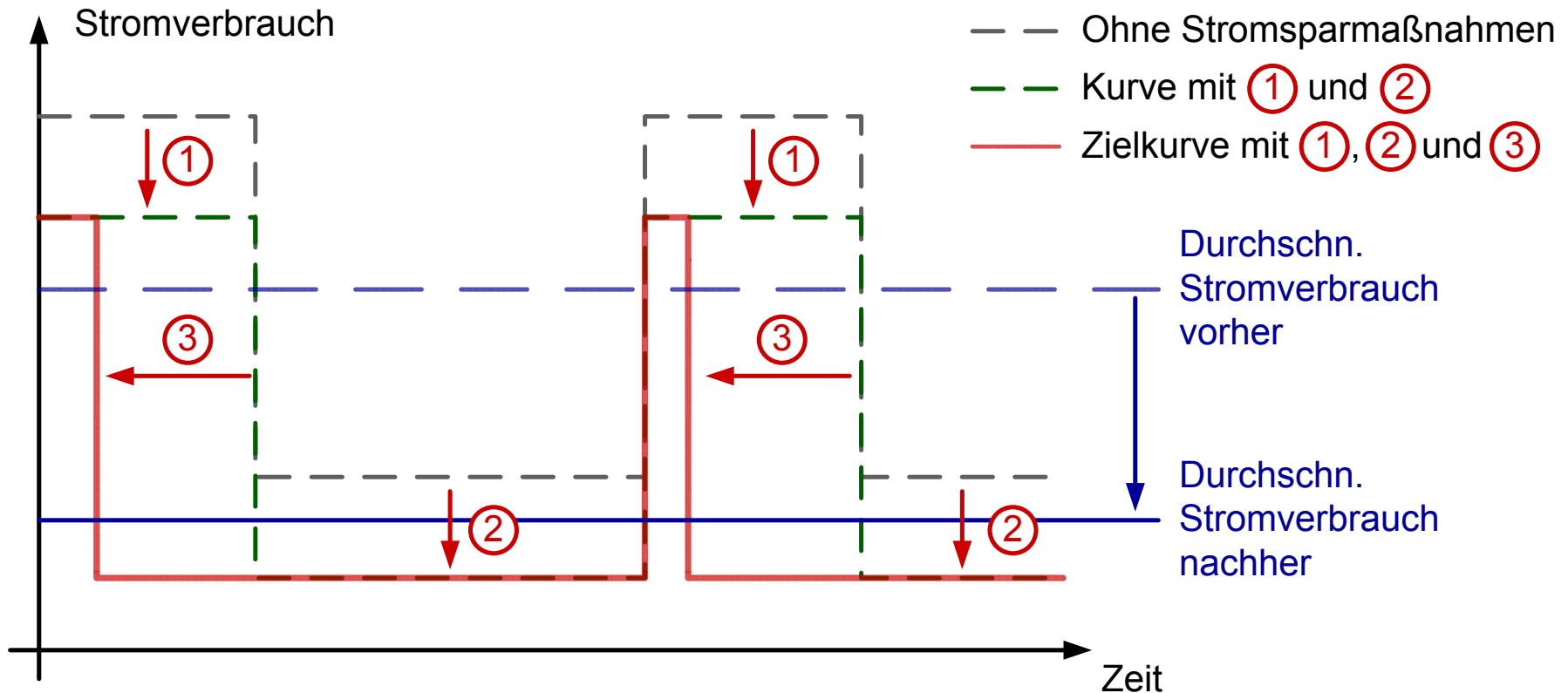




- | | | |
|-----|--------------------------------|--------------------|
| 1. | Einführung | - Prof. Zimmermann |
| 2. | Aspekte des Systementwurfs | - Prof. Zimmermann |
| 3. | Modellbasierter Entwurf | - Prof. Zimmermann |
| 4. | Echtzeitsysteme | - Prof. Zimmermann |
| 5. | Scheduling | - Prof. Zimmermann |
| 6. | Sicherheit und Zuverlässigkeit | - Prof. Zimmermann |
| 7. | Softwaretechnische Aspekte | - Prof. Fengler |
| 8. | Hardware-Software-Codesign | - Prof. Fengler |
| 9. | Rechnerarchitektur | - Prof. Fengler, |
| 10. | Kommunikation | - Prof. Fengler |
| 11. | Energieeffizienz | - Prof. Fengler |
| 12. | Domäne Automotive | - Prof. Fengler |

