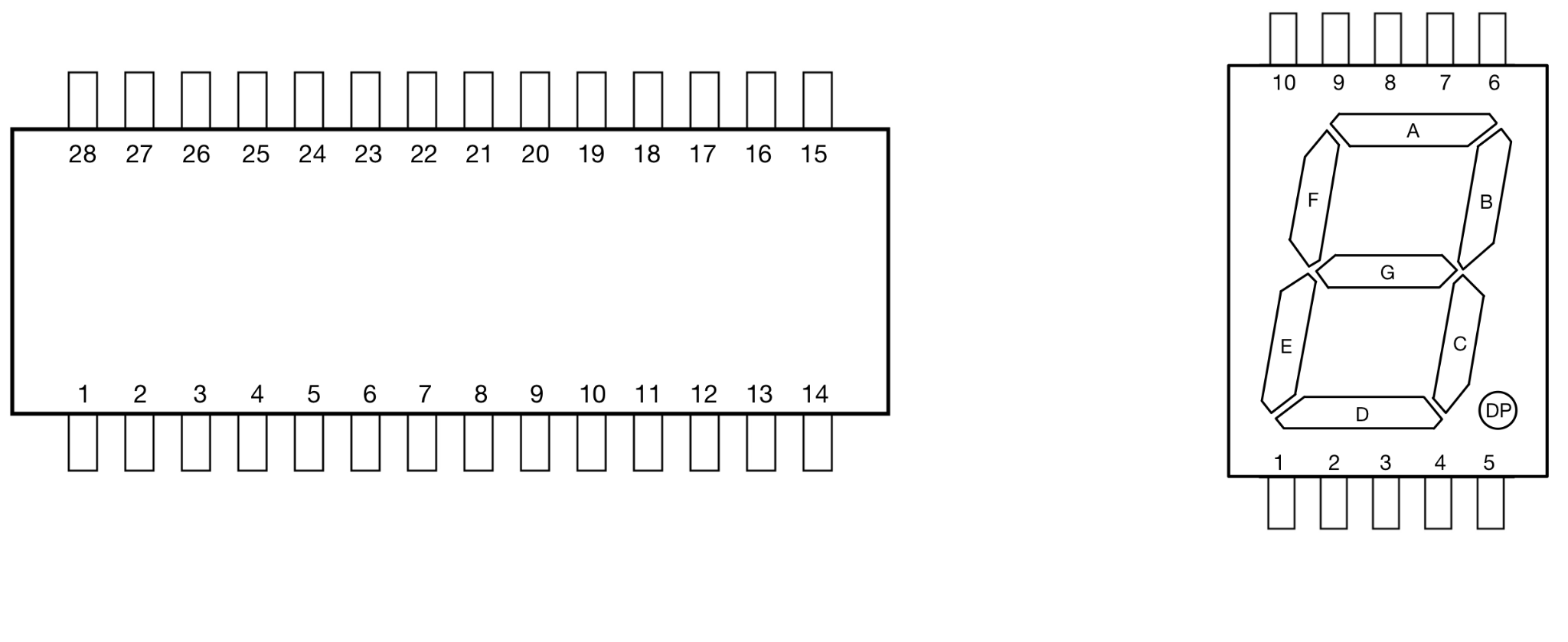
Name: Student ID:

Lab Section: Date:

# **Prelab**

1. Based on the AT28C64B datasheet and the 7-segment display datasheet, both linked below, make the necessary connections from the EEPROM to the display. This will use the pins I/O0 through I/O7 as the outputs from the EEPROM which will be in the order

[., g, f, e, d, c, b, a], respectively. Note: you do not need to make any connections other than mapping the I/O pins to the display.



1. In the following table, indicate the binary and hex values in the order [., g, f, e, d, c, b, a] to achieve the following displays on a common-cathode display.

| Display |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Binary |  |  |  |  |  |  |  |  |
| Hex |  |  |  |  |  |  |  |  |

| Display |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Binary |  |  |  |  |  |  |  |  |
| Hex |  |  |  |  |  |  |  |  |

# 

# 

# **Lab**

**4.1 Write the Script**

Show the TA your script to produce the binary file.

TA Initials:

**4.3 Load the Binary File**

Show the TA the table in Xgpro with the loaded encodings.

TA Initials:

**6.0 Testing**

After confirming this behavior, demonstrate to the TA that the binary inputs 0111, 1000, 1011, 1111, and 0000 produce the expected segment outputs on the display in counting mode. Then, switch to game mode and toggle each bit to ensure that every segment responds correctly.

TA Initials:

**7.0 Common-Anode vs Common-Cathode Displays**

Explain to the TA how you might modify your python script to produce the proper binary file for a common-anode display.

TA Initials: