

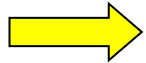
# Computer Engineering WS 2012

## CPU Ein- und Ausgabe

HTM – SHF - SWR

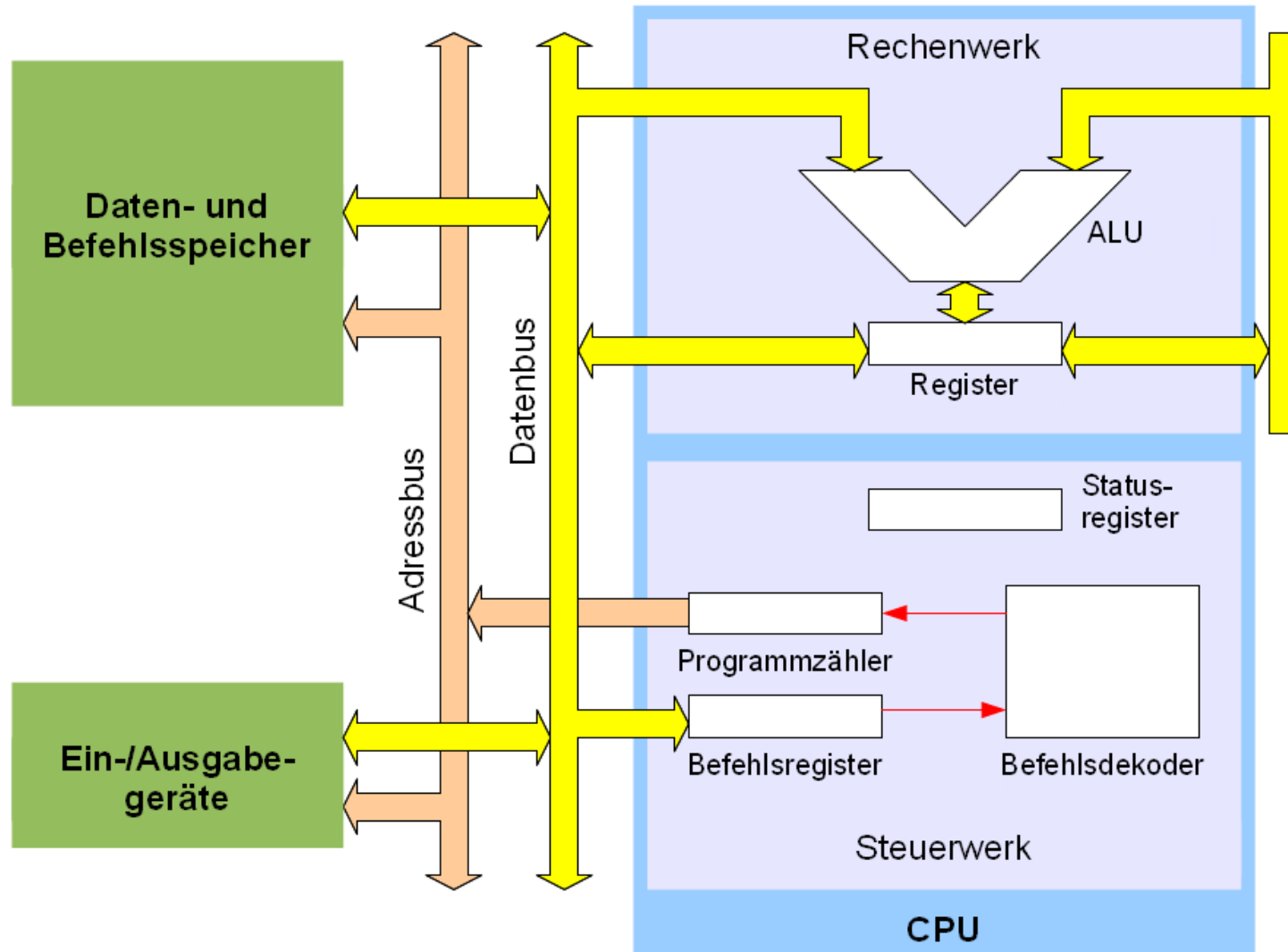


## Übersicht



- ▶ Ein- und Ausgabe
  - ▶ Memory-Mapped I/O
  - ▶ I/O-based I/O
- ▶ Bussysteme des LPC2468.
- ▶ Zugriff auf externe Komponenten.  
Typischer Ablauf eines Buszugriffs.
- ▶ Systemeinstellungen für LPC2468-Komponenten.
- ▶ General Purpose I/O (GPIO) LPC2468.

## Prinzipieller Aufbau eines Computers





## Wie greift man auf Register eines externen Gerätes zu?

- ▶ **Alternative 1: Memory Mapped**
  - ▶ Die Register sind auf Hauptspeicheradressen abgebildet (mapped).
  - ▶ Ein lesender / schreibender Zugriff auf diese Hauptspeicheradressen greift nicht auf den Speicher zu, sondern auf die entsprechenden Register des Devices.
- ▶ **Alternative 2: I/O Mapped (muss die CPU unterstützen, Intel tut dies)**
  - ▶ Es gibt einen weiteren Adressraum, so genannte I/O Adressen. Diese Adressen stehen in keiner Relation zu den Hauptspeicheradressen.
  - ▶ Über spezielle Befehle (in, out Assembler Befehle) wird über I/O Adressen auf die Register eines Devices zugegriffen.