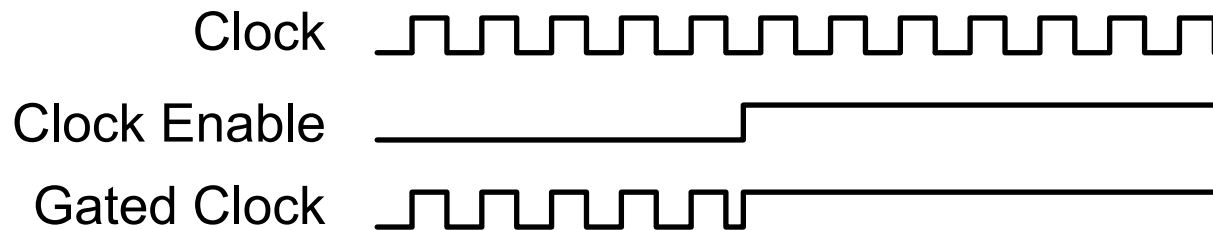
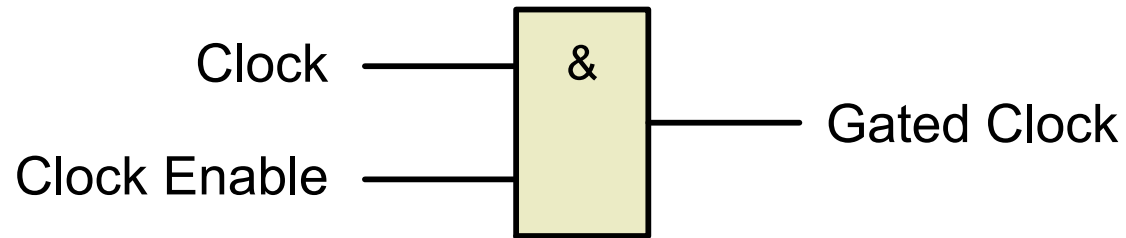
Verwendung nur als Lehrmaterial. Nicht zur Veröffentlichung!

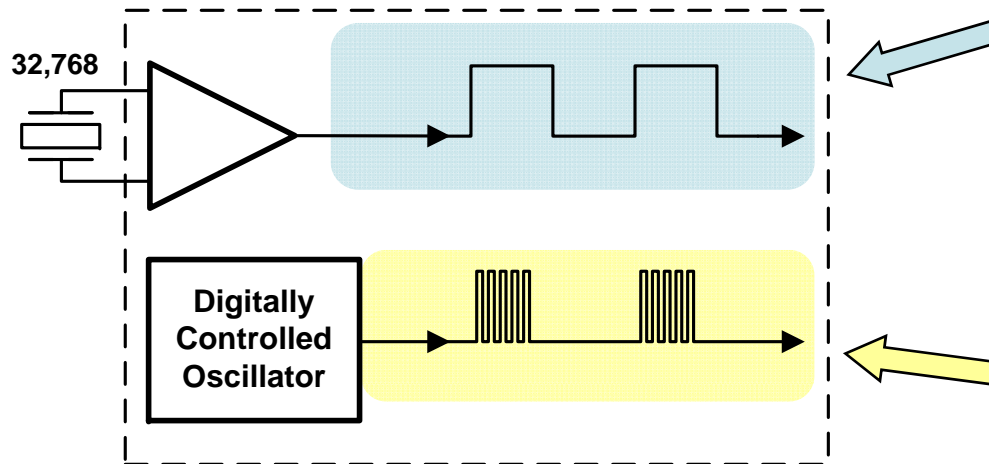


- ① Reduktion des Stromverbrauchs der Aktiv-Zeit, z.B. durch Clock Gating (s. Folie ?)
- ② Reduktion des Stromverbrauchs des Sleep-Modus, z.B. durch flexible, baugruppenspezifische, langsame Takte (s. Folie ?)
- ③ Verkürzung der Aktiv-Zeit, z.B. durch effiziente Programmierung, Interrupts, Einsatz von hochfunktionalen Peripheriebausteinen



