Light Hash algorithm (AES S-box based)

Referent

Email luca.crocetti@phd.unipi.it
Teams l.crocetti@studenti.unipi.it

Project

Design the following hash module based on the S-box of AES algorithm. The hash algorithm generates a 64-bit digest formed by the concatenation of 8 bytes $H[i]$, from $H[0]$ up to $H[7]$, being $H[0]$ the MSB. For each message the $H[i]$ variables are initialized with the following values:

	$\boldsymbol{H[0]}$	$\boldsymbol{H[1]}$	$\boldsymbol{H[2]}$	$\boldsymbol{H[3]}$	$\boldsymbol{H[4]}$	$\boldsymbol{H[5]}$	$\boldsymbol{H[6]}$	$\boldsymbol{H[7]}$
Init. value	8'h34	8'h55	8'h0F	8'h14	8'hDA	8'hC0	8'h2B	8'hEE

The, for each byte M of the input message (i.e. the 8-bit ASCII code of a message character) the hash module performs the following operation:

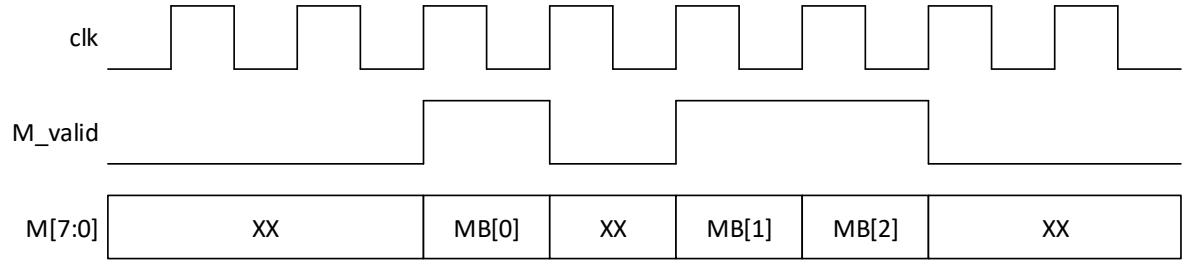
$$for\ (r = 0;\ r < 32;\ r++)$$
$$for(i = 0; i < 8; i++)$$
$$H[i] = S\big((H[(i+2)\ mod\ 8] \oplus M) \ll i\big)$$

where

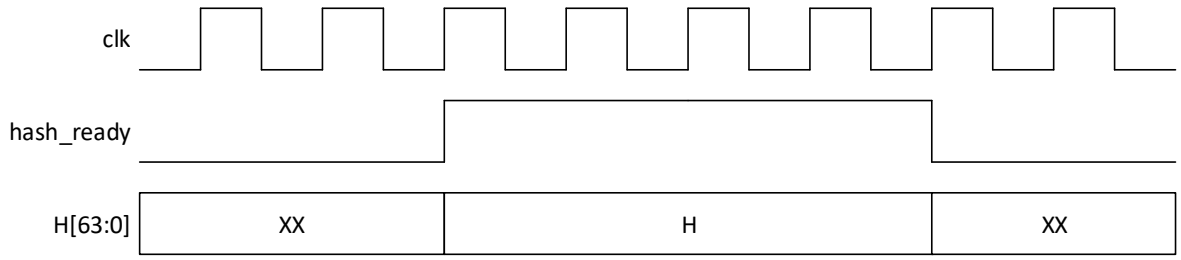
$mod\ n$	is the modulo operator by n.
$\oplus$	is the XOR operator.
$X \ll n$	is the left circular shift by n bits.
$S(\ \ )$	is the S-box transformation of AES algorithm, that works over a byte.

Additional design specifications

- The stream cipher shall have an asynchronous active-low reset port;
- The input message byte M can be any 8-bit ASCII character;
- The stream cipher shall feature an input port which has to be asserted when providing the input message byte M (*M_valid* port): 1'b1, when input character is valid and stable, 1'b0, otherwise; the following waveform is expected at input interface of hash module



- The stream cipher shall feature an output port which is asserted when the generated output digest (or hash value) is available at the corresponding output port (*hash_ready* port): 1'b1, when output digest is valid and stable, 1'b0, otherwise; this flag shall be kept to logic 1 until a new message digest computation is performed; the following waveform is expected at the output interface of hash module



Hints

- The AES S-box function is largely documented online: implement the LUT version (for faster developing).
  Below it is reported the S-box of AES algorithm, in hexadecimal format: it works on a byte, using the 4 MSb and the 4 LSb of input byte, respectively, as row and column coordinates to substitute it.

	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f
00	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
10	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
20	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
30	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
40	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
50	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
60	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
70	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
80	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
90	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
a0	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
b0	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
c0	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
d0	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
e0	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
f0	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

  For instance, assuming to apply the S-box transformation to the byte (hex) 8'hd3, the result (hex) is 8'h66, i.e. the cross between row d0 and column 03: thus $S(8'hd3) = 8'h66$.
- Design logic resources to perform the assignment of all the variables $H[i]$ in the same clock cycle.
- Logic resources (and maybe dedicated ports) are required to signal the beginning and the end of a message (by bytes): this is required to initialize the variables $H[i]$ and to trigger the last operation of the hash algorithm (thus to allow the module to signal when the digest is available at the output).
- Develop testbench with messages of different length and check that the same message generates the same hash value.