```
DD03 DIRECTION  DD01 PORT    6000 START   6008 TEXT    602F READDATA  6048 LOOP1

604C SENDLOW  6064 SENDHIGH  6079 CONT    608D WAITTV  6092 LOOPTV  6099 WAITTI
609E LOOPTI  60A5 WAITTS  60AA LOOPTS  01

6  A                ;NAME:    SER.SENDERV2.SRC
                    ;DATE:    OCTOBER 15TH 1989
                    ;AUTHOR:  OLIVER KALTSTEIN

                    ;VERSION:        V2 (NEEDS RAM)

                    ;PREPARATION: LOAD ACCU WITH CHARACTER,
                    ;             POSITION-COUNTER X WILL BE SAVED.

6000                ORG$6000

DD03         DIRECTION =  56579
DD01         PORT   =    56577


6000 78      START  SEI
6001 18             CLC
6002 D8             CLD
6003 A200           LDX #00
6005 4C2F60         JMP READDATA
6008 48414C TEXT    B   "HALLO THOMAS, HALLO OLIVER. ICH LEBE!  "
602F BC0860 READDATA LDY TEXT,X
6032 E8             INX
6033 8A             TXA
6034 48             PHA
6035 98             TYA
6036 A207           LDX #07
6038 8E03DD         STX DIRECTION                P0 TO P2 ARE OUTPUTS
603B 20A560         JSR WAITTS                   DELAY ALLOWS RECEIVER TO SYNCRONIZE I
F BYTE LOST
603E A204           LDX #04
6040 8E01DD         STX PORT                     SET ATN
6043 209960         JSR WAITTI
6046 A008           LDY #08
6048 18      LOOP1  CLC
6049 2A             ROL
604A B018           BCS SENDHIGH                 IF MSB IS ONE IT WILL BE SEND AS HIGH
604C A204    SENDLOW LDX #04                     SET DATA LOW
604E 8E01DD         STX PORT
6051 EA             NOP
6052 EA             NOP
6053 EA             NOP
6054 A206           LDX #06                      SET CLOCK HIGH
6056 8E01DD         STX PORT
6059 208D60         JSR WAITTV
605C A204           LDX #04
605E 8E01DD         STX PORT
6061 4C7960         JMP CONT

6064 A205    SENDHIGH LDX #05
6066 8E01DD         STX PORT                     SET DATA HIGH BUT CLOCK LOW
6069 EA             NOP
606A EA             NOP
606B EA             NOP
606C A207           LDX #07
606E 8E01DD         STX PORT                     PUT CLOCK HIGH
6071 208D60         JSR WAITTV
6074 A205           LDX #05
6076 8E01DD         STX PORT
6079 20A560 CONT    JSR WAITTS
607C 88             DEY
```

```
607D D0C9          BNE LOOP1
607F A200          LDX #00                   END OFF BYTE, ALL LINES LOW
6081 8E01DD        STX PORT
6084 68            PLA
6085 AA            TAX
```

