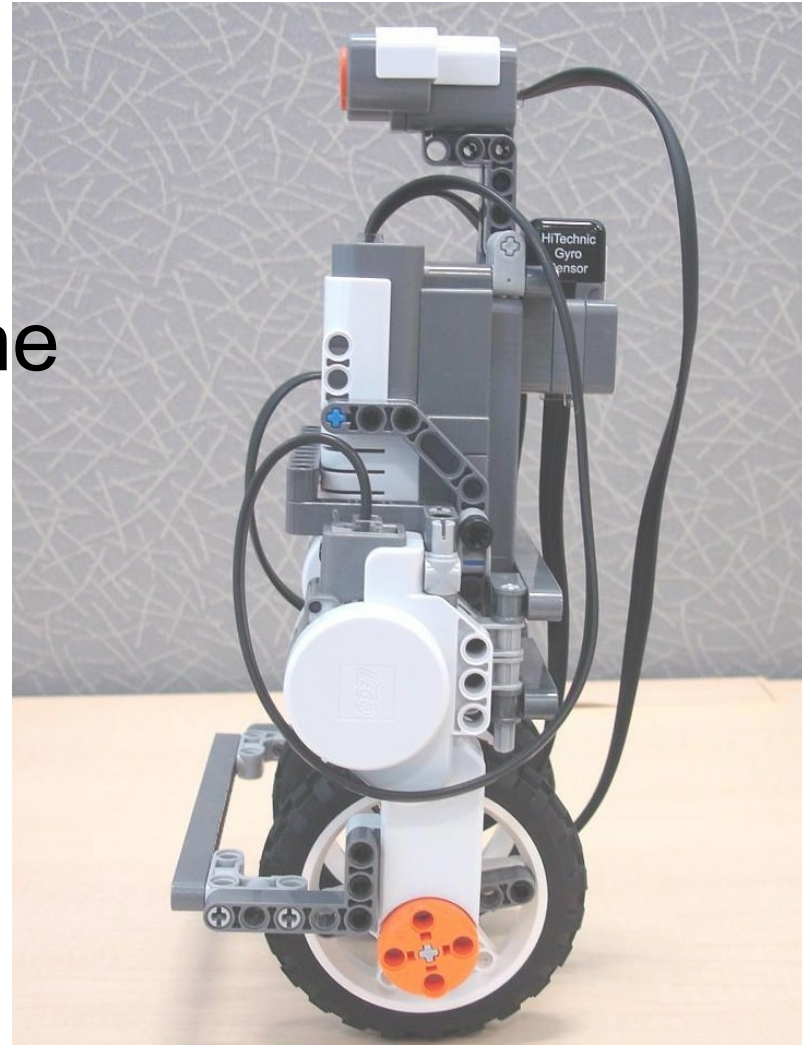
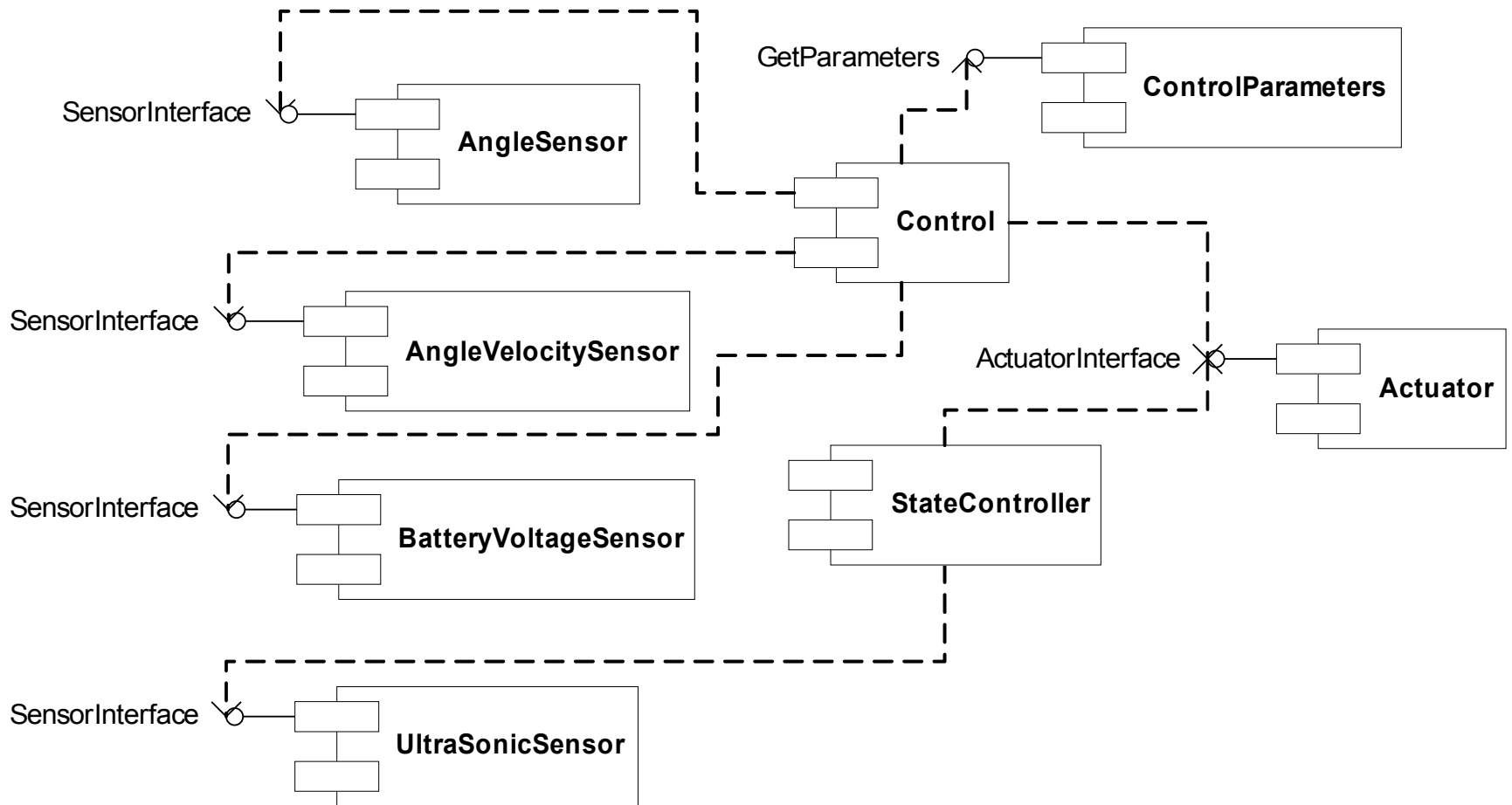


