M32 waveforms

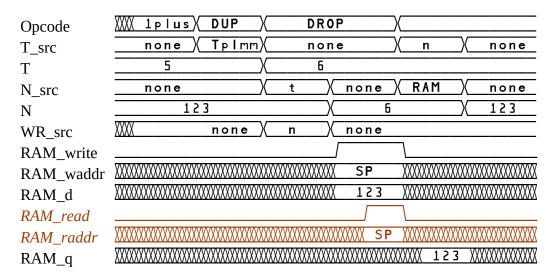
Block RAM

Block RAM uses synchronous read, whose signals are asynchronously decoded from **opcode**. Synchronous opcode decoding sets up the sources for various registers. For example, **new_T** indicates that **T** is to be loaded from 1 of 16 sources.

DROP followed by DUP:

Opcode	1 plus SWAP plus DUP	
T_src	none \ Tpimm\ n \ Tpn \ non	е
T	5 \ 6 \ 123 \ 129	
N_src	none t RAM t	none
N	123 / 6 / 321	129
WR_src	mone n	none
RAM_read		
RAM_raddr	\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
RAM_q		XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
RAM_write		
RAM_waddr		XX SP XX
RAM_d		XX 321 XX

DUP followed by DROP: To prevent DROP's read from occurring before DUP's write, "WR_src" must be "none" to allow decoding when is a read is expected. Simultaneous read/write to same address can be worked around.



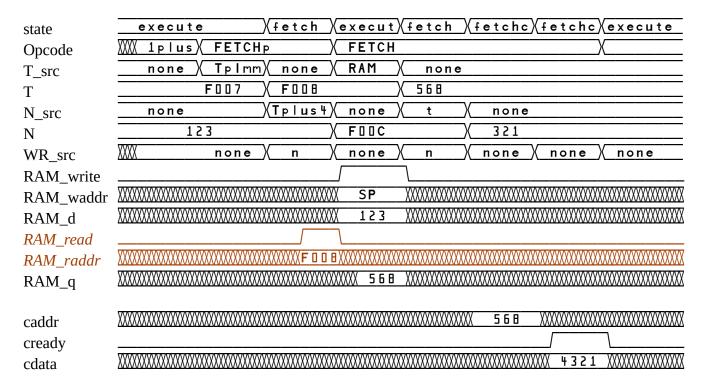
If single-port RAM is used, decoding must also be held off while **RAM_write** is high.

M32 Waveforms Page 1/2

Some opcodes request read and write operations. For example, >R.

Opcode	XXX toR X
T_src	none n none
T	5 321
N_src	none X RAM X none
N	321 \(123
WR_src	mone t none
_	
RAM_write	
RAM_write	\(\text{\lambda}\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
RAM_write RAM_waddr	
RAM_write RAM_waddr RAM_d	

Fetch needs T before it can start a read. @+(a-a+4n). The fetch state checks T and fetches from ROM space (instead) if necessary.



M32 Waveforms Page 2/2