Problem 6: [16 points]
Filename: hwAprob6.asm

```
       .ORG $2000
Y      .DW  $0
       .DW  $0
       .DW  $0
       .DW  $0
       .DW  $0
       .DW  $0
       .DW  $0
       .DW  $0

       .ORG $1000
init   MV R7, R0 ; 0 out error counter
       MV R6, R0 ; 0 out index counter
       LI R2, $1 ; set enable
       SW R0, R2, $0012 ; set just enable bit

wait   LI R2, $3 ; load enable and read bit
       SW R0, R2, $0012 ; set enable + read bit
       SLTI R0, R6, $10 ; check if index has hit 8 yet
       BRN skip
       MV R6, R0 ; 0 out index counter
skip   SLTI R0, R7, $3 ; check for 3 errors in parity bit in a row
       BRZ bad
       LW R2, R0, $0014 ; check if data is ready
       BRN check ; branch if data is ready
       BRA wait ; wait again if not ready
check  LI R2, $D ; load ack = 11, enable = 1
       SW R0, R2, $0012 ; set enable = 1, ack = 11
       MV R3, R0 ; 0 out the parity checker
       MV R5, R0 ; 0 out the loop counter
       LW R2, R0, $0020 ; load datapayload into R2
loop   LI R1, $1 ; load 1 into R1 to AND with 2
       AND R1, R1, R2 ; get last bit of R2
       XOR R3, R3, R1 ; XOR r3 with r1, running xor
       ADDI R5, R5, $1 ; add 1 to our counter
       SRLI R2, R2, $1 ; shift right by 1 to get next bit
       SLTI R0, R5, $8 ; if R5 = 8, then set Z; else, set N
       BRN loop
       SW R0, R0, $0014 ; 0 out the data ready bit
       SLTI R0, R3, $1 ; check if parity is 1 (which is odd)
       BRN good ; parity 0 then good, parity 1 then bad
       LI R2, $5 ; store ack = 01, enable in $0012
       SW R0, R2, $0012 ;
       ADDI R7, R7, $1 ; increment our error counter
       BRA wait
good   LW R2, R0, $0020 ; get good copy of payload back
       SW R6, R2, Y ; store in array at offset
       ADDI R6, R6, $2 ; add 2 to r6 offset
       MV R7, R0 ; re-zero r7
       LI R2, $1 ; set ack = 00, enable in $0012
       SW R0, R2, $0012 ;
       BRA wait
bad    LI R2, $FF ; got 3 errors, so we load FF into R2
       SW R6, R2, Y ; put FF in the array
       ADDI R6, R6, $2 ; add 2 to r6 offset
       LI R2, $9 ; set ack = 10, enable
       SW R0, R2, $0012 ;
       MV R7, R0 ; reset the error counter
       BRA wait
```