## Memory-Mapped

Hauptspeicher-  
adressen  
 $0x00000000$

$0xAFFFFFFF$

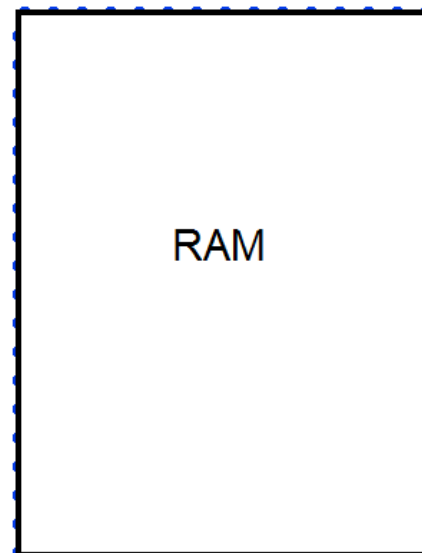
$0xFF000000$

$0xFF00000F$

$0xFF000100$

$0xFF0001FF$

$0xFFFFFFFF$



*Adressraum:  
von der CPU adressierbarer  
Bereich  
 $0x00000000-0xFFFFFFFF$*

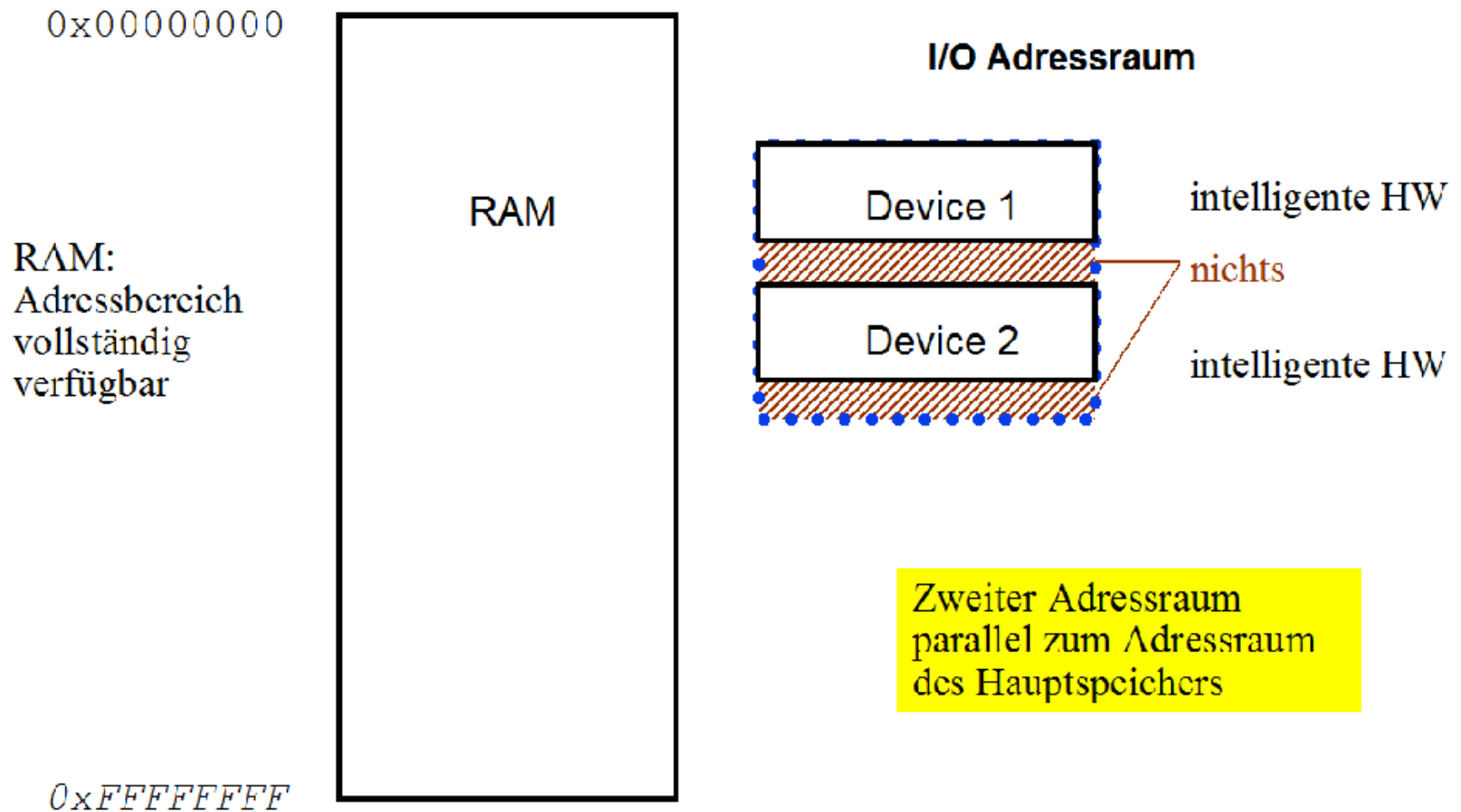
*RAM*

*intelligente HW*

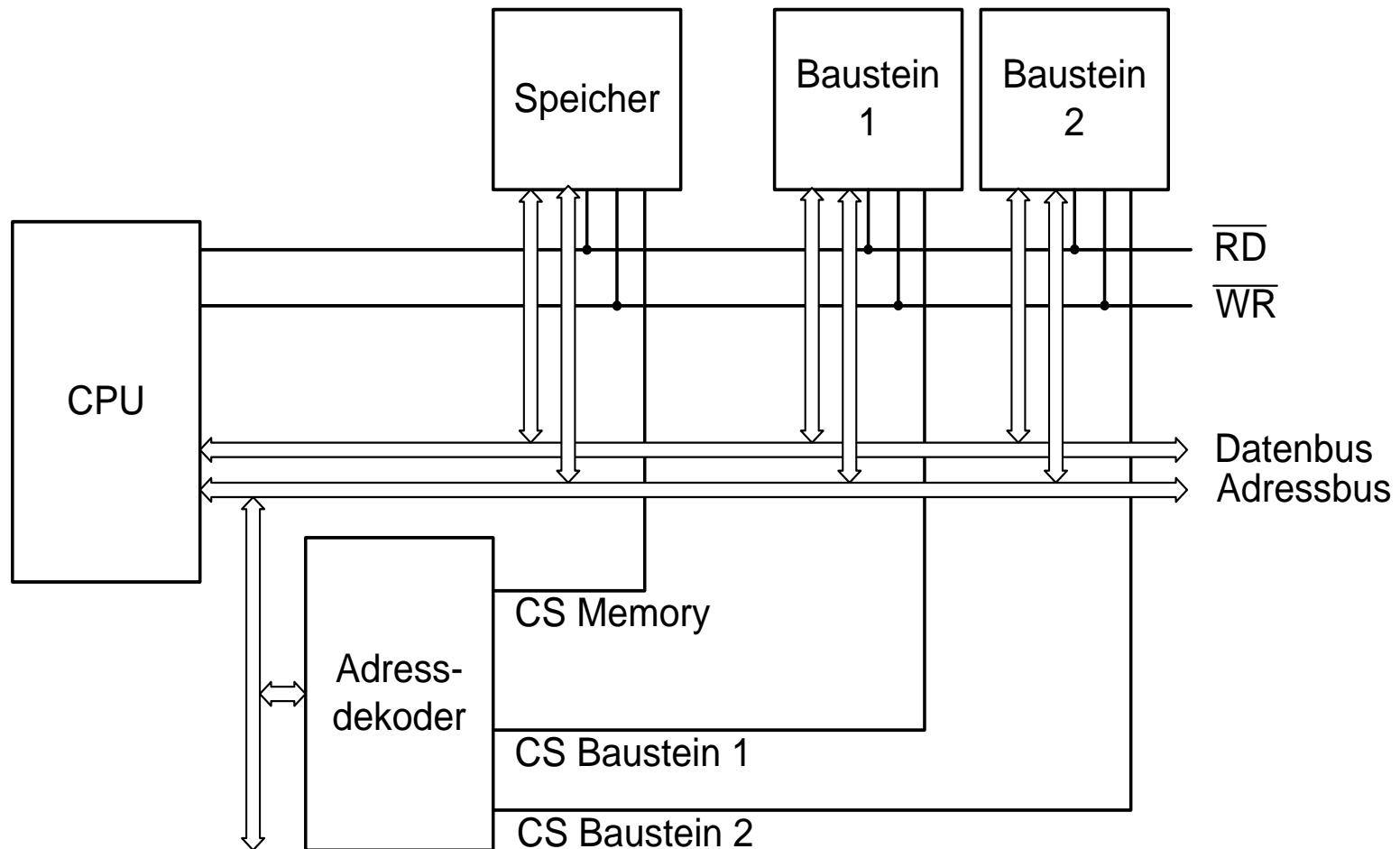
*nichts*

*intelligente HW*

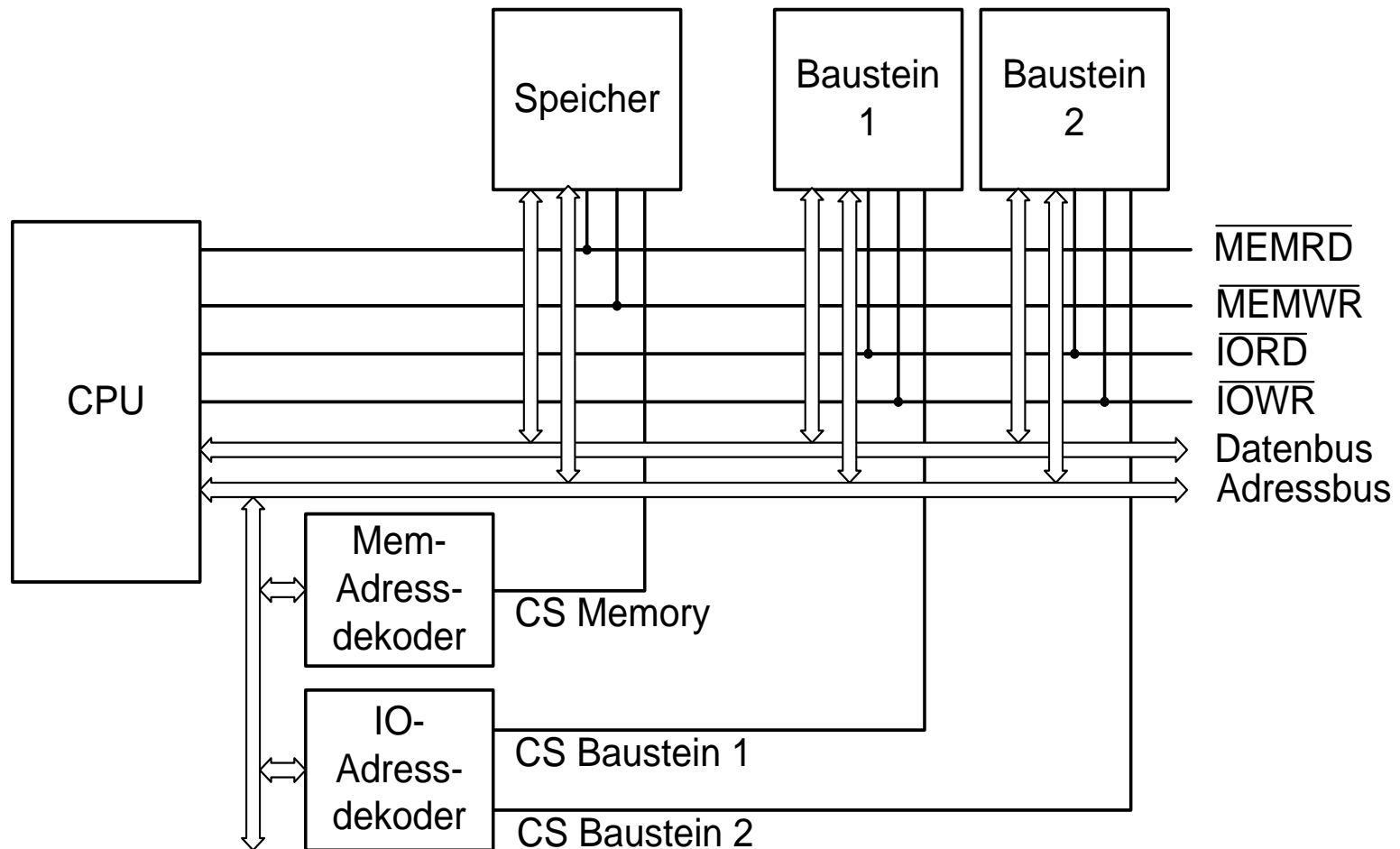
## IO-Mapped



## Einfaches Bussystem, Memory-Mapped



## Einfaches Bussystem, IO-Mapped





## Merkmale Bussysteme: Maximale Übertragungsrate, Einheit?

- ▶ hängt ab von:
  - ▶ Anzahl der gleichzeitig übertragbaren Bytes
    - gegeben durch Datenbusbreite
  - ▶ Bustaktfrequenz
    - maximale Frequenz, mit der die Signalleitungen eines Busses betrieben werden können
  - ▶ Anzahl der für die Übertragung notwendigen Bustakte



