

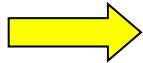
Computer Engineering WS 2012

Sytemeinstellungem

HTM – SHF - SWR



Übersicht

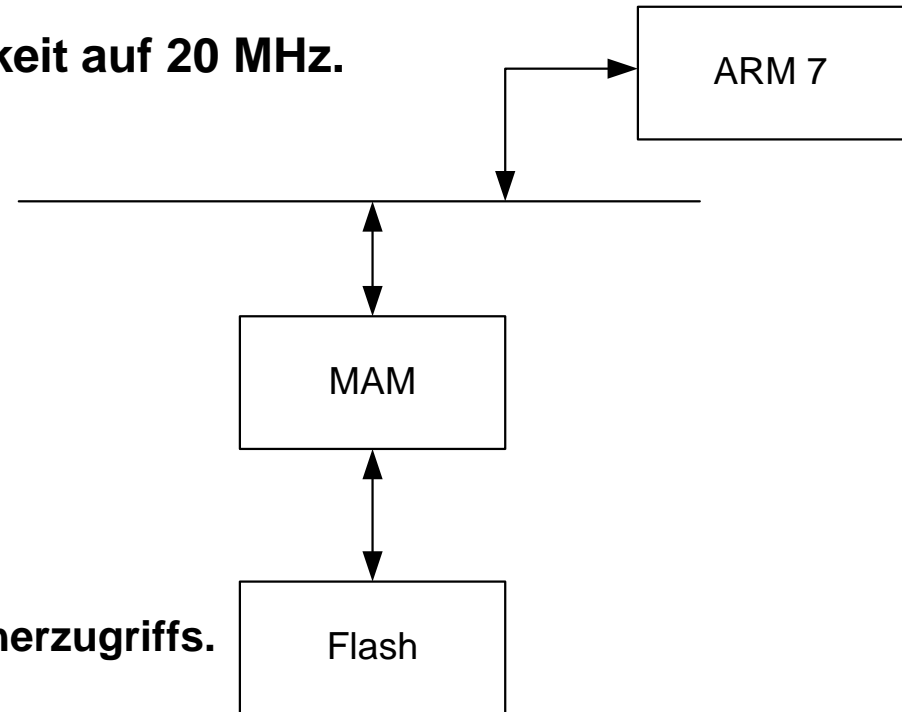


- ▶ Memory Accelerator Module
- ▶ Takterzeugung
- ▶ Stromverbrauch, Batteriebetrieb, Schlafzustände
- ▶ Reset
- ▶ Watchdog



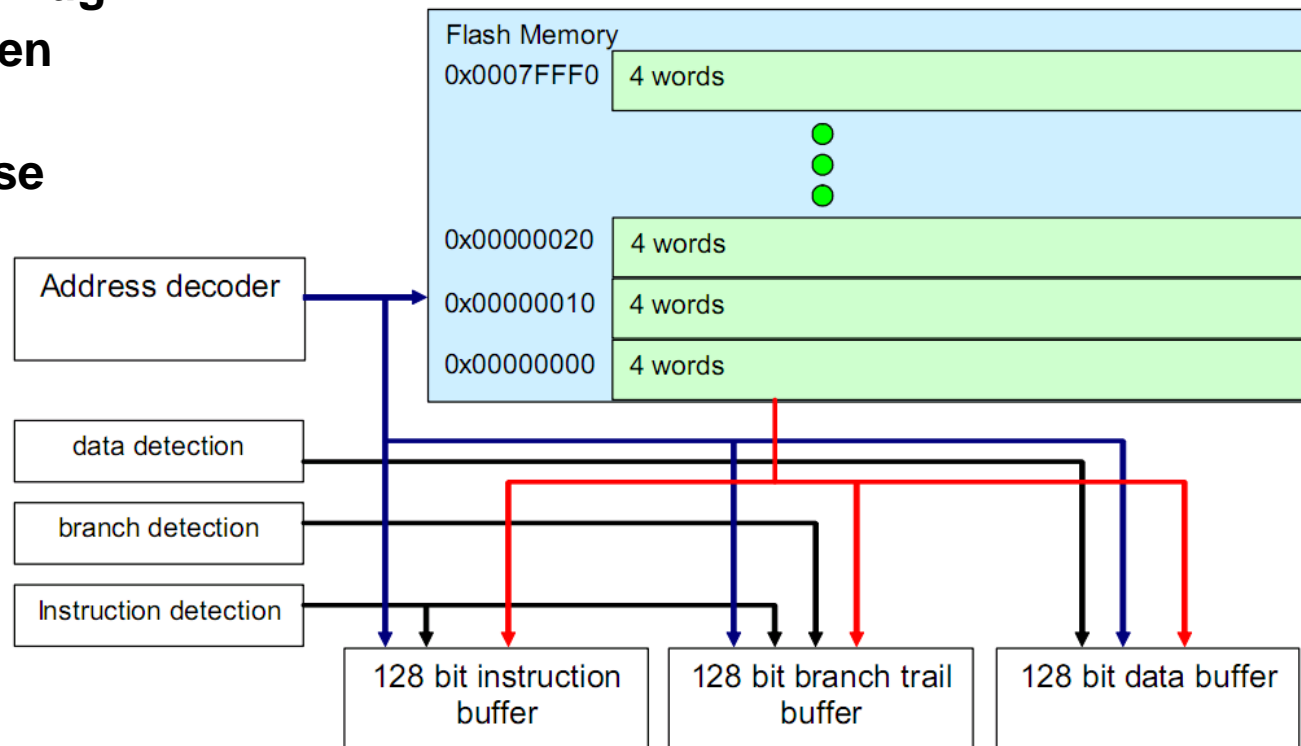
Memory Accelerator Module (MAM)

- ▶ Maschinencode ist im Flash gespeichert.
- ▶ Flash ist Flaschenhals bezüglich Programmausführungsgeschwindigkeit.
 - ▶ Zugriffszeit ca. 50 ns:
Begrenzung der Geschwindigkeit auf 20 MHz.
- ▶ Gängige Auswege:
 - ▶ Programm ins RAM kopieren.
 - ▶ Cache verwenden.
- ▶ LPC2000 verwendet **MAM**:
 - ▶ Kompromiss zwischen
 - Komplexität eines Caches und
 - Einfachheit des direkten Speicherzugriffs.



Memory Accelerator Module (MAM)

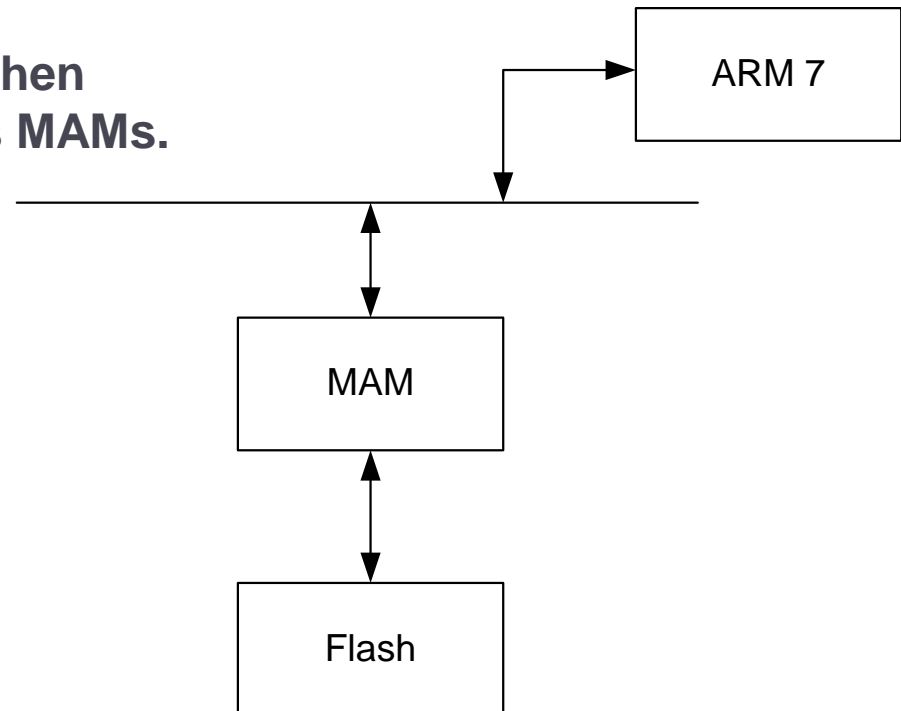
- ▶ Flash ist in 128-Bit Breite organisiert.
- ▶ Mit jedem Zugriff können 4 Instruktionen gelesen werden.
- ▶ Verbessert:
 - ▶ Sequentiellen Zugriff
 - ▶ Kurze Schleifen
 - ▶ Sprünge zur selben Adresse
- ▶ Wichtig:
Timing der Flashzugriffe muss beachtet werden.





Memory Accelerator Module (MAM)

- ▶ MAM befindet sich zwischen Flash und CPU
- ▶ MAM ist transparent für den Benutzer.
- ▶ Konfiguration über zwei Register:
 - ▶ Timing Register und
 - ▶ Control Register.
- ▶ Zusätzliche Register ermöglichen Statistiken zur Effektivität des MAMs.





Memory Accelerator Module (MAM)

- ▶ **Drei Betriebsarten:**
 - ▶ **Mode 0: Off**
 - Jeder Programmspeicherzugriff wirkt direkt auf das Flash
 - ▶ **Mode 1: Partially enabled**
 - Sequentielle Programmspeicherzugriffe werden vom Zwischenspeicher genommen.
 - Nichtsequentielle Zugriffe wirken direkt auf das Flash
 - ▶ **Mode 2: Fully enabled**
 - Alle Programmspeicherzugriffe werden nach Möglichkeit vom Zwischenspeicher genommen.