Bedarfsgerechtes baugruppenspezifisches
Zu- und Abschalten des Systemtaktes (Clock)





Langsamer 32kHz Takt
 → geringe Stromaufnahme
 → Möglichkeit die CPU aufzuwecken (Timer, RTC)

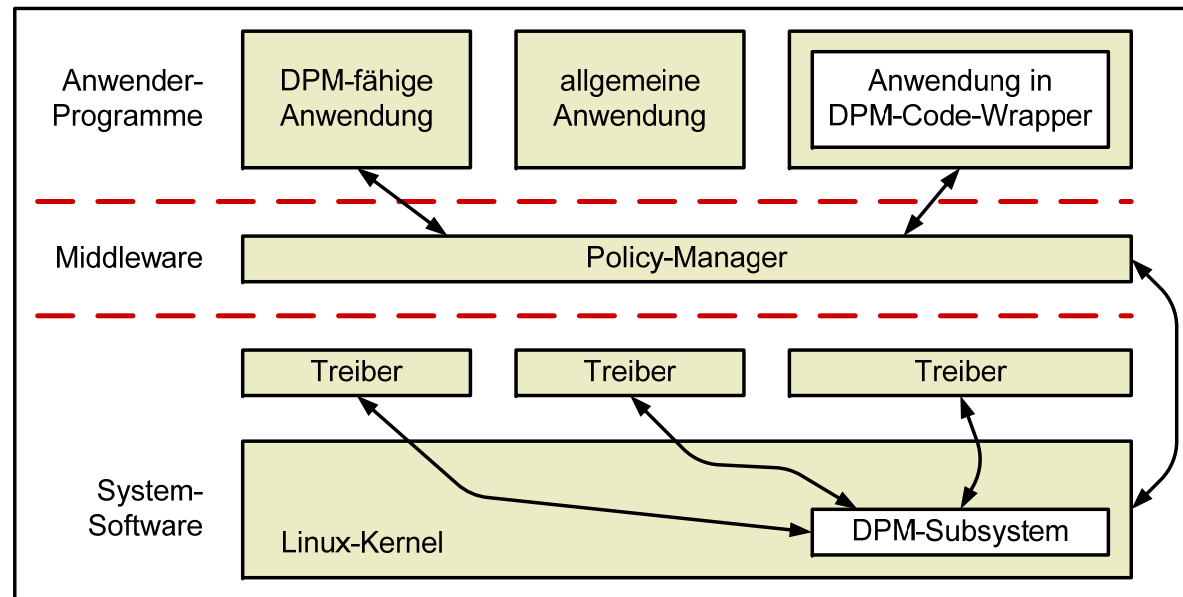
Schneller Takt
 → Schneller Start
 → Sofort stabil



- **Active Mode**
 - Controller arbeitet mit nominellem Systemtakt
 - Maximale Rechenleistung
 - Maximale Reaktionsgeschwindigkeit
 - Maximaler Stromverbrauch (ca. 12 bis 16 mA)
- **Power Save Mode**
 - Prozessor wird nur mit 32,768 kHz getaktet
 - nur ca. 1/400 der maximalen Rechenleistung
 - Stromverbrauch ca. 4 mA
- **Idle Mode**
 - CPU und Peripheriebausteine sind vom Takt getrennt
 - Zeitgesteuertes Aufwachen durch einen Timer
 - Ereignisgesteuertes Wecken durch Wake-Up-Modul
- **Halt Mode**
 - Alle Oszillatoren deaktiviert
 - Aufwecken nur durch Interrupt oder wake-Up-Modul möglich
 - Stromverbrauch ca. 700 μ A



