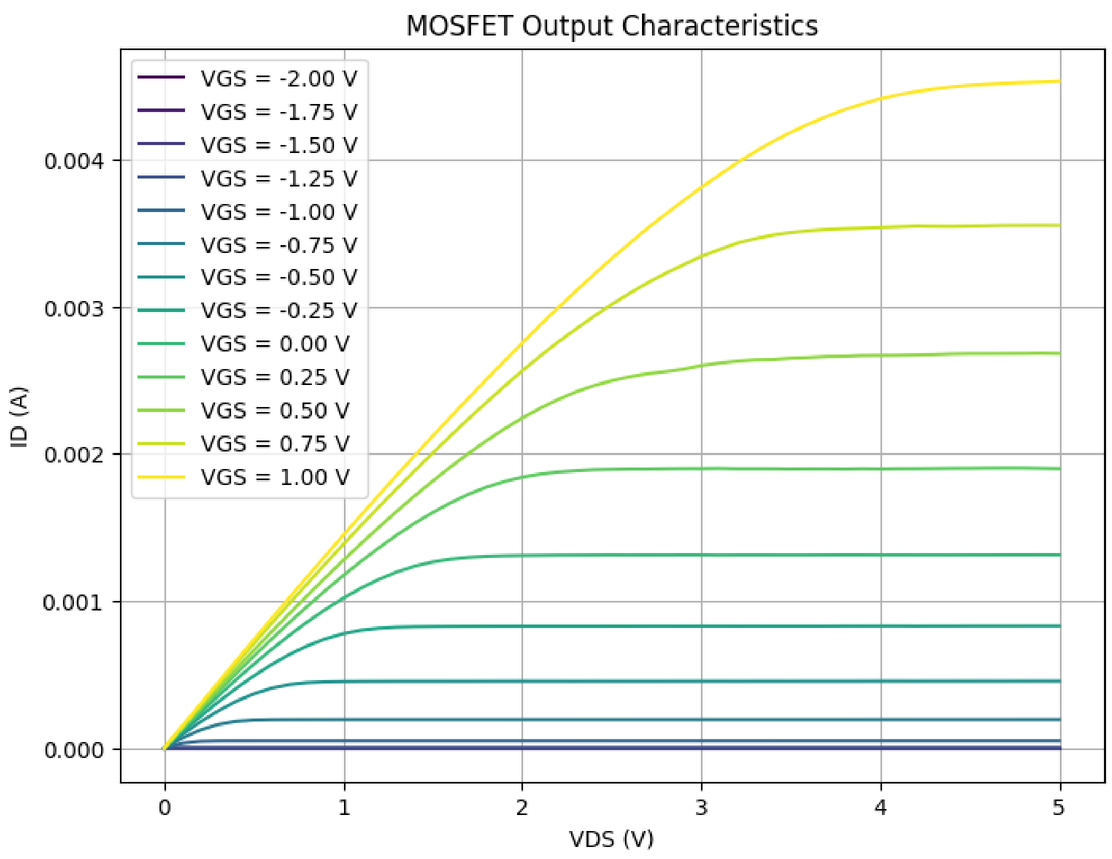
LND150K1-G_script

April 1, 2025

[4]:
```python
import numpy as np
import matplotlib.pyplot as plt

# Load the dataset
file_path = "Ouput 0-5V Gate -2-1V Step 0.25.txt"
data = np.loadtxt(file_path, usecols=(0, 1))  # Ignore the third column

# Extract VDS and ID
VDS = data[:, 0]  # First column is VDS
ID = data[:, 1]  # Second column is ID

# Define VGS values based on the dataset's step size
VGS_values = np.arange(-2, 1.25, 0.25)  # From -2V to 1V in steps of 0.25V
num_VGS = len(VGS_values)

# Compute number of data points per VGS sweep
num_points_per_sweep = len(VDS) // num_VGS  # Ensure correct reshaping

# Reshape data for plotting
VDS_reshaped = VDS[:num_points_per_sweep]  # Take first VDS sweep as reference
ID_reshaped = ID.reshape(num_VGS, num_points_per_sweep)

# Plot the output characteristics
plt.figure(figsize=(8, 6))
colors = plt.cm.viridis(np.linspace(0, 1, num_VGS))  # Assign different colors␣
 ↪for different VGS

for i, VGS in enumerate(VGS_values):
    plt.plot(VDS_reshaped, ID_reshaped[i, :], color=colors[i], label=f"VGS =␣
 ↪{VGS:.2f} V")

plt.xlabel("VDS (V)")
plt.ylabel("ID (A)")
plt.title("MOSFET Output Characteristics")
plt.legend()
plt.grid(True)
plt.show()
```



[5]:
```python
import numpy as np
import tensorflow as tf
from tensorflow import keras
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import MinMaxScaler
import matplotlib.pyplot as plt

# Load the dataset
file_path = "Ouput 0-5V Gate -2-1V Step 0.25.txt"
data = np.loadtxt(file_path, usecols=(0, 1))  # Ignore third column

# Extract VDS and ID
VDS = data[:, 0]  # First column is VDS
ID = data[:, 1]  # Second column is ID

# Define VGS values based on dataset's step size
VGS_values = np.arange(-2, 1.25, 0.25)  # VGS from -2V to 1V in steps of 0.25V
num_VGS = len(VGS_values)

# Compute number of data points per VGS sweep
```

```
num_points_per_sweep = len(VDS) // num_VGS

# Reshape VDS and ID correctly
VDS_reshaped = VDS[:num_points_per_sweep]  # Take first VDS sweep for reference
ID_reshaped = ID.reshape(num_VGS, num_points_per_sweep)

# Generate (VGS, VDS) pairs as input features
X = np.array([[VGS, vds] for VGS in VGS_values for vds in VDS_reshaped])
y = ID  # Target (ID values)

# Normalize input features and output target
scaler_X = MinMaxScaler()
scaler_y = MinMaxScaler()

scaler_X.fit(np.array([[min(VGS_values), 0], [max(VGS_values), max(VDS)]]))  #␣
 ↪Normalize within dataset range
scaler_y.fit(y.reshape(-1, 1))  # Normalize output (ID)

X_scaled = scaler_X.transform(X)
y_scaled = scaler_y.transform(y.reshape(-1, 1))

# Split data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y_scaled,␣
 ↪test_size=0.2, random_state=42)

# Define a new neural network model
model = keras.Sequential([
    keras.layers.Dense(128, activation='relu', input_shape=(2,)),  # Two input␣
 ↪features: VGS, VDS
    keras.layers.Dense(128, activation='relu'),
    keras.layers.Dense(1)  # Single output: ID
])

# Compile model
model.compile(optimizer='adam', loss='mse', metrics=['mae'])

# Train the model
history = model.fit(X_train, y_train, epochs=200, batch_size=16,␣
 ↪validation_data=(X_test, y_test), verbose=1)

# Save the trained model
model.save("/mnt/data/mosfet_model_5V.h5")
print("Model saved as mosfet_model_5V.h5")

# Evaluate model performance
loss, mae = model.evaluate(X_test, y_test)
print(f"Test MAE: {mae:.6f}")
```

```
# Plot training & validation loss
plt.figure(figsize=(8, 6))
plt.plot(history.history['loss'], label="Train Loss")
plt.plot(history.history['val_loss'], label="Validation Loss")
plt.xlabel("Epochs")
plt.ylabel("Loss (MSE)")
plt.title("Training & Validation Loss Over Epochs")
plt.legend()
plt.grid(True)
plt.show()
```

```
Epoch 1/200
```

```
c:\Users\asus1\anaconda3\envs\MOSFET_NN_mini\Lib\site-
packages\keras\src\layers\core\dense.py:87: UserWarning: Do not pass an
`input_shape`/`input_dim` argument to a layer. When using Sequential models,
prefer using an `Input(shape)` object as the first layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