MAM Control Register (MAMCR - address 0xE01F C000) bit description

Bit	Symbol	Value	Description	Reset value
1:0	MAM_mode _control		These bits determine the operating mode of the MAM.	0
		00	MAM functions disabled	
		01	MAM functions partially enabled	
		10	MAM functions fully enabled	
		11	Reserved. Not to be used in the application.	



Memory Accelerator Module (MAM)

► Timing-Einstellung

Suggestions for MAM timing selection

system clock	Number of MAM fetch cycles in MAMTIM
< 20 MHz	1 CCLK
20 MHz to 40 MHz	2 CCLK
40 MHz to 60 MHz	3 CCLK
> 60 MHz	4 CCLK

MAM Timing register (MAMTIM - address 0xE01F C004) bit description

Bit	Symbol	Value	Description
2:0	MAM_fetch_cycle_timing		These bits set the duration of MAM fetch operations.
		000	0 - Reserved
		001	1 - MAM fetch cycles are 1 processor clock (CCLK) in duration
		010	2 - MAM fetch cycles are 2 CCLKs in duration
		011	3 - MAM fetch cycles are 3 CCLKs in duration
		100	4 - MAM fetch cycles are 4 CCLKs in duration
		101	5 - MAM fetch cycles are 5 CCLKs in duration
		110	6 - MAM fetch cycles are 6 CCLKs in duration
		111	7 - MAM fetch cycles are 7 CCLKs in duration



MAM: Benchmarks

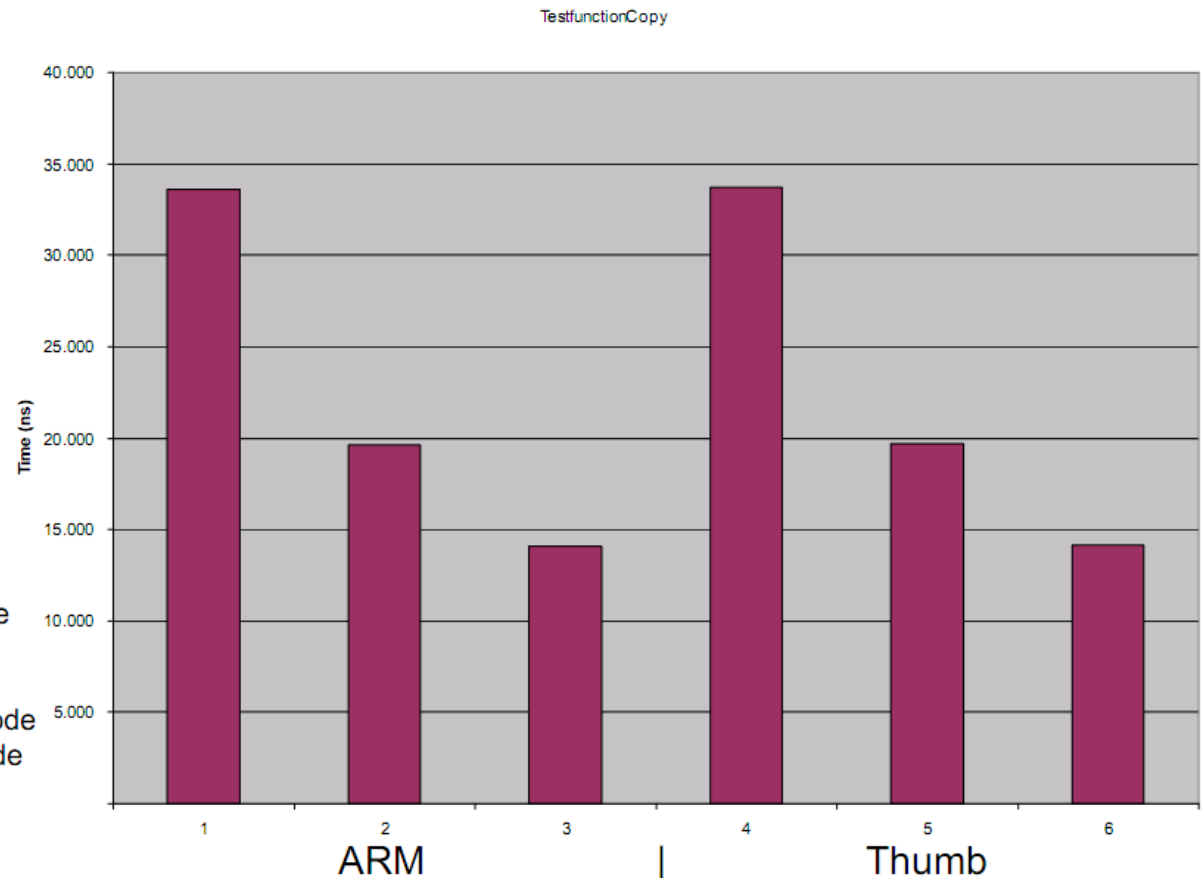


TestFunctionCopy

- ◆ Copy Data (256 Words)
- ◆ RAM to RAM

Test Configurations

1. MAM Disabled, ARM Code
2. MAM Partly Enabled, ARM Code
3. MAM Fully Enabled, ARM Code
4. MAM Disabled, Thumb Code
5. MAM Partly Enabled, Thumb Code
6. MAM Fully Enabled, Thumb Code



MAM: Benchmarks

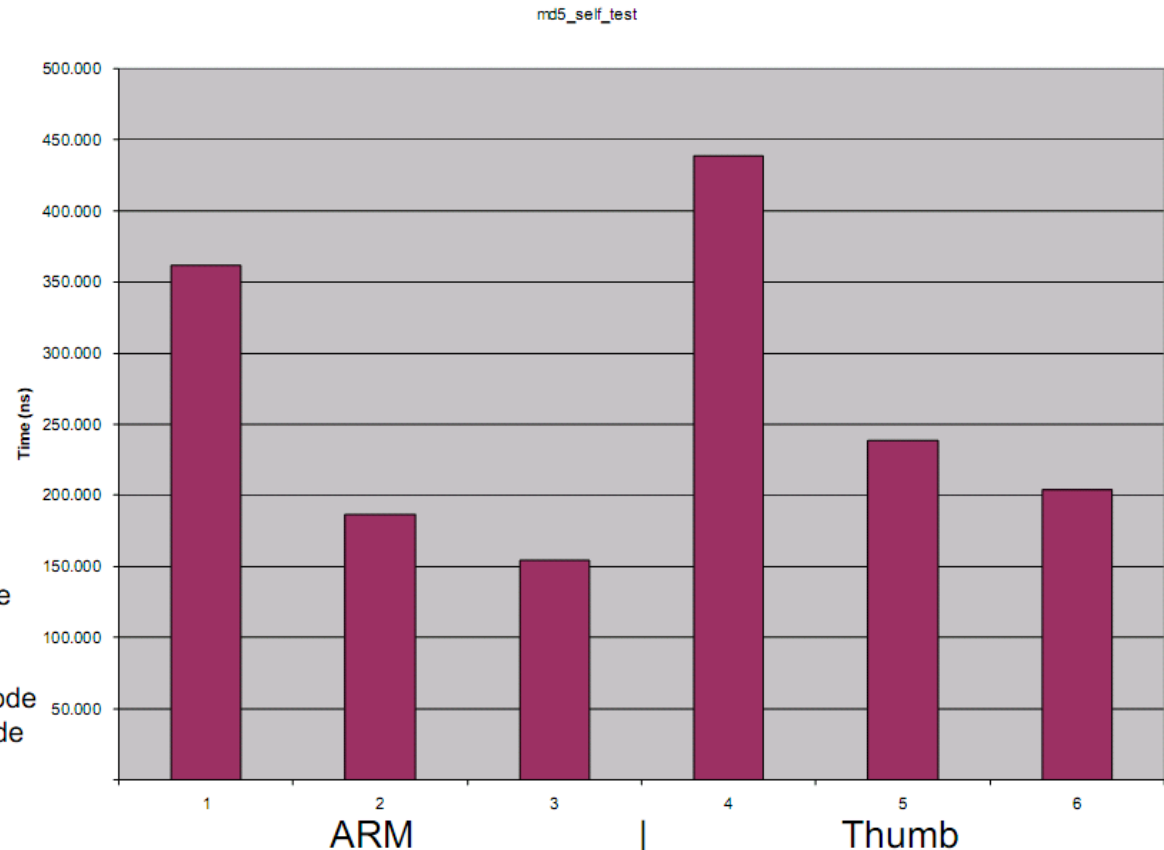
md5_self_test



- ◆ Use the self test function which is included in the md5 package

Test Configurations

1. MAM Disabled, ARM Code
2. MAM Partly Enabled, ARM Code
3. MAM Fully Enabled, ARM Code
4. MAM Disabled, Thumb Code
5. MAM Partly Enabled, Thumb Code
6. MAM Fully Enabled, Thumb Code





Übersicht

- ▶ Memory Accelerator Module
- ▶ Takterzeugung
- ▶ Stromverbrauch, Batteriebetrieb, Schlafzustände
- ▶ Reset
- ▶ Watchdog



Technologien Taktquellen

- ▶ Externer Oszillator
- ▶ Externer Quarz
- ▶ Externer Keramik-Resonator
- ▶ Interner RC-Oszillator

