state transaction functions - (transaction)  $\Upsilon \to$  (give rewards)  $\Omega \to$  (block)  $\Pi$   $\sigma \to \sigma_0 \to \sigma_P \to \sigma^* \to \sigma'$   $\sigma \to \sigma'_1 \to \sigma_1 \to \sigma^{**} \to \sigma'$ 

Greek	Αl	phabets

	Greek Alphabets	
α	stack items added	9.4.1, 9.5
$lpha_w$	!!!	9.4.2, 9.5
$\Delta$	!!!	9.5
δ	stack items removed	9.4.1, 9.5
δ	number of stack items for a given operation	3, 9.4.2
$\delta_w$	!!!	9.4.2, 9.5
$\epsilon$	exponential difficulty symbol for difficulty bomb	<b>4.3.4</b> (4.4.4)
	(or ice age)	- ( )
Γ()	function that maps a block to its initiation state	11.4
$\overline{\mu}$	machine-state	3, 9.4.0, 9.4.1,
		9.4.2, 9.5
$\mu'$	resultant machine state	9.4.0(), 9.5
$oldsymbol{\mu}_{ m g}$	!!!	9.4.0, 9.4.2
$oldsymbol{\mu}_{ m i}$	!!!	9.4.0
$oldsymbol{\mu}_{ ext{i}}'$	!!!	9.5
$\mu_{ m m}$	machine's memory	3, 9.4.0, 9.4.1
$\boldsymbol{\mu}_{\mathbf{m}}'$	!!!	9.5
$\frac{\mu_{\rm o}}{\mu_{\rm o}}$	!!!	9.4.0, 9.4.2()
$\mu_{ m pc}$	$(\mathbb{N}_{256})$	9.4.0, 9.4.1, 9.5
$\mu_{ m pc}'$	!!!	9.5
	machine's stack	3, 9.4.0, 9.4.2, 9.5
$\mu'_{s}$	!!!	9.5
$\mu_{\rm g}$	!!!	9.4.0()
$\frac{\mu_{\rm s}}{\mu_{\rm g}'}$ $\frac{\mu_{\rm g}}{\mu_{\rm g}'}$	remaining gas from the resultant machine state $(\mu')$	9.4.0()
Λ	!!!	(6.2), 7
$\frac{\Lambda}{\Lambda_4}$		6.2()
$\frac{\Omega(1)}{\Omega(1)}$	block-finalisation state transition function that	2.0, 11.3, 11.4
22()	gives rewards	2.0, 11.0, 11.1
Π()	block-level state transition function	2.0, 4.3.2(4.4.2),
		11.4
$\Phi$	block transition function which maps an incom-	11.4
	plete block to a complete block	
$\boldsymbol{\sigma}, \boldsymbol{\sigma}_t$	world-state, global state, sequence of accounts	2.0, $3,$
		4.3.2(4.4.2),
		6.0, 7, 8, 9.3,
		9.4.0, 9.4.2, 9.5,
	G., -1 (-, -+ +,+;, -1) -+-+-	11.3, 11.4
$oldsymbol{\sigma}'$	final (post-transactional) state	6.0, 6.2, 7, 8, 9.3,
$\sigma^*$	pre-final state	9.4.0(), 9.5, 11.3 <b>6.2</b> , <b>7</b>
$\frac{\sigma}{\sigma^{**}}$	resultant state	7, 8
$\frac{\boldsymbol{\sigma}}{\boldsymbol{\sigma}_0}$	checkpoint state	<b>6.2</b> , 11.4
$\sigma_1$	first transitional state	8
$\sigma_1$	!!!	8
$\frac{\boldsymbol{\sigma}_1}{\boldsymbol{\sigma}_{\mathrm{i}}}$	!!!	11.4
