System software

Code generation

Generating code

- What can we generate?
  - generate raw machine code
  - generate object code
  - print symbol table
  - print program (as .lst file)
  - print .log file
  - etc.

Generating raw code

- Visitation
  - `Node.emitCode(byte[] data, int pos)`
  - `byte[] Code.emitCode()`
- Memory image
  - can be copied into the main memory of SIC/XE
  - afterwards it can be executed
  - not relocatable

Generating object file

- Visitation
  - transform raw code to hex notation
  - some commands must be dealt with separately
- Object file
  - see also slides for „Loader“
  - start and end have **H** and **E** record
  - actual code is stored within **T** records

Generating lst file

- Contents
  - address, raw code in hex, label, command, operands
- Enables
  - ease of checking the generated code
  - ease of debugging
  - ease of optimization

Generating log file

- Contents of the file
  - list of blocks
  - list of symbols
  - list of literals
  - list of relocation

Relocatable code

- Resolving code
  - When do we need relocation?
  - Relocation table
- Generating object code
  - **M** records are generated from the relocation table

```
0000:                          . Rutine za zaslon
0000:              ScrSub      START  0
0000: B400         first       CLEAR  A
0002: 53201E       loop1       LDCH   =C'*'
0005: 4B203D                   JSUB   scrfill
0008: 4B2037                   JSUB   scrclear
000B: 3F2FF4                   J      loop1
000E: 010005                   LDA    #5
0011: 6D0004                   LDS    #4
0014: 4B2011                   JSUB   scrgoto
0017: 53200A       loop        LDCH   =C'A'
001A: 4B2016                   JSUB   scrch
001D: 3F2FF7                   J      loop
0020: 3F2FFD       halt        J      halt
0020:                          LTORG
0023: 2A           *0          BYTE   C'*'
0024: 41           *1          BYTE   C'A'
```

```
HScrSub00000000B853
T000000 1D B40053201E4B203D4B20373F2FF40100056D00044B201153200A4B2016
T00001D 1D 3F2FF73F2FFD2A41000000212025904 00F2FF54F0000072FEF5790B800
T00003A 1D B800132FE64F0000532011B4106D27865790B800B8403B2FF74F000020
T00B850 03 030001
M000037 05
M00004B 05
E000000
```

```
0025:                          . ********** Screen
0025:              scrcols      EQU     80
0025:              scrrows      EQU     25
0025:              scrlen       EQU     2000
0025: 000000       scrpos       WORD    X'000000'
0028:                          . screen goto A=row S=col
0028: 212025       scrgoto      MUL     #scrcols
002B: 9040                      ADDR    S,A
002D: 0F2FF5                    STA     scrpos
0030: 4F0000                    RSUB
0033:                          . print A=ch
0033: 072FEF       scrch        LDX     scrpos
0036: 5790B800                 +STCH    screen,X
003A: B800                      TIXR    A
003C: 132FE6                    STX     scrpos
003F: 4F0000                    RSUB
0042:                          . clear/fill screen
0042: 532011       scrclear     LDCH    =C' '
0045: B410         scrfill     CLEAR   X
0047: 6D2786                    LDS     #scrlen
004A: 5790B800     scrclearl   +STCH    screen,X
004E: B840                      TIXR    S
0050: 3B2FF7                    JLT     scrclearl
0053: 4F0000                    RSUB
0053:                           LTORG
0056: 20           *2           BYTE    C' '
0056:                           ORG     47104
B800: 00....00     screen       RESB    80
B850: 030001                    LDA     1
B850:                           END     first
```

```
HScrSub00000000B853
T000000 1D B40053201E4B203D4B20373F2FF40100056D00044B201153200A4B2016
T00001D 1D 3F2FF73F2FFD2A4100000021202590400F2FF54F0000072FEF5790B800
T00003A 1D B800132FE64F0000532011B4106D27865790B800B8403B2FF74F000020
T00B850 03 030001
M000037 05
M00004B 05
E000000
```