```
34/34                 2s 7ms/step - loss:
0.0675 - mae: 0.1871 - val_loss: 0.0182 - val_mae: 0.1074
Epoch 2/200
34/34                 0s 2ms/step - loss:
0.0159 - mae: 0.0975 - val_loss: 0.0079 - val_mae: 0.0689
Epoch 3/200
34/34                 0s 3ms/step - loss:
0.0067 - mae: 0.0616 - val_loss: 0.0028 - val_mae: 0.0346
Epoch 4/200
34/34                 0s 2ms/step - loss:
0.0025 - mae: 0.0338 - val_loss: 0.0013 - val_mae: 0.0243
Epoch 5/200
34/34                 0s 2ms/step - loss:
0.0014 - mae: 0.0271 - val_loss: 8.7224e-04 - val_mae: 0.0214
Epoch 6/200
34/34                 0s 2ms/step - loss:
8.4914e-04 - mae: 0.0204 - val_loss: 5.6413e-04 - val_mae: 0.0155
Epoch 7/200
34/34                 0s 2ms/step - loss:
4.8784e-04 - mae: 0.0153 - val_loss: 5.1668e-04 - val_mae: 0.0140
Epoch 8/200
34/34                 0s 2ms/step - loss:
6.1521e-04 - mae: 0.0174 - val_loss: 3.1277e-04 - val_mae: 0.0097
Epoch 9/200
34/34                 0s 2ms/step - loss:
2.3013e-04 - mae: 0.0099 - val_loss: 3.5482e-04 - val_mae: 0.0144
Epoch 10/200
34/34                 0s 2ms/step - loss:
2.4316e-04 - mae: 0.0110 - val_loss: 1.7398e-04 - val_mae: 0.0081
```

```
Epoch 11/200
34/34            0s 2ms/step - loss:
1.8860e-04 - mae: 0.0093 - val_loss: 1.4777e-04 - val_mae: 0.0093
Epoch 12/200
34/34            0s 3ms/step - loss:
1.6041e-04 - mae: 0.0093 - val_loss: 1.4465e-04 - val_mae: 0.0091
Epoch 13/200
34/34            0s 2ms/step - loss:
1.1435e-04 - mae: 0.0076 - val_loss: 5.2855e-04 - val_mae: 0.0197
Epoch 14/200
34/34            0s 2ms/step - loss:
7.4276e-04 - mae: 0.0223 - val_loss: 1.0304e-04 - val_mae: 0.0074
Epoch 15/200
34/34            0s 2ms/step - loss:
1.2987e-04 - mae: 0.0083 - val_loss: 9.9573e-05 - val_mae: 0.0075
Epoch 16/200
34/34            0s 3ms/step - loss:
1.3939e-04 - mae: 0.0089 - val_loss: 6.4648e-05 - val_mae: 0.0060
Epoch 17/200
34/34            0s 2ms/step - loss:
1.3389e-04 - mae: 0.0087 - val_loss: 8.3315e-05 - val_mae: 0.0071
Epoch 18/200
34/34            0s 2ms/step - loss:
6.6298e-05 - mae: 0.0062 - val_loss: 4.2105e-05 - val_mae: 0.0042
Epoch 19/200
34/34            0s 2ms/step - loss:
4.1341e-05 - mae: 0.0045 - val_loss: 9.7972e-05 - val_mae: 0.0086
Epoch 20/200
34/34            0s 2ms/step - loss:
4.1237e-05 - mae: 0.0049 - val_loss: 3.3517e-05 - val_mae: 0.0040
Epoch 21/200
34/34            0s 2ms/step - loss:
3.0982e-05 - mae: 0.0038 - val_loss: 2.9163e-04 - val_mae: 0.0142
Epoch 22/200
34/34            0s 2ms/step - loss:
2.3079e-04 - mae: 0.0117 - val_loss: 7.5533e-05 - val_mae: 0.0079
Epoch 23/200
34/34            0s 2ms/step - loss:
5.6797e-05 - mae: 0.0059 - val_loss: 7.4798e-05 - val_mae: 0.0068
Epoch 24/200
34/34            0s 2ms/step - loss:
4.6409e-05 - mae: 0.0052 - val_loss: 2.2252e-05 - val_mae: 0.0032
Epoch 25/200
34/34            0s 2ms/step - loss:
2.4360e-05 - mae: 0.0034 - val_loss: 2.9507e-05 - val_mae: 0.0035
Epoch 26/200
34/34            0s 2ms/step - loss:
2.1444e-05 - mae: 0.0033 - val_loss: 6.3419e-05 - val_mae: 0.0068
```

```
Epoch 27/200
34/34            0s 2ms/step - loss:
5.4233e-05 - mae: 0.0059 - val_loss: 2.1995e-05 - val_mae: 0.0033
Epoch 28/200
34/34            0s 3ms/step - loss:
2.3328e-05 - mae: 0.0034 - val_loss: 3.2437e-05 - val_mae: 0.0038
Epoch 29/200
34/34            0s 3ms/step - loss:
2.7177e-05 - mae: 0.0037 - val_loss: 2.1626e-05 - val_mae: 0.0033
Epoch 30/200
34/34            0s 2ms/step - loss:
1.9984e-05 - mae: 0.0034 - val_loss: 5.0133e-05 - val_mae: 0.0061
Epoch 31/200
34/34            0s 2ms/step - loss:
2.8329e-05 - mae: 0.0041 - val_loss: 2.1928e-05 - val_mae: 0.0038
Epoch 32/200
34/34            0s 2ms/step - loss:
1.5369e-05 - mae: 0.0029 - val_loss: 2.5879e-04 - val_mae: 0.0132
Epoch 33/200
34/34            0s 3ms/step - loss:
2.6670e-04 - mae: 0.0126 - val_loss: 8.1031e-05 - val_mae: 0.0060
Epoch 34/200
34/34            0s 2ms/step - loss:
4.7157e-05 - mae: 0.0052 - val_loss: 2.8533e-05 - val_mae: 0.0036
Epoch 35/200
34/34            0s 2ms/step - loss:
1.7085e-05 - mae: 0.0030 - val_loss: 1.6616e-05 - val_mae: 0.0027
Epoch 36/200
34/34            0s 2ms/step - loss:
2.5752e-05 - mae: 0.0037 - val_loss: 1.5230e-05 - val_mae: 0.0026
Epoch 37/200
34/34            0s 3ms/step - loss:
1.6407e-05 - mae: 0.0030 - val_loss: 1.1335e-05 - val_mae: 0.0024
Epoch 38/200
34/34            0s 2ms/step - loss:
1.9616e-05 - mae: 0.0033 - val_loss: 1.1520e-05 - val_mae: 0.0026
Epoch 39/200
34/34            0s 2ms/step - loss:
1.2029e-05 - mae: 0.0026 - val_loss: 2.3163e-05 - val_mae: 0.0035
Epoch 40/200
34/34            0s 3ms/step - loss:
1.2167e-05 - mae: 0.0026 - val_loss: 2.8298e-05 - val_mae: 0.0041
Epoch 41/200
34/34            0s 2ms/step - loss:
3.7047e-05 - mae: 0.0046 - val_loss: 2.1640e-05 - val_mae: 0.0039
Epoch 42/200
34/34            0s 2ms/step - loss:
1.7221e-05 - mae: 0.0033 - val_loss: 7.8835e-05 - val_mae: 0.0072
```

```
Epoch 43/200
34/34            0s 2ms/step - loss:
6.3422e-05 - mae: 0.0066 - val_loss: 4.5042e-05 - val_mae: 0.0060
Epoch 44/200
34/34            0s 3ms/step - loss:
2.6120e-05 - mae: 0.0041 - val_loss: 3.5205e-05 - val_mae: 0.0051
Epoch 45/200
34/34            0s 3ms/step - loss:
1.9133e-05 - mae: 0.0035 - val_loss: 4.9123e-05 - val_mae: 0.0062
Epoch 46/200
34/34            0s 3ms/step - loss:
3.0575e-05 - mae: 0.0042 - val_loss: 2.1355e-04 - val_mae: 0.0138
Epoch 47/200
34/34            0s 2ms/step - loss:
1.3735e-04 - mae: 0.0098 - val_loss: 5.3161e-05 - val_mae: 0.0057
Epoch 48/200
34/34            0s 2ms/step - loss:
2.6672e-05 - mae: 0.0038 - val_loss: 1.3020e-05 - val_mae: 0.0028
Epoch 49/200
34/34            0s 3ms/step - loss:
9.7483e-06 - mae: 0.0024 - val_loss: 1.8404e-05 - val_mae: 0.0034
Epoch 50/200
34/34            0s 3ms/step - loss:
1.7502e-05 - mae: 0.0032 - val_loss: 1.4613e-05 - val_mae: 0.0028
Epoch 51/200
34/34            0s 2ms/step - loss:
1.6390e-05 - mae: 0.0032 - val_loss: 1.0551e-05 - val_mae: 0.0027
Epoch 52/200
34/34            0s 2ms/step - loss:
1.9631e-05 - mae: 0.0034 - val_loss: 3.1076e-05 - val_mae: 0.0045
Epoch 53/200
34/34            0s 2ms/step - loss:
1.9235e-05 - mae: 0.0034 - val_loss: 1.6469e-05 - val_mae: 0.0030
Epoch 54/200
34/34            0s 2ms/step - loss:
1.2838e-05 - mae: 0.0027 - val_loss: 3.0633e-05 - val_mae: 0.0041
Epoch 55/200
34/34            0s 2ms/step - loss:
3.5654e-05 - mae: 0.0048 - val_loss: 1.2948e-04 - val_mae: 0.0104
Epoch 56/200
34/34            0s 2ms/step - loss:
4.5801e-05 - mae: 0.0053 - val_loss: 2.1427e-05 - val_mae: 0.0037
Epoch 57/200
34/34            0s 2ms/step - loss:
1.6915e-05 - mae: 0.0032 - val_loss: 1.0815e-05 - val_mae: 0.0027
Epoch 58/200
34/34            0s 2ms/step - loss:
2.3657e-05 - mae: 0.0039 - val_loss: 2.9191e-05 - val_mae: 0.0045
```