Auswahlkriterien

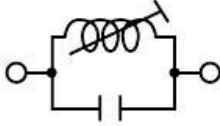
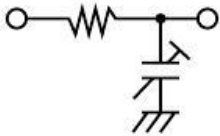


- ▶ Preis
- ▶ Genauigkeit
- ▶ Platzbedarf
- ▶ Strombedarf





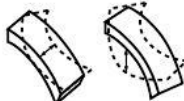
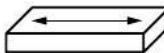
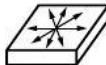

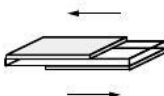
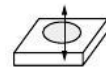
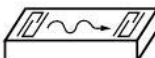
Oszillatoren


■ Characteristics of Various Oscillator Elements

Name	Symbol	Price	Size	Adjust-ment	Oscillation Frequency Initial Tolerance	Long-term Stability
LC		Inexpensive	Big	Required	$\pm 2.0\%$	Fair
CR		Inexpensive	Small	Required	$\pm 2.0\%$	Fair
Quartz Crystal		Expensive	Big	Not required	$\pm 0.001\%$	Excellent
Ceramic Resonator		Inexpensive	Small	Not required	$\pm 0.5\%$	Excellent

Schwingquarz

■ Vibration Mode and Frequency Range

Frequency (Hz)		1k	10k	100k	1M	10M	100M	1G	
Vibration Mode									
1 Flexural mode		■							
2 Length mode				■					
3 Area expansion mode				■					
4 Radius vibration				■					
5 Shear thickness mode					■				
6 Thickness expander mode					■				
7 Surface acoustic wave						■			

[Note] :  show the direction of vibration

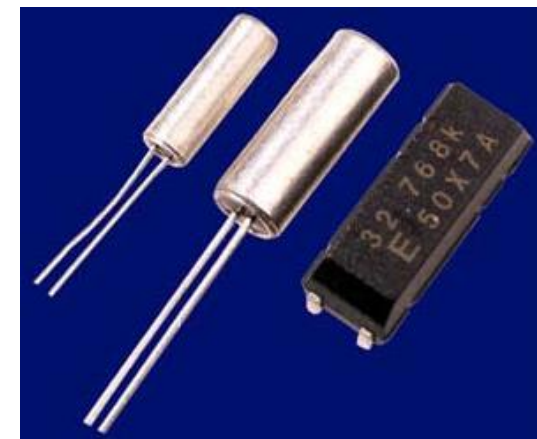
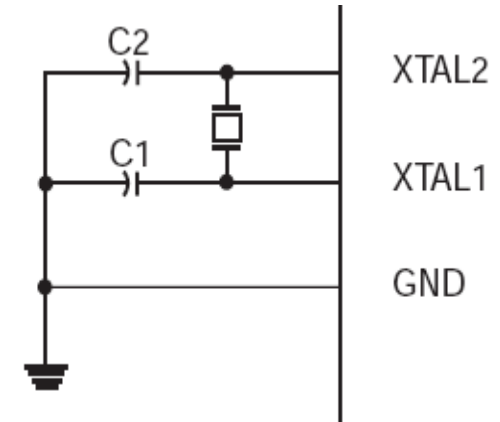
Externer Taktoszillator

	CXO-11	CXO-21
Package	14 Pin DIP	8 Pin DIP
Frequency Range	1.000 ~ 100.000 MHz	
Frequency Stability	A=±25ppm, B=± 50ppm, C=± 100ppm	
Operating Temperature Range	0°C - 70°C (-40°C - +85°C -> Option 'S')	
Storage Temperature Range	-55°C - 125°C	
Supply Voltage	5.0 VDC ±10%	
Aging (at 25°C)	±5ppm / year max.	
Supply Current	1.000MHz to 23.999MHz 24.000MHz to 70.000MHz 70.000MHz to 100.000MHz	20mA max. 30mA max. 40mA max.
Waveform Symmetry	40/60 % Normal, 45/ 55% Tight	
Rise/Fall Time	<div>< 9 MHz 10ns max.</div> <div>< 32 MHz 5ns max.</div> <div>> 32 MHz 4ns max.</div>	
Output Voltage	Logic Low Logic High	0.4 V max. 2.4 V min.
Output Load	1 to 10 TTL Load	
Start-up Time	10 ms	



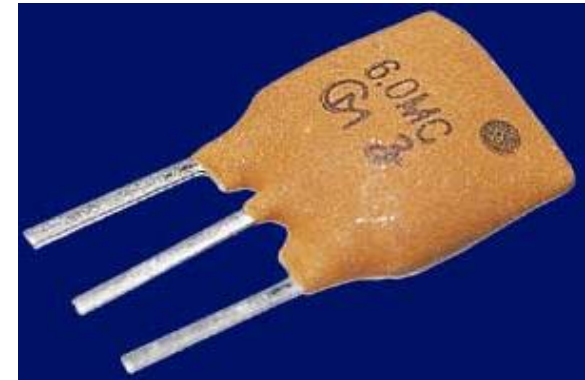
Externer Quarz

Frequenzbereich:	4 ... 40 MHz
Frequenztoleranz (25°C):	± 50 ppm
Shunt-Kapazität (max.):	5 pF
Betriebstemperaturbereich:	- 20 ... + 70 °C
Temperaturstabilität < 5,5 MHz:	± 50 ppm
Temperaturstabilität > 5,5 MHz:	± 30 ppm
Isolationswiderstand (min):	500 MΩ
Belastung (empfohlen):	10 ... 100 μW
Alterung:	± 5 ppm / Jahr
Lötwärmebeständigkeit bis 3 Min.:	< 230 °C
bis 10 s:	> 260 °C
Schockbeständigkeit (max.):	± 10 ppm



Externer Keramik-Resonator

- ▶ 8,00 MHz Keramik-Resonator, 3-pin, für integrierte Ladekapazität.
- ▶ Frequenztoleranz: $\pm 0,5\%$
- ▶ Frequenzstabilität bis 6 MHz: $\pm 0,3\%$
- ▶ Frequenzstabilität ab 8 MHz: $\pm 0,4\%$

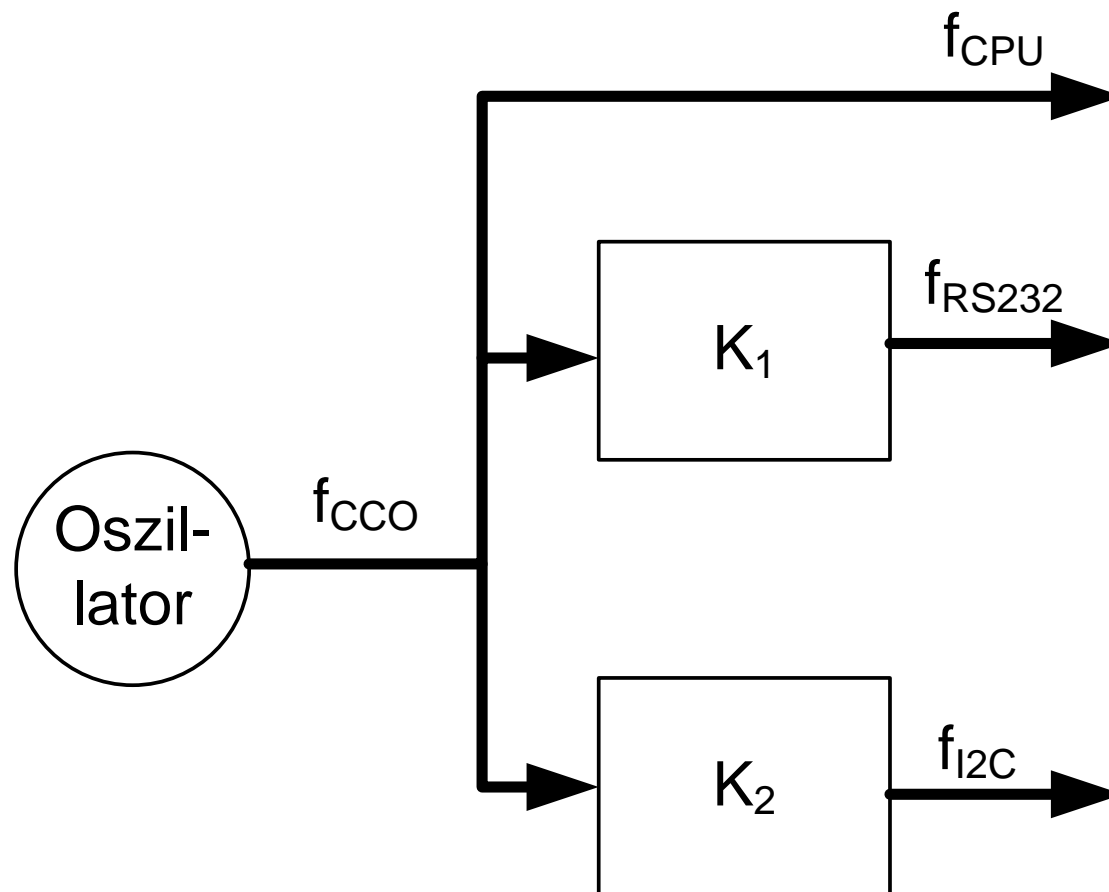


Interner RC-Oszillator

- ▶ Feste Taktfrequenz: 8 MHz
- ▶ Genauigkeit:
- ▶ Standard: 10 % (bei 5 V und 25 °C)
- ▶ Kann kalibriert werden: 2%