$\frac{\boldsymbol{\sigma}_1}{\boldsymbol{\sigma}_P}$	tuple of post-execution provisional state	6.2
— P	tuple of post-execution provisional state	0.4

$\sigma[a]$	account state	4.1, 6.2, 11.3(),
- [~]	2220410 20000	11.4()
$\sigma[a]_{ m b}$	balance, number of Wei owned by this address $(\mathbb{N}_{256})$	4.1, 6.2, 7, 8, 11.3
$\sigma[a]_{ m c}$	codeHash, hash of the EVM code of this address $(\mathbb{N}_{32})$	4.1, 7, 8, 11.3()
$\sigma[a]_{\mathrm{n}}$	nonce, (non-contract account) number of trans-	4.1, 6.2, 7, 8(),
	actions sent from this address or (contract ac-	11.3()
	count) number of contract-creations made by	
	this accout $(\mathbb{N}_{256})$	
$\sigma[a]_{\rm s}$	storageRoot, root node hash of a Merkle Patri-	4.1, 7(), 8(),
	cia tree that encodes the storage contents of this	11.3()
	accout $(\mathbb{N}_{32})$	
1	pre-Homestead difficulty parameter	(4.4.4)
2	Homestead difficulty parameter	4.3.4(4.4.4)
)	!!!	8
$\Theta_3$	!!!	(6.2)
$\Theta_4$	!!!	6.2()
ſ	transaction-level state transition function	2.0, 3, 6.0, 11.4
$C^{\mathrm{g}}()$	total gas used in the execution of a transaction	6.0, 6.2, 11.4
$\Gamma^{1}()$	log items created by this transaction	6.0, 6.2, 11.4
^z()	status code resulting from the transaction	6.0(), 6.2(), 11.4()
Ξ(),Ξ()	code execution function	7, 8, 9.3, 9.4.0
0,		
_	Mathematical Symbols	
=	defined to be	•••
	sequence length	•••
	size of the code	7
	and	•••
/	or	•••
/	bitwise or	4.3.1(4.4.1),
`		4.3.2(4.4.2)
)	empty string	4.1, 6.1, 7, 9.4.0,
y	empty set	9.4.4
<u>-</u>	!!!	6.2()
7	not	9.4.2()
		3.1.2()
	Latin Alphabets	
$\overline{A}$	transaction substate	6.1, 6.2, 7, 8,
		9.3(), 9.4.0, 9.5
4'	!!!	9.5
40	empty substate	<b>6.1</b> , 9.4.0
$4_1$	log series, series of archived and indexable 'checkpoints'	6.1
$A_{\rm r}$	refund balance which increases through using	6.1, 6.2
21 <sub>T</sub>	the SSTORE instruction in order to reset con-	
$A_{\mathbf{s}}$	the SSTORE instruction in order to reset contract storage self-destruct (suicide) set: a set of accounts that will be discarded	6.1, 6.2

a	address of the new account	7
	!!!	11.3()
$rac{\mathbf{a}'}{\mathbf{a}^*}$	updated contract creation sender (see also $s$ )	7
$f a_1$	!!!	