```
Epoch 59/200
34/34            0s 2ms/step - loss:
2.5357e-05 - mae: 0.0040 - val_loss: 1.1153e-05 - val_mae: 0.0026
Epoch 60/200
34/34            0s 2ms/step - loss:
1.4178e-05 - mae: 0.0029 - val_loss: 2.2757e-05 - val_mae: 0.0039
Epoch 61/200
34/34            0s 2ms/step - loss:
1.6943e-05 - mae: 0.0033 - val_loss: 1.5846e-05 - val_mae: 0.0031
Epoch 62/200
34/34            0s 2ms/step - loss:
1.0323e-05 - mae: 0.0025 - val_loss: 1.6956e-05 - val_mae: 0.0035
Epoch 63/200
34/34            0s 2ms/step - loss:
1.8527e-05 - mae: 0.0033 - val_loss: 2.0708e-05 - val_mae: 0.0039
Epoch 64/200
34/34            0s 3ms/step - loss:
1.1057e-05 - mae: 0.0025 - val_loss: 1.2305e-05 - val_mae: 0.0029
Epoch 65/200
34/34            0s 2ms/step - loss:
1.0603e-05 - mae: 0.0026 - val_loss: 1.9559e-05 - val_mae: 0.0039
Epoch 66/200
34/34            0s 2ms/step - loss:
1.9027e-05 - mae: 0.0035 - val_loss: 6.7532e-06 - val_mae: 0.0020
Epoch 67/200
34/34            0s 2ms/step - loss:
9.3102e-06 - mae: 0.0022 - val_loss: 4.3828e-05 - val_mae: 0.0053
Epoch 68/200
34/34            0s 2ms/step - loss:
2.0748e-05 - mae: 0.0034 - val_loss: 1.6834e-05 - val_mae: 0.0028
Epoch 69/200
34/34            0s 2ms/step - loss:
2.0538e-05 - mae: 0.0035 - val_loss: 3.5867e-05 - val_mae: 0.0040
Epoch 70/200
34/34            0s 2ms/step - loss:
8.0349e-05 - mae: 0.0071 - val_loss: 1.5014e-05 - val_mae: 0.0027
Epoch 71/200
34/34            0s 3ms/step - loss:
3.5946e-05 - mae: 0.0042 - val_loss: 1.4373e-05 - val_mae: 0.0028
Epoch 72/200
34/34            0s 2ms/step - loss:
1.3758e-05 - mae: 0.0029 - val_loss: 3.1327e-05 - val_mae: 0.0041
Epoch 73/200
34/34            0s 2ms/step - loss:
1.2976e-05 - mae: 0.0027 - val_loss: 2.4549e-05 - val_mae: 0.0036
Epoch 74/200
34/34            0s 3ms/step - loss:
2.0913e-05 - mae: 0.0033 - val_loss: 7.1184e-05 - val_mae: 0.0059
```

```
Epoch 75/200
34/34            0s 2ms/step - loss:
3.3644e-05 - mae: 0.0043 - val_loss: 9.3219e-05 - val_mae: 0.0078
Epoch 76/200
34/34            0s 2ms/step - loss:
5.0895e-05 - mae: 0.0056 - val_loss: 4.4738e-05 - val_mae: 0.0047
Epoch 77/200
34/34            0s 2ms/step - loss:
1.8305e-05 - mae: 0.0033 - val_loss: 9.3224e-06 - val_mae: 0.0025
Epoch 78/200
34/34            0s 2ms/step - loss:
7.8447e-06 - mae: 0.0022 - val_loss: 1.2946e-05 - val_mae: 0.0024
Epoch 79/200
34/34            0s 2ms/step - loss:
1.2800e-05 - mae: 0.0027 - val_loss: 9.6677e-06 - val_mae: 0.0020
Epoch 80/200
34/34            0s 3ms/step - loss:
1.7552e-05 - mae: 0.0032 - val_loss: 1.0405e-05 - val_mae: 0.0023
Epoch 81/200
34/34            0s 3ms/step - loss:
9.6325e-06 - mae: 0.0024 - val_loss: 1.1483e-05 - val_mae: 0.0025
Epoch 82/200
34/34            0s 2ms/step - loss:
1.2067e-05 - mae: 0.0026 - val_loss: 3.9257e-05 - val_mae: 0.0041
Epoch 83/200
34/34            0s 2ms/step - loss:
5.9185e-05 - mae: 0.0056 - val_loss: 1.0839e-04 - val_mae: 0.0091
Epoch 84/200
34/34            0s 3ms/step - loss:
2.1128e-04 - mae: 0.0115 - val_loss: 2.5927e-04 - val_mae: 0.0110
Epoch 85/200
34/34            0s 2ms/step - loss:
2.0550e-04 - mae: 0.0111 - val_loss: 2.4941e-05 - val_mae: 0.0037
Epoch 86/200
34/34            0s 3ms/step - loss:
3.4149e-05 - mae: 0.0045 - val_loss: 7.1571e-06 - val_mae: 0.0020
Epoch 87/200
34/34            0s 2ms/step - loss:
8.2050e-06 - mae: 0.0022 - val_loss: 3.0452e-05 - val_mae: 0.0049
Epoch 88/200
34/34            0s 2ms/step - loss:
3.1251e-05 - mae: 0.0044 - val_loss: 1.9353e-04 - val_mae: 0.0127
Epoch 89/200
34/34            0s 2ms/step - loss:
1.4391e-04 - mae: 0.0093 - val_loss: 1.7434e-05 - val_mae: 0.0032
Epoch 90/200
34/34            0s 2ms/step - loss:
4.2790e-05 - mae: 0.0051 - val_loss: 4.6814e-05 - val_mae: 0.0052
```

```
Epoch 91/200
34/34            0s 2ms/step - loss:
1.6386e-04 - mae: 0.0099 - val_loss: 7.5669e-04 - val_mae: 0.0218
Epoch 92/200
34/34            0s 2ms/step - loss:
2.9236e-04 - mae: 0.0136 - val_loss: 3.9762e-04 - val_mae: 0.0153
Epoch 93/200
34/34            0s 2ms/step - loss:
2.7525e-04 - mae: 0.0130 - val_loss: 3.9260e-04 - val_mae: 0.0159
Epoch 94/200
34/34            0s 2ms/step - loss:
3.4990e-04 - mae: 0.0147 - val_loss: 1.3329e-04 - val_mae: 0.0095
Epoch 95/200
34/34            0s 3ms/step - loss:
1.0832e-04 - mae: 0.0075 - val_loss: 1.6620e-04 - val_mae: 0.0105
Epoch 96/200
34/34            0s 2ms/step - loss:
7.7199e-05 - mae: 0.0062 - val_loss: 3.4992e-05 - val_mae: 0.0047
Epoch 97/200
34/34            0s 2ms/step - loss:
3.1221e-05 - mae: 0.0041 - val_loss: 4.2390e-05 - val_mae: 0.0053
Epoch 98/200
34/34            0s 2ms/step - loss:
3.0574e-05 - mae: 0.0043 - val_loss: 1.6231e-04 - val_mae: 0.0099
Epoch 99/200
34/34            0s 2ms/step - loss:
1.4587e-04 - mae: 0.0098 - val_loss: 5.4221e-04 - val_mae: 0.0203
Epoch 100/200
34/34            0s 2ms/step - loss:
2.8378e-04 - mae: 0.0134 - val_loss: 5.4558e-04 - val_mae: 0.0211
Epoch 101/200
34/34            0s 3ms/step - loss:
3.7970e-04 - mae: 0.0160 - val_loss: 9.8233e-05 - val_mae: 0.0069
Epoch 102/200
34/34            0s 2ms/step - loss:
6.7915e-05 - mae: 0.0060 - val_loss: 3.3755e-05 - val_mae: 0.0048
Epoch 103/200
34/34            0s 2ms/step - loss:
7.8757e-05 - mae: 0.0069 - val_loss: 5.1139e-05 - val_mae: 0.0055
Epoch 104/200
34/34            0s 2ms/step - loss:
3.7974e-05 - mae: 0.0046 - val_loss: 1.7814e-05 - val_mae: 0.0034
Epoch 105/200
34/34            0s 2ms/step - loss:
3.1842e-05 - mae: 0.0044 - val_loss: 5.8536e-05 - val_mae: 0.0066
Epoch 106/200
34/34            0s 2ms/step - loss:
3.3030e-05 - mae: 0.0045 - val_loss: 7.2909e-06 - val_mae: 0.0020
```