## Merkmale Bussysteme: Buszyklenarten

- ▶ **Einzelzyklusbus (single cycle)**
  - ▶ **Führt nur eine Datenübertragung durch**
- ▶ **Blockzyklus (burst cycle)**
  - ▶ **mehrfache aufeinander folgende Zugriffe**
  - ▶ **Adresse wird nur einmal übertragen**

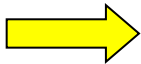
## Merkmale Bussysteme: Bustypen

- ▶ **Split-Bus**
  - ▶ **Getrennte Adress- und Datenleitungen**
  - ▶ **Gleichzeitige Übertragung von Adressen und Daten möglich.**
- ▶ **Gemultiplexer Bus**
  - ▶ **Adressen und Daten werden über die gleichen Leitungen übertragen.**
  - ▶ **Während eines Speicherzugriffs werden zuerst die Adressen, dann die Daten übertragen.**
  - ▶ **Nachteil: Langsamer.**
  - ▶ **Vorteil: Geringerer Verdrahtungsaufwand, geringere Pin-Zahl.**

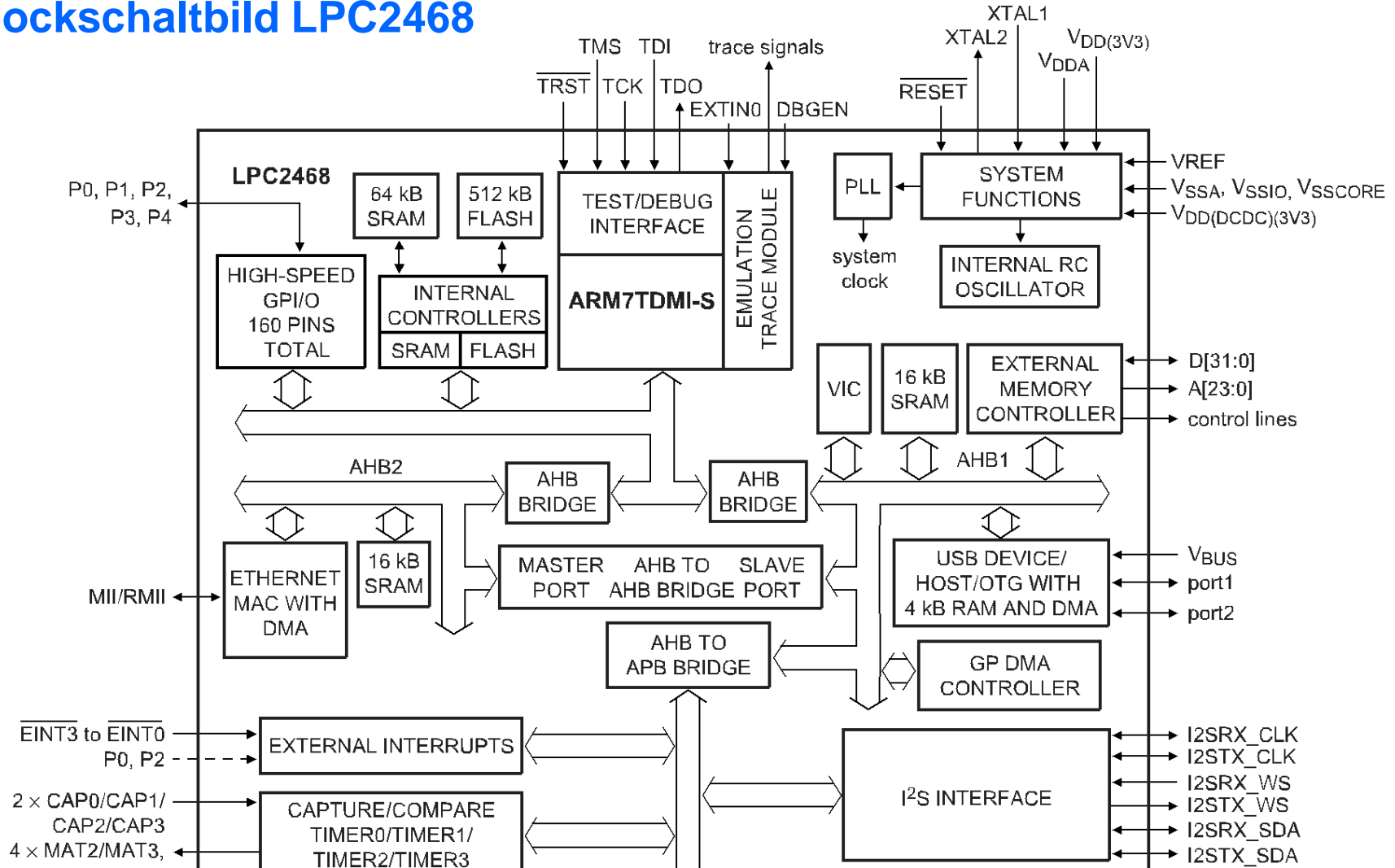


## Übersicht

- ▶ Ein- und Ausgabe
  - ▶ Memory-Mapped I/O
  - ▶ I/O-based I/O
- ▶ Bussysteme des LPC2468.
- ▶ Zugriff auf externe Komponenten.  
Typischer Ablauf eines Buszugriffs.
- ▶ Systemeinstellungen für LPC2468-Komponenten.
- ▶ General Purpose I/O (GPIO) LPC2468.