Takterzeugung





Takterzeugung, Beispiel Atmel AVR

- ▶ Im Mikrocontroller verwendete Takte werden meist von einer gemeinsamen Quelle abgeleitet.
- ▶ Kann bei niedrigen Taktraten problematisch werden, z.B. bei gleichzeitiger Nutzung von I2C und RS232 (Beispiel AVR):
 - ▶ I2C mit Bitrate 400kHz

$$Teiler = \frac{f_{CLK}}{2 \cdot Bitrate} - 8 \qquad Teiler \geq 10$$

- Mögliche Teiler sind 10, 11 und 12 bei 14.4 MHz, 15.2 MHz und 16 MHz

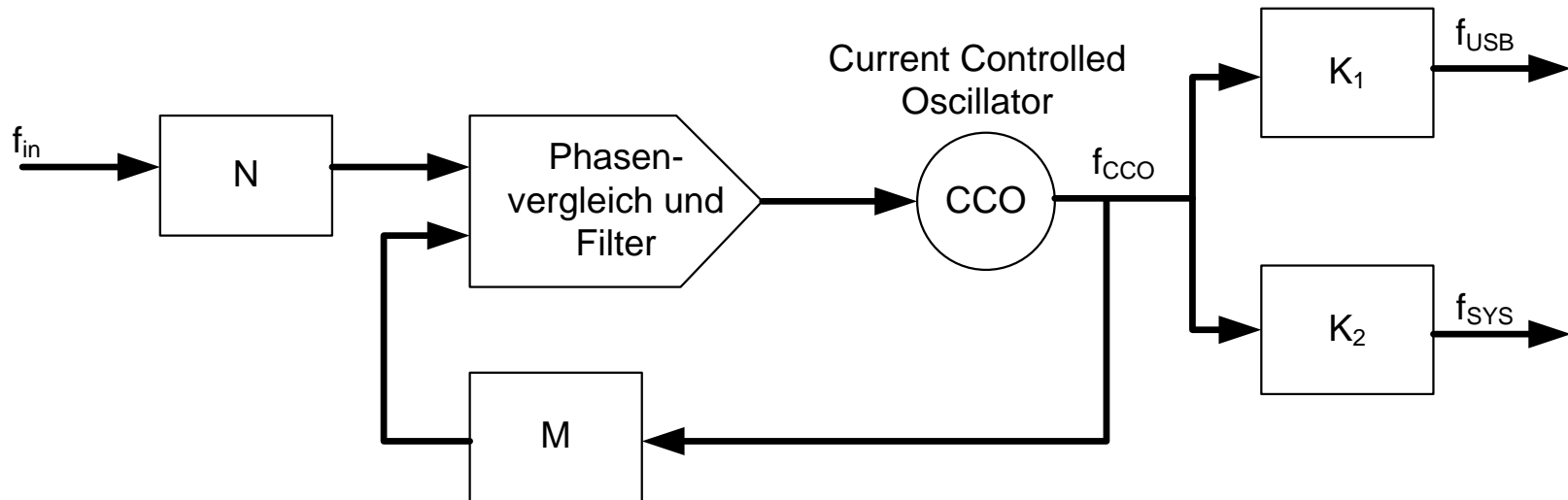
- ▶ RS232 mit Baudrate 115200

$$Teiler = \frac{f_{CLK}}{8 \cdot Baudrate} - 1$$

- Fehler bei bester Einstellung:
-2.6% bei 14.4 MHz, 3.2% bei 15.2 MHz, -2.2% bei 16 MHz,

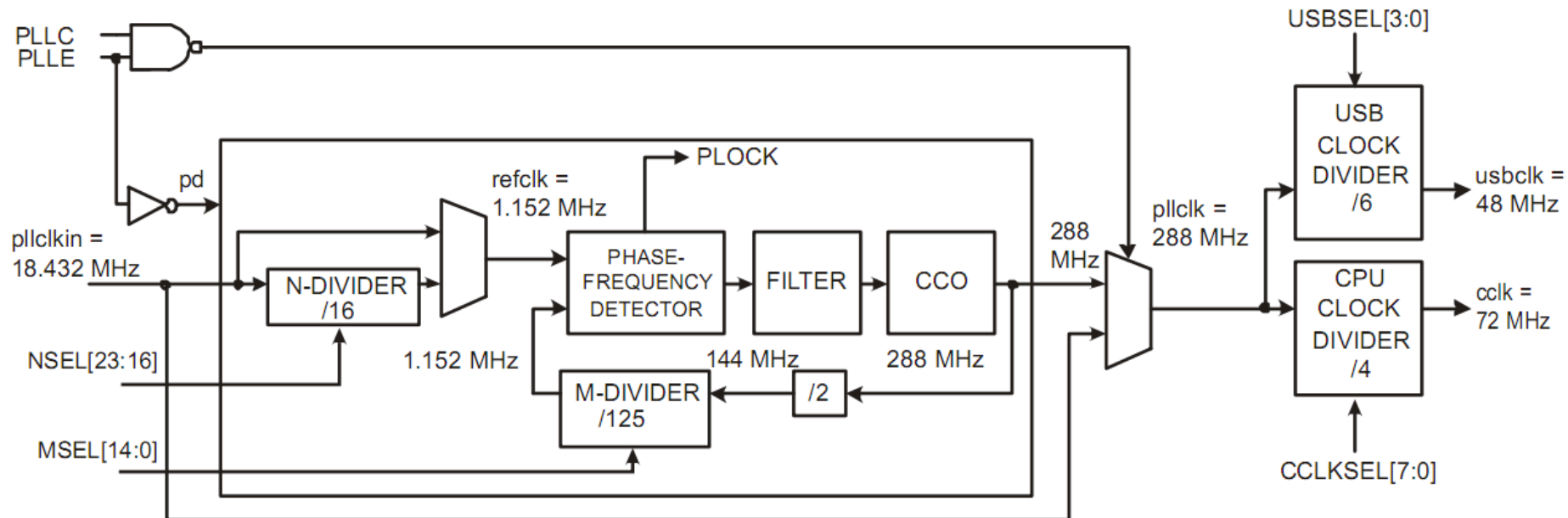
Takterzeugung

- ▶ **Frequenzvervielfachung:**
 - ▶ **höhere Oszillatorfrequenz ermöglicht flexiblere Takteinstellung**
 - ▶ **Verwendung von Phase-locked loops (PLL)**