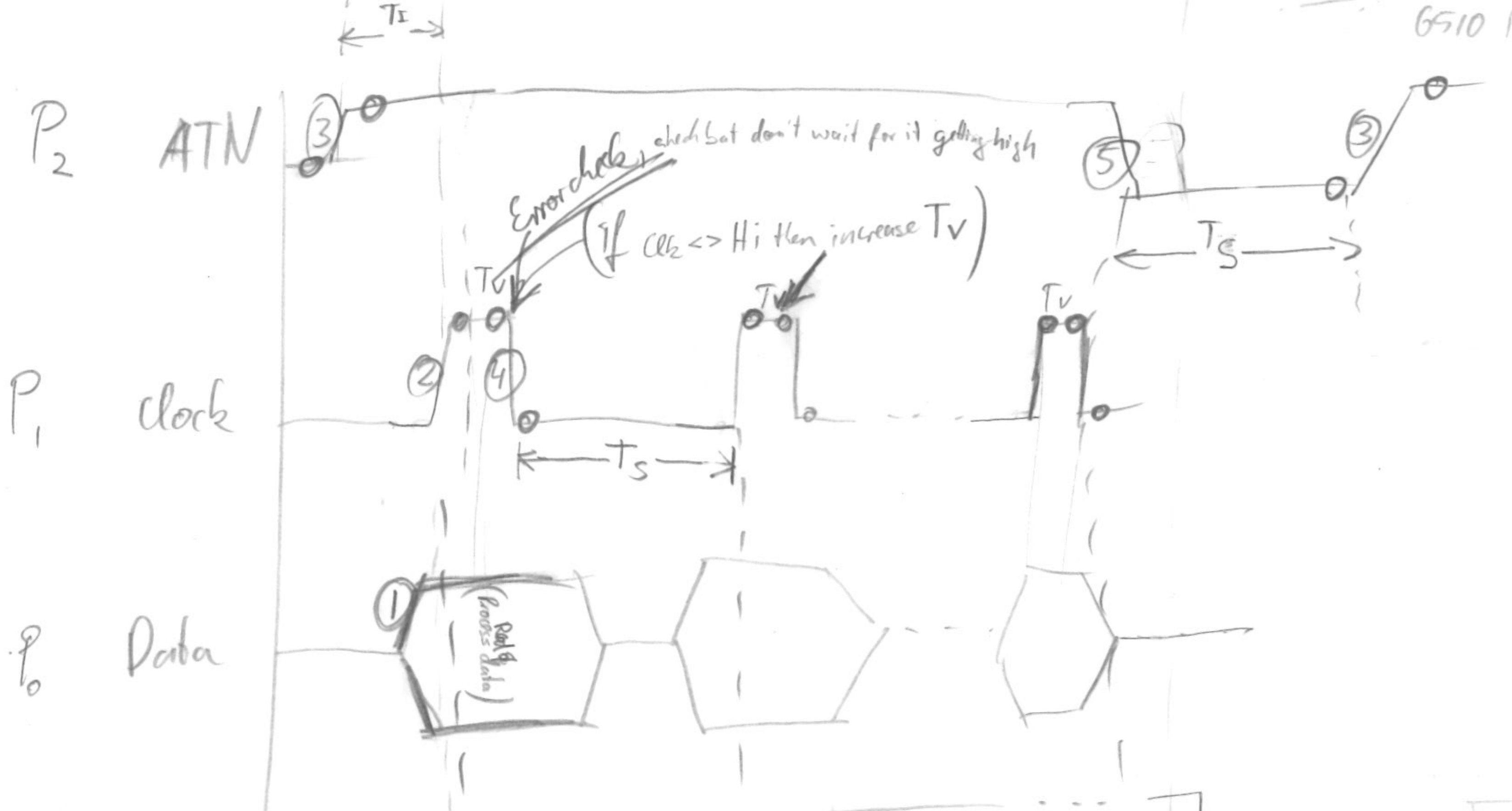
02

```
6086 E027          CPX #39
6088 D0A5          BNE READDATA
608A 4C0060        JMP START

608D 48     WAITTV PHA
608E 8A            TXA
608F 48            PHA
6090 A2A0          LDX #$A0
6092 CA     LOOPTV DEX
6093 D0FD          BNE LOOPTV
6095 68            PLA
6096 AA            TAX
6097 68            PLA
6098 60            RTS
6099 48     WAITTI PHA
609A 8A            TXA
609B 48            PHA
609C A240          LDX #$40
609E CA     LOOPTI DEX
609F D0FD          BNE LOOPTI
60A1 68            PLA
60A2 AA            TAX
60A3 68            PLA
60A4 60            RTS
60A5 48     WAITTS PHA
60A6 8A            TXA
60A7 48            PHA
60A8 A2FF          LDX #$FF
60AA CA     LOOPTS DEX
60AB D0FD          BNE LOOPTS
60AD 68            PLA
60AE AA            TAX
60AF 68            PLA
60B0 60            RTS
```

OCT 10th 1989

SERIAL LINK Between
6510 Project & C64 Terminal


T_I
P2
ATN
③
⑤
③
Errorcheck, check but don't wait for it getting high
(If Clk <> Hi then increase Tv)
Tv
Tv
Tv
T_S
P1
clock
②
④
T_S
P0
Data
①
(Read & process data)

Tv = Data Valid

T_I = Time for intrupt to realise that byte has to be received

T_S = Time for synchronisation

O = check = will wait until potential reached

③ ATN on
- Wait T_I
① Data (1st)
② Clock on
~~③ ATN on~~
- Wait Tv
④ Clock off
- Wait T_S

① Data (...)
② Clock on
- Wait Tv
④ Clock off
- Wait T_S

① Data (8th)
② Clock on
- Wait Tv
④ Clock off
⑤ ATN off
- Wait T_S

repeat process
(JMP start)

- Grenze der Kabelqualität finden, dann aber nur 1/5 Geschwindigkeit gebrauchen

note: - Die Leistung dieser Routine ist schwach, dafür braucht sie aber kein RAM.
- Bitte noch mal schreiben wenn Computer auch RAM besitzt!
- Stack Pointer wird zur Datenspeicherung mißbraucht da kein RAM/Stack vorhanden

Pseudocode for SER.SENDER.SRC

OKT 10th 1989

preparations: - Load accu with Databyte
- X will be stored in SP temporarly and returned into X after transmition

Hole nächstes Zeichen

```
~~TAX~~
- TXS
  - set port to Output $P_0 - P_2$
 - LD~~X~~ #04
- ATN on (ST~~X~~ Direction ~~PORT~~)
          LDY #08
~~TXA~~   TAX
          TXA
Y=8to1 - ROL        clear carry
    if carry = 1 then
        LDX #05 (CLK=1)
        STX PORT
        NOP
        NOP
        LDX # 07
         STX PORT
          JMP cont
         LDX # 06 (CLK=1)
         STX PORT
cont  - Wait NOP1
        LD~~X~~ #04          lösche CLK zwangsweise im port
        ST~~X~~ Port
       - Wait NOP2
         Dey                   TAX
                               ~~TXA~~ TYA
                               BNE
        LDX #00
        STX PORT
      - ~~Wait~~NOP3
        ~~Wait~~ TSX
NOP1  2k NOP            = 2 ms
      JMP NOP1 Cont

NOP2  2k NOP            = 2ms

      Jmp NOP2 Cont
NOP3  2k NOP            2ms
      JMP NOP3 cont
- ~~Input when not sending!~~
```