Table 7: Run-Times (in milliseconds unless stated otherwise) for different atomic operations and comparison with prior art. Each experiment is performed for 1,000 operations on 32-bit numbers in parallel. The detailed performance results for ABY [35] are provided for three different modes of operation: GC, GMW, and Additive. Minimum values marked in bold.

	TinyGarble [83]	**ABY-GC** [35]		**ABY-GMW** [35]		**ABY-A** [35]		**Sharemind** [18]	**Chameleon**	
Op	Online	Offline	Online	Offline	Online	Offline	Online	Online	Offline	Online
ADD	1.57 s	11.71	2.73	25.78	4.73	**0.00**	**0.00**	1 μs	**0.00**	**0.00**
MULT	2.31 s	423.82	112.29	174.52	14.25	10.46	0.59	17	**4.24**	**0.13**
XOR	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**			1 μs	**0.00**	**0.00**
AND	1.58 s	11.83	2.34	9.27	**0.52**			17	**1.50**	0.56
CMP	1.57 s	11.90	2.63	17.39	1.63			2.5 s	**2.46**	**1.48**
EQ	1.56 s	11.60	2.42	9.11	1.15			5 s	**1.54**	**1.09**
MUX	1.59 s	11.91	2.49	**1.06**	0.68			34	1.52	**0.63**

Table 8: Communication (in kilobytes unless stated otherwise) for different atomic operations and comparison with prior art. Each experiment is performed for 1,000 operations on 32-bit numbers in parallel. The detailed performance results of the ABY framework [35] is provided for three modes of operation: GC, GMW, and Additive. Minimum values marked in bold.

	TinyGarble [83]	**ABY-GC** [35]		**ABY-GMW** [35]		**ABY-A** [35]		**Sharemind** [18]	**Chameleon**	
Op	Total	Offline	Online	Offline	Online	Offline	Online	Total	Offline	Online
ADD	7936	992	**0**	3593	76	**0**	**0**	**0**	**0**	**0**
MULT	318 K	47649	**0**	37900	840	1280	16	192	**8**	16
XOR	**0**	**0**	**0**	**0**	**0**			**0**	**0**	**0**
AND	8192	1024	**0**	1028	16			192	**12**	8
CMP	8192	1024	**0**	2851	45				**23**	33
EQ	7936	992	**0**	995	16				**8**	12
MUX	8192	1024	**0**	33	8			384	**8**	4

Table 9: Run-Times (in milliseconds) for conversion operations and comparison with prior art. Each experiment is performed for 1,000 operations on 32-bit numbers in parallel. Minimum values marked in bold.

	ABY [35]		**Chameleon**	
Op	Offline	Online	Offline	Online
GC2GMW	**0.00**	**0.00**	**0.00**	**0.00**
GMW2A	9.47	2.44	**3.45**	**2.33**
GMW2GC	17.05	1.30	**13.24**	**1.15**
A2GC	19.75	14.03	**15.83**	**12.91**

Table 10: Communication (in bits) in the offline phase in Chameleon compared to prior art ABY [35].

	ABY [35]	**Chameleon**	**Improvement**
OT	128	128	-
B-MT	256	1	256×
A-MT (bitlength $\ell = 16$)	4,368	16	273×
A-MT (bitlength $\ell = 32$)	9,248	32	289×
A-MT (bitlength $\ell = 64$)	20,544	64	321×

D FURTHER RELATED WORKS ON PRIVACY-PRESERVING MACHINE LEARNING

One of the earliest solutions for obliviously evaluating a neural network was proposed by Orlandi et al. [71]. They suggest adding fake neurons to the hidden layers in the original network and evaluating the network using HE. Chabanne et al. [28] also approximate the ReLu non-linear activation function using low-degree polynomials and provide a normalization layer prior to the activation layer. However, they do not report experimental results. Sadeghi and Schneider proposed to utilize universal circuits to securely evaluate neural networks and fully hide their structure [79]. Privacy-preserving classification of electrocardiogram (ECG) signals using neural networks has been addressed in [10]. The recent work of Shokri and Shmatikov [81] is a Differential Privacy (DP) based approach for the distributed training of a Neural Network and they do not provide secure DNN or CNN inference. Due to the added noise in DP, any attempt to implement secure inference suffers from a significant reduction in accuracy of the prediction. Phong et al. [58] propose a mechanism for privacy-preserving deep learning based on additively homomorphic encryption. They do not consider secure deep learning inference (classification). There are also limitations of deep learning when an adversary can craft malicious inputs in the training phase [72]. Moreover, deep learning can be used to break semantic image CAPTCHAs [82].