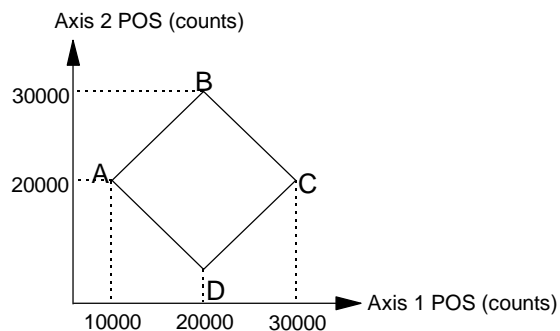


# 4 Linear & Circular Moves

## Constant Acceleration Linear Move

---

The linear motion commands are used in motions where the velocity connecting point A to point B is linear. The starting position/velocity (defining point A) are those of an axis at the commencement of this command. The ending position and velocity are the command's arguments. The following example will trace a square shape as illustrated below.



## *Linear & Circular Moves*

```
plc_program:
    run_m_program (square)
end_program

square:
    var23=1
    ctrl(0x3,0,1000,1000,1000,0,1000,1000,1000)
        ;set control gains for motor 1
    pos_preset(0x3,10000,20000) ;point A
    while(var23==1)

        linear_move(0x3,15000,5,25000,5) ;point AB/2
        linear_move(0x3,20000,0,30000,0) ;point B

        linear_move(0x3,25000,5,25000,-5) ;point BC/2
        linear_move(0x3,30000,0,20000,0) ;point C

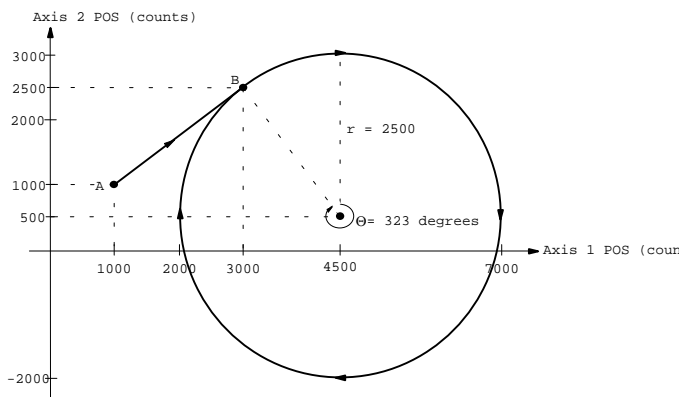
        linear_move(0x3,25000,-5,15000,-5) ;point CD/2
        linear_move(0x3,20000,0,10000,0) ;point D

        linear_move(0x3,15000,-5,15000,5) ;point DA/2
        linear_move(0x3,10000,0,20000,0) ;point A

    wend
end
```

## Combined S-Curve Linear & Circular Moves

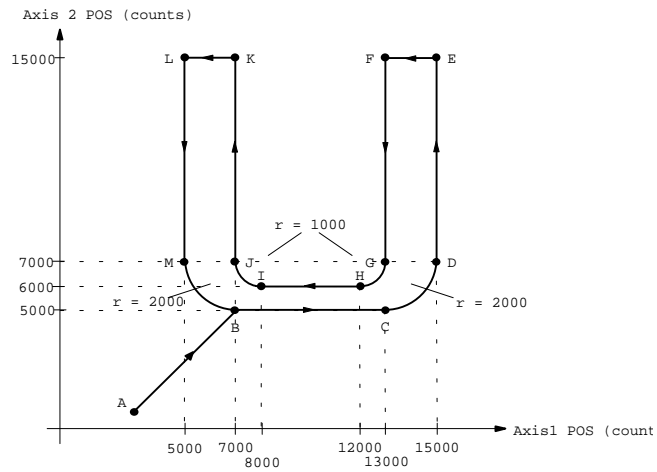
From position A (1000,1000) counts, move axes one and two to position (3000,2500) where the axes complete 360° of a circle centered at (4500,500). The linear move is completed in 1000 mS (5000 (200µs ticks)). The circle feedrate is 1.0 counts/200µs.



```
circular:
    pos_preset(0x3,1000,1000)    ;preset position counters to
                                ;point A
    linear_move_s(0x3,0.00030,3000,0.8,5000,
                 0.00023,2500,0.6,5000)
                                ;linear move from A to B
    circle(0x3,1500,-2000,2500,1.0,0,0)
                                ;circle from B to B (360
                                ;degrees clockwise)
end
```

## Combined Linear & Arc Moves

This example demonstrates how to move an x-y table according to the shape illustrated below.



U-Shape :

```

pos_preset(0x3,3000,1000)           ;preset position counters
axmove(0x3,0.004,7000,1.0,0.004,5000,1.0) ;from A to B
wait_until(CPOS1==7000)             ;wait for completion of A
                                     ;to B motion

linear_move_s(0x1,0.01,13000,1.0,6000) ;from B to C
circle(0x3,0,2000,2000,-1.0,2000,2000) ;from C to D
linear_move_s(0x2,0.01,15000,1.0,8000) ;from D to E
wait_until((CPOS2==15000) and (CVEL2==0)) ;wait for completion of
;D to E motion

axmove(0x1,0.004,13000,-1.0)         ;from E to F
wait_until((CPOS1==13000) and (CVEL1==0)) ;wait for completion of
;E to F motion

linear_move_s(0x2,0.01,7000,-0.4,20000) ;from F to G
circle(0x3,-1000,0,1000,0.4,-1000,-1000) ;from G to H
linear_move_s(0x1,0.01,8000,-0.4,10000) ;from H to I
circle(0x3,0,1000,1000,0.4,-1000,1000) ;from I to J

linear_move_s(0x2,0.01,15000,0.4,20000) ;from J to K
wait_until((CPOS2==15000) and (CVEL2==0)) ;wait for completion of
;J to K motion

axmove(0x1,0.004,5000,-1.0)         ;from K to L

```

### *Linear & Circular Moves*

```
wait_until((CPOS1==5000) and (CVEL1==0))      ;wait for completion of
linear_move_s(0x2,0.01,7000,-1.0,8000)        ;K to L motion
circle(0x3,2000,0,2000,-1.0,2000,-2000)      ;from L to M
end                                             ;from M to N
```

*Linear & Circular Moves*

This page intentionally blank.