Net Micro Framework での JIT 機能実装について

DeviceCode\Drivers\BlockStorage\Flash の WriteX

// enable programming

FLASH->CR = FLASH\_CR\_PG | FLASH\_CR\_PSIZE\_BITS;

// --> ::Flash\_ChipReadOnly( FALSE );

while(ChipAddress < EndAddress) {

if (\*ChipAddress != \*pBuf) {

// write data

\*ChipAddress = \*pBuf;

// wait for completion

while (FLASH->SR & FLASH\_SR\_BSY);

// check

if (\*ChipAddress != \*pBuf) {

debug\_printf( "Flash\_WriteToSector failure @ 0x%08x, wrote 0x%08x, read 0x%08x\r\n", (UINT32)ChipAddress, \*pBuf, \*ChipAddress );

return FALSE;

}

}

ChipAddress++;

pBuf++;

}

// reset & lock the controller

FLASH->CR = FLASH\_CR\_LOCK;

// --> ::Flash\_ChipReadOnly( TRUE );

-------------------------------------------------

// if same, nothing to do, continue nextword.

if(\*ChipAddress != \*pData)

{

// check for having to move bits from 0->1, a failure case for a write

if(0 != (\*pData & ~(\*ChipAddress)))

{

debug\_printf( "erase failure: 0x%08x=0x%04x\r\n", (size\_t)ChipAddress, \*ChipAddress );

ASSERT(0);

result =FALSE;

break;

}

Action\_WriteWord( config, ChipAddress, \*pData );

if (\*ChipAddress != \*pData)

{

debug\_printf( "Flash\_WriteToSector failure @ 0x%08x, wrote 0x%08x, read 0x%08x\r\n", (UINT32)ChipAddress, \*pData, \*ChipAddress );

result = FALSE;

break;

}

}