Digital Systems

Prof. Manfred Schimmler
Lehrstuhl für Technische Informatik
Institut für Informatik
Christian Albrechts Universität zu Kiel
Tel.: 8804480
E-Mail: masch@informatik.uni-kiel.de

Prof. Dr. Manfred Schimmler

..... may I introduce myself? ......

1980-1981	Siemens AG Berlin
1981-1985	Assistent, CAU Kiel
1985-1987	Research Fellow University Aarhus
1987-1989	Research Fellow ANU Canberra
1989-1996	Geschäftsführer, ISATEC GmbH, Kiel
1996-1997	Fachhochschulprofessor, FH Stralsund
1997-2004	Professor, TU Braunschweig
2004-?	Professor, CAU Kiel

Materials for Lecture

1. Script: The script is new. Therefore, there are still some mistakes in it. These mistakes will be found and corrected during the semester. For every mistake that you have shown me (first time), I will donate a chocolate bar.
2. Assignments are normally available in the website on Wednesday from 20:00 hrs. These assignments are to be submitted up to the second of the following Mondays till 08:00 hrs in the cabinet, situated at the ground floor of the building in Hermann-Rodewald-Str.3.
3. Books and www-Links: Will be given in the lecture

http://www.techinf.informatik.uni-kiel.de/de/lehre/vorlesungen/digital-systems

Assignments

1. Every week, you must work on the assignments by yourselves (no copying). The solutions will be marked with up to 100 Points.
2. The assignment series are given in fixed groups of two. Both partners should thereby, have equal share of the homework.
3. Please ensure that the names, matriculation numbers and the group number are clearly stated in your assignments. If the submitted homework consists of several pages, they have to be clipped or stapled together.

Exam Requirements

1. At the end of the semester, there is an oral examination.

2. Registration for the examination: informally

Examination Materials

- Allowed materials for the examination are: Pen, plain paper
- Not allowed are: Pocket calculators, scripts, books, mobiles, PDAs, computers, tables, printed formula collections

Topics Overview

1. Number Representations in Computers
2. Fundamentals of Digital Circuits
3. The MOS Transistor
4. CMOS Technology and CMOS Gate
5. Combinatorial Circuit
6. Computer Arithmetic
7. Flip-Flops
8. Sequential Circuit
9. Arithmetic Logic Unit
10. Memories
11. The DLX Processor
12. Assembler

1. Number representations in computers

Literatur:

Waldschmidt, K.: Schaltungen der Datenverarbeitung, Teubner, 1980, ISBN 3-519-06108-2

Klar, R.: Digitale Rechenautomaten, de Gruyter, 1976, ISBN 3-110-04194-4

Leonhard, E.: Grundlagen der Digitaltechnik, Hanser Verlag, 1976, ISBN 3-446-12158-7

Polyadic representation of numbers

$$n = \sum_{i=0}^{N-1} b_i * B^i$$

$$= b_{N-1}B^{N-1} + b_{N-2}B^{N-2} + \ldots + b_1 B^1 + b_0 B^0$$

known as B-adic representation of *n*

$b_i \in \{0,1,\ldots,B-1\}$ known as digits

Polyadic representation of numbers

Abbreviations for B-adic representations:

$$(b_{N-1}b_{N-2}...b_1b_0)_B$$

or, when it is clear on which basis it is about:

$$b_{N-1}b_{N-2}...b_1b_0$$

Clause:

The N-digit B-adic representation enables every whole number of $\{0,1,...,B^N-1\}$ to be represented in exactly one way.

Proof:

Every number can be represented

- in at least one way (induction on N)
- in at most one way (counting the combination of numbers)

Hornerscheme

$$n = \sum_{i=0}^{N-1} b_i * B^i$$

$$= b_{N-1}B^{N-1} + b_{N-2}B^{N-2} + ... + b_1B^1 + b_0B^0$$

$$= ((..(b_{N-1}B + b_{N-2})*B + b_{N-3})*B ... + b_1)*B + b_0$$

Conversion of numbers between polyadic systems

1. The method of iterative division with remainder:

n	:	B	=	q_1	Rem	b_0
q_1	:	B	=	q_2	Rem	b_1
q_2	:	B	=	q_3	Rem	b_2
q_3	:	B	=	q_4	Rem	b_3
q_4	:	B	=	q_5	Rem	b_4