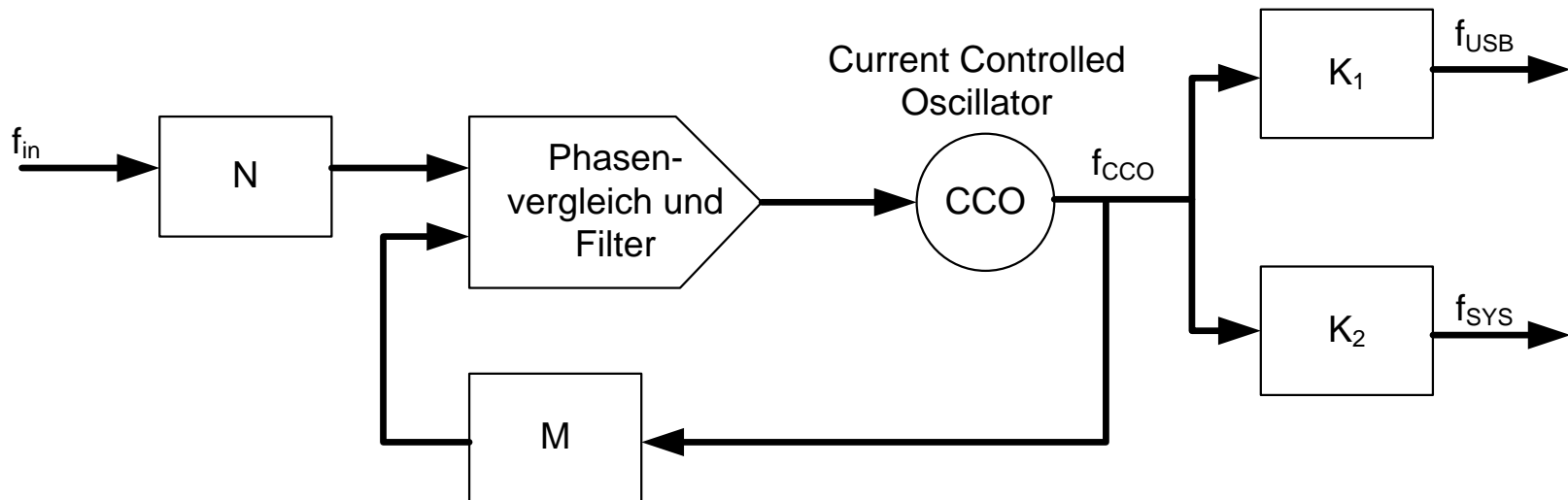
Takterzeugung

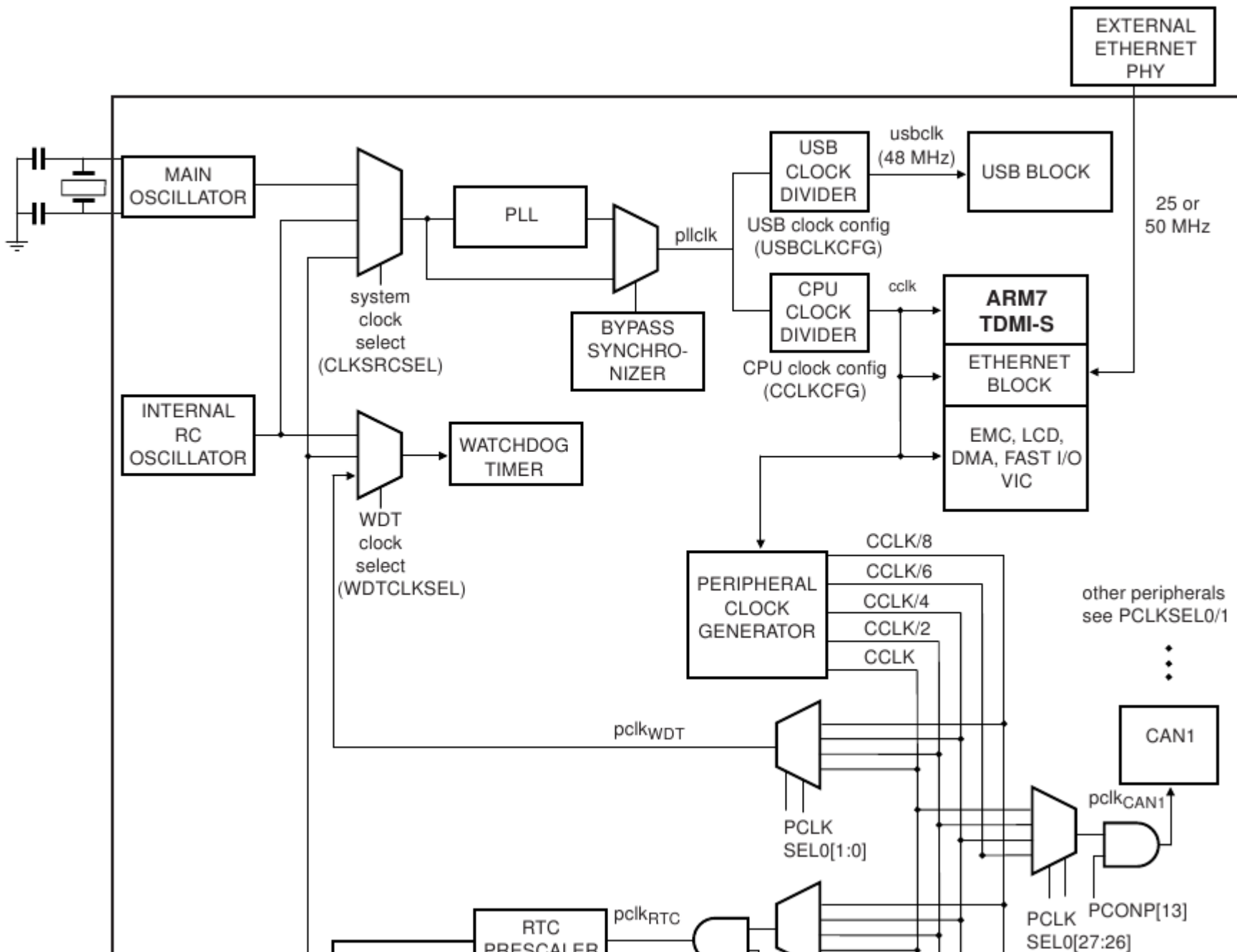
- ▶ **Getrennter Takt für**
 - ▶ **System (cclk) und**
 - ▶ **Usb (usbclk)**
- ▶ Takte werden mittels Frequenzvervielfacher von langsamer externer Taktquelle abgeleitet

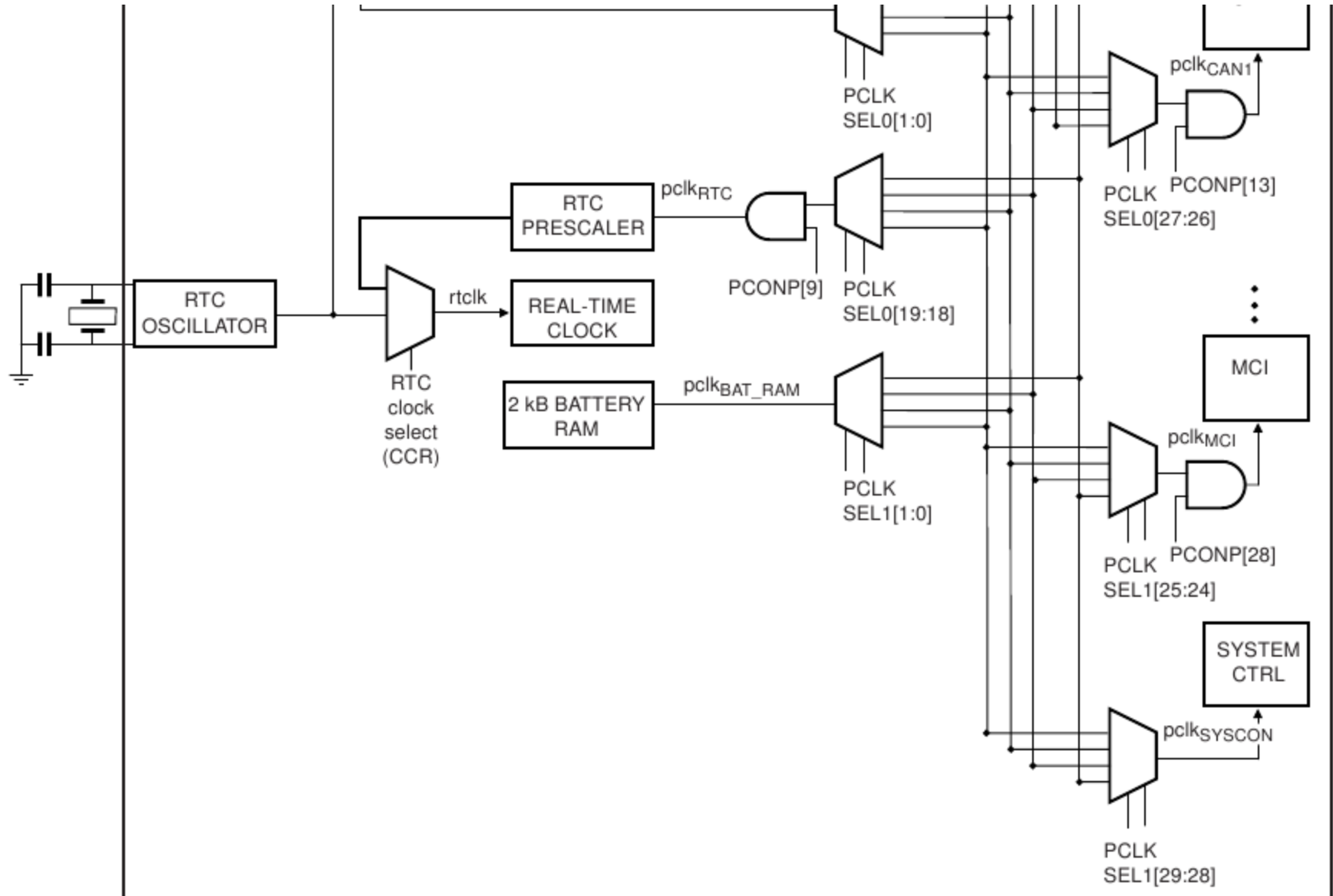


Takterzeugung LPC2468

- ▶ f_{in} : 32kHz bis 24 MHz
 - ▶ f_{CCO} : 275 MHz bis 550 MHz
 - ▶ f_{USB} : 48 MHz (4*12 MHz)
 - ▶ f_{SYS} : maximal 72 MHz
- ▶ K_1, K_2 : gerade
 - ▶ N : 1 bis 32
 - ▶ M : 6 bis 32768



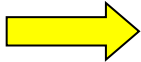






Übersicht

- ▶ Memory Accelerator Module
- ▶ Takterzeugung
- ▶ Stromverbrauch, Batteriebetrieb, Schlafzustände
- ▶ Reset
- ▶ Watchdog





CE WS12

Stromverbrauch LPC2468

$T_{amb} = -40\text{ °C to }+85\text{ °C}$ for commercial applications, unless otherwise specified.

Symbol	Parameter	Conditions	Min	Typ ^[1]	Max	Unit
$I_{DD(DCDC)act(3V3)}$	active mode DC-to-DC converter supply current (3.3 V)	$V_{DD(DCDC)(3V3)} = 3.3\text{ V};$ $T_{amb} = 25\text{ °C};$ code <code>while(1){}</code> executed from flash; no peripherals enabled; PCLK = CCLK				
		CCLK = 10 MHz	-	15	-	mA
		CCLK = 72 MHz	-	63	-	mA
		all peripherals enabled; PCLK = CCLK / 8				
		CCLK = 10 MHz	-	21	-	mA
		CCLK = 72 MHz	-	92	-	mA
		all peripherals enabled; PCLK = CCLK				
		CCLK = 10 MHz	-	27	-	mA
$I_{DD(DCDC)pd(3V3)}$	power-down mode DC-to-DC converter supply current (3.3 V)	$V_{DD(DCDC)(3V3)} = 3.3\text{ V};$ $T_{amb} = 25\text{ °C}$	-	150	-	μA
I_{BATact}	active mode battery supply current	DC-to-DC converter on	^[10] -	20	-	μA
		DC-to-DC converter off	^[10] -	28	-	μA



Betriebsarten der CPU, Schlafzustände

- ▶ Zum Stromsparen hat die CPU 3 verschiedene Schlafzustände:
 - ▶ **Idle**
 - ▶ **Sleep**
 - ▶ **Power-down**
- ▶ Unterscheiden sich
 - ▶ in den möglichen **Quellen** zum Aufwecken
 - ▶ benötigte **Zeit** zum Aufwecken
 - ▶ Höhe der **Stromersparnis**



