The scenario you describe is an example of a known-plaintext attack on a linear block cipher, where the attacker has access to chosen ciphertexts. In this case, because the linear block cipher is linear (i.e., it satisfies the equation EL(k, [m1 ⊕ m2]) = EL(k, m1) ⊕ EL(k, m2)), an adversary can use these properties to recover the secret key k with 128 chosen ciphertexts. Here's how the attack works:

Choose 128 different ciphertexts (C1, C2, ..., C128) that you want to decrypt. These ciphertexts are your chosen ciphertexts.

For each chosen ciphertext Ci (1 ≤ i ≤ 128), submit it for decryption to the encryption oracle with the unknown key k, which will give you the corresponding plaintext Pi: Pi = EL(k, Ci).

Create a system of linear equations using the knowledge that EL(k, [m1 ⊕ m2]) = EL(k, m1) ⊕ EL(k, m2) for all 128-bit patterns m1 and m2. You now have 128 pairs of (Ci, Pi), and you know that Ci = EL(k, Pi).

Set up a system of linear equations based on the XOR properties:

For each pair (Ci, Pi), you can write Ci = EL(k, Pi) as Ci ⊕ EL(k, Pi) = 0.

You now have a system of 128 linear equations with 128 unknown bits of the key k.

Solve this system of linear equations to find the unknown key k. With 128 equations and 128 unknowns, you have enough information to uniquely determine the key.