Assembly Listing

```
                ;++++++++++++++++++++
                ;++++++++++++++++++++
                ;++                ++
                ;++   LINESUB 64   ++
                ;++                ++
                ;++++++++++++++++++++
                ;++++++++++++++++++++

                PLTSUB = $C183
                XLO    = $C103
                XHI    = $C104
                YLO    = $C105
                *      = $C300

                ;++++ LINESUB VARIABLES ++++
                ;
                X1LO   *=*+1
                X1HI   *=*+1
                X2LO   *=*+1
                X2HI   *=*+1
                Y1     *=*+1
                Y2     *=*+1
                DXLO   *=*+1
                DXHI   *=*+1
                DY     *=*+1
                TEMP   *=*+1
                TOTLO  *=*+1
                TOTHI  *=*+1
                NEGREG *=*+1
                DECFLG *=*+1
                ;
                ;++++ PUSH REGISTERS ONTO STACK ++++
                ;
48              PHA
8A              TXA
48              PHA
98              TYA
48              PHA
                ;
                ;++++ INITIALISE VARIABLES ++++
                ;
                START
A9 00           LDA #$00
8D 0D C3        STA DECFLG
8D 0C C3        STA NEGREG
8D 0A C3        STA TOTLO
8D 0B C3        STA TOTHI
                ;
                ;++++ CALCULATE DX ++++
AD 02 C3        LDA X2LO
38              SEC
ED 00 C3        SBC X1LO
8D 06 C3        STA DXLO
AD 03 C3        LDA X2HI
4C 86 C4        JMP FINISH
                ;
                NODEY
C8              INY
CC 05 C3        CPY Y2
90 98           BCC PLOT
                ;
4C 86 C4        JMP FINISH
                ;++++ GRADIENTS LESS THAN ONE ++++
                ;
                GLESS
AD 00 C3        LDA X1LO
8D 03 C1        STA XLO
AD 01 C3        LDA X1HI
8D 04 C1        STA XHI
98              TYA
8D 05 C1        STA YLO
20 83 C1        JSR PLTSUB
                ;
AD 0A C3        LDA TOTLO
18              CLC
6D 08 C3        ADC DY          ; TOTAL=TOTAL+DY
8D 0A C3        STA TOTLO
AD 0B C3        LDA TOTHI
69 00           ADC #$00        ; CATCH CARRY
8D 0B C3        STA TOTHI
                ;
AD 0A C3        LDA TOTLO
38              SEC
ED 06 C3        SBC DXLO
8D 0A C3        STA TOTLO   TOTAL=TOTAL-DX
AD 0B C3        LDA TOTHI
ED 07 C3        SBC DXHI
8D 0B C3        STA TOTHI
30 0F           BMI ADD         ;NEGATIVE RESULT
                ;
                ;
AD 0D C3        LDA DECFLG
C9 01           CMP #$01
F0 04           BEQ DECY
C8              INY
4C 68 C4        JMP PAST
                DECY
88              DEY
4C 68 C4        JMP PAST
                ;
                ADD
AD 0A C3        LDA TOTLO
18              CLC
6D 06 C3        ADC DXLO        ; ADD BACK DX
8D 0A C3        STA TOTLO
AD 0B C3        LDA TOTHI
6D 07 C3        ADC DXHI
8D 0B C3        STA TOTHI
ED 01 C3        SBC Y1HI
8D 07 C3        STA DXHI
10 08           BPL NBRWDX
AD 0C C3        LDA NEGREG
09 02           ORA #$02        ;FLAG DX NEGATIVE
8D 0C C3        STA NEGREG
                ;
                ;++++ CALCULATE DY ++++
                ;
                NBRWDX
AD 05 C3        LDA Y2
38              SEC
ED 04 C3        SBC Y1
8D 08 C3        STA DY
B0 08           BCS NBRWDY
AD 0C C3        LDA NEGREG
09 01           ORA #$01        ;FLAG DY NEGATIVE
8D 0C C3        STA NEGREG
                ;
                NBRWDY
                ;++++ TEST STATE OF NEGREG ++++
                ;
AD 0C C3        LDA NEGREG
C9 01           CMP #$01
F0 14           BEQ NEG1
                ;
C9 02           CMP #$02
D0 06           BNE FALSE
20 8C C4        JSR SWOP
4C 13 C3        JMP START
                ;
                FALSE
C9 03           CMP #$03
D0 13           BNE NEG0
20 8C C4        JSR SWOP
4C 13 C3        JMP START
                ;
                NEG1
AD 04 C3        LDA Y1
38              SEC
ED 05 C3        SBC Y2      ;RECALCULATE DY
8D 08 C3        STA DY          ;AS Y1-Y2
EE 0D C3        INC DECFLG;SET DECFLG
                ;
                NEG0
AD 06 C3        LDA DXLO
18              CLC
69 01           ADC #$01        ; ADD 1 TO DX
8D 06 C3        STA DXLO
AD 07 C3        LDA DXHI
69 00           ADC #$00        ; CATCH CARRY
8D 07 C3        STA DXHI
EE 08 C3        INC DY          ; ADD ONE TO DY
AD 04 C3        LDA Y1
A8              TAY             ;SET Y=Y1
                ;
                ;++++ TEST FOR DX >=DY ++++
                ;
AD 07 C3        LDA DXHI
C9 01           CMP #$01
F0 73           BEQ GLESS
AD 06 C3        LDA DXLO
CD 08 C3        CMP DY
B0 6B           BCS GLESS
                ;
                ;++++ PLOT POINT ++++
                ;
                PLOT
AD 00 C3        LDA X1LO
8D 03 C1        STA XLO         ;PASS LOBYTE OF X
AD 01 C3        LDA X1HI
8D 04 C1        STA XHI         ;PASS HIBYTE OF X
98              TYA
8D 05 C1        STA YLO         ;PASS Y
20 83 C1        JSR PLTSUB
AD 0A C3        LDA TOTLO
18              CLC
6D 06 C3        ADC DXLO        ;TOTAL=TOTAL+DX
B0 08           BCS DOINX       ;SUM EXCEEDS 1 BYTE?
8D 0A C3        STA TOTLO
CD 08 C3        CMP DY
90 18           BCC NOINX       ;TOTAL REACHED DY?
                DOINX
38              SEC
ED 08 C3        SBC DY          ;TOTAL=TOTAL-DY
8D 0A C3        STA TOTLO
AD 00 C3        LDA X1LO
18              CLC
69 01           ADC #$01
8D 00 C3        STA X1LO        ; INCREMENT X1
AD 01 C3        LDA X1HI
69 00           ADC #$00
8D 01 C3        STA X1HI
                ;
                NOINX
AD 0D C3        LDA DECFLG
C9 01           CMP #$01
D0 1F           BNE NODEY       ;DECFLG=1?
88              DEY
CC 05 C3        CPY Y2
F0 03           BEQ FINAL       ; IF Y=Y2 PLOT LAST
4C A1 C3        JMP PLOT        ;PLOT LAST POINT
                FINAL
98              TYA
8D 05 C1        STA YLO
AD 00 C3        LDA X1LO
8D 03 C1        STA XLO
AD 01 C3        LDA X1HI
8D 04 C1        STA XHI
20 83 C1        JSR PLTSUB
                ;
                PAST
AD 00 C3        LDA X1LO
18              CLC
69 01           ADC #$01
8D 00 C3        STA X1LO        ; INCREMENT X
AD 01 C3        LDA X1HI
69 00           ADC #$00
8D 01 C3        STA X1HI        ; CATCH CARRY
                ;
CD 03 C3        CMP X2HI
D0 0E           BNE GLESS
AD 00 C3        LDA X1LO        ; HAS X REACHED X2?
CD 02 C3        CMP X2LO
D0 06           BNE GLESS
                ;
                ;++++ PULL REGISTERS OFF STACK ++++
                ;
                FINISH
68              PLA
A8              TAY
68              PLA
AA              TAX
68              PLA
                ;
60              RTS
                ;
                ;++++ END OF PROGRAM ++++
                ;
                ;++++ SWOP POINTS SUBROUTINE ++++
                ;
                SWOP
AD 02 C3        LDA X2LO
8D 09 C3        STA TEMP
AD 00 C3        LDA X1LO
8D 02 C3        STA X2LO
AD 09 C3        LDA TEMP        ;SWOP LOBYTES OF X
8D 00 C3        STA X1LO
                ;
AD 03 C3        LDA X2HI
8D 09 C3        STA TEMP
AD 01 C3        LDA X1HI
8D 03 C3        STA X2HI
AD 09 C3        LDA TEMP        ;SWOP HIBYTES OF X
8D 01 C3        STA X1HI
                ;
AD 05 C3        LDA Y2
8D 09 C3        STA TEMP
AD 04 C3        LDA Y1
8D 05 C3        STA Y2
AD 09 C3        LDA TEMP        ; SWOP Y
8D 04 C3        STA Y1
                ;
60              RTS
                ;
                ;++++ END OF SUBROUTINE ++++
```