Betriebsarten der CPU, Schlafzustände

- ▶ **Idle Mode**
 - ▶ **Takt der CPU wird abgeschaltet.**
 - ▶ **Peripherie funktioniert weiter und kann Interrupts auslösen.**
 - ▶ **Stromersparnis dadurch,
dass CPU, Speicher und interne Busse nicht mehr getaktet werden.**



Betriebsarten der CPU, Schlafzustände

- ▶ **Sleep Mode**
 - ▶ **Ausschalten des Hauptoszillators.**
 - ▶ **Interner RC-Oszillator läuft weiter, ist aber nicht mit System verbunden**
 - ermöglicht schnelles Aufwachen
 - ▶ **Real-time Clock funktioniert weiter**
 - Echtzeituhr kann CPU wieder aufwecken.
 - ▶ **Aufwecken kann durch Echtzeituhr oder durch andere Interrupts, die keinen Takt benötigen, erfolgen.**
 - ▶ **Nach Aufwecken muss Hauptoszillator gestartet werden:**
 - 4096 Takte Wartezeit
 - ▶ **Zusätzlicher Zeitbedarf für das Neukonfigurieren der PLL:**
 - 500µsec



Betriebsarten der CPU, Schlafzustände

▶ Power-down Mode

▶ Wie Sleep Mode, zusätzlich:

- Power-down des Flash
- Ausschalten des RC-Oszillators.

▶ Real-time Clock funktioniert weiter

→ Echtzeituhr kann CPU wieder aufwecken.

▶ Aufwecken kann durch Echtzeituhr oder durch andere Interrupts, die keinen Takt benötigen, erfolgen.

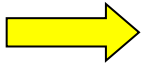
▶ Nach Aufwecken:

- Start des internen RC-Oszillators: 60 μ sec
- Start des Flash: 100 μ sec
- Start des Hauptoszillators: 4096 Takte
- Neukonfiguration der PLL 500 μ sec



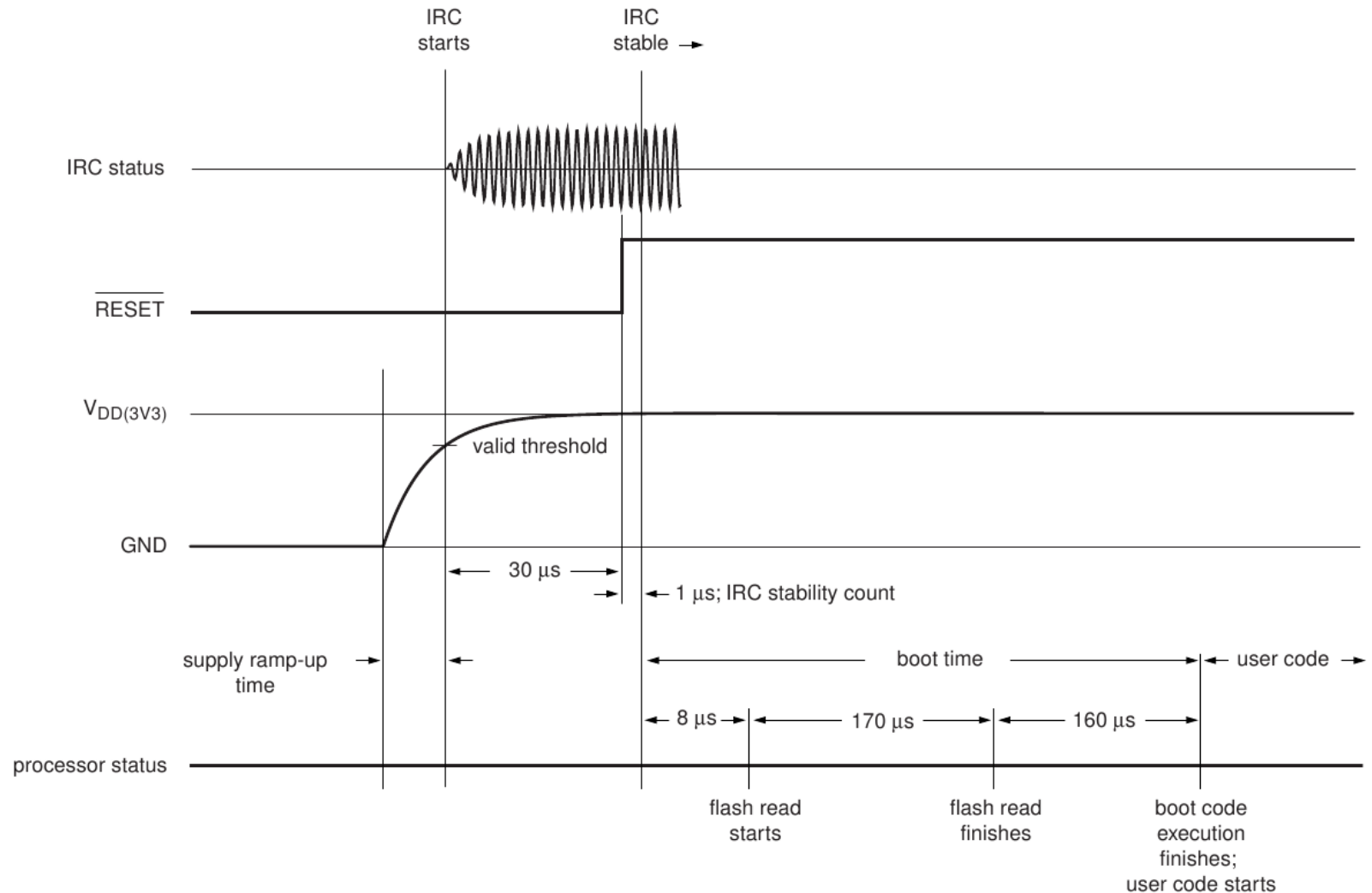
Übersicht

- ▶ Memory Accelerator Module
- ▶ Takterzeugung
- ▶ Stromverbrauch, Batteriebetrieb, Schlafzustände
- ▶ Reset
- ▶ Watchdog



CE WS12

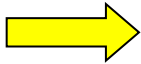
LPC2468: Reset



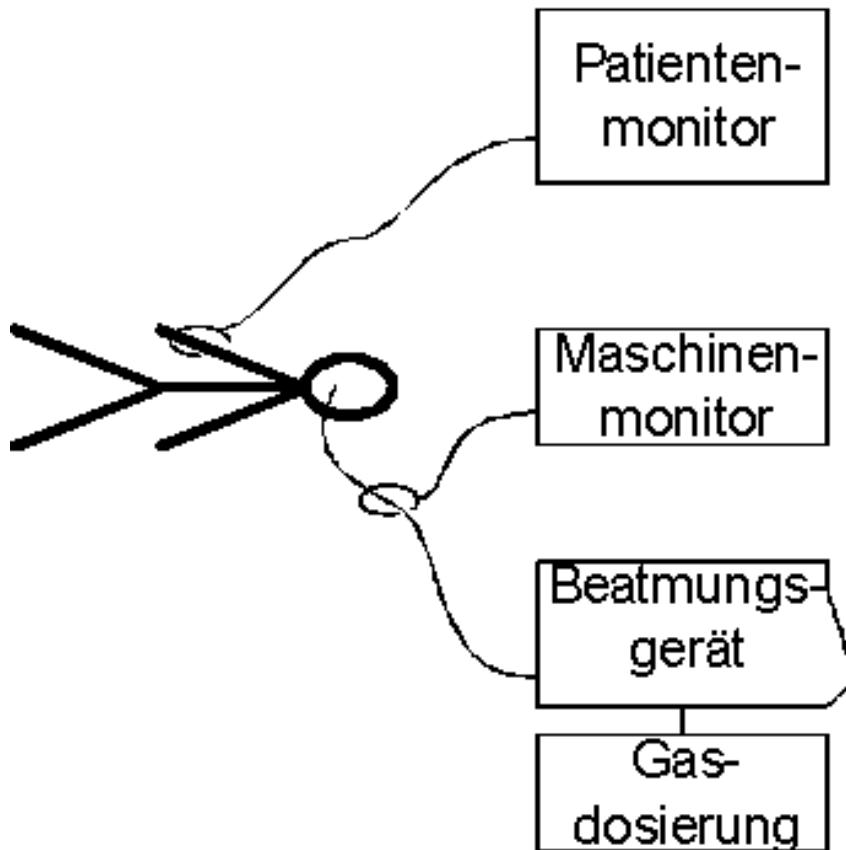


Übersicht

- ▶ Memory Accelerator Module
- ▶ Takterzeugung
- ▶ Stromverbrauch, Batteriebetrieb, Schlafzustände
- ▶ Reset
- ▶ Watchdog



Sicherheitsaspekte: Beispiel Medizintechnik



Überwachung:

Kreislauf (EKG, Herzfrequenz)
Sauerstoffsättigung des Blutes
Blutdruck

Überwachung:

