HASH Generator Functional Specification

Rev. 1.00
2021.05.09

This document describes functional specification of HASH generator module (HAGE).

【References】

- N/A

Rev.	Issued on	Section	Revised on	Revised by
0.01	2021.05.09	ALL	Newly created	Bac Huynh

Table of Contents

1. Overview 3

1.1 Features 3
1.2 HASH function 3
1.3 Block Diagram 4
1.4 Interface 4
1.5 Register Configuration 4

2. Register Description 5

2.1 HAGE Control Register (HAGECR) 5
2.2 HAGE Seed Register (HAGESEED) 6
2.3 HAGE Input Data Register (HAGEIDAT) 6
2.4 HAGE Output Data Register (HAGEODAT) 7
2.5 HAGE Source Address Register (HAGESRCADD) 7
2.6 HAGE Destination Address Register (HAGEDSTADD) 8
2.7 HAGE DMA Data Length Register (HAGEDL) 8

3. Operation 9

3.1 HASH generator performance 9
3.2 Block by block mode 9
3.3 DMA mode 9

1. Overview

1.1 Features

HAGE is a simple 32-bits HASH function generator

- Configurable seed with register
- Single block mode
- DMA mode (pre-fetch 128-bit data equivalent to 4 data blocks)
- Interrupt when computation finish (1 block in single block mode, whole message in DMA mode)

1.2 HASH function

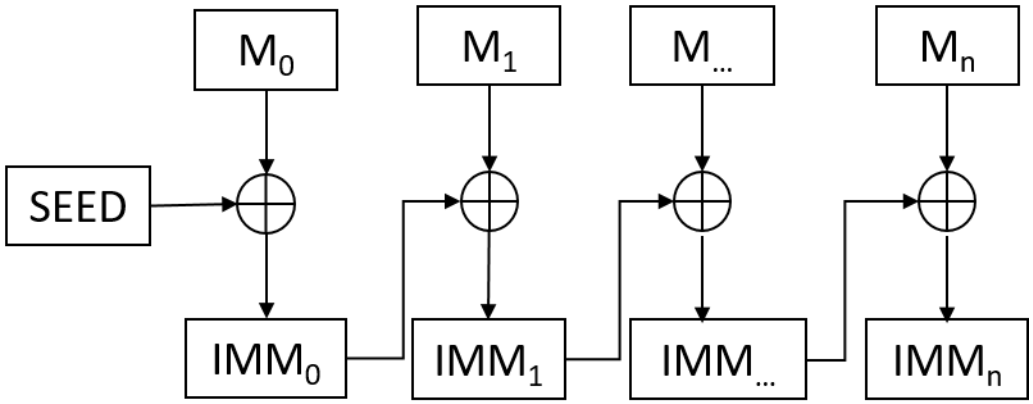


Figure 1-1 Hash function

Explanation:

- SEED: initial block (32 bit)
- M_0, M_1…M_n: input data block (32 bit)
- IMM_0, IMM_1…IMM_n: immediate result of block (32 bit)
- Formula:

  If (n = 0)

  $IMM_n = M_n$ XOR SEED

  Else

  $IMM_n = M_n$ XOR IMM_{n-1}

Example:

- Message: 0x0001020304050607
- Seed: 0x12345678
- Result:
  - IMM_0 = 0x00010203 XOR 0x12345678 = 0x1235547B
  - IMM_1 = 0x04050607 XOR 0x1235547B = 0x1630527C