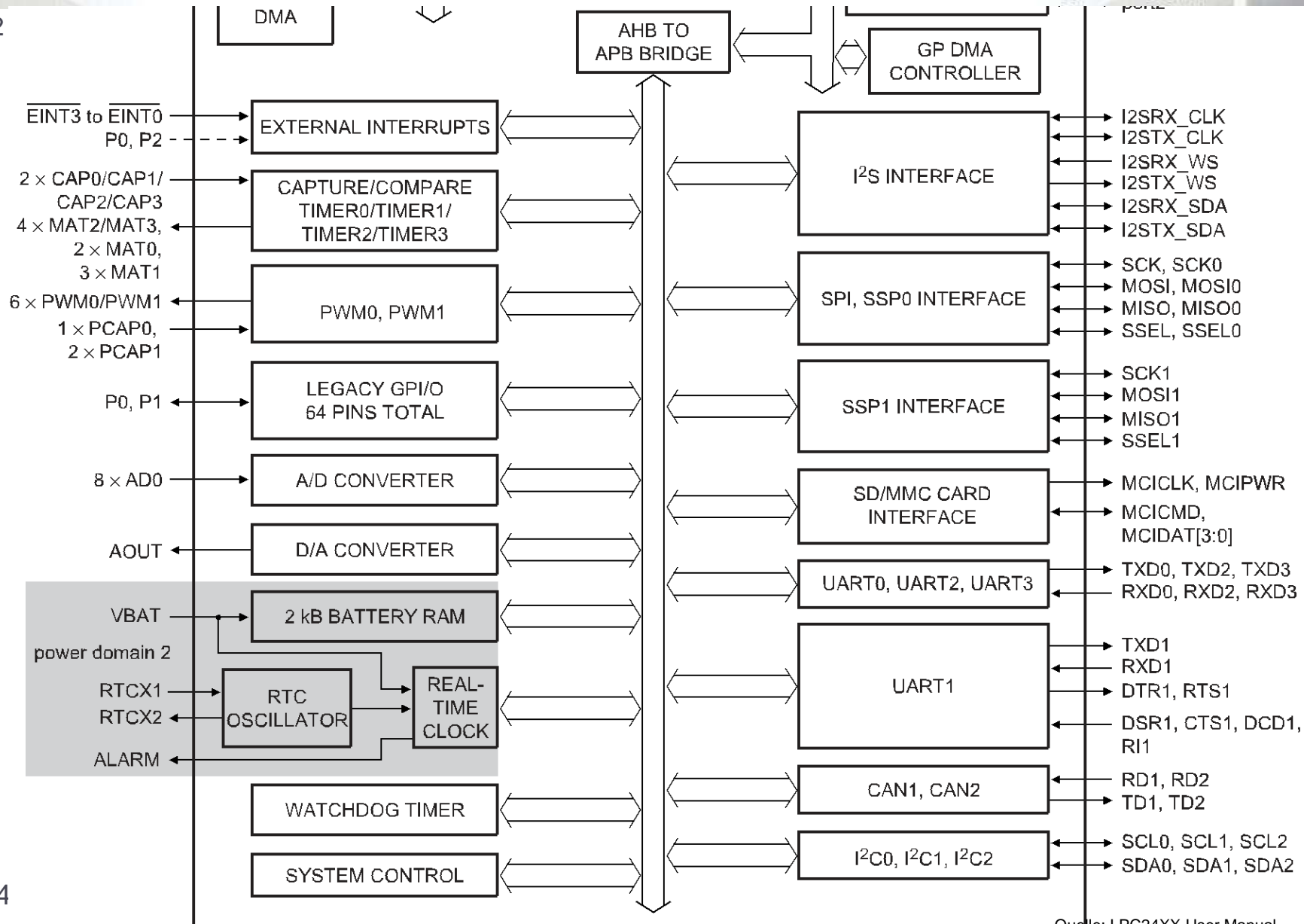
## Blockschaltbild LPC2468





CE WS12

## Blockschaltbild LPC2468 (Fortsetzung)





## Bushierarchie

► **LPC2468 enthält 3 Hierarchieebenen:**

**1. ARM 7 local bus**

- High Speed
- Anschluss von On-chip-Speicher (Flash, RAM)
- Single Master: CPU

**2. „Advanced High-performance BUS“ (AHB)**

- Multimasterfähig (z.B.: Bridge, USB, Ethernet)
- burst transfers
- single clock operation

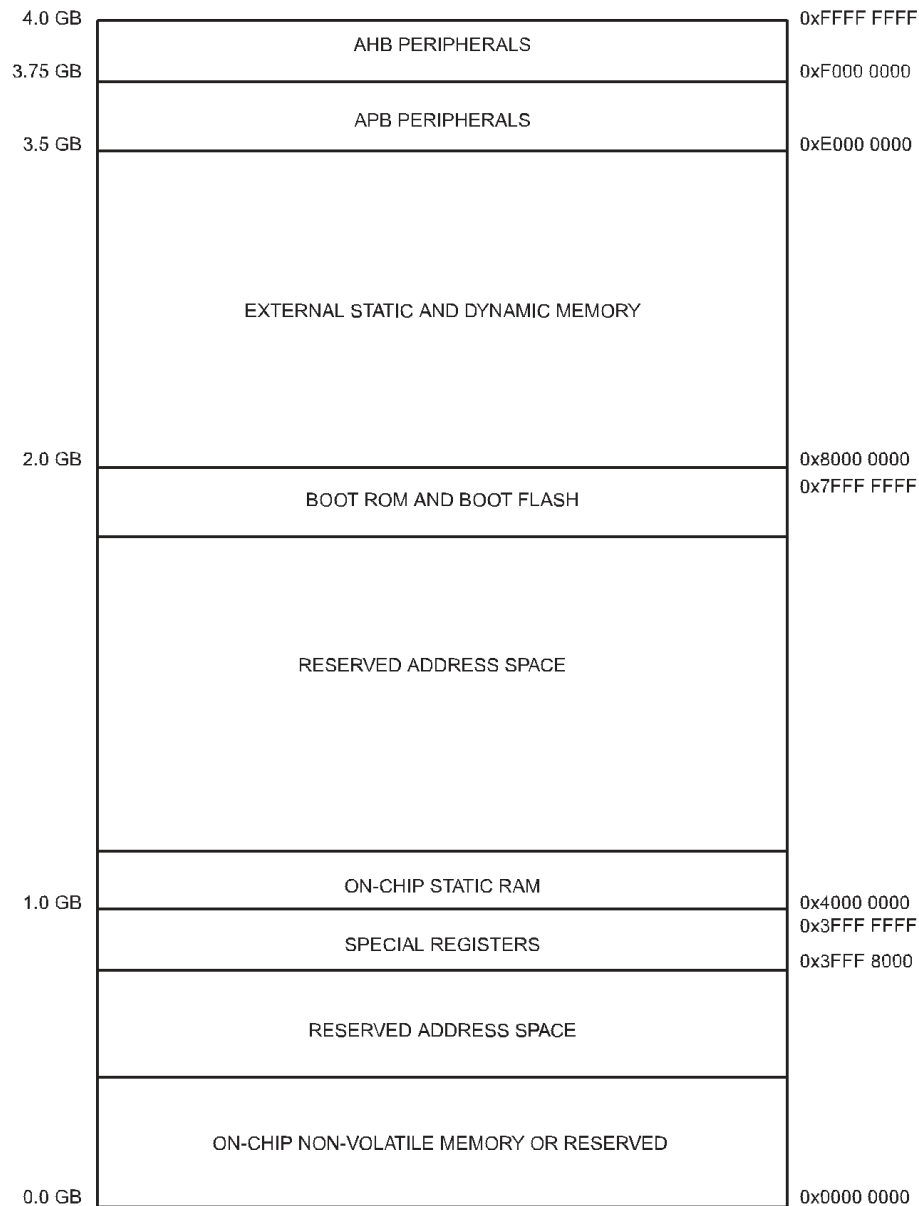
**3. „Advanced Peripheral Bus“ (APB)**

- Low Speed
- Single Master: Bridge
- Sehr einfacher Steuerbus

# LPC2468

CE WS12

## Speicherbelegung







## Adressaufteilung APB (Ausschnitt)

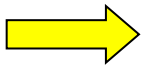
- Jeder Baustein belegt 16 KBytes

APB Peripheral	Base Address	Peripheral Name
0	0xE000 0000	Watchdog Timer
1	0xE000 4000	Timer 0
2	0xE000 8000	Timer 1
3	0xE000 C000	UART0
4	0xE001 0000	UART1
5	0xE001 4000	PWM0
6	0xE001 8000	PWM1
7	0xE001 C000	I <sup>2</sup> C0
8	0xE002 0000	SPI
9	0xE002 4000	RTC
10	0xE002 8000	GPIO
11	0xE002 C000	Pin Connect Block
12	0xE003 0000	SSP1
13	0xE003 4000	ADC
14	0xE003 8000	CAN Acceptance Filter RAM
15	0xE003 C000	CAN Acceptance Filter Registers
16	0xE004 0000	CAN Common Registers
17	0xE004 4000	CAN Controller 1
18	0xE004 8000	CAN Controller 2
19 to 22	0xE004 C000 to 0xE005 8000	Not used
23	0xE005 C000	I <sup>2</sup> C1