Gaszusammensetzung
(O₂, CO₂, Narkosemittel)
Beatmungsdruck

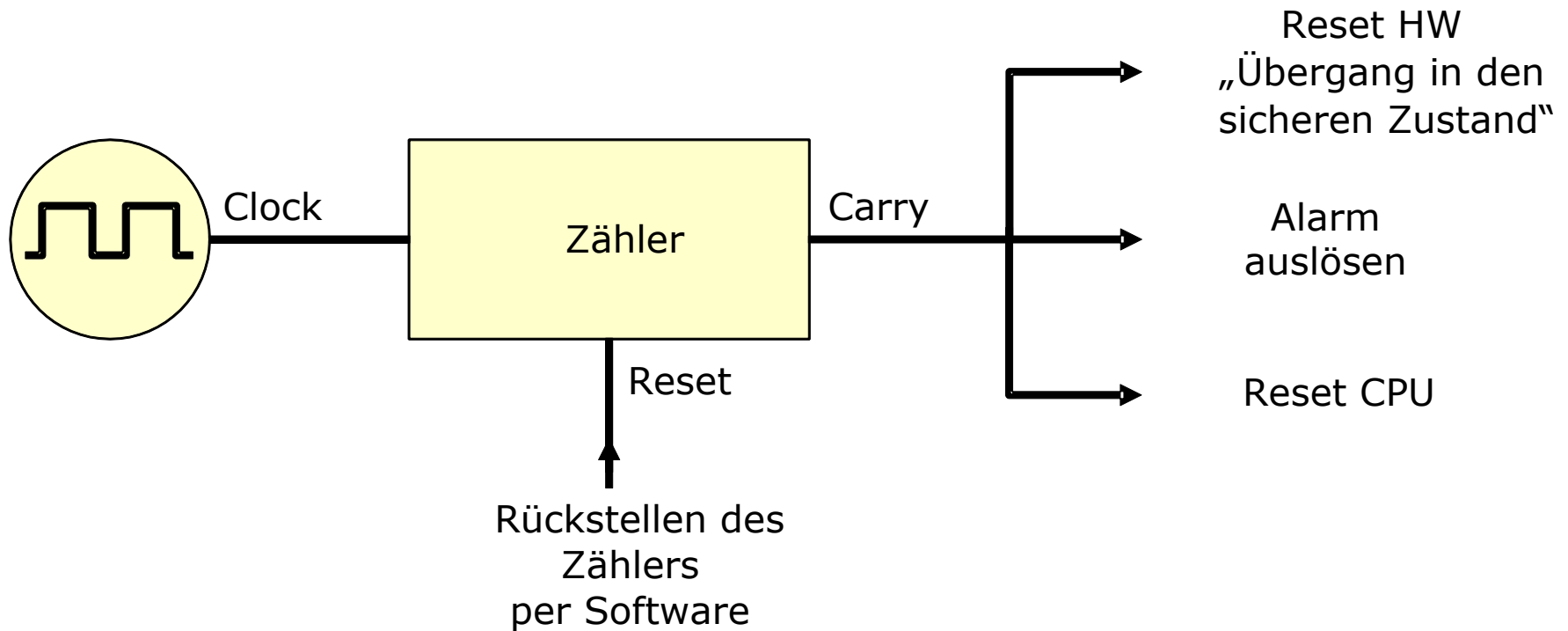
Einstellung:

Beatmungsfrequenz, -volumen
Sauerstoffgehalt, Narkosemittel



CE WS12

Prinzipieller Aufbau





Erkennbare Fehler

- ▶ Programmpfad, welcher Zurücksetzen des Watchdogs beinhaltet, wird nicht mehr durchlaufen, z. B. wg.
 - ▶ CPU hängt in Endlosschleife
 - ▶ Interrupt-Hardware umprogrammiert
 - ▶ CPU ist im Sleep-Zustand
 - ▶ CPU ist aus dem Programm gesprungen

Nicht erkennbare Fehler

- ▶ Falsche Daten
- ▶ Zu häufiges Durchlaufen der Schleife
- ▶ versehentliches Triggern des Zählers
- ▶ Falscher Takt (wenn keine unabhängigen Taktquellen)



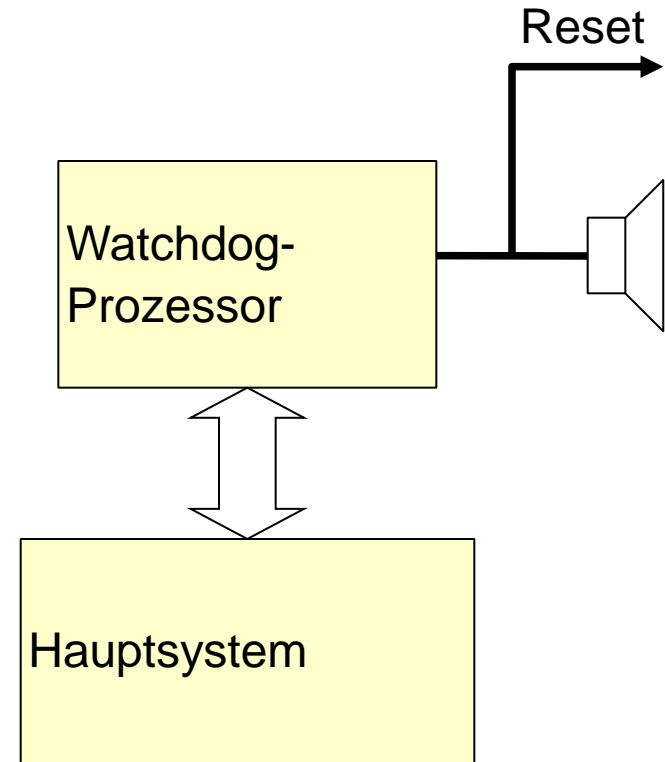
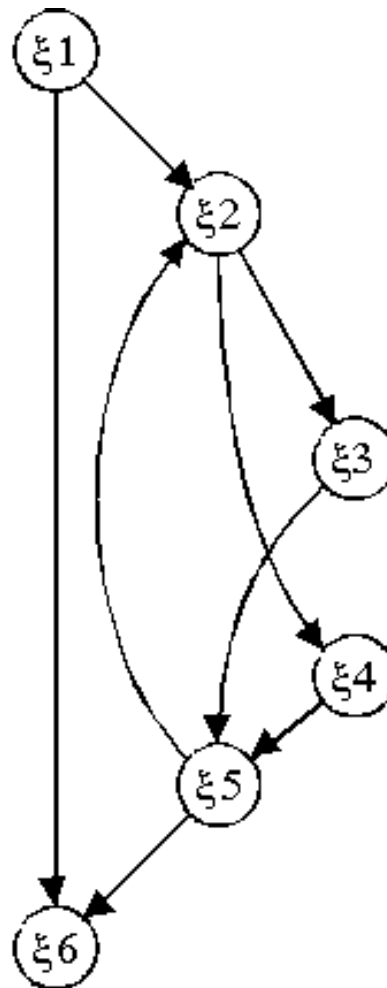
Design-Hinweise

- ▶ Zeitbasis muss unabhängig von CPU-Takt sein.
- ▶ Vorsicht bei programmierbarer Periode.
- ▶ Schutz gegen versehentliches Triggern, z. B. mit spezieller Sequenz: (0x55, 0xAA)
- ▶ Besser: Verwendung komplexer Sequenzen, die z. B. den Programmablauf beschreiben.
- ▶ Unabhängige Überwachung von
 - ▶ Hauptschleife und
 - ▶ Interruptserviceroutinen.
- ▶ Watchdog-Funktion muss beim Einschalten des Systems überprüft werden.

Watchdog Prozessor

```
ξ1  
while( x > 0 )  
{  
  ξ2  
  x = x - 1;  
  if( y < x )  
  {  
    ξ3  
  }  
  else  
  {  
    ξ4  
  }  
  ξ5  
  z = y + x;  
}
```

