**Listing 1. Konfigurowanie QUADSPI**

// QSPI - GD25Q32 connected as block2

QUADSPI**->**DCR **=** 21 **<<** QUADSPI\_DCR\_FSIZE\_Pos **|** 7 **<<** QUADSPI\_DCR\_CSHT\_Pos**;** // 22 addr bits, 8 idle cycles between accesses

QUADSPI**->**CR **=** QUADSPI\_CR\_APMS **|** 0 **<<** QUADSPI\_CR\_PRESCALER\_Pos **|** QUADSPI\_CR\_FSEL **|** QUADSPI\_CR\_EN**;**

QUADSPI**->**PIR **=** 100**;**

QUADSPI**->**PSMKR **=** 1**;** //status match mask - write in progress bit

//QUADSPI->PSMAR = 0; // status match pattern - write in progress = 0 (default val)

QUADSPI**->**LPTR **=** 200**;** // memory mapped mode timeout

**Listing 2. Procedury dla trybu podstawowego**

static \_Bool fl\_writeop**,** fl\_mmread**;**

static void sf\_waitrdy**(**void**)**

**{**

**while** **(**QUADSPI**->**SR **&** QUADSPI\_SR\_BUSY**);**

QUADSPI**->**FCR **=** QUADSPI\_FCR\_CTOF **|** QUADSPI\_FCR\_CSMF **|** QUADSPI\_FCR\_CTCF **|** QUADSPI\_FCR\_CTEF**;**

**}**

// enable write access

static void sf\_wren**(**void**)**

**{**

sf\_waitrdy**();**

QUADSPI**->**CCR **=** QUADSPI\_CCR\_FMODE\_IW **|** QUADSPI\_CCR\_IMODE\_0 **|** 0x06**;**

sf\_waitrdy**();**

**}**

// enable quad mode

void sfll\_enquad**(**void**)**

**{**

sf\_wren**();**

QUADSPI**->**DLR **=** 0**;**

QUADSPI**->**CCR **=** QUADSPI\_CCR\_FMODE\_IW **|** QUADSPI\_CCR\_DMODE\_0 **|** QUADSPI\_CCR\_IMODE\_0 **|** 0x31**;**

QUADSPI\_DRB **=** 0x02**;**

sf\_waitrdy**();**

**}**

// write memory page

static void sf\_mbwrite**(**uint32\_t addr**,** const uint8\_t **\***data**,** uint32\_t dlen**)**

**{**

**if** **(!**dlen**)** **return;**

sf\_wren**();**

// send command, then send and receive data from/to data buffer

QUADSPI**->**DLR **=** dlen **-** 1**;**

// quad page program

QUADSPI**->**CCR **=** QUADSPI\_CCR\_FMODE\_IW **|** QUADSPI\_CCR\_DMODE4 // 4 lines

**|** QUADSPI\_CCR\_ADSIZE24 **|** QUADSPI\_CCR\_ADMODE1 // 1 line

**|** QUADSPI\_CCR\_IMODE\_0 **|** 0x32**;**

QUADSPI**->**AR **=** addr**;**

**while** **(**dlen **--)**

**{**

**while** **(~**QUADSPI**->**SR **&** QUADSPI\_SR\_FTF**);**

QUADSPI\_DRB **=** **\***data**++;**

**}**

fl\_writeop **=** 1**;**

**}**

// erase sector

void sfll\_mserase**(**uint32\_t addr**)**

**{**

sf\_wren**();**

QUADSPI**->**CCR **=** QUADSPI\_CCR\_FMODE\_IW

**|** QUADSPI\_CCR\_ADMODE\_0 **|** QUADSPI\_CCR\_ADSIZE24

**|** QUADSPI\_CCR\_IMODE\_0 **|** 0x20**;**

QUADSPI**->**AR **=** addr**;**

fl\_writeop **=** 1**;**

**}**

**Listing 3. Odpytywanie – oczekiwanie na gotowość pamięci po zapisie lub kasowaniu**

static void sf\_waitwrdy**(**void**)**

**{**

sf\_waitrdy**();**

**if** **(**fl\_writeop**)**

**{**

QUADSPI**->**DLR **=** 0**;** // 1 byte

QUADSPI**->**CCR **=** QUADSPI\_CCR\_FMODE\_AP **|** QUADSPI\_CCR\_DMODE\_0 **|** QUADSPI\_CCR\_IMODE\_0 **|** 0x05**;**

**while** **((**QUADSPI**->**SR **&** QUADSPI\_SR\_SMF**)** **==** 0**);**

QUADSPI**->**FCR **=** QUADSPI\_FCR\_CTOF **|** QUADSPI\_FCR\_CSMF **|** QUADSPI\_FCR\_CTCF **|** QUADSPI\_FCR\_CTEF**;**

fl\_writeop **=** 0**;**

**}**

**}**

**Listing 4. Programowanie odczytu w trybie przezroczystym**

void sfll\_startmm**(**void**)**

**{**

sf\_waitwrdy**();**

sf\_waitrdy**();**

// quad io fast read - ok @ 80 MHz FAST

QUADSPI**->**CCR **=** QUADSPI\_CCR\_FMODE\_MM **|** QUADSPI\_CCR\_DMODE4 // 4 lines

**|** 4 **<<** QUADSPI\_CCR\_DCYC\_Pos // 4 clocks

**|** QUADSPI\_CCR\_ABMODE4 // 4 lines, 1 byte

**|** QUADSPI\_CCR\_ADSIZE24 **|** QUADSPI\_CCR\_ADMODE4 // 4 lines

**|** QUADSPI\_CCR\_IMODE\_0 **|** 0xeb**;**

fl\_mmread **=** 1**;**

**}**