## Übersicht

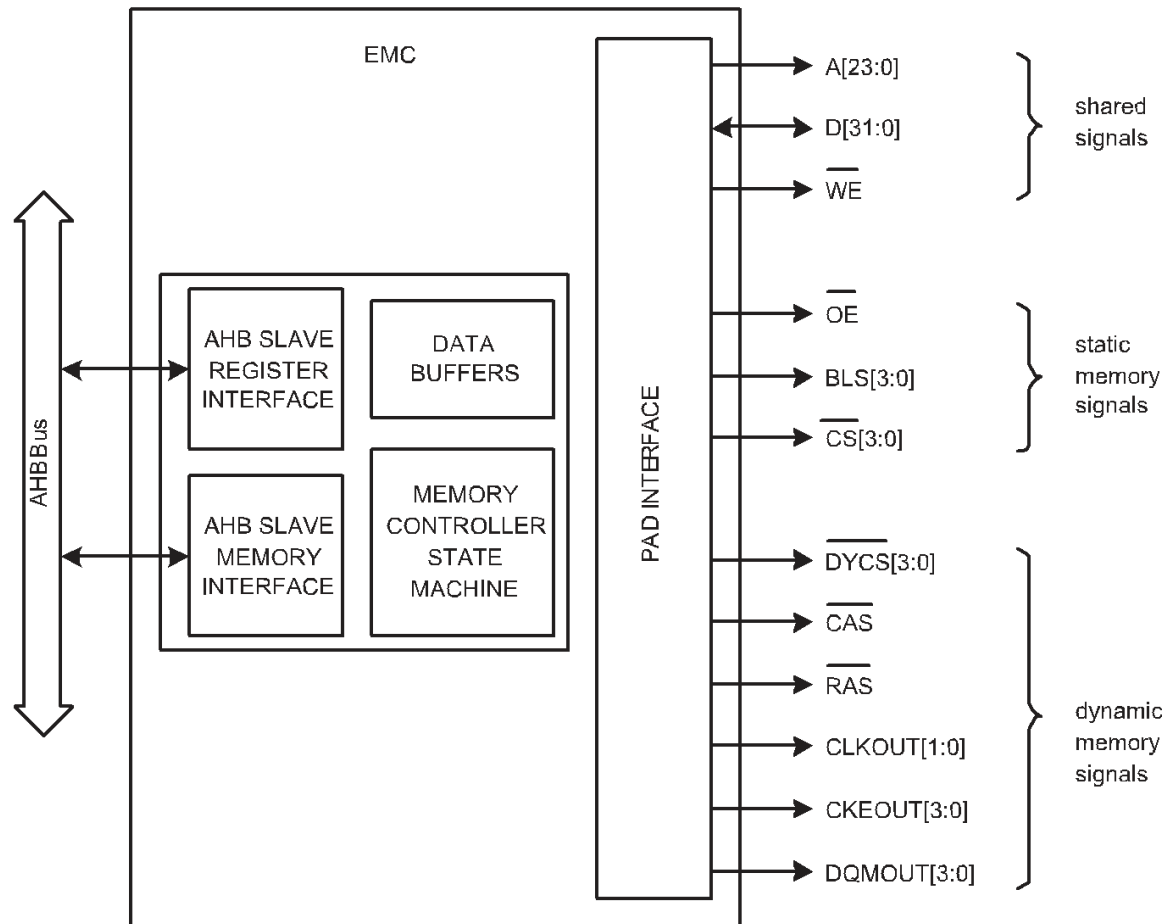
- ▶ Ein- und Ausgabe
  - ▶ Memory-Mapped I/O
  - ▶ I/O-based I/O
- ▶ Bussysteme des LPC2468.
- ▶ Zugriff auf externe Komponenten.  
Typischer Ablauf eines Buszugriffs.
- ▶ Systemeinstellungen für LPC2468-Komponenten.
- ▶ General Purpose I/O (GPIO) LPC2468.





## Externe Speicherschnittstelle

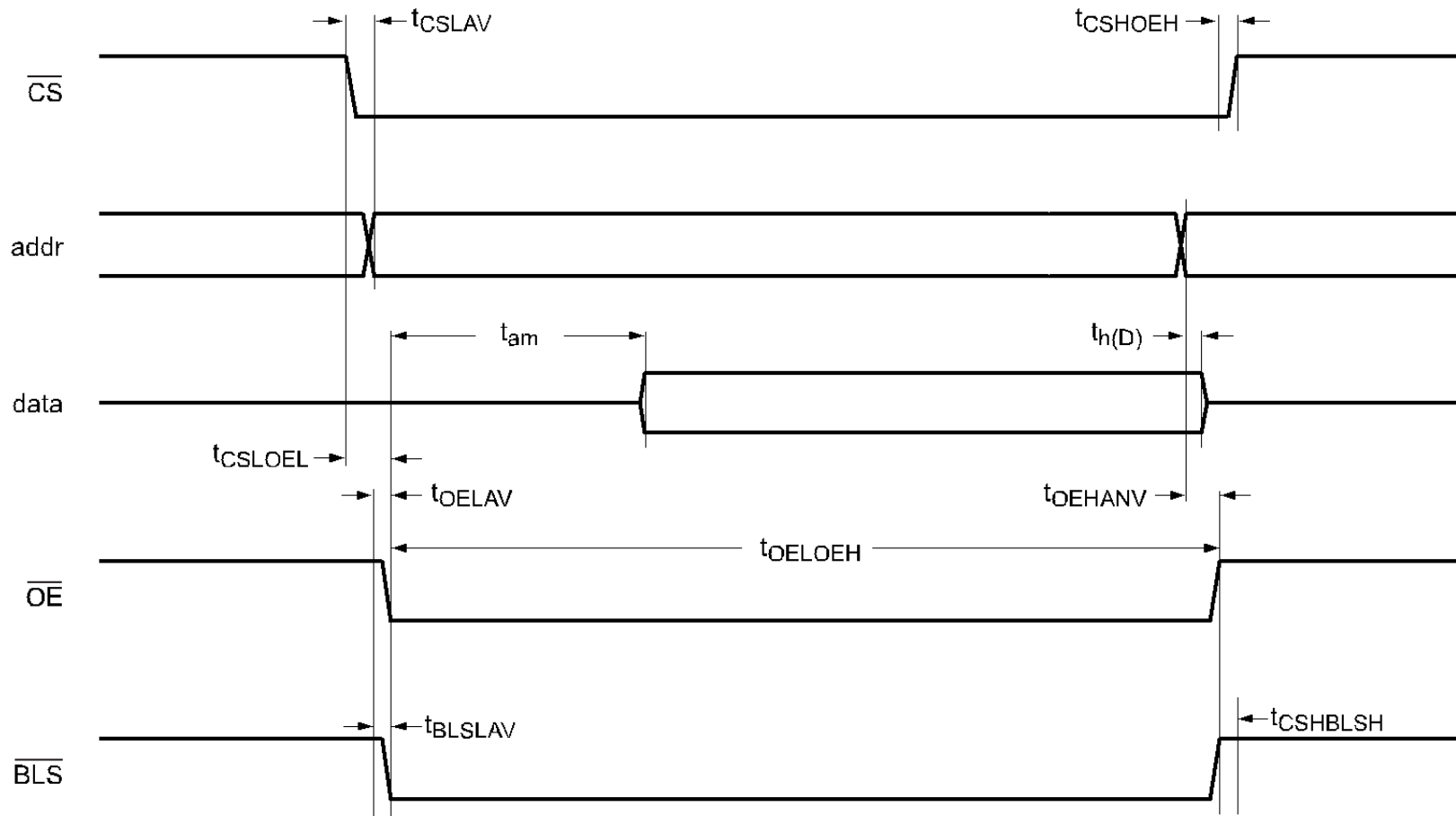
- ▶ Zugriff über AHB1
- ▶ Schnittstelle für
  - ▶ Statisches RAM
  - ▶ Dynamisches RAM
- ▶ Unterschiedliche Busbreiten:
  - ▶ 8 bit
  - ▶ 16 bit
  - ▶ 32 bit



## Speicherbereiche für externe Speicherschnittstelle 8 bit

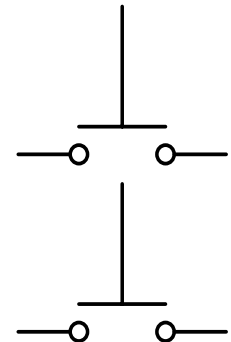
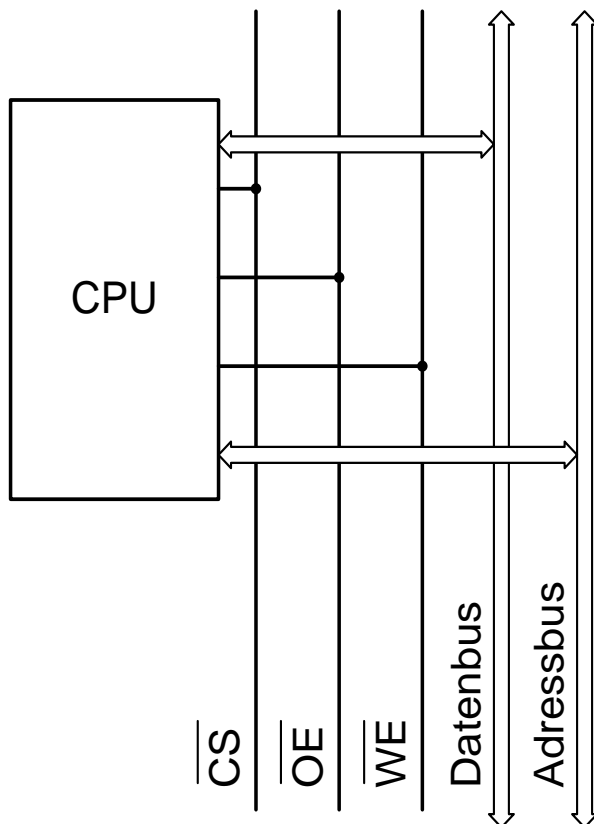
Chip select pin	Address range	Memory type	Size of range
$\overline{\text{CS0}}$	0x8000 0000 - 0x80FF FFFF	Static	16 MB
$\overline{\text{CS1}}$	0x8100 0000 - 0x81FF FFFF	Static	16 MB
$\overline{\text{CS2}}$	0x8200 0000 - 0x82FF FFFF	Static	16 MB
$\overline{\text{CS3}}$	0x8300 0000 - 0x83FF FFFF	Static	16 MB
$\overline{\text{DYCS0}}$	0xA000 0000 - 0xAFFF FFFF	Dynamic	256 MB
$\overline{\text{DYCS1}}$	0xB000 0000 - 0xBFFF FFFF	Dynamic	256 MB
$\overline{\text{DYCS2}}$	0xC000 0000 - 0xCFFF FFFF	Dynamic	256 MB
$\overline{\text{DYCS3}}$	0xD000 0000 - 0xDFFF FFFF	Dynamic	256 MB