	!!!	8()
$\mathbf{a}_1'$	!!!	8()
$\frac{\mathbf{a}_1}{\prime}$		8()
$\mathbf{a}_1'$	<u>   </u>	8()
ADD	!!!	9.4.1
B	this block	2.0,  4.3.0(4.4.0),
		4.3.2(4.4.2),
		4.3.3(4.4.3), 6.2,
		10, 11.3, 11.4
$B_H$	block header (see $H$ )	4.3.0(4.4.0),
		4.3.3(4.4.3),
	1 (77)	4.3.4(4.4.4), 11.1
$B_{Hc}$	beneficiary $(H_c)$	6.2
$B_{\rm d}$	block's difficulty	10
${ m B}_{H{ m c}}$	!!!	11.3
$B_{H_g}$	total gas used in the block $(H_{\rm g})$	11.2
$B_{H\mathrm{i}}$	$!!! (H_i)$	11.3
$B_{H_1}$	block's gasLimit $(H_1)$	6.2
$B_{\mathbf{R}}$	transaction receipts in this block	4.3.1(4.4.1),
		4.3.2(4.4.2), 6.2
$B_{\mathbf{T}}$	a series of the transactions in this block	4.3.0(4.4.0),
		4.3.2(4.4.2),
		4.3.3(4.4.3), 11.4
$B_{ m t}$	total difficulty of block	10
$B'_{\mathrm{t}}$	total difficulty of the parent block	10
$B_{\mathbf{U}}$	list of ommer (uncle) block headers	4.3.0(4.4.0),
	, ,	4.3.2(4.4.2),
		4.3.3(4.4.3),
		4.3.4(4.4.4), 11.1,
		11.3
B'	parent block	10
B'	!!! complete block	11.4
$B'_{\rm n}$	!!!	11.4
$B'_{\rm m}$	!!!	11.4
$B^*$	!!!	11.4
$B_{\aleph}^*$	!!!	11.4
$B_{\mathrm{r}}^*$		11.4
$\mathbb{B}, \mathbb{B}_{number}$	byte sequence, of length number	
$\mathbb{D}, \mathbb{D}_{number}$	by te sequence, or length wanteer	3, 4.1, 4.2(4.3),
		4.3.1(4.4.1),
B.() B .()	hit of index i or hits of indices in the names [a 1]	4.3.3(4.4.3)
$\mathcal{B}_{\mathtt{j}}(),\mathcal{B}_{ab}()$	bit of index $j$ or bits of indices in the range $[a, b]$	<b>4.3.1</b> (4.4.1), 7
<u> </u>	in the byte array	4.1
b Di OGRII AGII	EVM code	4.1
BLOCKHASH	!!!	12.2
$C, C_{\text{OPERATION}}$	cost function	3, 9.4.1
C()	<u>!!!</u>	9.4.0, 9.4.2, 9.5
	!!!	9.4.3
3	code-deposit cost	7

	against whose ends is to be avointed (the same	0
c	account whose code is to be executed (the same	8
CALL	as message call recipient) !!!	0.2.0.2.0.4.2()
	!!!	9.2, 9.3, 9.4.2()
CALLCODE	!!!	9.2, 9.4.2()
CREATE		9.2, 9.3, 9.4.2()
D()	canonical difficulty of a block of the given	<b>4.3.4</b> (4.4.4)
D()	header (see $H_d$ )	0.4.0.0.4.0
D()	function to determine the set of valid jump des-	9.4.2, <b>9.4.3</b>
D ()	tinations given the code that is being run !!!	0.4.9
$D_J()$		9.4.3
$D_0$	difficulty of the genesis block	<b>4.3.4</b> (4.4.4)
Đ	Ether, ETH	2.1
$\mathbf{d}$	current DAG for computing the mix-hash	4.3.4(4.4.4), 11.4,
.1	input data of the call	11.5.0
d PEAD()	input data of the call	8
DEAD()	whether an account is dead, when its account	(4.1, 6.2, 7)
DDIEGIEDGIII	state is non-existent or empty (EMPTY())	
DELEGATECALL	!!!	8
e	present depth of the message-call/contract-	7, 8
EMDTV()	creation stack	(4.1)
EMPTY()	whether an account is empty, when it has no	(4.1)
$\overline{F}$	code, zero nonce and zero balance out-of-gas exception condition	(7)
G		(7)
	specific gas cost	6.2
g	$\cdots$	9.4.1