1.3 Block Diagram

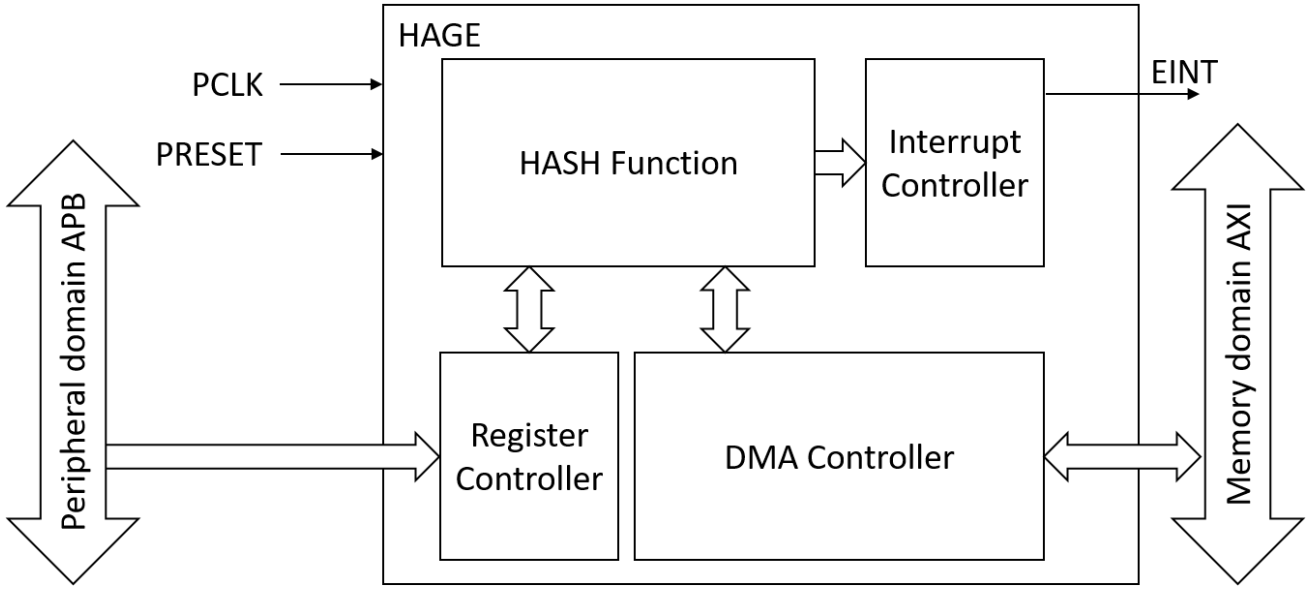


Figure 1-2 Block diagram

1.4 Interface

Interface list of HAGE.

I/F	Direction	Sync	Active	Initial	Description
PCLK	Input	-	-	-	System clock
PRESET	Input	-	HIGH	-	Asynchronous reset
EINT	Output	PCLK	HIGH	LOW	HASH generate complete interrupt
APB Slave	Input	-	-	-	32 bit APB slave (register controller)
AXI Master	Input/Output	-	-	-	128 bit AXI master (read/write memory)

Table 1-1 HAGE Interfaces

1.5 Register Configuration

HAGE base address: 0xFFBF0000

Symbol	Offset	R/W	Access Size			Description
			8	16	32	
HAGECR	xxx_base+0x00	R/W	✓	✓	✓	Control register
HAGESEED	xxx_base+0x04	R/W	-	-	✓	Seed setup register
HAGEIDAT	xxx_base+0x08	R/W	-	-	✓	Input data register
HAGEODAT	xxx_base+0x0C	R/W	-	-	✓	Output data register
HAGESRCADD	xxx_base+0x10	R/W	-	-	✓	DMA source address
HAGEDSTADD	xxx_base+0x14	R/W	-	-	✓	DMA destination address
HAGEDL	xxx_base+0x18	R/W	-	-	✓	DMA data length

Table 1-2 HAGE Register Configuration

2. Register Description

Explanation of abbreviation of register

Initial value: Value of the register after power-on reset

- : Undefined value

R/W : The bit or field is readable and writable.

R : The bit or field is readable only. When writing to the register, write 0 to it.