## Externe Speicherschnittstelle, Lesezugriff auf statischen Speicher





## Lesezugriff, Anschließen von Tasten



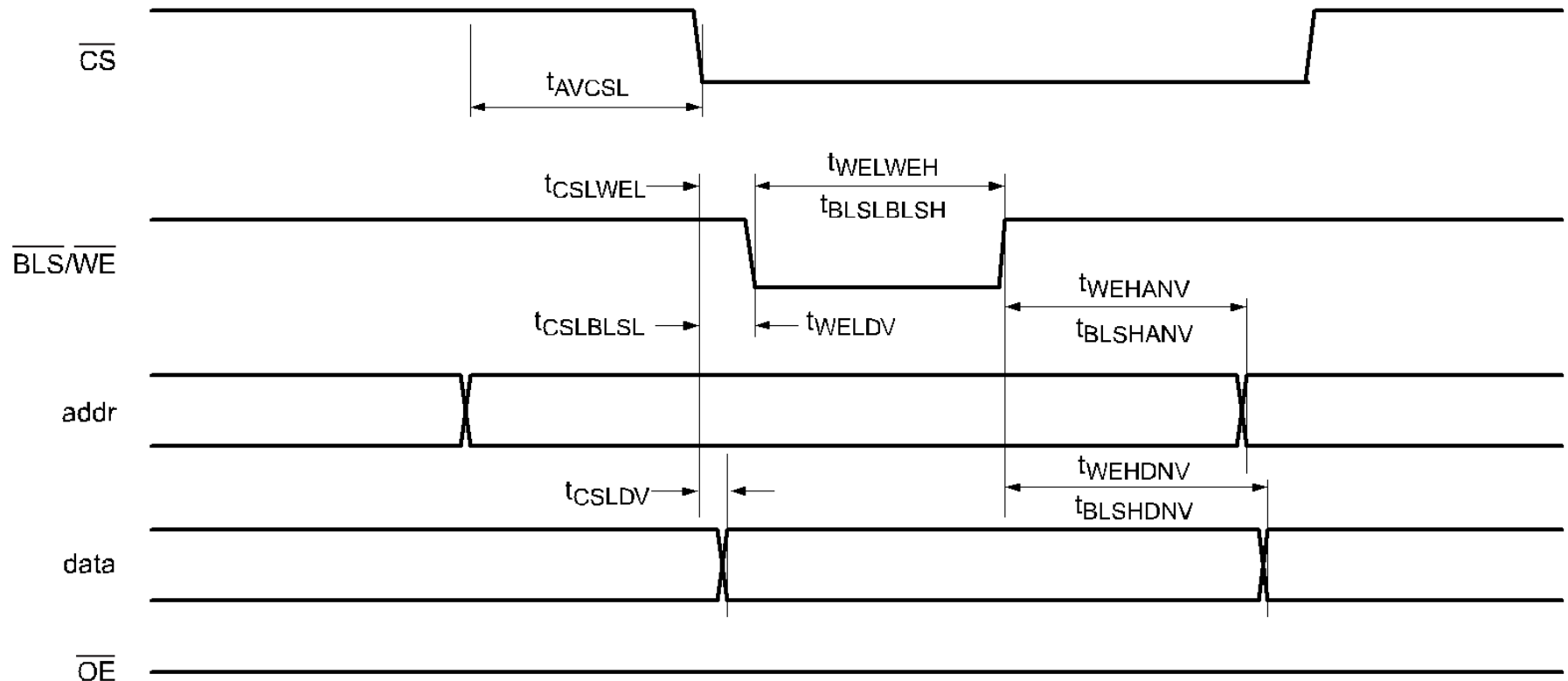


## Lesezugriff

- ▶ **Ausgewählte Speicheradresse:  $CS0 + 0x9 = 0x8000\ 0009$**
- ▶ **Assembler**

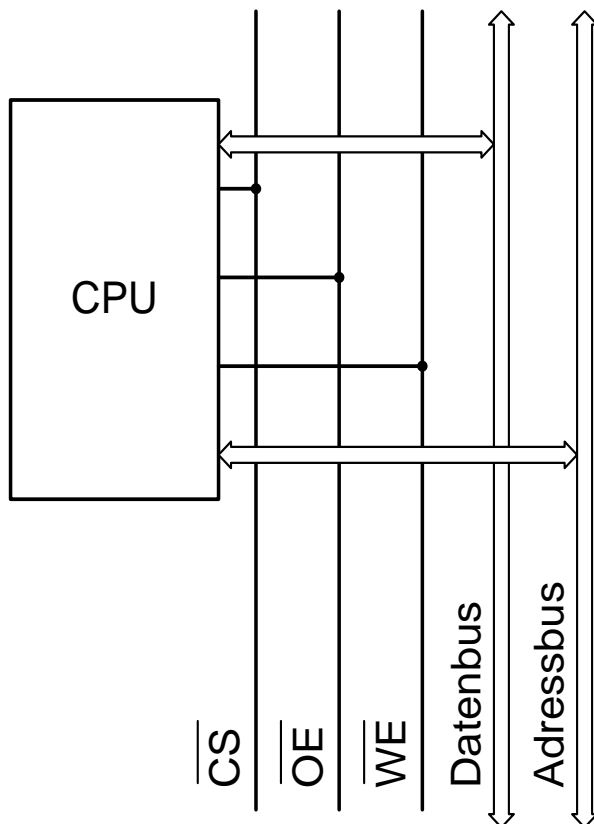
▶ **C**

## Externe Speicherschnittstelle, Schreibzugriff auf statischen Speicher





## Schreibzugriff, Anschließen von Lampen



## Schreibzugriff

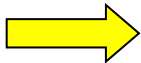
- ▶ **Ausgewählte Speicheradresse:  $CS0 + 0x9 = 0x8000\ 0009$**
- ▶ **Assembler**

▶ **C**



## Übersicht

- ▶ Ein- und Ausgabe
  - ▶ Memory-Mapped I/O
  - ▶ I/O-based I/O
- ▶ Bussysteme des LPC2468.
- ▶ Zugriff auf externe Komponenten.  
Typischer Ablauf eines Buszugriffs.
- ▶ Systemeinstellungen für LPC2468-Komponenten.
- ▶ General Purpose I/O (GPIO) LPC2468.





## Steuerung des Stromverbrauchs

- ▶ **Betriebszustände zum Reduzieren des Stromverbrauchs:**
  - ▶ **Idle mode,**
  - ▶ **Sleep mode,**
  - ▶ **Power-down mode,**
  - ▶ **Deep power-down mode.**
- ▶ **Stromverbrauch wird wesentlich durch die **Taktfrequenz** bestimmt**
  - ▶ **CPU-Taktfrequenz kann per Software verändert werden**
- ▶ **Steuerung des Leistungsverbrauchs von **Peripherie-Bausteinen**:**
  - ▶ **Nicht benötigte Bausteine können ausgeschaltet werden (siehe PCONP-Register).**
  - ▶ **Taktfrequenzen der einzelnen Komponenten lassen sich individuell einstellen (siehe PCLKSEL0 and PCLKSEL1).**



## Ein- und Ausschalten von Peripherie-Komponenten

- ▶ **Einstellung über PCONP:**
  - ▶ **Für jede Komponente steht 1 Bit zur Verfügung:**
    - 0: Komponente ist deaktiviert.
    - 1: Komponente wird mit Takt versorgt.

