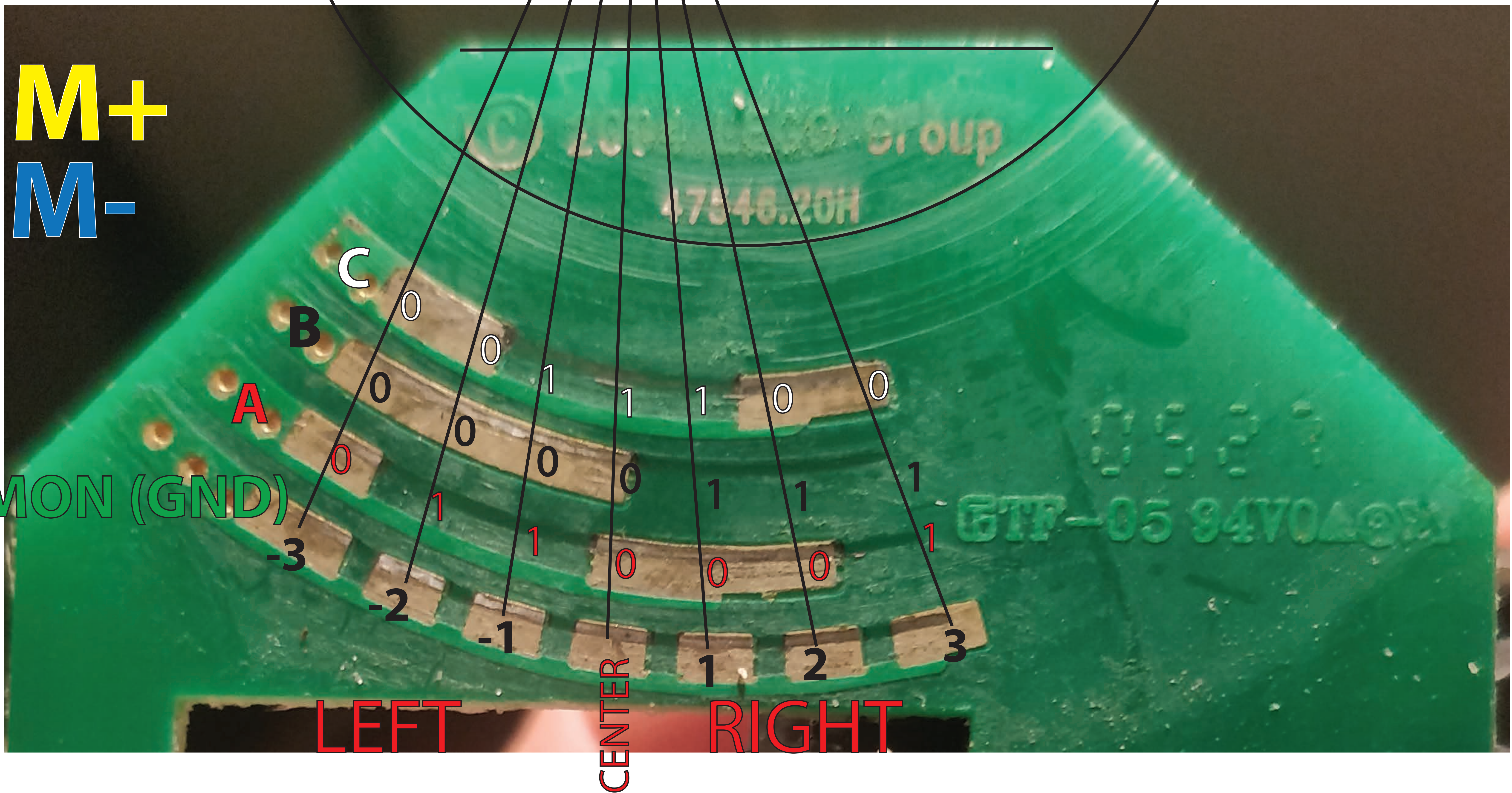


# Steering feedback circuit:



```
//on RESET:
pinMode(A,INPUT_PULLUP);
pinMode(B,INPUT_PULLUP);
pinMode(C,INPUT_PULLUP);
int previous_state=0;

int READ(){

int aState = digitalRead(A);
int bState = digitalRead(B);
int cState = digitalRead(C);

if( aState & bState & cState ) return previous_state; //non-discrete in-between-state with all pins high