```
Epoch 107/200
34/34            0s 2ms/step - loss:
1.7343e-05 - mae: 0.0033 - val_loss: 7.7865e-06 - val_mae: 0.0022
Epoch 108/200
34/34            0s 3ms/step - loss:
1.4089e-05 - mae: 0.0029 - val_loss: 1.2285e-05 - val_mae: 0.0023
Epoch 109/200
34/34            0s 2ms/step - loss:
8.3720e-06 - mae: 0.0021 - val_loss: 5.7558e-06 - val_mae: 0.0016
Epoch 110/200
34/34            0s 2ms/step - loss:
5.5298e-06 - mae: 0.0016 - val_loss: 5.3649e-06 - val_mae: 0.0018
Epoch 111/200
34/34            0s 2ms/step - loss:
5.2556e-06 - mae: 0.0017 - val_loss: 1.0076e-05 - val_mae: 0.0022
Epoch 112/200
34/34            0s 2ms/step - loss:
7.1505e-06 - mae: 0.0020 - val_loss: 7.5982e-06 - val_mae: 0.0018
Epoch 113/200
34/34            0s 2ms/step - loss:
7.6654e-06 - mae: 0.0021 - val_loss: 1.9625e-05 - val_mae: 0.0035
Epoch 114/200
34/34            0s 2ms/step - loss:
1.3245e-05 - mae: 0.0027 - val_loss: 1.5618e-05 - val_mae: 0.0033
Epoch 115/200
34/34            0s 2ms/step - loss:
2.0331e-05 - mae: 0.0035 - val_loss: 8.8476e-05 - val_mae: 0.0084
Epoch 116/200
34/34            0s 2ms/step - loss:
3.2573e-05 - mae: 0.0045 - val_loss: 3.3760e-04 - val_mae: 0.0154
Epoch 117/200
34/34            0s 2ms/step - loss:
3.7134e-04 - mae: 0.0147 - val_loss: 3.2177e-05 - val_mae: 0.0041
Epoch 118/200
34/34            0s 2ms/step - loss:
7.5801e-05 - mae: 0.0059 - val_loss: 1.7169e-05 - val_mae: 0.0031
Epoch 119/200
34/34            0s 2ms/step - loss:
1.8960e-05 - mae: 0.0032 - val_loss: 9.1936e-06 - val_mae: 0.0019
Epoch 120/200
34/34            0s 3ms/step - loss:
1.0450e-05 - mae: 0.0023 - val_loss: 7.1723e-05 - val_mae: 0.0064
Epoch 121/200
34/34            0s 2ms/step - loss:
3.7472e-05 - mae: 0.0049 - val_loss: 7.2490e-05 - val_mae: 0.0066
Epoch 122/200
34/34            0s 2ms/step - loss:
3.7107e-05 - mae: 0.0049 - val_loss: 1.0875e-04 - val_mae: 0.0068
```

```
Epoch 123/200
34/34            0s 2ms/step - loss:
7.4288e-05 - mae: 0.0066 - val_loss: 1.7828e-05 - val_mae: 0.0031
Epoch 124/200
34/34            0s 2ms/step - loss:
1.8932e-05 - mae: 0.0031 - val_loss: 3.8251e-05 - val_mae: 0.0057
Epoch 125/200
34/34            0s 2ms/step - loss:
2.1453e-05 - mae: 0.0038 - val_loss: 2.9330e-05 - val_mae: 0.0044
Epoch 126/200
34/34            0s 2ms/step - loss:
2.6099e-05 - mae: 0.0038 - val_loss: 1.7292e-05 - val_mae: 0.0035
Epoch 127/200
34/34            0s 2ms/step - loss:
1.2710e-05 - mae: 0.0026 - val_loss: 1.6851e-05 - val_mae: 0.0024
Epoch 128/200
34/34            0s 2ms/step - loss:
1.7305e-05 - mae: 0.0030 - val_loss: 5.8276e-05 - val_mae: 0.0052
Epoch 129/200
34/34            0s 2ms/step - loss:
3.2693e-05 - mae: 0.0044 - val_loss: 4.8120e-05 - val_mae: 0.0055
Epoch 130/200
34/34            0s 2ms/step - loss:
3.9551e-05 - mae: 0.0051 - val_loss: 4.9650e-05 - val_mae: 0.0045
Epoch 131/200
34/34            0s 2ms/step - loss:
2.6201e-05 - mae: 0.0037 - val_loss: 2.8774e-05 - val_mae: 0.0044
Epoch 132/200
34/34            0s 2ms/step - loss:
7.5965e-05 - mae: 0.0068 - val_loss: 1.2546e-04 - val_mae: 0.0088
Epoch 133/200
34/34            0s 2ms/step - loss:
1.9290e-04 - mae: 0.0095 - val_loss: 4.5182e-05 - val_mae: 0.0054
Epoch 134/200
34/34            0s 2ms/step - loss:
3.8283e-05 - mae: 0.0046 - val_loss: 1.2758e-05 - val_mae: 0.0025
Epoch 135/200
34/34            0s 2ms/step - loss:
1.8715e-05 - mae: 0.0030 - val_loss: 1.0345e-05 - val_mae: 0.0023
Epoch 136/200
34/34            0s 2ms/step - loss:
9.1234e-06 - mae: 0.0021 - val_loss: 6.6462e-06 - val_mae: 0.0018
Epoch 137/200
34/34            0s 2ms/step - loss:
7.5726e-06 - mae: 0.0021 - val_loss: 9.3295e-06 - val_mae: 0.0019
Epoch 138/200
34/34            0s 2ms/step - loss:
1.6426e-05 - mae: 0.0031 - val_loss: 4.2706e-05 - val_mae: 0.0050
```

