	0 1 2 3 4 5 6 7 8 9 1 1 1 1 1 1 1 4 8 9 1 1 1 2 3 4 5 6 7 8 9 1 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7 0 1 2 3 4 5 6	1 1 1 1 2 2 2 2 2 5 6 7 8 9 0 1 2 3 4 5 6 7 0	2 2 2 2 2 2 3 3 3 4 5 6 7 8 9 0 1 Byte 3	3 3 3 3 3 3 3 3 3 3 3 4 4 4 4 4 4 4 4 4	4 4 5 5 5 5 5 5 5 5	
			ATOMIC Req	uest Packets		
Byte 0 >	ackID S E prio tt ftype 0 0 1 0	1	destina	ationID	sourceID [0:15]	
Byte 8 >	sourceID [16:31]	transaction wrsize	srcTID	Extended Address [0:15]	Address [0:15]	
Byte 16 >	Address [16:29] 법 xai	n s		double-word 0 [0:47]		
Byte 24 >	double-word 0 [48:63] Final CRC		 			
			STOUT, FLUSH w	rith data Request Packets (Where	N < 8)	
Byte 0 >	ackID S B prio tt of type of 1 0	1	destina	ationID	sourceID [0:15]	
Byte 8 >	sourceID [16:31]	transaction wrsize	srcTID	Extended Address [0:15]	Address [0:15]	
Byte 16 >	Address [16:29]	Address [16:29] $\frac{\xi}{\xi} \left \begin{array}{c} xam \\ sbs \end{array} \right $ double-word 0 [0:47]				
Byte (16 + (N-1)*8) >	double-word 0 [48:63]	double-word n [0:47]				
Byte (16 + N*8) >	double-word n [48:63]	Final C	:RC	 		
	NWF	ITE, NWRITE_R, CA	STOUT, FLUSH v	vith data Request Packets (Where	e N = 8)	
Byte 0 >	ackID S B prio 0 0 0 1 0	1	destina	ationID	sourceID [0:15]	
Byte 8 >	sourceID [16:31]	transaction wrsize	srcTID	Extended Address [0:15]	Address [0:15]	
Byte 16 >	Address [16:29]	n s		double-word 0 [0:47]		
Byte 72 >	double-word 0 [48:63]			double-word 7 [0:47]		
Byte 80 >	Early CRC	double-word	17 [48:63]	double-word 8 [0:31]		
Byte 88 >	double-v	vord 8 [32:63]		Final CRC	Logic 0 Pad	
			İ		1 1 1	
 	NWRITE, NWRITE_R, CASTOUT, FLUSH with data Request Packets (Where N > 8)					
Byte 0 >	ackID	1	destina	ationID	sourcelD [0:15]	
Byte 8 >	sourceID [16:31]	transaction wrsize	srcTID	Extended Address [0:15]	Address [0:15]	
Byte 16 >	Address [16:29]	n s		double-word 0 [0:47]		
Byte 72 >	double-word 0 [48:63]			double-word 7 [0:47]		
Byte 80 >	Early CRC double-word 7 [48:63]		double-word 8 [0:31]			
Byte 88 >	double-word 8 [32:63]			double-word 9 [0:31]		
				,		
Byte (16 + (N-1)*8) >	double-word n-1 [32:63]			double-word <i>n</i> [0:31]		
Byte (16 + N*8) >	double-word <i>n</i> [32:63]			Final CRC	Logic 0 Pad	

Notes: N is the number of double-words in the payload. n = N-1