### Ausschnitt von PCONP:

Bit	Symbol	Description	Reset value
0	-	Unused, always 0	0
1	PCTIM0	Timer/Counter 0 power/clock control bit.	1
2	PCTIM1	Timer/Counter 1 power/clock control bit.	1
3	PCUART0	UART0 power/clock control bit.	1
4	PCUART1	UART1 power/clock control bit.	1
5	PCPWM0	PWM0 power/clock control bit.	1
6	PCPWM1	PWM1 power/clock control bit.	1
7	PCI2C0	The I <sup>2</sup> C0 interface power/clock control bit.	1
8	PCSPI	The SPI interface power/clock control bit.	1

## Takteinstellung Peripherie-Komponenten

- ▶ Einstellung über PCLKSEL0 und PCLKSEL1:
  - ▶ In PCLKSELx stehen für jede Komponente 2 Bits zur Verfügung.
  - ▶ Damit kann jede Komponente auf 4 verschiedene Taktfrequenzen eingestellt werden (siehe Tabelle).
  - ▶ Standard ist CCLK/4, CCLK ist die CPU-Frequenz, z.B.: 48 MHz

### Ausschnitt von PCLKSEL0:

Bit	Symbol	Description
1:0	PCLK_WDT	Peripheral clock selection for WDT.
3:2	PCLK_TIMER0	Peripheral clock selection for TIMER0.
5:4	PCLK_TIMER1	Peripheral clock selection for TIMER1.
7:6	PCLK_UART0	Peripheral clock selection for UART0.
9:8	PCLK_UART1	Peripheral clock selection for UART1.

### Bedeutung der Bits:

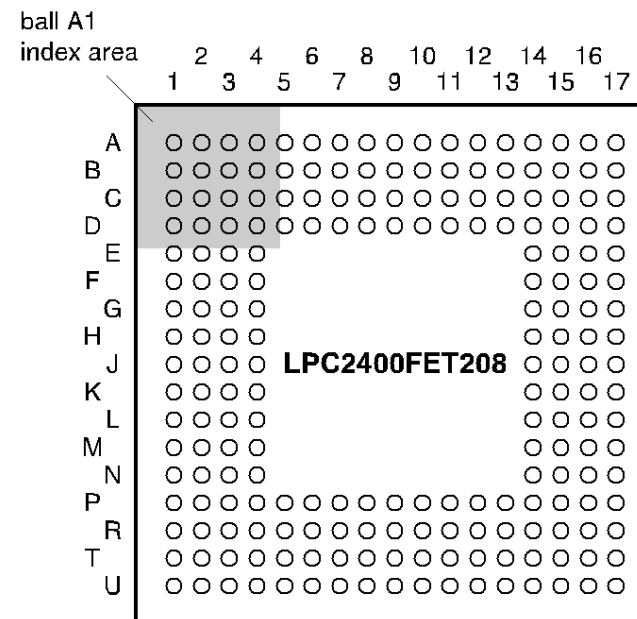
00	PCLK_xyz = CCLK/4
01	PCLK_xyz = CCLK
10	PCLK_xyz = CCLK/2
11	PCLK_xyz = CCLK/8 PCLK_xyz = CCLK/6 CAN1, CAN2, CAN filtering



## Anschlüsse (Pins)

- ▶ Baustein hat 208 Pins
- ▶ Möglich wird dies durch ein Ball Grid Array (BGA)
- ▶ Anschlüsse über kleine Lotkugelnchen (Balls).
- ▶ Dadurch Anordnung in mehreren Reihen und Spalten möglich.

**LPC2400 pinning  
TFBGA208 package**





## Anschlüsse (Pins)

- ▶ Die meisten Pins können zwischen 4 verschiedene Funktionen per Software umgeschaltet werden.
- ▶ Hierzu dienen die 10 Register PINSEL0 bis PINSEL9.
- ▶ Standardmäßig ist die „General Purpose IO“ (GPIO)-Funktion aktiviert.

PINSEL4	Pin name	Function when 00	Function when 01	Function when 10	Function when 11	Reset value
1:0	P2[0]	GPIO Port 2.0	PWM1[1]	TXD1	TRACECLK	00
3:2	P2[1]	GPIO Port 2.1	PWM1[2]	RXD1	PIPESTAT0	00
5:4	P2[2]	GPIO Port 2.2	PWM1[3]	CTS1	PIPESTAT1	00
7:6	P2[3]	GPIO Port 2.3	PWM1[4]	DCD1	PIPESTAT2	00
9:8	P2[4]	GPIO Port 2.4	PWM1[5]	DSR1	TRACESYNC	00
11:10	P2[5]	GPIO Port 2.5	PWM1[6]	DTR1	TRACEPKT0	00
13:12	P2[6]	GPIO Port 2.6	PCAP1[0]	RI1	TRACEPKT1	00
15:14	P2[7]	GPIO Port 2.7	RD2	RTS1	TRACEPKT2	00





## Anschlüsse (Pins)

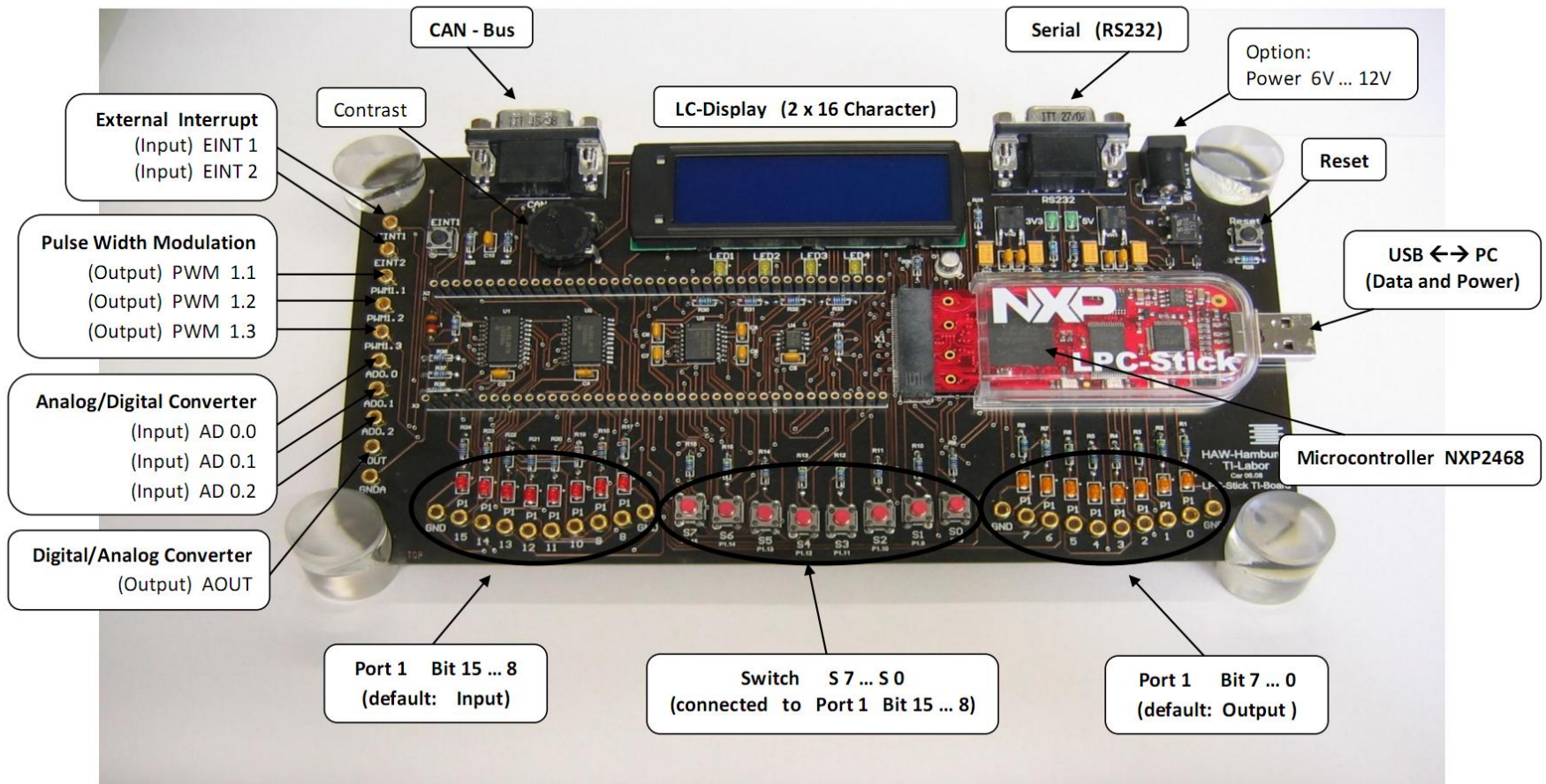
- ▶ Viele Pins können mit einem chipinternen Pull-up oder Pull-down Widerstand beschaltet werden.
- ▶ Hierzu dienen die 10 Register PINMODE0 bis PINMODE9.
- ▶ Standardmäßig ist der Pull-up Widerstand aktiviert.

PINMODE0 to PINMODE9 Values	Function
00	Pin has an on-chip pull-up resistor enabled.
01	Reserved. This value should not be used.
10	Pin has neither pull-up nor pull-down resistor enabled.
11	Pin has an on-chip pull-down resistor enabled.