```
Epoch 139/200
34/34            0s 2ms/step - loss:
6.3337e-05 - mae: 0.0058 - val_loss: 8.7866e-05 - val_mae: 0.0086
Epoch 140/200
34/34            0s 2ms/step - loss:
5.0855e-05 - mae: 0.0057 - val_loss: 5.6784e-05 - val_mae: 0.0054
Epoch 141/200
34/34            0s 2ms/step - loss:
3.6360e-05 - mae: 0.0048 - val_loss: 1.8615e-05 - val_mae: 0.0033
Epoch 142/200
34/34            0s 3ms/step - loss:
1.5856e-05 - mae: 0.0030 - val_loss: 1.0167e-05 - val_mae: 0.0024
Epoch 143/200
34/34            0s 2ms/step - loss:
1.6920e-05 - mae: 0.0032 - val_loss: 6.6082e-06 - val_mae: 0.0019
Epoch 144/200
34/34            0s 2ms/step - loss:
8.3273e-06 - mae: 0.0021 - val_loss: 1.1037e-05 - val_mae: 0.0025
Epoch 145/200
34/34            0s 2ms/step - loss:
7.0757e-06 - mae: 0.0021 - val_loss: 1.8728e-05 - val_mae: 0.0036
Epoch 146/200
34/34            0s 2ms/step - loss:
1.2126e-05 - mae: 0.0027 - val_loss: 3.3895e-05 - val_mae: 0.0049
Epoch 147/200
34/34            0s 2ms/step - loss:
1.8666e-05 - mae: 0.0033 - val_loss: 2.9782e-05 - val_mae: 0.0039
Epoch 148/200
34/34            0s 2ms/step - loss:
1.3865e-05 - mae: 0.0028 - val_loss: 2.2089e-05 - val_mae: 0.0031
Epoch 149/200
34/34            0s 2ms/step - loss:
2.8313e-05 - mae: 0.0042 - val_loss: 9.6757e-06 - val_mae: 0.0020
Epoch 150/200
34/34            0s 2ms/step - loss:
9.5626e-06 - mae: 0.0022 - val_loss: 4.5598e-06 - val_mae: 0.0014
Epoch 151/200
34/34            0s 2ms/step - loss:
8.1403e-06 - mae: 0.0021 - val_loss: 6.7549e-06 - val_mae: 0.0017
Epoch 152/200
34/34            0s 2ms/step - loss:
1.4013e-05 - mae: 0.0026 - val_loss: 9.1349e-06 - val_mae: 0.0027
Epoch 153/200
34/34            0s 2ms/step - loss:
1.0641e-05 - mae: 0.0024 - val_loss: 8.8438e-06 - val_mae: 0.0022
Epoch 154/200
34/34            0s 3ms/step - loss:
1.5184e-05 - mae: 0.0031 - val_loss: 4.5169e-05 - val_mae: 0.0050
```

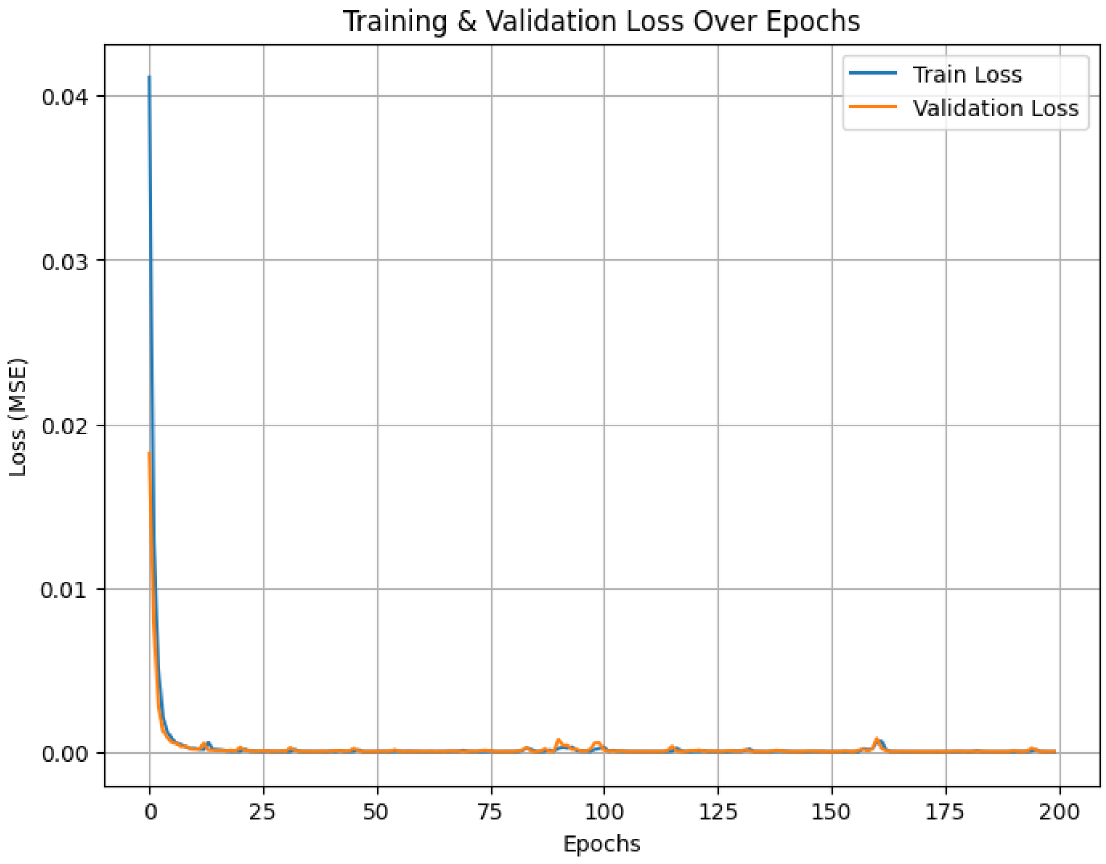
```
Epoch 155/200
34/34            0s 2ms/step - loss:
1.4427e-05 - mae: 0.0029 - val_loss: 1.1882e-05 - val_mae: 0.0028
Epoch 156/200
34/34            0s 2ms/step - loss:
1.1258e-05 - mae: 0.0026 - val_loss: 5.0826e-06 - val_mae: 0.0018
Epoch 157/200
34/34            0s 3ms/step - loss:
6.2980e-06 - mae: 0.0019 - val_loss: 9.9660e-05 - val_mae: 0.0087
Epoch 158/200
34/34            0s 2ms/step - loss:
1.4460e-04 - mae: 0.0093 - val_loss: 1.5050e-04 - val_mae: 0.0095
Epoch 159/200
34/34            0s 2ms/step - loss:
1.2861e-04 - mae: 0.0087 - val_loss: 5.1336e-05 - val_mae: 0.0049
Epoch 160/200
34/34            0s 2ms/step - loss:
1.4398e-04 - mae: 0.0091 - val_loss: 1.8159e-04 - val_mae: 0.0074
Epoch 161/200
34/34            0s 2ms/step - loss:
2.9397e-04 - mae: 0.0125 - val_loss: 8.1622e-04 - val_mae: 0.0240
Epoch 162/200
34/34            0s 3ms/step - loss:
9.1449e-04 - mae: 0.0242 - val_loss: 2.1945e-04 - val_mae: 0.0112
Epoch 163/200
34/34            0s 2ms/step - loss:
1.6400e-04 - mae: 0.0097 - val_loss: 6.2457e-05 - val_mae: 0.0070
Epoch 164/200
34/34            0s 2ms/step - loss:
3.3304e-05 - mae: 0.0048 - val_loss: 1.8816e-05 - val_mae: 0.0036
Epoch 165/200
34/34            0s 2ms/step - loss:
1.3746e-05 - mae: 0.0029 - val_loss: 6.7339e-06 - val_mae: 0.0020
Epoch 166/200
34/34            0s 2ms/step - loss:
1.0250e-05 - mae: 0.0025 - val_loss: 1.3422e-05 - val_mae: 0.0029
Epoch 167/200
34/34            0s 2ms/step - loss:
1.2532e-05 - mae: 0.0027 - val_loss: 7.9469e-06 - val_mae: 0.0021
Epoch 168/200
34/34            0s 2ms/step - loss:
7.6830e-06 - mae: 0.0021 - val_loss: 1.2862e-05 - val_mae: 0.0028
Epoch 169/200
34/34            0s 2ms/step - loss:
7.0908e-06 - mae: 0.0021 - val_loss: 1.1046e-05 - val_mae: 0.0024
Epoch 170/200
34/34            0s 2ms/step - loss:
7.6645e-06 - mae: 0.0020 - val_loss: 1.2521e-05 - val_mae: 0.0026
```

```
Epoch 171/200
34/34            0s 2ms/step - loss:
9.4356e-06 - mae: 0.0024 - val_loss: 4.7584e-06 - val_mae: 0.0017
Epoch 172/200
34/34            0s 2ms/step - loss:
6.3124e-06 - mae: 0.0018 - val_loss: 6.7615e-06 - val_mae: 0.0020
Epoch 173/200
34/34            0s 2ms/step - loss:
4.0819e-06 - mae: 0.0016 - val_loss: 4.1743e-06 - val_mae: 0.0015
Epoch 174/200
34/34            0s 3ms/step - loss:
7.4199e-06 - mae: 0.0021 - val_loss: 1.4708e-05 - val_mae: 0.0027
Epoch 175/200
34/34            0s 2ms/step - loss:
8.3091e-06 - mae: 0.0021 - val_loss: 5.7734e-06 - val_mae: 0.0018
Epoch 176/200
34/34            0s 3ms/step - loss:
5.5995e-06 - mae: 0.0018 - val_loss: 1.2352e-05 - val_mae: 0.0027
Epoch 177/200
34/34            0s 3ms/step - loss:
1.1644e-05 - mae: 0.0026 - val_loss: 7.9286e-06 - val_mae: 0.0020
Epoch 178/200
34/34            0s 2ms/step - loss:
1.2174e-05 - mae: 0.0025 - val_loss: 2.8458e-05 - val_mae: 0.0035
Epoch 179/200
34/34            0s 2ms/step - loss:
3.0480e-05 - mae: 0.0042 - val_loss: 2.8790e-05 - val_mae: 0.0039
Epoch 180/200
34/34            0s 3ms/step - loss:
2.2623e-05 - mae: 0.0035 - val_loss: 6.1035e-06 - val_mae: 0.0017
Epoch 181/200
34/34            0s 2ms/step - loss:
6.4466e-06 - mae: 0.0019 - val_loss: 2.0155e-05 - val_mae: 0.0037
Epoch 182/200
34/34            0s 2ms/step - loss:
1.2410e-05 - mae: 0.0026 - val_loss: 1.3199e-05 - val_mae: 0.0026
Epoch 183/200
34/34            0s 2ms/step - loss:
4.6730e-05 - mae: 0.0049 - val_loss: 1.3903e-05 - val_mae: 0.0031
Epoch 184/200
34/34            0s 2ms/step - loss:
1.8469e-05 - mae: 0.0033 - val_loss: 1.7837e-05 - val_mae: 0.0030
Epoch 185/200
34/34            0s 2ms/step - loss:
9.0362e-06 - mae: 0.0022 - val_loss: 9.6056e-06 - val_mae: 0.0026
Epoch 186/200
34/34            0s 2ms/step - loss:
8.1912e-06 - mae: 0.0022 - val_loss: 4.8006e-06 - val_mae: 0.0017
```

