```
0A9C  9F  GHI   RF ;D=RF.1
  9D  F3  XOR      ;Compare RA.0:RF.1
  9E  12  INC   R2 ;Reset stack pointer
  9F  3A  BNZ      ;If ≠, branch to 0A8E

0AA0  8E
  A1  F8  LDI      ;Else hand is a straight
  A2  05
  A3  BE  PHI   RE ;RE.1=05 to mark straight

                          (FLUSHES)

0AA4  F8  LDI
  A5  07
  A6  AE  PLO   RE ;RE.0=07  (initialize RE.0)
  A7  9F  GHI   RF ;D=RF.1
  A8  FF  SMI      ;Subtract 04
  A9  04
  AA  AA  PLO   RA ;RA.0=D   (RA points to first card)
  AB  0A  LDN   RA ;D=M(R(A))  (get card)
  AC  52  STR   R2 ;Push
  AD  1A  INC   RA ;RA=RA+1  (next card)
  AE  0A  LDN   RA ;D=M(R(A))  (get card)
  AF  F3  XOR      ;Compare with byte on stack

0AB0  FA  ANI      ;"AND" result with F0 to strip card type
  B1  F0
  B2  3A  BNZ      ;If ≠, branch to 0ABF  (no flush)
  B3  BF
  B4  22  DEC   R2 ;Decrement stack pointer preserves value
  B5  8A  GLO   RA ;D=RA.0
  B6  52  STR   R2 ;Push for comparing
  B7  9F  GHI   RF ;D=RF.1
  B8  F3  XOR      ;Compare RA.0:RF.1
  B9  12  INC   R2 ;Reset stack pointer
  BA  3A  BNZ      ;If RA.0≠RF.1, branch to 0AAD to continue
  BB  AD
  BC  F8  LDI      ;Else hand is a flush (at least)
  BD  06
  BE  AE  PLO   RE ;RE.0=06  (to mark flush)
  BF  8E  GLO   RE ;D=RE.0  (either 06 or 00)

0AC0  52  STR   R2 ;Push for adding
  C1  9E  GHI   RE ;D=RE.1  (either 05 or 00 -- straight or no straight)
  C2  F4  ADD      ;D=RE.1 + RE.0
  C3  BE  PHI   RE ;RE.1=D  (RE.1=05=straight/06=flush/0B=straight
                             flush/00=bust hand)
```