Phase 2: Komponenten und Schnittstellen

NXT Standalone



Komponentenansicht



Actuator

class Actuator

- private:
 - Motor *m_pMotorA
 - Motor *m_pMotorB
- public:
 - Actuator(Motor &a_motorA, Motor &a_motorB)
 - Drive(bool a_bForward)
 - Stop()
 - Turn(bool a_bLeft)

Control

class Control

- private:
 - ControllerParameters m_CP
 - Actuator *m_pActuator
- public:
 - Control(ControllerParameters &a_Cp, Actuator &a_Actuator)
 - void Perform(S16 DesiredAngle, S16 a_Angle, S16 a_AngleVelocity, S16 a_BatVoltmv)

ControlParameters

class ControlParameters

- private:
 - S32 m_P
 - S32 m_I
 - S32 m_D
- public:
 - ControlParameters(S32 a_P, S32 a_I, S32 a_D)

StateController

class StateController

- private:
 - Actuator *m_pActuator
 - State m_State
- public:
 - StateController()
 - void Handle()

AngleSensor

class AngleSensor

- public:
 - AngleSensor(U16 a_Period)
 - void Calibrate()
 - void Sample()
 - S16 GetValue()

AngleVelocitySensor

class AngleVelocitySensor

- public:
 - AngleVelocitySensor(U16 a_Period)
 - void Sample()
 - S16 GetValue()

BatteryVoltageSensor

class BatteryVoltageSensor

- public:
 - BatteryVoltageSensor()
 - void Sample()
 - S16 GetValue()

UltraSonicSensor

class UltraSonicSensor

- public:
 - UltraSonicSensor()
 - void Sample()
 - S16 GetValue()