//Now we know the feedback is in a discreet state:

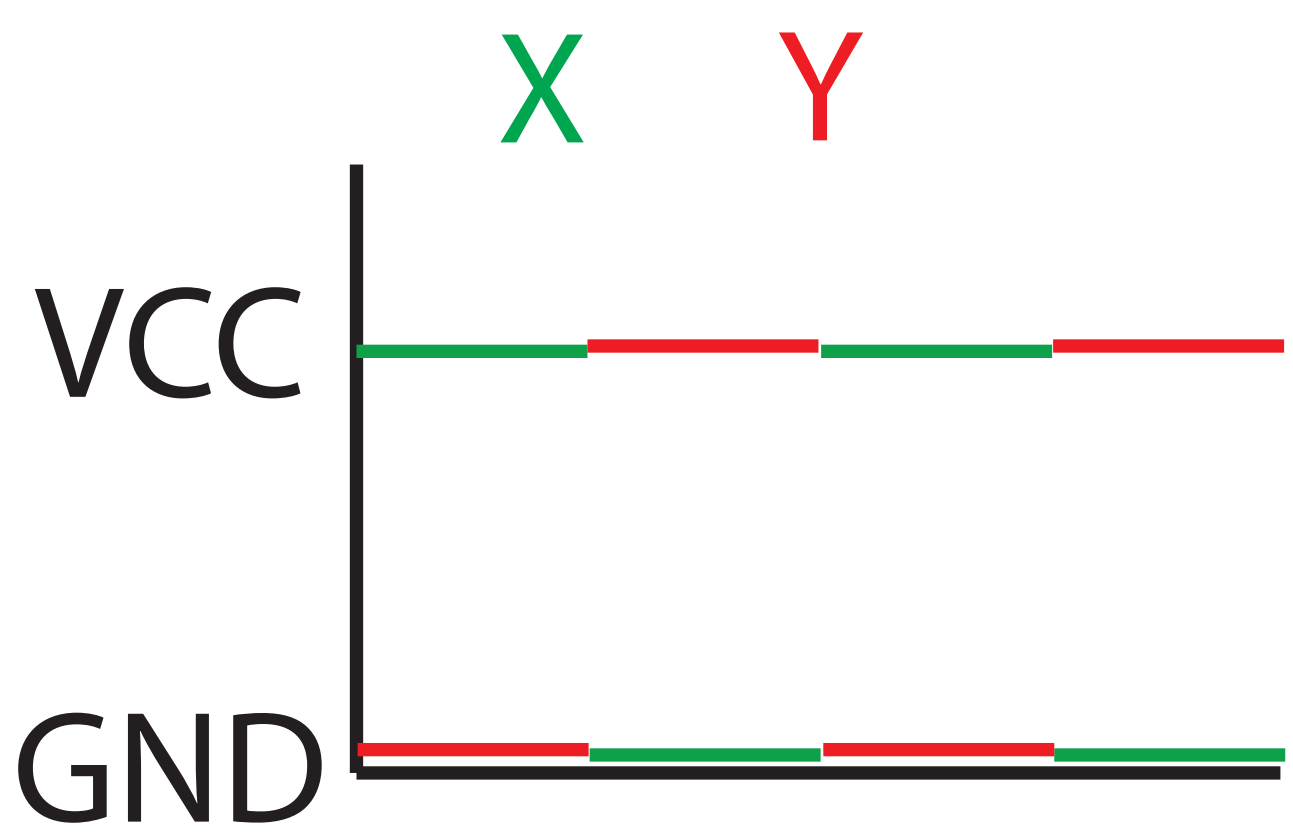
if( bState ) return 2-cState+aState //Steering feedback is positive (1,2,3)

else if ( cState & !aState ) return 0; // A=0, B=0; C=1

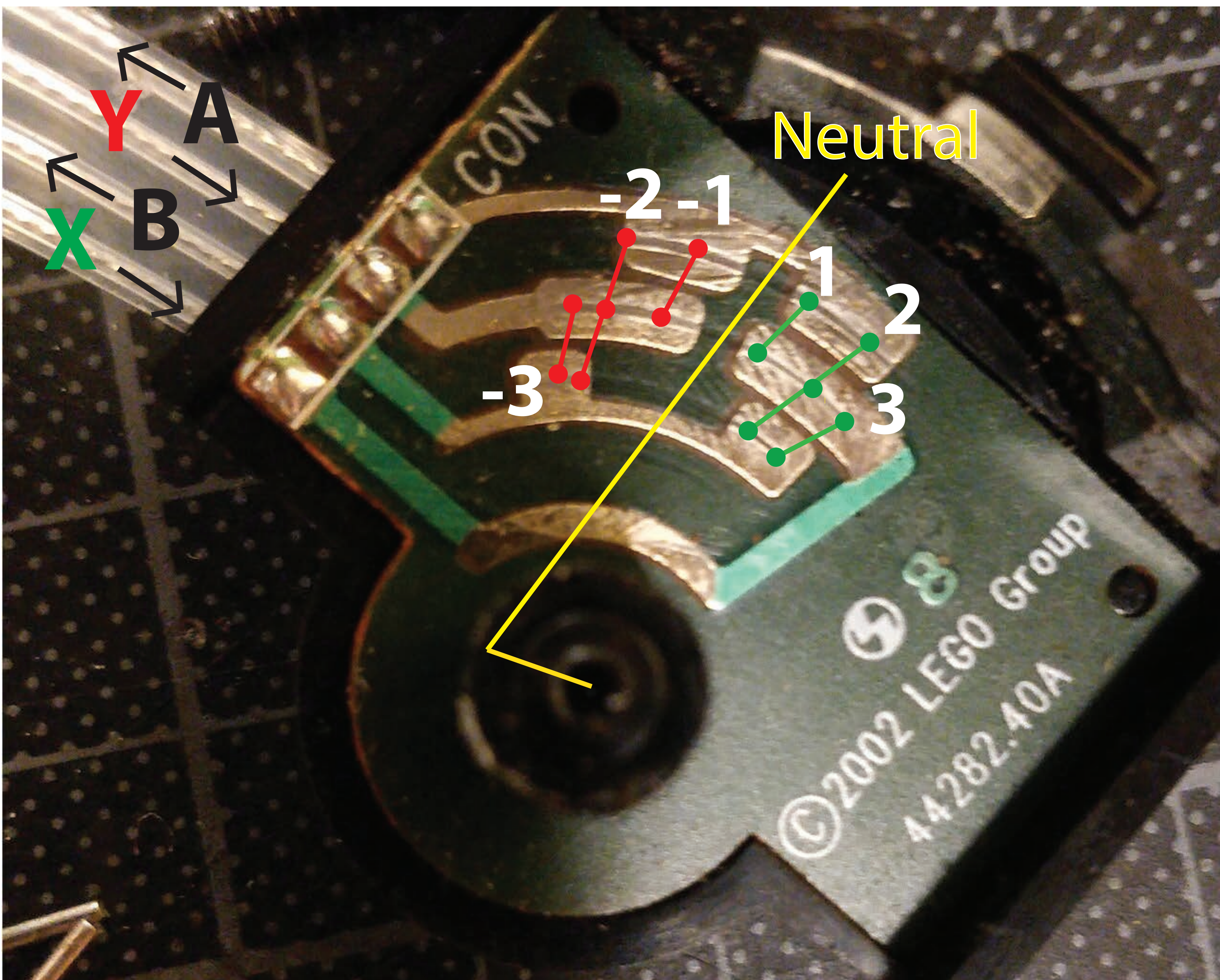
else return -3+aState+cState; //-3,-2,-1
}
```