. . . .

q_{N-2}	:	B	=	q_{N-1}	Rem	b_{N-2}
q_{N-1}	:	B	=	0	Rem	b_{N-1}

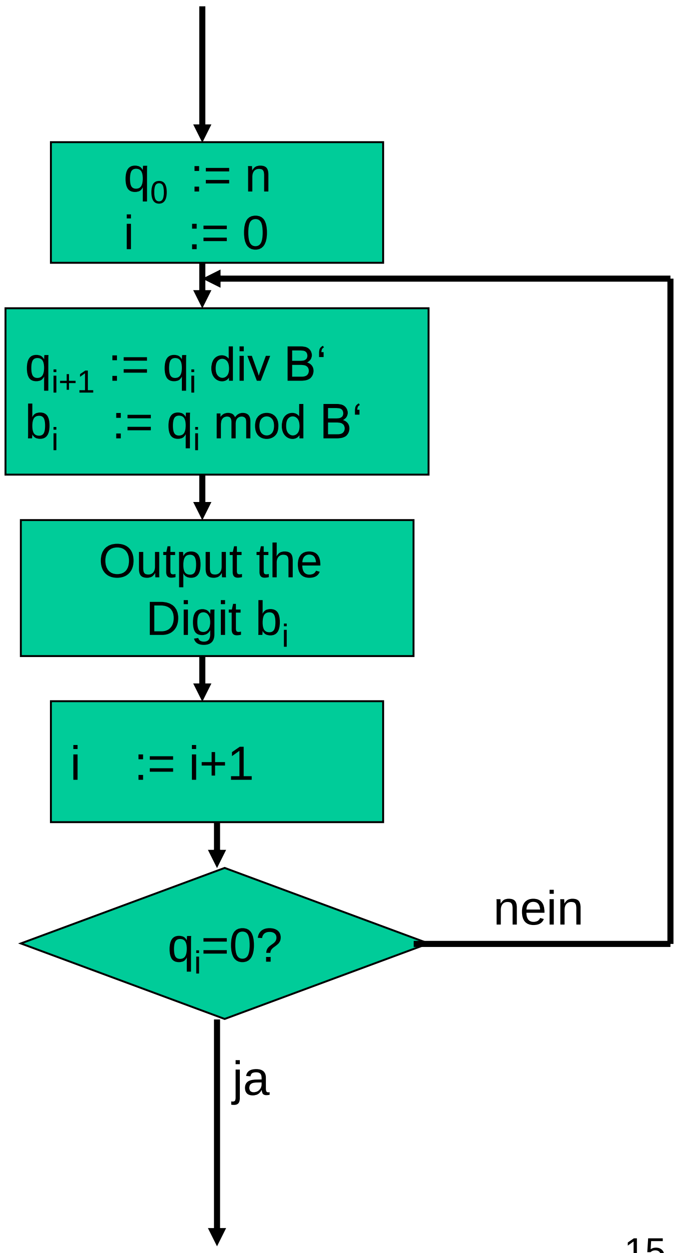
Conversion of numbers between polyadic systems

Calculation in the rootsystem:

$(x)_B \rightarrow (y)_{B'}$

1. Represent the basis B' of the desired system in the root system.
2. $q_0=n$
3. Repeat for ascending i:
   $q_{i+1} = q_i$ div B‘; $r_i = q_i$ mod B‘
   till $q_{i+1}=0$.
4. The r_i are the B‘-adic representation of y

Conversion of numbers from B-adic in the B‘-adic number system through calculations in the root system (B-adic number system)



