```
RAM Assembler V.3

Pass... 2          ORG &8000

8000:              .number=&0e
8000:              .charout=&bb5a
8000:3E 0E         LD A,number
8002:0E 30         LD C,&30
8004:              .hundreds
8004:D6 64         SUB &64
8006:0C            INC C
8007:30 FB         JR NC,hundreds
8009:0D            DEC C
800A:C6 64         ADD A,&64
800C:47            LD B,A
800D:79            LD A,C
800E:CD 5A BB      CALL charout
8011:0E 30         LD C,&30
8013:78            LD A,B
8014:              .tens
8014:D6 0A         SUB &a
8016:0C            INC C
8017:30 FB         JR NC,tens
8019:0D            DEC C
801A:C6 0A         ADD A,&a
801C:47            LD B,A
801D:79            LD A,C
801E:CD 5A BB      CALL charout
8021:0E 30         LD C,&30
8023:78            LD A,B
8024:              .units
8024:D6 01         SUB &1
8026:0C            INC C
8027:30 FB         JR NC,units
8029:0D            DEC C
802A:79            LD A,C
802B:CD 5A BB      CALL charout
802E:C9            RET
802F:              END
```

code assembled from here

the number to be decimalised

output routine

number placed in A register

ASC II offset

100 is subtracted and a count kept in C

loop if carry flag not set

hundreds routine

corrects "overshoot"

tens and units stored

hundreds count displayed

remain of number retrieved and ASC II offset restored

ten subtracted from accumulator

tens routine

units routine

one is subtracted

display units figure

return to where called

locations

opcodes

assembler