Bitwise Operations Test

a = ?
 	
int orig = 0x9B50;
int insert = 0x00EB;
int a = orig & (insert << 21);
	a = ?
	
	int orig = 0xA3C9;
	int insert = 0x00F4;
	int a = orig | (insert << 2);

AND = ?
 	
int orig = 0x924C;
int insert = 0x00BC;
int a = orig | (insert << 10);
int b = orig | (insert << 22);
int AND = a & b;
	OR = ?
	
	int orig = 0x3E91;
	int insert = 0x00E5;
	int a = orig | (insert << 29);
	int b = orig | (insert << 7);
	int OR = a ^ b;

result = ?
 	
long value1 = 0x5AEEF776;
long value2 = 0xF6A44BDA;
int result = (value1 << 7) ^ (value2 >> 3);
	result = ?
	
	long value1 = 0xDD5D9988;
	long value2 = 0x33880C6D;
	int result = (value1 << 25) ^ (value2 >> 26);

result = ?
 	
long value1 = 9104;
long value2 = 1383;
int result = (value1 << 15) ^ (value2 >> 5);
	result = ?
	
	long value1 = 4291;
	long value2 = 4495;
	int result = (value1 << 21) | (value2 >> 11);

right = ?
 	
int i = 0x514D;
int right = i ^ (1 << 10);
	a = ?
	
	long testValue = 0xA0F07040;
	int a = 0;
	if (testValue & (1 << 10))
	{
	a = 1;
	}
	else
	{
	a = 2;
	}

a = ?
 	
``c
long testValue = 0x5E008B00;
int a = 0;
if (testValue
	testValue & (1 << 30))
	{
	a = 1;
	}
	else
	{
	a = 2;
	}
	``
		a = ?, result = ?
		
		``c
		long testValue = 0x500F800F;
		int a = 0;
		if ((result = testValue
			testValue & testValue
			^ (1 << 17)))
			{
			a = 1;
			}
			else
			{
			a = 2;
			}
			``