ξ6
38

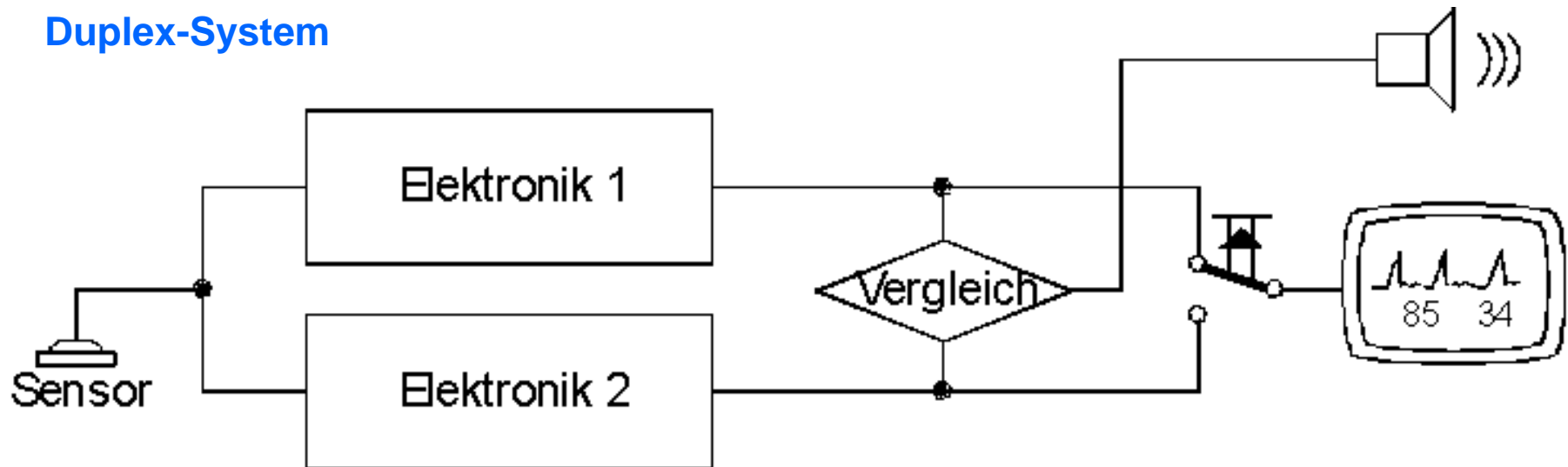




Watchdog Prozessor

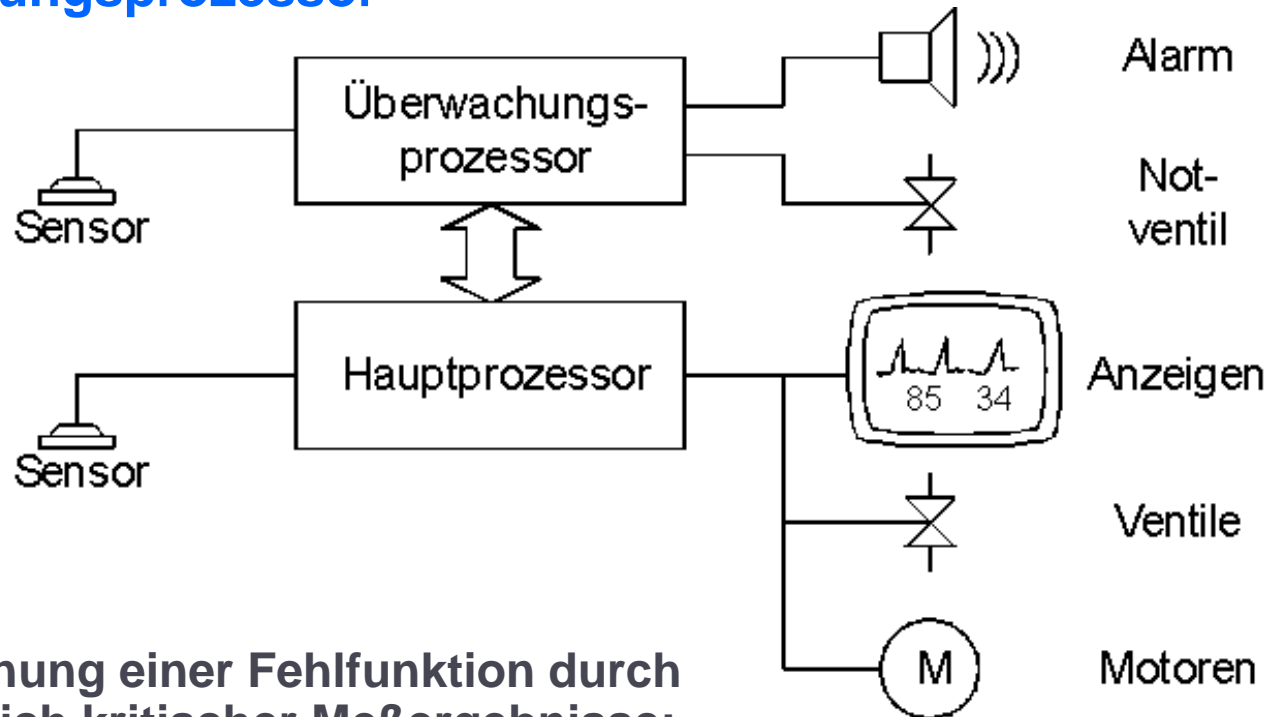
- ▶ Überwachung des Programmflusses
- ▶ Darstellung des Programmablaufs als Graph
- ▶ Jedem Knoten wird eine Signatur zugeordnet:
z.B. mittels automatischem Verfahren
- ▶ Signaturen, erlaubte Pfade und Verweildauer sind im Watchdog-Prozessor gespeichert
- ▶ Während der Programmausführung werden Signaturen und Pfade mit den gespeicherten verglichen.
- ▶ Im Fehlerfall: CPU-Reset, Übergang in den "sicheren Zustand" und Alarmierung
- ▶ Zu beachten:
 - ▶ erlaubte, aber falsche Übergänge werden nicht entdeckt
 - ▶ Interruptroutinen können Kontrollfluß jederzeit unterbrechen
 - ▶ Besondere Behandlung von Bibliotheksfunktionen und Betriebssystemaufrufen (z.B. Multitasking)
 - ▶ Erkennung von fehlerhaften Daten nur wenn sie zu unerlaubten Übergängen führen

Duplex-System



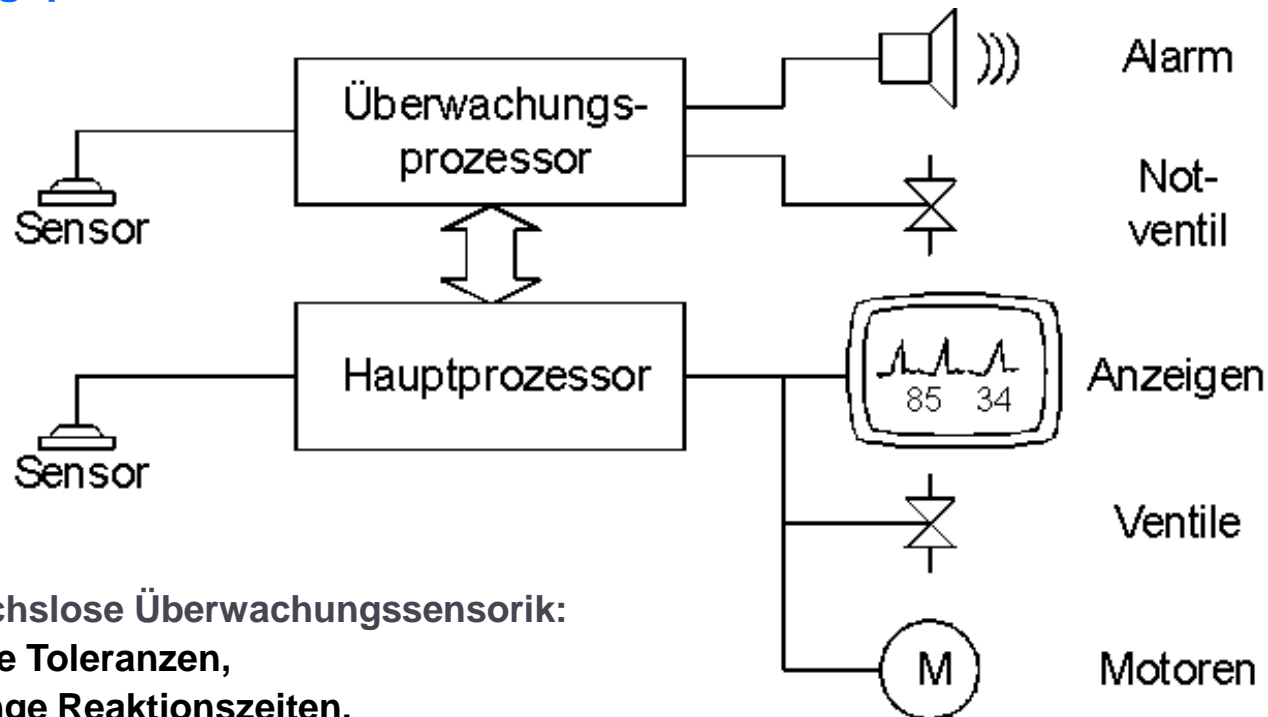
- ▶ Zwei gleichwertige Kanäle zur Signalaufbereitung und –darstellung.
- ▶ Einfache Sensorik.
- ▶ Bei unterschiedlichen Ergebnissen erfolgt Alarmgebung.
- ▶ Bediener muss entscheiden, welche Ergebnisse richtig sind.
- ▶ **Sicherstellung der Funktion bei Ausfall einer Elektronik.**
- ▶ **Geringer Entwicklungsaufwand.**
- ▶ **Keine Sicherheit gegen Ausfall eines Sensors.**
- ▶ **Hohe Herstellkosten.**

Überwachungsprozessor



- ▶ **Erkennung einer Fehlfunktion durch Vergleich kritischer Meßergebnisse:**
 - ▶ **Übergang in den sicheren Zustand**
 - ▶ **Alarmierung**
 - ▶ **Aber keine Sicherstellung der Gerätefunktion**

Überwachungsprozessor



- ▶ **Anspruchslose Überwachungssensorik:**
 - ▶ große Toleranzen,
 - ▶ geringe Reaktionszeiten,
 - ▶ einfacher zu verifizieren.
- ▶ Geringer Softwareaufwand, einfache Algorithmen im Überwachungsprozessor
- ▶ Anforderungen:
 - ▶ Alle Überwachungseinrichtungen müssen überprüfbar sein (z.B. während des Einschalt-Selbsttest).
 - ▶ Hardwareausfall während des Betriebes kann toleriert werden (führt zum doppelten Fehlerfall).



LPC2468: Watchdog

- ▶ **Initialisierung:**
 - ▶ **Auswahl der Taktquelle: Interner RC Oszillator.**
 - Wichtig wegen Unabhängigkeit vom Haupttakt.
 - ▶ **Programmierung der Betriebsart:**
 - Interrupt oder Reset
 - ▶ **Festlegung der Timeout-Zeit.**
 - 4-facher Vorteiler und 32-Bit Zähler.
 - ▶ **Aktivierung des Watchdog durch Schreiben der Sequenz:**
 - 0xAA, 0x55 (darf nicht unterbrochen werden!)
- ▶ **Danach regelmäßiges Nachtriggern der Watchdog-Timers**
 - ▶ **Schreiben der Sequenz 0xAA, 0x55.**