PINMODE4	Symbol	Description	Reset value
1:0	P2.00MODE	PORT2 pin 0 on-chip pull-up/down resistor control.	00
...			
31:30	P2.15MODE	PORT2 pin 15 on-chip pull-up/down resistor control.	00



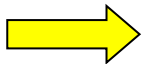
## LPC-Stick TI-Board





## Übersicht

- ▶ Ein- und Ausgabe
  - ▶ Memory-Mapped I/O
  - ▶ I/O-based I/O
- ▶ Bussysteme des LPC2468.
- ▶ Zugriff auf externe Komponenten.  
Typischer Ablauf eines Buszugriffs.
- ▶ Systemeinstellungen für LPC2468-Komponenten.
- ▶ General Purpose I/O (GPIO) LPC2468.





## General Purpose Input/Output

- ▶ Chipselect wird aktiviert beim Zugriff auf die Variable GPIO1\_IOPIN.
- ▶ GPIO1\_IOPIN ist in der Headerdatei lpc24xx.h definiert.

- ▶ **Eingabe:** Alle 32 Bit lesen:

```
int x;  
x = GPIO1_IOPIN;  
printf( "%08X", x );
```

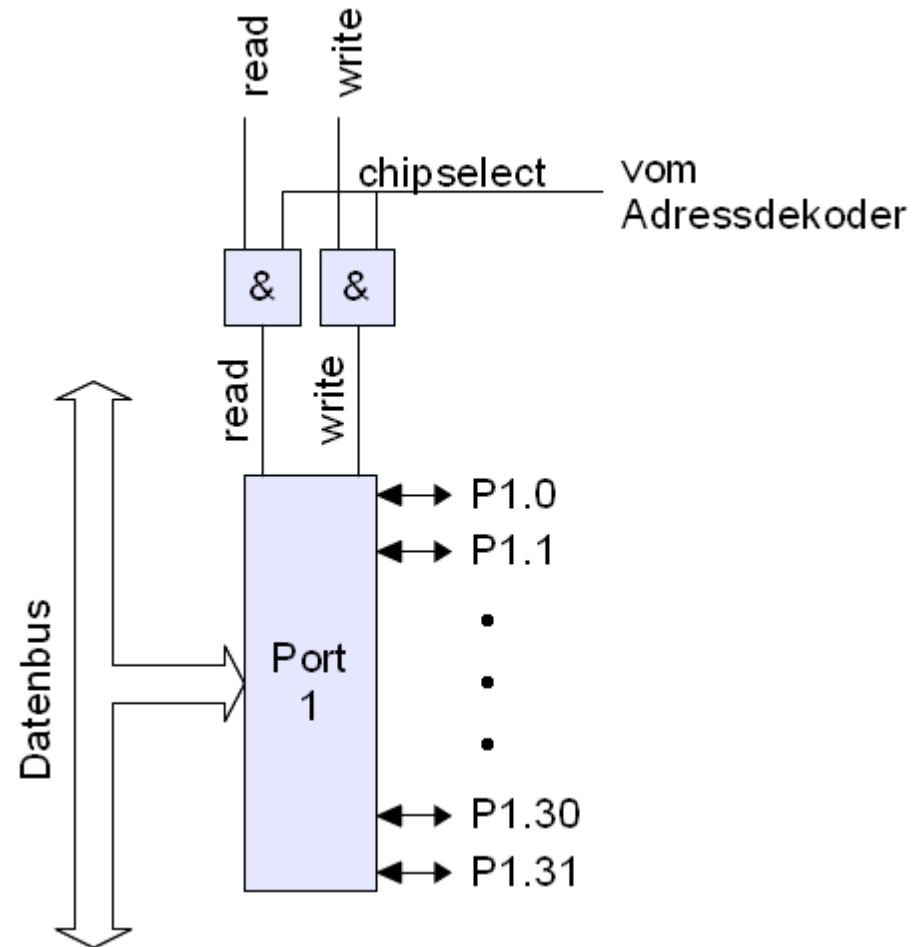
- ▶ **Ausgabe:** Alle 32 Bit verändern:

```
GPIO1_IOPIN = 0x88FF0011;
```

- ▶ **Fazit:**

- ▶ Mit GPIO1\_IOPIN können nur alle 32 Ein-/Ausgabelleitungen gleichzeitig bearbeitet werden.

- ▶ Wie können einzelne Bits behandelt werden?





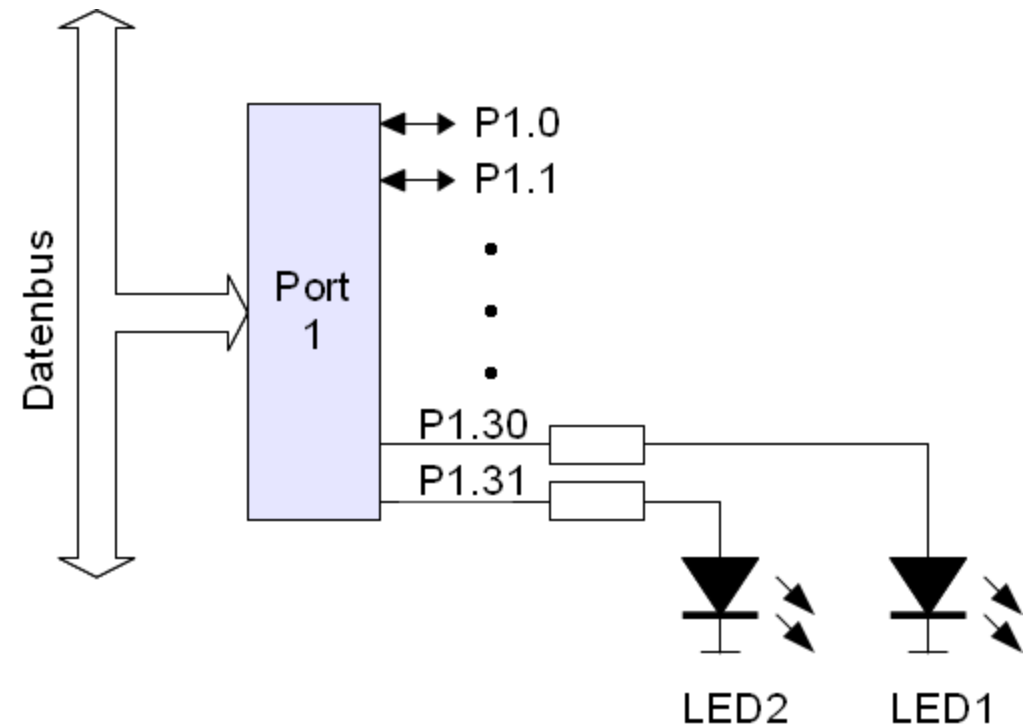
## LPC2468: General Purpose Input/Output

- ▶ **GPIOx\_IODIR**  
steuert individuell die Ein- und Ausgaberrichtung jedes einzelnen Pins.
- ▶ **GPIOx\_IOMASK**  
steuert den Zugriff über die Ports  
GPIOx\_IOPIN, GPIOx\_IOSET und GPIOx\_IOCLR.  
Ermöglicht Auswahl einzelner Bits.  
Bit n == 0: Bit n kann gelesen oder beschrieben werden  
Bit n == 1: Beim Lesen ist dieses Bit immer 0  
Beim Schreiben wird dieses Bit nicht verändert
- ▶ **GPIOx\_IOPIN**  
Lesen des momentanen Zustandes der mit GPIOx\_IOMASK  
ausgewählten Bits.  
Schreiben der mit GPIOx\_IOMASK ausgewählten Bits.
- ▶ **GPIOx\_IOSET**  
Setzen der mit GPIOx\_IOMASK ausgewählten Bits.  
Bit n == "1": Ausgang n wird High (logisch 1).  
Bit n == "0": Ausgang n wird nicht verändert.
- ▶ **GPIOx\_IOCLR**  
Löschen der mit GPIOx\_IOMASK ausgewählten Bits.  
Bit n == "1": Ausgang n wird Low (logisch 0).  
Bit n == "0": Ausgang n wird nicht verändert.

## Ansteuerung einzelner Bits

► LED1 einschalten:

► LED1 ausschalten:



## ÜBUNG: Ein- und Ausgabe, Bitmanipulation

- Erstellen Sie ein Unterprogramm zum Ansteuern der LEDs:

```
void setLed( int leds );
```

mit folgendem Verhalten:

leds	LED1	LED2
0	aus	aus
1	ein	aus
2	aus	ein
3	ein	ein

- a) ohne Ausnutzung der LPC2xxx Hardware
- b) mit Hardwareunterstützung