2.1 HAGE Control Register (HAGECR)

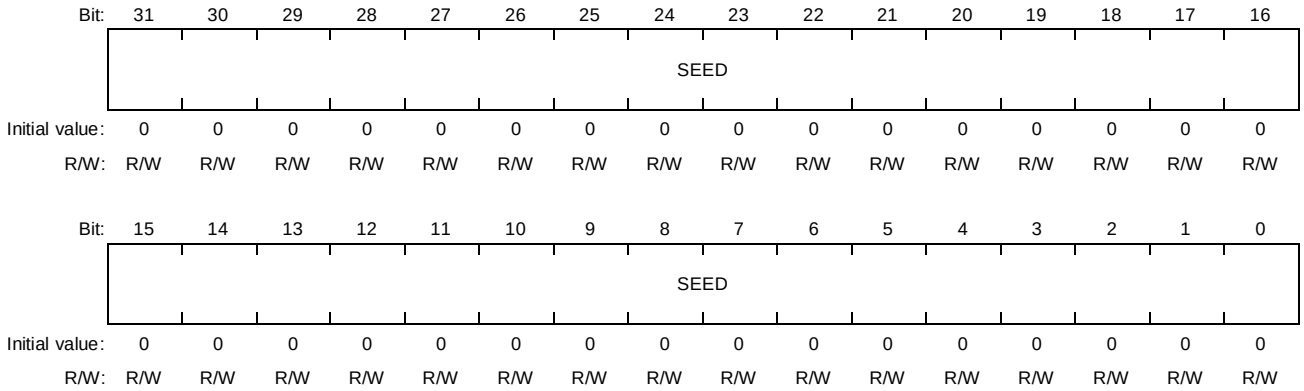
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	DMASTR	-	-	-	-	-	-	PLL	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	EINT_MSK	-	-	-	-	-	HASHOUT	MODSEL	NEW_SEED
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	R	Reserved. These bits are always read as 0. The write value should always be 0.If a value other than 0 is written, correct operation cannot be guaranteed.
24	DMASTR	0	R/W	Start DMA transfer 0: No DMA transfer, write 0 when operating will force stop DMA operation 1: Start DMA transfer, auto set to 0 when DMA operation finish
23 to 18	-	All 0	R	Reserved. These bits are always read as 0. The write value should always be 0.If a value other than 0 is written, correct operation cannot be guaranteed.
17, 16	PLL	All 0	R/W	Increate HASH block computation speed by PLL 0b00: 1 x PCLK 0b01: 4 x PCLK 0b10: 8 x PCLK Others are prohibited
15 to 9	-	All 0	R	Reserved. These bits are always read as 0. The write value should always be 0.If a value other than 0 is written, correct operation cannot be guaranteed.
8	EINT_MSK	0	R/W	Mask EINT interrupt 0: No mask 1: Masked interrupt
7 to 2	-	All 0	R	Reserved. These bits are always read as 0. The write value should always be 0.If a value other than 0 is written, correct operation cannot be guaranteed.
2	HASHOUT	0	R/W	Write HASH final result to destination memory (DMA mode only) 0: No transfer data to memory 1: Transfer data to memory
1	MODSEL	0	R/W	Mode select 0: Block by block mode 1: DMA mode

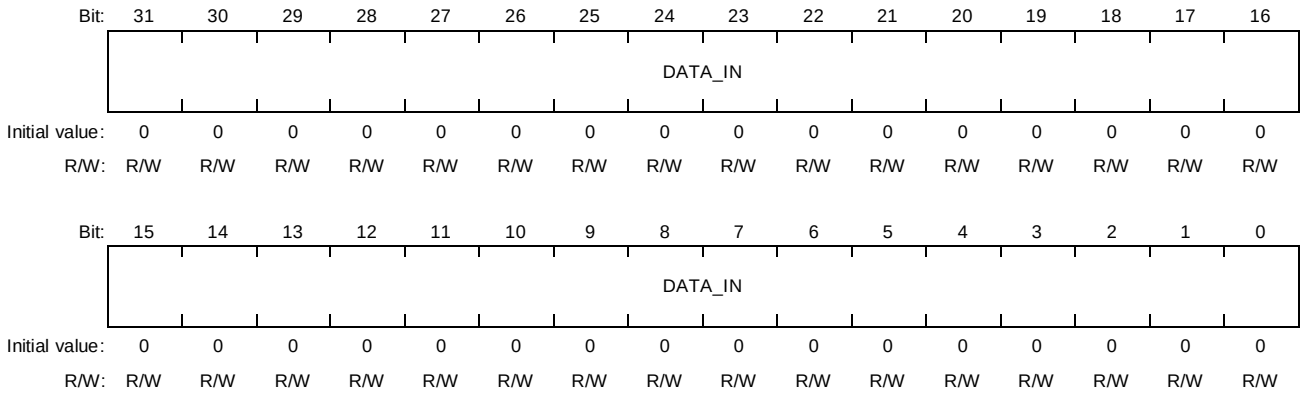
0	NEW_SEED	0	R/W	New seed indicator 0: No new seed available. Compute on previous intermediate data 1: New seed available. Clear to 0 when seed is used