Linesub Machine Code Routine

The machine code routine draws lines with gradients greater than, equal to, or less than one. There is no problem with vertical lines as, using the method we have described, there is no division by zero required. For lines with gradients less than one, the same principles apply as those outlined. All that is required is to insert a test to determine whether or not DX is greater than DY (indicating a gradient less than one), and branching accordingly.

- The routine uses two bytes for each X co-ordinate since X may exceed 255. Y however, cannot exceed 199 and can therefore be held in a single byte.
- The locations used by Plotsub to hold the co-ordinates to be plotted are assigned at the beginning of the Assembly listing. The start address of Plotsub is also assigned.
- Bits 1 and 0 of NEGREG are set according to the results of calculating DX and DY. In order to access each bit separately, without affecting the other, use is made of the logical ORA instruction.
- As the Y co-ordinate can be held in a single byte, the Y register is used to hold its value during plotting. The register values are not corrupted by Plotsub.

N.B. In order to ensure that Plotsub can be used with Linesub, the following change should be made to your source code listing (on page 339). A branch instruction must be inserted between SBC REMX and TAX.
The equivalent change can be made to the BASIC Demonstration program by inserting these lines:

5010 FOR I=HRSFLG TO HRSFLG+314:READ A
5380 DATA 169,7,237,8, 193,240,6,170,14,13,193
5410 DATA 38944: REM*CHECKSUM*