```
; Subroutine to add two 3D coordinates.
; X points to the first coordinate.
; Y points to the second coordinate.
; D points to the place to store the sum.

; Save the registers so we can use them freely. There's no pshd instruction,
; so we do it in two parts.
add3D:  pshx
        pshy
        psha
        pshb

; Add the two x coordinates. Push the sum on the stack until we're ready to
; deal with it. The syntax 'xcoord,X' specifies the indexed addressing mode
; using the X register. The effective address will be X+xcoord = X+2.
; 'xcoord,Y' will use the Y register as the base.
        ldaa  xcoord,X
        adda  xcoord,Y
        psha

; Repeat for the y and z coordinates, pushing the y coordinate on the stack
; for temporary storage.
        ldaa  ycoord,X
        adda  ycoord,Y
        psha
        ldaa  zcoord,X
        adda  zcoord,Y

; Now pop the y and x coordinate sums into Y (y is in the most significant
; byte), and the destination address into X. Then store all three sums into
; the destination. We have to transfer the sums from Y to A,B to store them
; one at a time.
        pulY
        pulX
        staa  zcoord,X
        xgdy                    ; exchange A,B <-> Y
        staa  ycoord,X
        stab  xcoord,X

; Now restore the registers to their original values and return.
        xgdx
        puly
        pulx
        rts
```