2.2 HAGE Seed Register (HAGESEED)



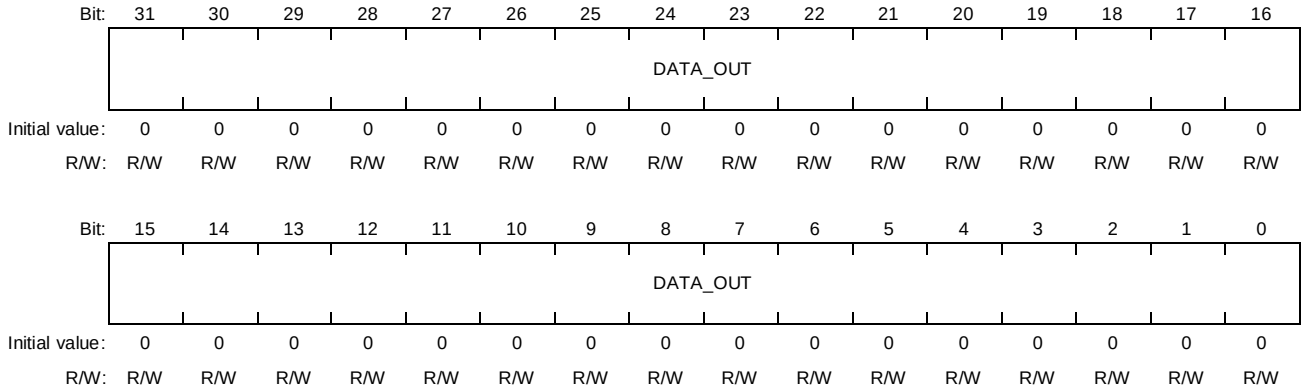
Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SEED	All 0	R/W	HASH seed data

2.3 HAGE Input Data Register (HAGEIDAT)



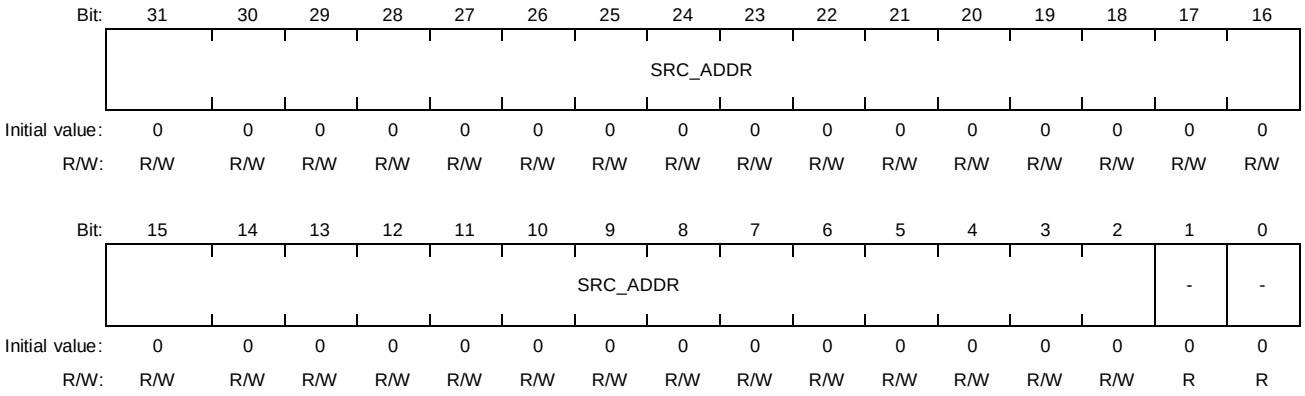
Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA_IN	All 0	R/W	Data in use in block by block mode