Both steering and inputs have 7 states.  
( Steering technically has another (non-discrete) state where A,B,C = 1 )

# Input circuit:



Read A and B  
while alternating X and Y



```
//on RESET:
pinMode(A,INPUT_PULLUP);
pinMode(B,INPUT_PULLUP);
pinMode(Y,INPUT);
pinMode(X,INPUT);

int READ(){
state=0;

pinMode(X,OUTPUT);
digitalWrite(X,LOW); //perhaps redundant if state never changes?
aState = digitalRead(A);
bState = digitalRead(B);
pinMode(X,INPUT);

if( !aState ) state = 1+(1-bState); //A=low (state 1 or 2)
else if ( !bState ) state = 3; //A=high, B=low

if( state ) return state; //drop the rest if state has changed

pinMode(Y,OUTPUT);
digitalWrite(Y,LOW); //perhaps redundant if state never changes?
aState = digitalRead(A);
bState = digitalRead(B);
pinMode(Y,INPUT);

if( !aState ) state = -1*(1+(1-bState)); //A=low (state -1 or -2)
else if ( !bState ) state = -3; //A=high, B=low

return state;
}
```

TX 1 byte:  
/\*  
0bxxxxxxx  
||||| \\_ Throttle STICK BIT #0  
||||| \\_ Throttle STICK BIT #1  
|||| \\_ Throttle STICK BIT #2  
||| \\_ Steering STICK BIT #0  
|| \\_ Steering STICK BIT #1  
|| \\_ Steering STICK BIT #2  
| \\_ SPEED BIT 1  
| \\_ SPEED BIT 2  
\*/

tree steering bits  
tree speed bits  
  
2 speedmodifier bits (1-3)