```
Epoch 187/200
34/34            0s 3ms/step - loss:
5.6395e-06 - mae: 0.0018 - val_loss: 4.6925e-06 - val_mae: 0.0017
Epoch 188/200
34/34            0s 2ms/step - loss:
3.4609e-06 - mae: 0.0014 - val_loss: 3.8879e-06 - val_mae: 0.0016
Epoch 189/200
34/34            0s 2ms/step - loss:
5.1220e-06 - mae: 0.0018 - val_loss: 1.3007e-05 - val_mae: 0.0030
Epoch 190/200
34/34            0s 2ms/step - loss:
6.3257e-06 - mae: 0.0020 - val_loss: 5.8776e-06 - val_mae: 0.0019
Epoch 191/200
34/34            0s 2ms/step - loss:
7.9224e-06 - mae: 0.0021 - val_loss: 6.5853e-05 - val_mae: 0.0058
Epoch 192/200
34/34            0s 2ms/step - loss:
3.3280e-05 - mae: 0.0043 - val_loss: 1.5679e-05 - val_mae: 0.0036
Epoch 193/200
34/34            0s 2ms/step - loss:
1.4381e-05 - mae: 0.0031 - val_loss: 9.1554e-06 - val_mae: 0.0020
Epoch 194/200
34/34            0s 2ms/step - loss:
1.4985e-05 - mae: 0.0029 - val_loss: 2.8515e-05 - val_mae: 0.0045
Epoch 195/200
34/34            0s 3ms/step - loss:
5.4815e-05 - mae: 0.0053 - val_loss: 2.2855e-04 - val_mae: 0.0119
Epoch 196/200
34/34            0s 2ms/step - loss:
1.4055e-04 - mae: 0.0093 - val_loss: 1.1897e-04 - val_mae: 0.0080
Epoch 197/200
34/34            0s 2ms/step - loss:
4.0161e-05 - mae: 0.0042 - val_loss: 3.2698e-06 - val_mae: 0.0013
Epoch 198/200
34/34            0s 3ms/step - loss:
8.7082e-06 - mae: 0.0022 - val_loss: 1.1763e-05 - val_mae: 0.0030
Epoch 199/200
34/34            0s 2ms/step - loss:
8.2573e-06 - mae: 0.0023 - val_loss: 8.7490e-06 - val_mae: 0.0025
Epoch 200/200
34/34            0s 2ms/step - loss:
6.7663e-06 - mae: 0.0021 - val_loss: 2.0894e-05 - val_mae: 0.0041
```

```
WARNING:absl:You are saving your model as an HDF5 file via `model.save()` or
`keras.saving.save_model(model)`. This file format is considered legacy. We
recommend using instead the native Keras format, e.g.
`model.save('my_model.keras')` or `keras.saving.save_model(model,
'my_model.keras')`.
```

```
Model saved as mosfet_model_5V.h5
5/5             0s 4ms/step - loss:
2.1147e-05 - mae: 0.0041
Test MAE: 0.004104
```



```
[7]: model.save("LND150K1-G.keras")
```

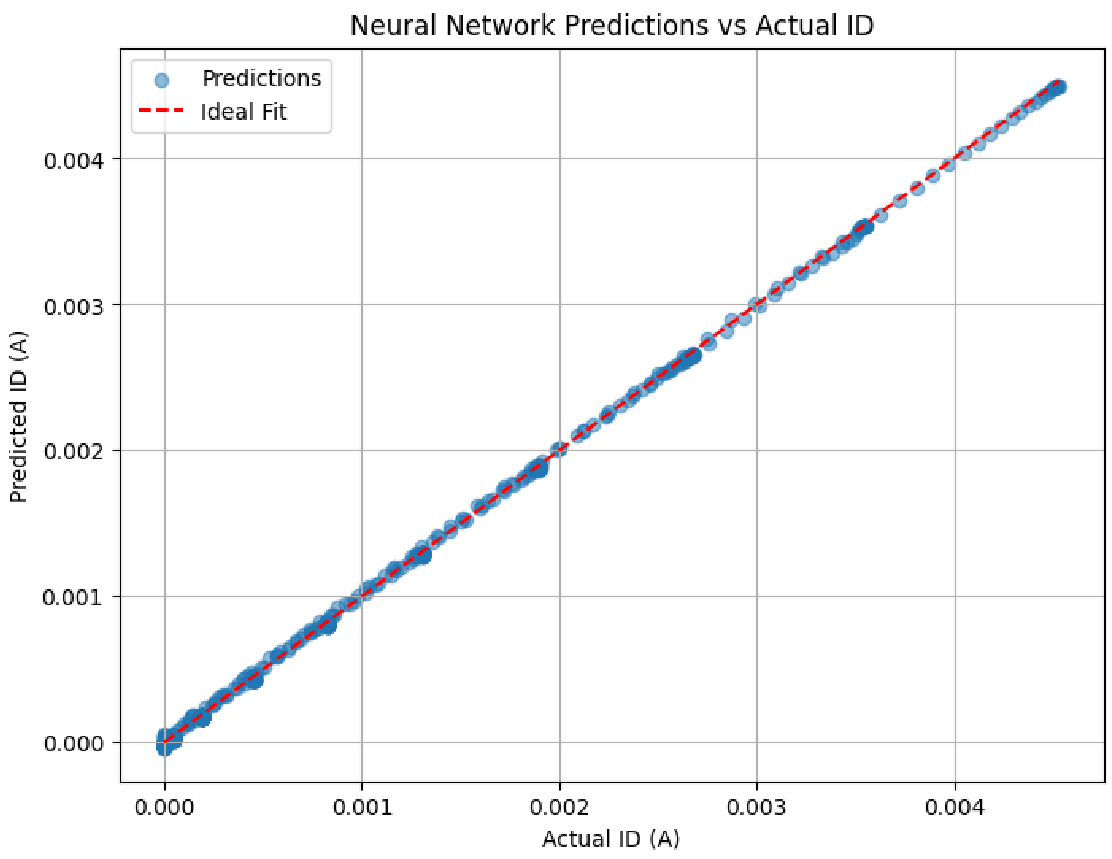
```
[ ]: import numpy as np
     import matplotlib.pyplot as plt
     from tensorflow.keras.models import load_model
     from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
     from sklearn.preprocessing import MinMaxScaler

     # Load the trained model
     model_path = "LND150K1-G.keras"
     try:
         model = load_model(model_path)
         print(f"Model '{model_path}' loaded successfully!")
     except:
         print(f"Error: Model '{model_path}' not found. Train the model first.")
```

```
    exit()

# Load the dataset
file_path = "Ouput 0-5V Gate -2-1V Step 0.25.txt"
data = np.loadtxt(file_path, usecols=(0, 1))  # Ignore third column

# Extract VDS and ID
VDS = data[:, 0]  # First column is VDS
ID = data[:, 1]  # Second column is ID

# Define VGS values based on dataset's step size
VGS_values = np.arange(-2, 1.25, 0.25)  # VGS from -2V to 1V in steps of 0.25V
num_VGS = len(VGS_values)

# Compute number of data points per VGS sweep
num_points_per_sweep = len(VDS) // num_VGS
VDS_reshaped = VDS[:num_points_per_sweep]  # First VDS sweep reference

# Generate input data for evaluation
X_test = np.array([[VGS, vds] for VGS in VGS_values for vds in VDS_reshaped])

# Normalize the test data using MinMaxScaler
scaler_X = MinMaxScaler()
scaler_y = MinMaxScaler()
scaler_X.fit(np.array([[min(VGS_values), 0], [max(VGS_values), max(VDS)]]))  #␣
 ↪Fit using dataset range
scaler_y.fit(ID.reshape(-1, 1))  # Fit output scaler using ID range

