

---

## Background Processor Instruction Summary

---

000-jk Exit  
001-jk Exit

002i-k Jump to parcel Ak and hold return parcel Ai  
003---- Jump to constant parcel

004---- Jump to constant parcel if semaphore clear: Set semaphore  
005---- Jump to constant parcel if semaphore set: Set semaphore

006---- Set semaphore  
007---- Clear semaphore

010---k Jump to constant parcel if Ak is zero  
011---k Jump to constant parcel if Ak is nonzero

012---k Jump to constant parcel if Ak is positive  
013---k Jump to constant parcel if Ak is negative

014-j- Jump to constant parcel if Sj is zero  
015-j- Jump to constant parcel if Sj is nonzero

016-j- Jump to constant parcel if Sj is positive  
017-j- Jump to constant parcel if Sj is negative

020ijk Enter Ai with integer sum  $A_j + A_k$   
021ijk Enter Ai with integer difference  $A_j - A_k$

022ijk Enter Ai with integer product  $A_j * A_k$   
023ijk (same as above)

024ij- Enter Ai from Sj  
025i-- Enter Ai from L

026ijk Enter Ai with positive jk  
027ijk Enter Ai with negative jk

030---k Enter M bits where Vk has zero elements  
031---k Enter M bits where Vk has nonzero elements

032---k Enter M bits where Vk has positive elements  
033---k Enter M bits where Vk has negative elements

034-j- Enter M from Sj  
035---k Alter status register enable flags

036---k Enter L from Ak  
037---k (same as above)