2.4 HAGE Output Data Register (HAGEODAT)



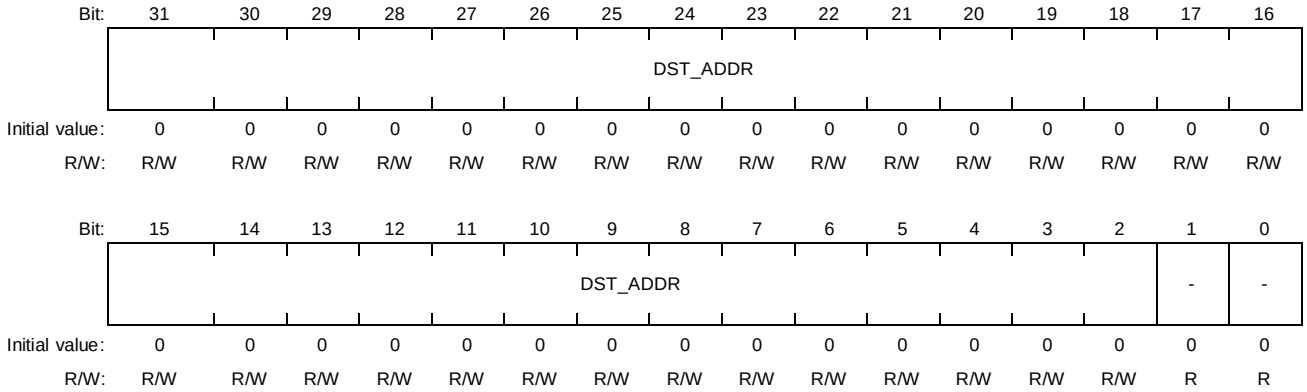
Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA_OUT	All 0	R/W	Result of each data block calculation. Result is updated to this register for each block HASH computation

2.5 HAGE Source Address Register (HAGESRCADD)



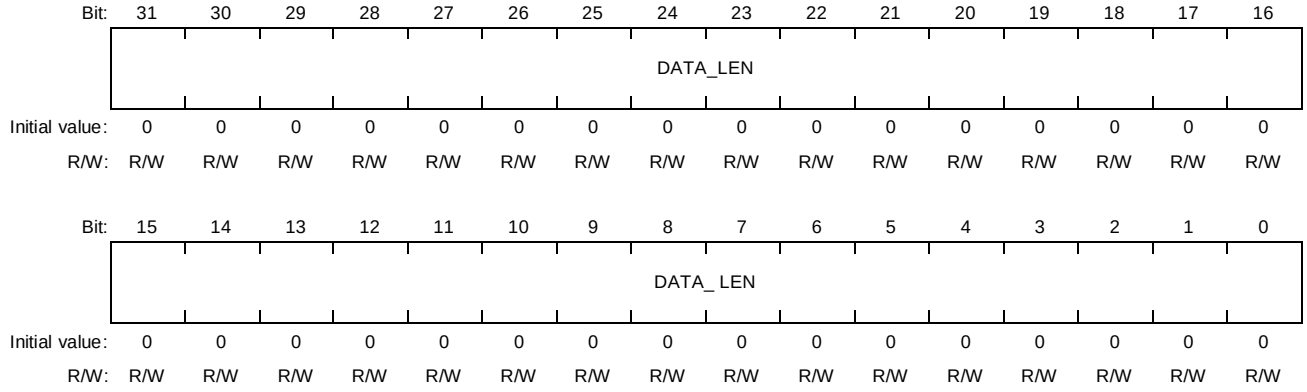
Bit	Bit Name	Initial Value	R/W	Description
31 to 2	SRC_ADDR	All 0	R	Source address
1 to 0	-	All 0	R	Reserved. These bits are always read as 0. The write value should always be 0.If a value other than 0 is written, correct operation cannot be guaranteed.

2.6 HAGE Destination Address Register (HAGEDSTADD)



Bit	Bit Name	Initial Value	R/W	Description
31 to 2	DST_ADDR	All 0	R	Destination address
1 to 0	-	All 0	R	Reserved. These bits are always read as 0. The write value should always be 0.If a value other than 0 is written, correct operation cannot be guaranteed.