g	amount of gas remaining after deducting the ba-	<b>6.2</b> , 7, 8, 9.3, 9.4.0
,	sic amount, $T_{\rm g} - g_0$	
<u>g'</u>	remaining gas	6.2, 7, 8, 9.3
$g^*$	amount of gas to be refunded	6.2
$g^{**}$	available gas remaining	7
$g_0$	intrinsic gas, the amount of gas this transaction	6.2
0	requires to be paid prior to execution	
H()	function which specifies the output data of the	9.4.0, 9.4.1, 9.4.4
	instruction if and only if the present state is a	
	normal halting state of the machine (cf. $Z()$ )	
H	block header	4.3.0(4.4.0),
		4.3.3(4.4.3),
		4.3.4(4.4.4),
TT		11.5.0
$H_{ m b}$	logsBloom, Bloom filter for each log entry from	4.3.0(4.4.0),
	the each transaction receipt	4.3.2(4.4.2),
		4.3.3(4.4.3)
$H_{\mathrm{c}}$	beneficiary, address to which all fees collected	4.3.0(4.4.0),
77	be transferred	4.3.3(4.4.3), 11.3
$H_{ m d}$	difficulty, difficulty level of this block	4.3.0(4.4.0),
		4.3.3(4.4.3),
		4.3.4(4.4.4), 11.4,
TT	magaintDoot most models -b -f b twee	11.5.0
$H_{ m e}$	receiptRoot, root node hash of each transaction	4.3.0(4.4.0),
	receipt	4.3.1(4.4.1),
		4.3.2(4.4.2),
		4.3.3(4.4.3)

$H_{ m g}$	gasUsed, total gas used in transactions in this	4.3.0(4.4.0),
	block	4.3.3(4.4.3),
		4.3.4(4.4.4)
$H_{ m i}$	number, number of ancestor blocks, block num-	4.3.0(4.4.0),
	ber	4.3.3(4.4.3),
		4.3.4(4.4.4), (6.2),
		(7), 11.3
$H_{ m i}' \ H_{ m l}$	fake block number to delay the ice age	(4.3.4)
$H_1$	gasLimit, current limit of gas expenditure per	4.3.0(4.4.0),
	block	4.3.3(4.4.3),
		4.3.4(4.4.4)
$H_{ m m}$	mixHash, hash which proves a sufficient amount	4.3.0(4.4.0),
	of computation	4.3.3(4.4.3),
	•	4.3.4(4.4.4),
		11.5.0
$H_{ m n}$	nonce, value which proves a sufficient amount	4.3.0(4.4.0),
	of computation	4.3.3(4.4.3),
	•	4.3.4(4.4.4),
		11.5.0
$H_{\mathbb{M}}$	block's header without the nonce $(H_n)$ and mix-	4.3.4(4.4.4),
	hash $(H_{\rm m})$ components	11.5.0
$H_{\rm o}$	ommersHash, hash of the ommers (uncles) list	4.3.0(4.4.0),
	•	4.3.2(4.4.2),
		4.3.3(4.4.3)
$H_{\rm p}$	parentHash, hash of the parent block's header	4.3.0(4.4.0),
r	1	4.3.3(4.4.3),
		4.3.4(4.4.4)
$H_{\rm r}$	stateRoot, root node hash of the state trie	4.3.0(4.4.0),
- T	source to the state that of the state the	4.3.2(4.4.2),
		4.3.3(4.4.3), 11.4
$H_{\mathrm{s}}$	timstamp, block creation time in seconds	4.3.0(4.4.0),
118	timbuamp, block creation time in seconds	
		4.3.3(4.4.3),
$H_{ m t}$	transactionsRoot, root node hash of each trans-	4.3.4(4.4.4)
ıı <sub>t</sub>		4.3.0(4.4.0),
	action	4.3.2(4.4.2),
U	extraData, arbitrary data relevant to this block	4.3.3(4.4.3)
$H_{\mathrm{x}}$	eamapana, arbinary data relevant to this block	4.3.0(4.4.0),
		4.3.3(4.4.3),
TT	an axial function for DEWILDN 1 DEVEROR	4.3.4(4.4.4)
H <sub>RETURN</sub>	special function for RETURN and REVERT	9.4.4
I	parameters of the execution environment	7, 8, 9.3, 9.4.0,