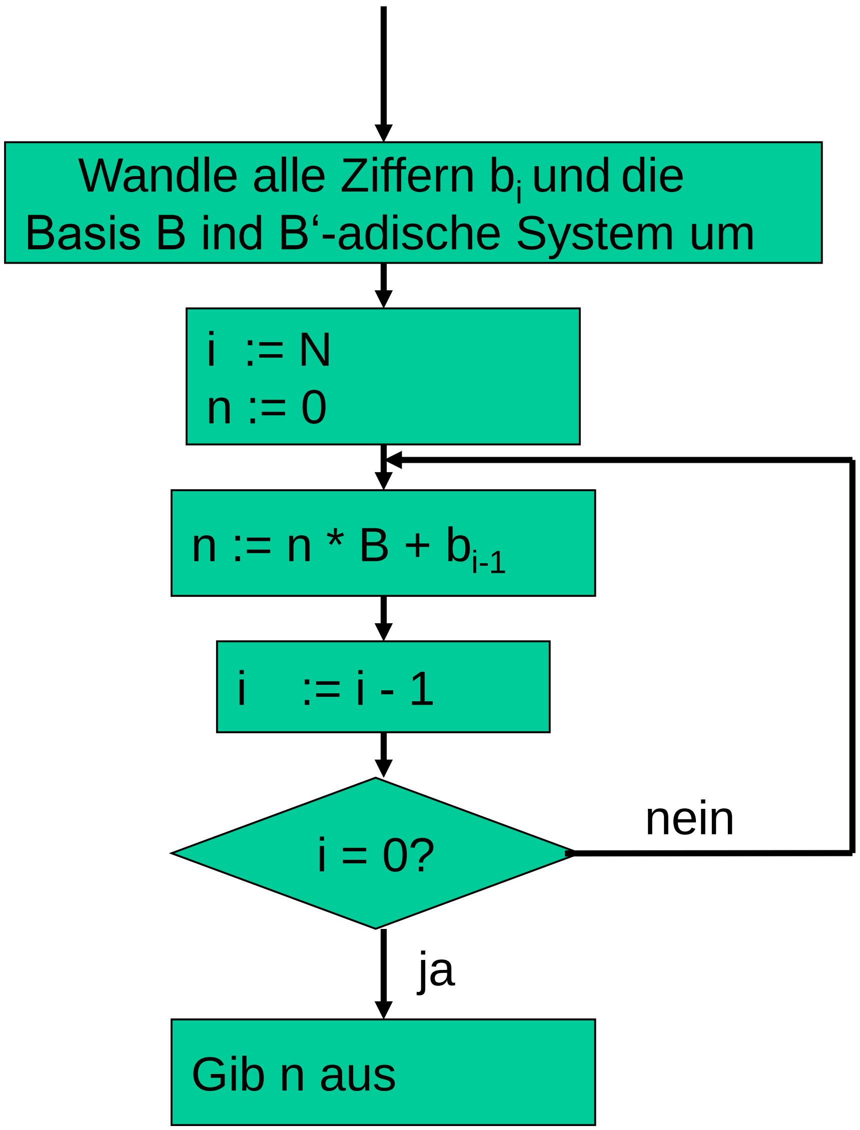
Conversion of numbers between polyadic systems
2. Processing Hornerschemes from left to right:

$$((..(b_{N-1}B+ b_{N-2})*B+ b_{N-3})*B...+b_1)*B+ b_0$$

Calculation in the target system:

1. Conversion of all b_i in the B‘-adic system
2. Conversion of B in the B‘-adic system
3. Compute in the B‘-adic system

Conversion of numbers from the B-adic in the B‘-adic number system through calculations in the target system (B‘-adic number system)



Conversion of numbers between polyadic Systems whose bases are powers of 2

1. Conversion of all digits into the binary system
2. Convert the source number (digit by digit) into a binary number
3. Group up appropriate bits together (LSB first) for always one digit at the desired system
4. Produce all digits of the target system like that

(LSB least significant bit, hence LSB first means: start with the bit with the least priority)

Binary 2-adisch	Ternary 3-adisch	Octal 8-adisch	Decimal 10-adisch	Hexadecimal 16-adisch
0	0	0	0	0
1	1	1	1	1
10	2	2	2	2
11	10	3	3	3
100	11	4	4	4
101	12	5	5	5
110	20	6	6	6
111	21	7	7	7
1000	22	10	8	8
1001	100	11	9	9
1010	101	12	10	A
1011	102	13	11	B
1100	110	14	12	C
1101	111	15	13	D
1110	112	16	14	E
1111	120	17	15	F
10000	121	20	16	10

Definition:

Let n be a natural number, represented as N-digits B-adic number. The B-complement of n is the N-digits B-adic number formed from the last N digit of B^N-n. The B-complement will be interpreted as -n

Conversion of a number into negative number (B-complement):

$$(b_{N-1}b_{N-2}...b_1b_0)_B$$

Each digit b_i will be replaced by the digit $(B-1-b_i)$. After that, one is added to the number that was produced.