Bsp.: Embedded Linux



Quelle: Weinberg, W: Gut verwaltete Power in Elektronik 25/2004

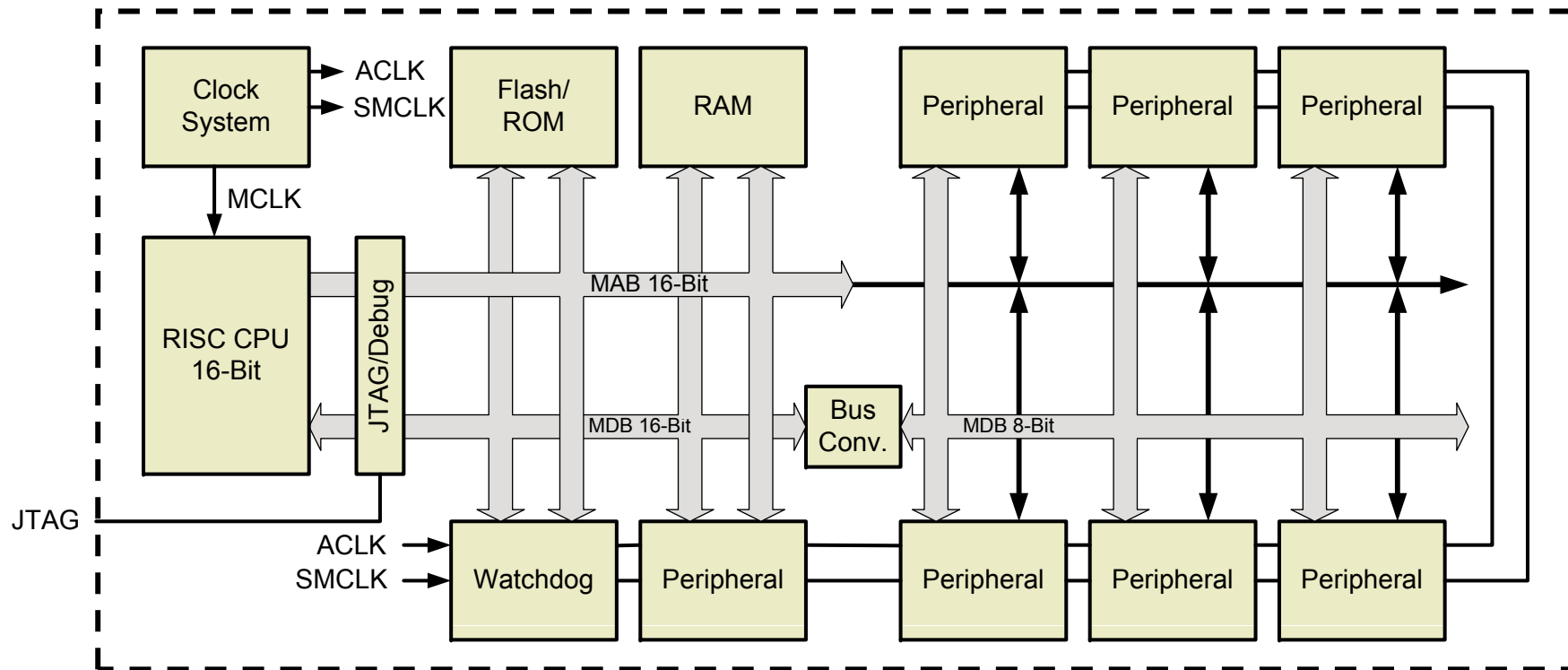
DPM: Dynamic Power Management

- Treiber müssen an DPM angepasst sein
- Policy-Manager wählt Powermanagement-Strategie
- Allgemeine Anwendungen ohne DPM-Fähigkeit können System ausbremsen

→ System konsequent aus Powermanagement-bewussten Teilen aufbauen



Blockbild



Quelle: <http://focus.ti.com/mcu/docs/>



Power Management

- **Takterzeugung und Clock-Gating**
nach Bedarf drei unterschiedliche Quellen, können abgeschaltet werden
- **DMA-Controller**
für schnellen Datentransfer ohne CPU
- **Tiefe und unterschiedliche Schlafmodi**
- **Low-Leakage-Prozesse**
- **Hardwaremultiplizierer**
Resultat steht 3 Takte nach Ablegen des zweiten Operanden bereit
- **48 Portpins**
z.T. interruptfähig
- **Steuerung durch Interrupts**

Ultra-low Power

- 0.1µA power down
- 0.8µA standby mode
- 250µA / 1MIPS @ 3V
- <1µs clock start-up
- <50nA port leakage
- Zero-power BOR
- 1.8-3.6V operation
- -40 - +85C/+105°C temp range