X_test_scaled = scaler_X.transform(X_test)

# Predict ID
ID_pred_scaled = model.predict(X_test_scaled)
ID_pred = scaler_y.inverse_transform(ID_pred_scaled)  # Convert back to real␣
 ↪scale

# Extract actual values from dataset
ID_actual = ID.reshape(num_VGS, num_points_per_sweep)
ID_predicted = ID_pred.reshape(num_VGS, num_points_per_sweep)

# Compute performance metrics
mae = mean_absolute_error(ID_actual.flatten(), ID_predicted.flatten())
mse = mean_squared_error(ID_actual.flatten(), ID_predicted.flatten())
rmse = np.sqrt(mse)
r2 = r2_score(ID_actual.flatten(), ID_predicted.flatten())

# Print performance metrics
print(f"Test MAE: {mae:.6e}")
print(f"Test MSE: {mse:.6e}")
print(f"Test RMSE: {rmse:.6e}")
print(f"Test R² Score: {r2:.6f}")

# Plot actual vs predicted values
plt.figure(figsize=(8, 6))
plt.scatter(ID_actual.flatten(), ID_predicted.flatten(), alpha=0.5,␣
 ↪label="Predictions")
plt.plot([min(ID_actual.flatten()), max(ID_actual.flatten())],
         [min(ID_actual.flatten()), max(ID_actual.flatten())], 'r',␣
 ↪linestyle='dashed', label="Ideal Fit")
plt.xlabel("Actual ID (A)")
plt.ylabel("Predicted ID (A)")
plt.title("LND150K1-G: Neural Network Predictions vs Actual ID")
plt.legend()
plt.grid(True)
plt.show()
```

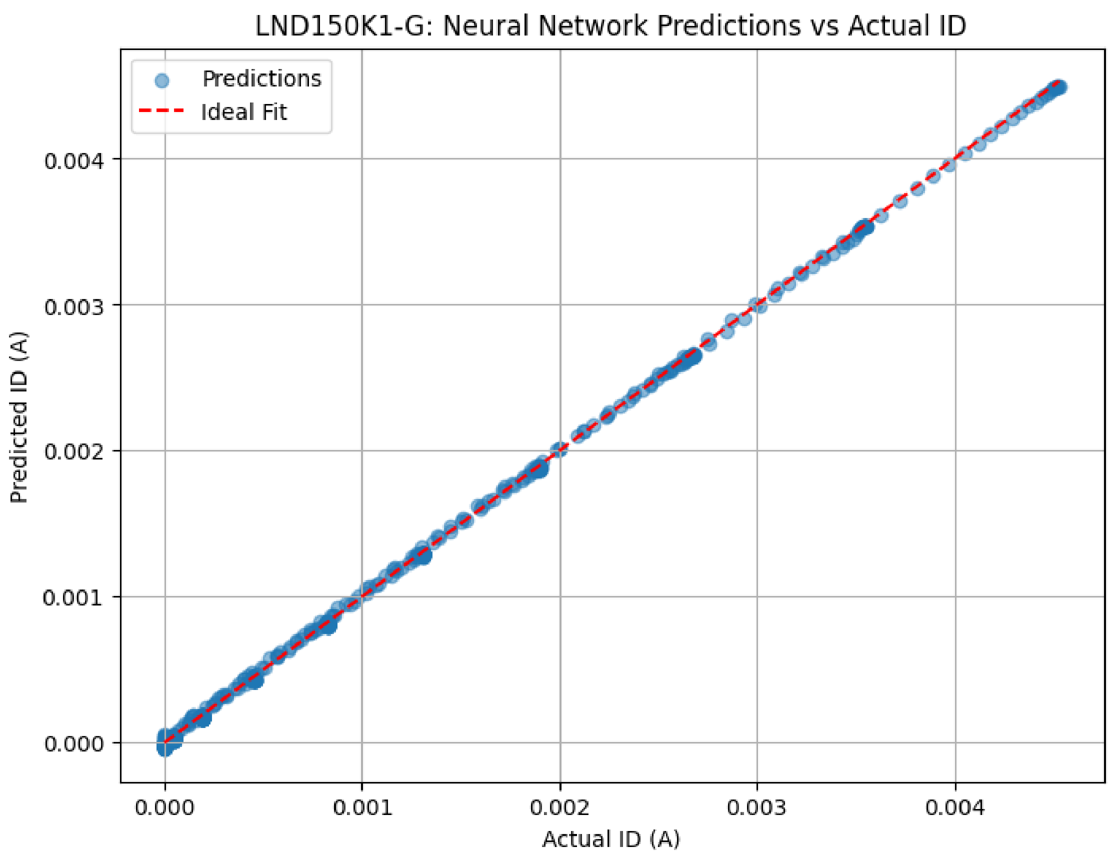
```
Model 'LND150K1-G.keras' loaded successfully!
21/21             0s 3ms/step
```

```
c:\Users\asus1\anaconda3\envs\MOSFET_NN_mini\Lib\site-
packages\keras\src\saving\saving_lib.py:719: UserWarning: Skipping variable
loading for optimizer 'rmsprop', because it has 8 variables whereas the saved
optimizer has 14 variables.
  saveable.load_own_variables(weights_store.get(inner_path))
```

```
Test MAE: 1.783722e-05
Test MSE: 4.051142e-10
Test RMSE: 2.012745e-05
Test R² Score: 0.999704
```



```
[13]: plt.figure(figsize=(8, 6))
      plt.scatter(ID_actual.flatten(), ID_predicted.flatten(), alpha=0.5,␣
       ↪label="Predictions")
      plt.plot([min(ID_actual.flatten()), max(ID_actual.flatten())],
               [min(ID_actual.flatten()), max(ID_actual.flatten())], 'r',␣
       ↪linestyle='dashed', label="Ideal Fit")
      plt.xlabel("Actual ID (A)")
      plt.ylabel("Predicted ID (A)")
      plt.title("LND150K1-G: Neural Network Predictions vs Actual ID")
      plt.legend()
      plt.grid(True)
      plt.show()
```

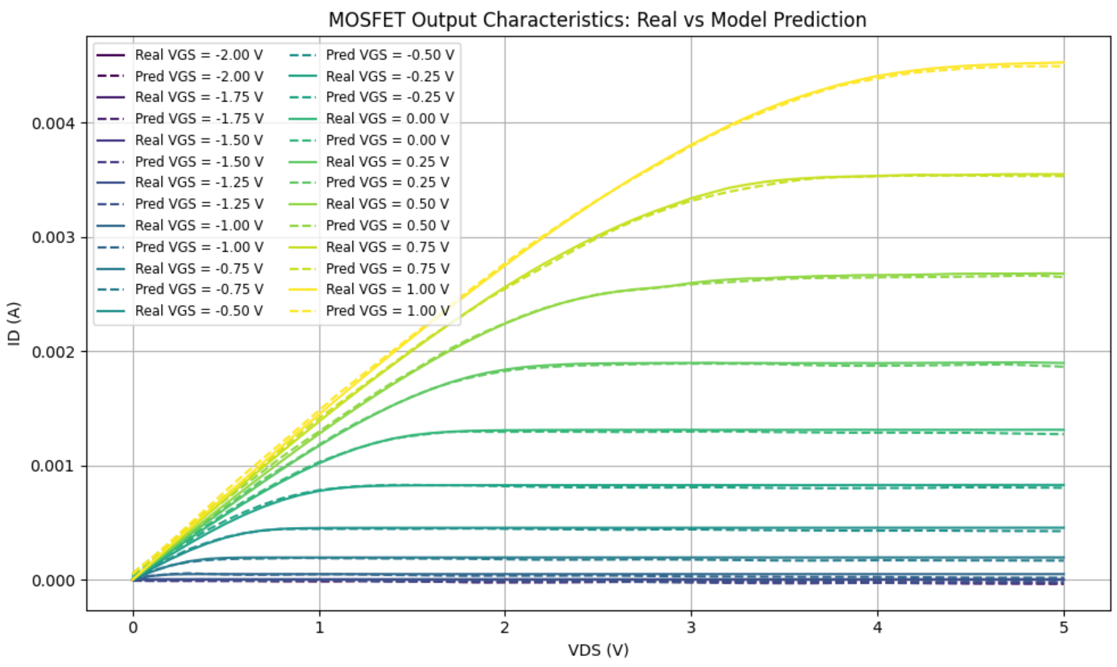


[9]:
```
import numpy as np
import matplotlib.pyplot as plt
from tensorflow.keras.models import load_model
from sklearn.preprocessing import MinMaxScaler