4-digits 2's complement-numbers

Decimal 10-adisch	Binary 2-adisch
-8	1000
-7	1001
-6	1010
-5	1011
-4	1100
-3	1101
-2	1110
-1	1111
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111

Representable range of N-digits B-adic numbers in B-complement for even B

$$\{-(B/2)B^{N-1},\ldots\ldots,+(B/2)B^{N-1}-1\}$$

Exactly those numbers beginning with a digit ≥ B/2 are negative.

Clause:

The result of an addition of two N-digits 2-adic numbers will again be (with N positions) within the representable range, if for the sum after the addition, the pre-sign position (position N-1) is in accordance with the securing position (position N).

Representation of Rational Numbers in the Fixed Point Format

$$n = \sum_{i=-M}^{N-1} b_i * B^i$$

$$= b_{N-1}B^{N-1} + b_{N-2}B^{N-2} + \ldots + b_1 B^1 + b_0 B^0$$

$$+ b_{-1}B^{-1} + b_{-2}B^{-2} + \ldots + b_{-M+1}B^{-M+1} + b_{-M}B^{-M}$$

Representable range (if B is even):

$$[-(B/2)*B^{N-1} .. +(B/2)*B^{N-1} - B^{-M}]$$

Hornerscheme for rational numbers

$$n = \sum_{i=-M}^{-1} b_i * B^i$$

$$= b_{-1}B^{-1} + b_{-2}B^{-2} + \ldots + b_{-M+1}B^{-M+1} + b_{-M}B^{-M}$$

$$= ((..(b_{-M}B^{-1} + b_{-M+1})*B^{-1} + b_{-M+2})*B^{-1} \ldots + b_{-1})*B^{-1}$$

Conversion of numbers between polyadic systems

1. Method of the iterative multiplication with truncation:

$$n_1 := n \cdot B \qquad b_{-1} := \lfloor n_1 \rfloor \qquad n'_1 := n_1 - \lfloor n_1 \rfloor$$

$$n_2 := n'_1 \cdot B \qquad b_{-2} := \lfloor n_2 \rfloor \qquad n'_2 := n_2 - \lfloor n_2 \rfloor$$

$$n_3 := n'_2 \cdot B \qquad b_{-3} := \lfloor n_3 \rfloor \qquad n'_3 := n_3 - \lfloor n_3 \rfloor$$

....

$$n_M := n'_{M-1} \cdot B \qquad b_{-M} := \lfloor n_M \rfloor \qquad n'_M := n_M - \lfloor n_M \rfloor$$

Floating point numbers

n = V * 0,Mantissa * 2$^{\text{Exponent}}$

When V = +1 if the sign is + and V = -1, if the sign is –.

The range of representable numbers at m mantissa bits and e exponent bits is

$$\left[\ -\left(1-2^{-m}\right)\cdot 2^{2^{e-1}-1} \bullet\bullet\bullet +\left(1-2^{-m}\right)\cdot 2^{2^{e-1}-1}\ \right]$$

Floating point numbers have the advantage that they cover a much bigger number range as compared to fixed point numbers of the same length.

Furthermore, they are much more precise around or near to zero.

Multiplication of Floating Point Numbers

$N_1 = V_1 * 0{,}M_1 * 2^{E1}$, $N_2 = V_2 * 0{,}M_2 * 2^{E2}$

$N_1 * N_2 = (V_1 * V_2) * 0{,}(M_1 * M_2) * 2^{E1+E2}$

Addition of Floating Point Numbers

$N_1 = V_1 * 0,M_1 * 2^{E1}$, $N_2 = V_2 * 0,M_2 * 2^{E2}$

1. Calculate exponent differences. (e.g. E1>E2). d=E1-E2
2. Shifting of the mantissa M_2 by d positions to the right. $M'_2=M_2 >> d$
3. Addition of the mantissas M_1 and M'_2
4. Calculation of the sign of the result
5. Normalization

$N_1+N_2=(V)*0,(M_1+M'_2)*2^{E1}$

IEEE 754 Format 32-Bit (float, single)