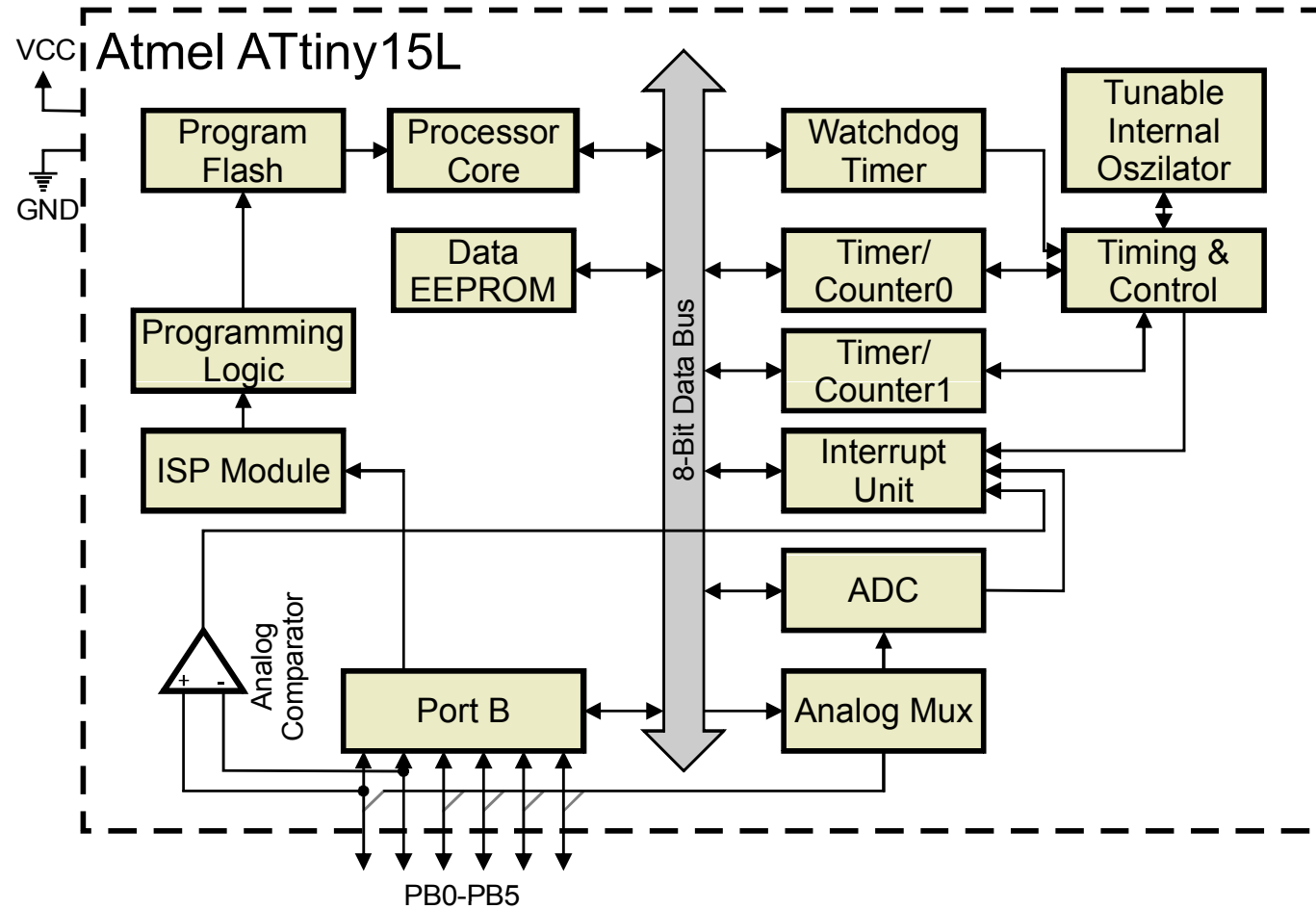
High-Performance

- Modern 16-bit RISC
- Signal Chain-on Chip integration

Besonders geeignet für Anwendungen mit niedrigem Stromverbrauch,
z.B. Blutdruckmessgeräte, Tauchcomputer, Wärmezähler



Blockbild



Quelle: Atmel Corporation: *ATtiny15L Datenblatt*. Revision F, San Jose, USA, Juni 2005. - Firmenschrift

KES → 11. → **Bsp.: Infineon C167CR (1)** → 10