2.7 HAGE DMA Data Length Register (HAGEDL)



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA_LEN	All 0	R/W	Number of data block to digest in HASH function

3. Operation

3.1 HASH generator performance

HASH block generating capacity depends on it HASHCLK = PCLK x PLL. Each block computation and update to HAGEODAT register after 1 HASHCLK period.

3.2 Block by block mode

Manual HASH digest by writing each block to data register

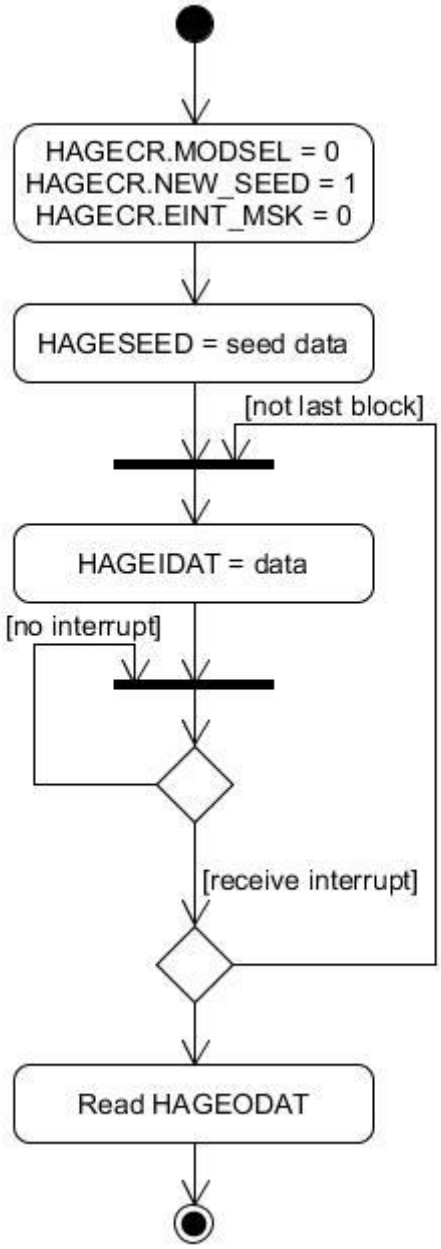


Figure 3-1 Block by block mode operation flow

3.3 DMA mode

HASH digest by DMA transfer. Result can be selected to transfer to memory or read directly in HAGEODAT register

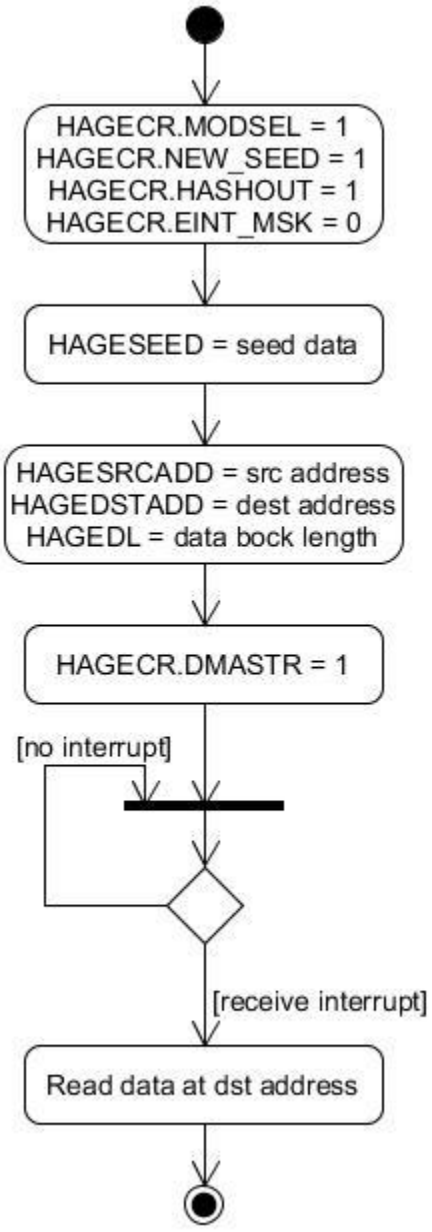


Figure 3-2 DMA mode with result to memory