1 sign bit
8 Exponent bits (MSB first)
23 Mantissa bits (MSB first)

The value w of such a number can be calculated by:

$w = (-1)^V*(1,M)*2^{E-127}$,	when E>0 and E<255
$w = (-1)^V*(0,M)*2^{-126}$,	when E=0 and M≠0
$w = (-1)^V*0$,	when E=0 and M=0
$w = (-1)^V*\text{Infinity}\ (\infty)$,	when E=255 and M=0
w = NaN (Not a number),	when E=255 and M≠0

Representable range approximately $[-10^{38}..+10^{38}]$

IEEE 754 Format 64-Bit (double)

1 sign bit
11 Exponent bits (MSB first)
52 Mantissa bits (MSB first)

The value w of such a number can be calculated by:

$w = (-1)^V * (1,M) * 2^{E-1023}$,	when $E>0$ and $E<2047$
$w = (-1)^V * (0,M) * 2^{-1022}$,	when $E=0$ and $M \neq 0$
$w = (-1)^V * 0$,	when $E=0$ and $M=0$
$w = (-1)^V *$ Infinity (∞),	when $E=2047$ and $M=0$
w = NaN (Not a number),	when $E=2047$ and $M \neq 0$

Representable range approximately $[-10^{300} .. +10^{300}]$

IEEE 754 Format 80-Bit (extended)

1 sign bit
15 Exponent bits (MSB first)
64 Mantissa bits (MSB first)

The value w of such a number can be calculated by:

$w = (-1)^V*(0,M)*2^{E-16383}$, when $E>0$ and $E<32767$
$w = (-1)^V*\text{Infinity}\ (\infty)$, when $E=32767$ and $M=0$
w = NaN (Not a number), when $E=32767$ and $M\neq 0$

Representable Range approximately $[-10^{5000}..+10^{5000}]$

Coding of the Decimal Numbers

Decimal-Digit	Binary	Aiken	3-Excess	2aus5
0	0000	0000	0011	11000
1	0001	0001	0100	00011
2	0010	0010	0101	00101
3	0011	0011	0110	00110
4	0100	0100	0111	01001
5	0101	1011	1000	01010
6	0110	1100	1001	01100
7	0111	1101	1010	10001
8	1000	1110	1011	10010
9	1001	1111	1100	10100
Weights	8421	2421	None	74210

EBCDIC (Extended Binary Coded Decimal Interchange Code)

		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
0	0000																
1	0001																
2	0010																
3	0011																
4	0100	blank										§	.	<	(	+	|
5	0101	&										!	$	●	)	;	
6	0110	-	/									^	,	%		>	?
7	0111											:	#	@	'	*	"
8	1000		a	b	c	d	e	f	g	h	i						
9	1001		j	k	l	m	n	o	p	q	r						
A	1010			s	t	u	v	w	x	y	z						
B	1011																
C	1100		A	B	C	D	E	F	G	H	I						
D	1101		J	K	L	M	N	O	P	Q	R						
E	1110			S	T	U	V	W	X	Y	Z						
F	1111	0	1	2	3	4	5	6	7	8	9						

ASCII (American Standard Code for Information Interchange)

		000	001	010	011	100	101	110	111
0	0000	NUL	DLE	SP	0	@	P	‘	p
1	0001	SOH	DC1	!	1	A	Q	a	q
2	0010	STX	DC2	“	2	B	R	b	r
3	0011	ETX	DC3	#	3	C	S	c	s
4	0100	EOT	DC4	$	4	D	T	d	t
5	0101	ENQ	NAK	%	5	E	U	e	u
6	0110	ACK	SYN	&	6	F	V	f	v
7	0111	BEL	ETB	´	7	G	W	g	w
8	1000	BS	CAN	(	8	H	X	h	x
9	1001	SKIP	EM	)	9	I	Y	i	y
A	1010	LF	SUB	*	:	J	Z	j	z
B	1011	VT	ESC	+	;	K	[	k	{
C	1100	FF	FS	,	<	L	\	l	|
D	1101	CR	GS	-	=	M	]	m	}
E	1110	SO	HOME	.	>	N	^	n	~
F	1111	SI	NL	/	?	O	_	o	DEL