# Load the dataset
file_path = "Ouput 0-5V Gate -2-1V Step 0.25.txt"
data = np.loadtxt(file_path, usecols=(0, 1))

VDS = data[:, 0]
ID = data[:, 1]
VGS_values = np.arange(-2, 1.25, 0.25)
num_VGS = len(VGS_values)
num_points_per_sweep = len(VDS) // num_VGS

VDS_reshaped = VDS[:num_points_per_sweep]
ID_reshaped = ID.reshape(num_VGS, num_points_per_sweep)

# Prepare input features
X = np.array([[VGS, vds] for VGS in VGS_values for vds in VDS_reshaped])
```

```python
# Normalize input
scaler_X = MinMaxScaler()
scaler_y = MinMaxScaler()
scaler_X.fit(np.array([[min(VGS_values), 0], [max(VGS_values), max(VDS)]]))
scaler_y.fit(ID.reshape(-1, 1))
X_scaled = scaler_X.transform(X)

# Load trained model
model = load_model("LND150K1-G.keras")

# Predict and inverse transform
y_pred_scaled = model.predict(X_scaled)
y_pred = scaler_y.inverse_transform(y_pred_scaled).flatten()
ID_pred_reshaped = y_pred.reshape(num_VGS, num_points_per_sweep)

# Plotting
plt.figure(figsize=(10, 6))
colors = plt.cm.viridis(np.linspace(0, 1, num_VGS))

for i, VGS in enumerate(VGS_values):
    plt.plot(VDS_reshaped, ID_reshaped[i], color=colors[i], linestyle='-',␣
 ↪label=f"Real VGS = {VGS:.2f} V")
    plt.plot(VDS_reshaped, ID_pred_reshaped[i], color=colors[i],␣
 ↪linestyle='--', label=f"Pred VGS = {VGS:.2f} V")

plt.xlabel("VDS (V)")
plt.ylabel("ID (A)")
plt.title("MOSFET Output Characteristics: Real vs Model Prediction")
plt.legend(ncol=2, fontsize='small')
plt.grid(True)
plt.tight_layout()
plt.show()
```

```
21/21              0s 3ms/step
```

```
c:\Users\asus1\anaconda3\envs\MOSFET_NN_mini\Lib\site-
packages\keras\src\saving\saving_lib.py:719: UserWarning: Skipping variable
loading for optimizer 'rmsprop', because it has 8 variables whereas the saved
optimizer has 14 variables.
  saveable.load_own_variables(weights_store.get(inner_path))
```



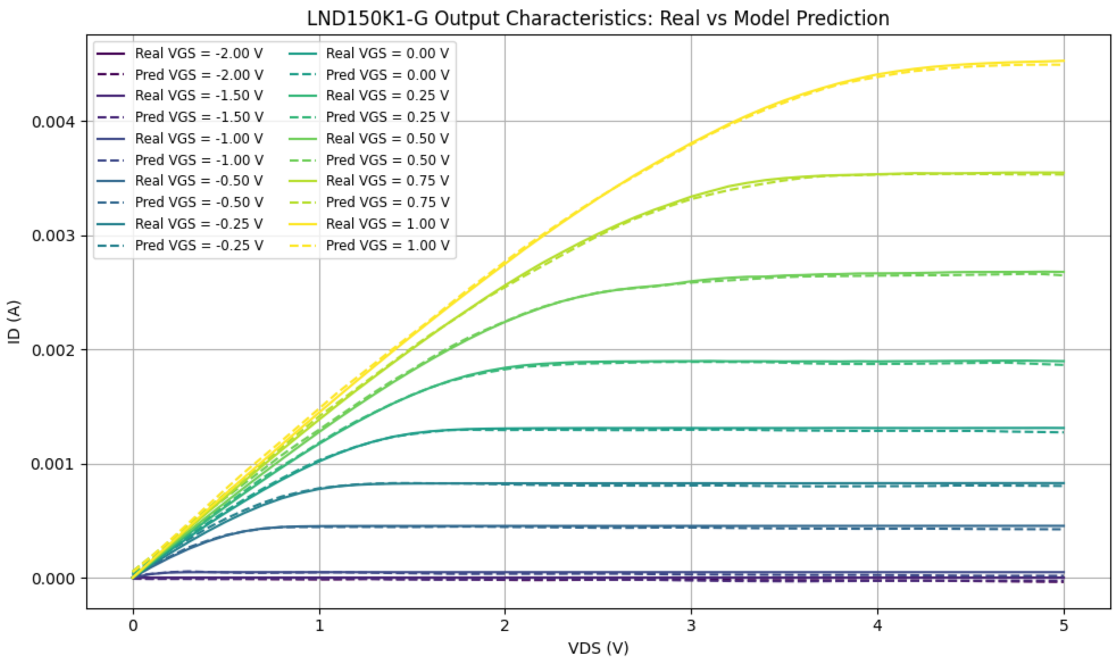
[12]:
```
import numpy as np
import matplotlib.pyplot as plt
from tensorflow.keras.models import load_model
from sklearn.preprocessing import MinMaxScaler

# Load dataset
file_path = "Ouput 0-5V Gate -2-1V Step 0.25.txt"
data = np.loadtxt(file_path, usecols=(0, 1))

VDS = data[:, 0]
ID = data[:, 1]
VGS_values = np.arange(-2, 1.25, 0.25)
excluded_VGS = {-1.75, -1.25, -0.75}

num_VGS = len(VGS_values)
num_points_per_sweep = len(VDS) // num_VGS

VDS_reshaped = VDS[:num_points_per_sweep]
ID_reshaped = ID.reshape(num_VGS, num_points_per_sweep)

# Filter VGS values and corresponding data
included_indices = [i for i, vgs in enumerate(VGS_values) if vgs not in␣
↪excluded_VGS]
filtered_VGS = [VGS_values[i] for i in included_indices]
filtered_ID_real = ID_reshaped[included_indices, :]
```

```python
# Prepare model input
X = np.array([[vgs, vds] for vgs in filtered_VGS for vds in VDS_reshaped])

# Normalize
scaler_X = MinMaxScaler()
scaler_y = MinMaxScaler()
scaler_X.fit(np.array([[min(VGS_values), 0], [max(VGS_values), max(VDS)]]))
scaler_y.fit(ID.reshape(-1, 1))
X_scaled = scaler_X.transform(X)

# Load model and predict
model = load_model("LND150K1-G.keras")
y_pred_scaled = model.predict(X_scaled)
y_pred = scaler_y.inverse_transform(y_pred_scaled).flatten()
ID_pred_reshaped = y_pred.reshape(len(filtered_VGS), num_points_per_sweep)

# Plotting
plt.figure(figsize=(10, 6))
colors = plt.cm.viridis(np.linspace(0, 1, len(filtered_VGS)))

for i, VGS in enumerate(filtered_VGS):
    plt.plot(VDS_reshaped, filtered_ID_real[i], color=colors[i], linestyle='-',␣
  ↪label=f"Real VGS = {VGS:.2f} V")
    plt.plot(VDS_reshaped, ID_pred_reshaped[i], color=colors[i],␣
  ↪linestyle='--', label=f"Pred VGS = {VGS:.2f} V")

plt.xlabel("VDS (V)")
plt.ylabel("ID (A)")
plt.title("LND150K1-G Output Characteristics: Real vs Model Prediction")
plt.legend(ncol=2, fontsize='small')
plt.grid(True)
plt.tight_layout()
plt.show()
```

```
 1/16              1s 69ms/step
```

```
c:\Users\asus1\anaconda3\envs\MOSFET_NN_mini\Lib\site-
packages\keras\src\saving\saving_lib.py:719: UserWarning: Skipping variable
loading for optimizer 'rmsprop', because it has 8 variables whereas the saved
optimizer has 14 variables.
  saveable.load_own_variables(weights_store.get(inner_path))
```

```
16/16              0s 4ms/step
```


LND150K1-G Output Characteristics: Real vs Model Prediction
Real VGS = -2.00 V
Pred VGS = -2.00 V
Real VGS = -1.50 V
Pred VGS = -1.50 V
Real VGS = -1.00 V
Pred VGS = -1.00 V
Real VGS = -0.50 V
Pred VGS = -0.50 V
Real VGS = -0.25 V
Pred VGS = -0.25 V
Real VGS = 0.00 V
Pred VGS = 0.00 V
Real VGS = 0.25 V
Pred VGS = 0.25 V
Real VGS = 0.50 V
Pred VGS = 0.50 V
Real VGS = 0.75 V
Pred VGS = 0.75 V
Real VGS = 1.00 V
Pred VGS = 1.00 V
ID (A)
VDS (V)