

# P89LPC9xx CRC Calculation

## © Philips Semiconductors

A 32-bit CRC may be performed on either an individual sector (Sector CRC) or the entire user code memory (Global CRC). Both use the same method for calculating the 32-bit CRC result. The following algorithm is used over the desired memory range.

Define a 32-bit CRC result register (CRC) and set its contents = 0.

Define a 32-bit temporary variable (TAP) and set its contents = 0.

Define a single-bit variable (CRC\_FLAG).

Starting with the first byte in code memory, and for each byte in the memory, perform the following CRC calculation:

1. Save the MSB in the CRC\_FLAG and shift the CRC result (CRC) to the left one bit.
2. Read the byte from code memory and distribute the eight bits of the code-byte into the 32 bits of the TAP variable as shown in the table, below. Unused bits of the TAP variable must be filled with zeros.

Code-byte bit position	Is copied into TAP variable bit position
0	0
1	3
2	5
3	8
4	10
5	13
6	16
7	18

3. XOR the 32-bit TAP variable with the 32-bit CRC variable and save the result in the CRC variable.
4. If the CRC\_FLAG (saved in step 1) was a zero, proceed to step 5, else, XOR the CRC variable with 00400007H. Store the result in the CRC variable.
5. The CRC calculation for THIS byte is finished and the CRC variable holds the current CRC result. Repeat, starting with step 1, for each additional byte of code memory.

If the saved MSB = 1, the byte from the code memory is XOR with 00400007H. This result is XOR with the 32-bit CRC result. The result of this operation is stored as the CRC result.

If the saved MSB = 0, the byte from the code memory is XOR with the 32-bit CRC result. The result of this operation is stored as the CRC result.