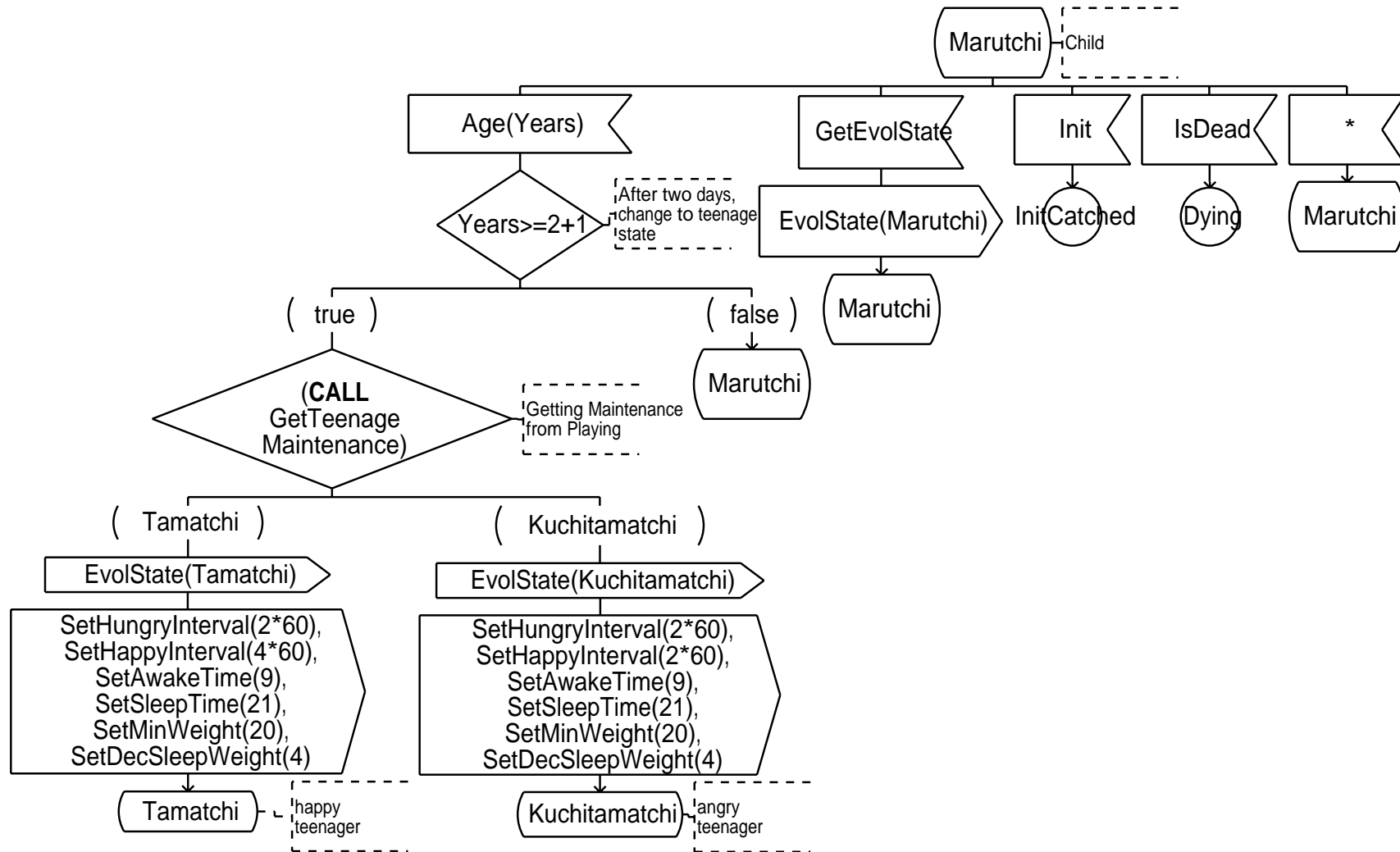


4. Spezifikation

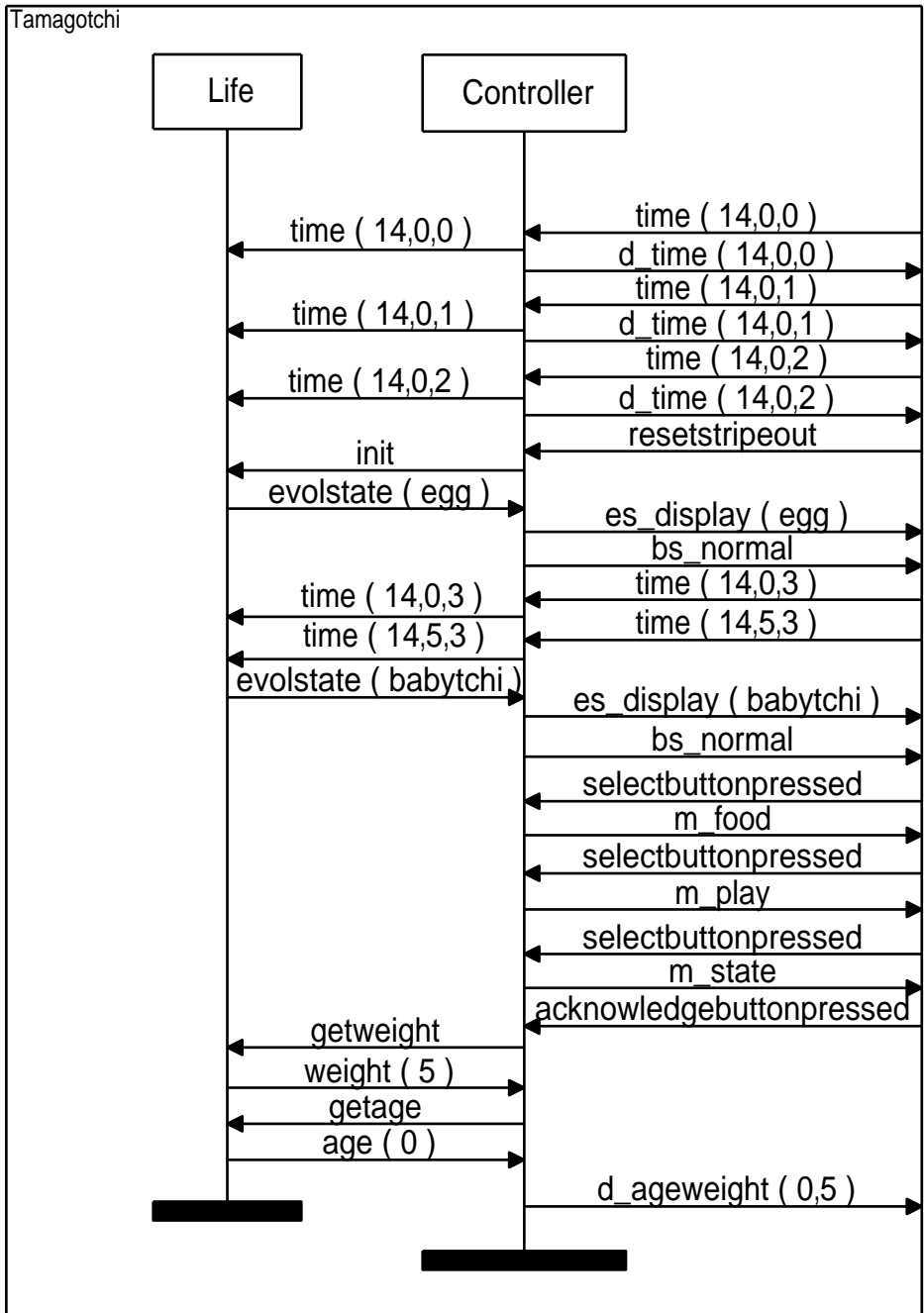
FSM der Komponente "Life Cycle" - Zustand "Babytchi"



FSM der Komponente "Life Cycle" - Zustand "Marutchi"



MSC der "Tamagotchi" Komponente



Erfahrungen mit SDL

- Positiv
 - Interne Timer nützlich
 - Durch Signalpufferung weniger Transitionen in den FSMs
 - Spezifiziertes Verhalten in der graphischen Notation sichtbar
 - Spezielle Transitionen für alle Zustände eines FSM gültig
- Negativ
 - Keine Wiedereinreihung eingelesener Signale in die Signalwarteschlange
 - Nur “Punkt zu Punkt” Kommunikation - Kein Bus
 - Unterautomaten werden nicht unterstützt
 - Platzbedarf der Notation

Erfahrungen mit ObjectGEODE

- Positiv
 - Spezifikation von komplexeren eingebetteten Systemen mit akzeptablen Zeitaufwand möglich
 - Schnittstelle des Simulators zu anderen Programmen
 - Anpassung des Simulators an spezielle Anforderungen
- Negativ
 - Schlechte Dokumentation
 - Benutzeroberflächen der Editoren nicht intuitiv und z.T. fehlerhaft
 - Keine Ablaufsteuerung der Simulation durch MSCs möglich
 - Exportfunktionen für Diagramme nicht ausreichend
 - Keine Berücksichtigung von Echtzeit während der Simulation
 - Codegenerierung und UNIX-Runtime-Bibliothek fehlerhaft
 - Konsistenz-Check unzureichend
 - “Cut and Paste” Funktionen nicht ausgereift

Anmerkungen

- Modellierung des Tamagotchi in SDL sinnvoll
- Gutes Zeichenprogramm (z.B. Visio) für SDL Diagramme besser geeignet
- Simulator rechtfertigt den Einsatz von ObjectGEODE

- Wir würden das Tool zur Spezifikation von Systemen wieder verwenden
 - Praxisorientierter Ansatz von SDL
 - Trotz der vorhandenen Mängel besser als andere CASE-Tools (z.B. PEP, AutoFocus)
 - Durch Simulation und Codegenerierung schnelle Erstellung von Prototypen