<i>T</i>	. 11	9.4.2, 9.5
$I_{ m a}$	address of the account which owns the code that	7, 8, 9.3
	is executing (a)	
$I_{\mathbf{b}}$	byte array that is the machine code to be exe-	7, 8, 9.3, 9.4.1,
	cuted (i)	9.4.2
$I_{\mathbf{d}}$	!!! empty tuple as there is no input data to	7, 8, 9.3
	this call (()), byte array that is the input data	
	to this execution; if the execution agent is a	
	transaction, this would be the transaction data	
$I_{ m e}$	depth of the present message-call or contract-	7, 8, 9.3
	creation $(e)$	
$I_H$	block header of the present block	7, 9.3
	*	<u> </u>

$I_{\mathrm{o}}$	sender address of the transaction that origi-	7, 8, 9.3
_	nated this execution (o)	
$I_{ m p}$	price of gas in the transaction that originated	7, 8, 9.3
	this execution $(p)$	
$I_{ m s}$	address of the account which caused the code to	7, 8, 9.3
	be executing; if the execution agent is a trans-	
	action, this would be transaction sender $(s)$	
$I_{ m v}$	value in Wei passed to this account as part of	7, 8, 9.3
	the same procedure as execution $(v)$	
$I_{ m w}$	permission to make modifications to the state	(7, 8, 9.3, 9.4.2)
	(w)	
I'	dropped fourth element from $X()$	9.4.0()
i	initialisation EVM code for creating a contract	7
i	!!! $(oldsymbol{\mu}_{ m i})$	9.4.1
$J, J_{\text{JUMP}}, J_{\text{JUMPI}}$	!!!	9.5
$J_{cacheinit}, J_{datasetin}$	$_{iit}, \!\!\!\downarrow\!\!\!\!\downarrow_{epoch}$	11.5.1
JUMP	!!!	9.4.2
JUMPDEST	!!!	9.4.3
JUMPI	!!!	9.4.2
k()	check "is-kin" property	11.1
KEC()	Keccak-256 hash function, plain Keccak	3, 4.1,
MDO()	recear 200 hash rancoon, plant recear	4.3.1(4.4.1),
		, , , ,
		4.3.2(4.4.2),
KEC512()	Keccak 512 hash function	4.3.4(4.4.4), 7, 8
$L_B()$	preparation function for a block	4.3.3(4.4.3)
**		
$L_H()$	preparation function for a block header	4.3.3(4.4.3)
$L_H^*()$	!!!	4.3.2(4.4.2),
T ()	collance function for a least/solue points the trie	4.3.3(4.4.3)
$L_I()$	collapse function for a key/value pair in the trie,	4.1
T*/\	$L_I((k,v)) \equiv (\text{KEC}(k), \text{RLP}(v))$	
$L_I^*()$	element-wise transformation of $L_I$ (see $\sigma[a]_s$ )	4.1
$L_R()$	function that converts a transaction receipt into	4.3.1(4.4.1),
T ()	an RLP-serialised byte array	4.3.2(4.4.2)
$L_S()$	world-state collapse function	4.1, (4.4.1),
<i>T</i> ()	111	4.3.2(4.4.2), 11.4
$L_T()$	!!!	4.2(4.3),
T * /\	111	4.3.2(4.4.2)
$L_T^*()$	!!!	4.3.3(4.4.3)
1	$\log \text{ series } (A_1)$	9.3
$\ell()$	last item in the sequence	<b>3</b> , 11.2, 11.4
LOG0	!!!	9.4.2()
LOG4	!!!	9.4.2()
M()	Bloom filter function to reduce a log entry into	4.3.1(4.4.1)
· ·	a single 256-byte hash	
$M_{3:2048}()$	specialised Bloom filter that sets three bits out	<b>4.3.1</b> (4.4.1)
<del>V</del>	of 2048	
m	$(\mu_{ m m})$	9.4.1
$\overline{m}$	beneficiary (miner, $H_c$ )	6.2
m()	fetch the given index after hashing	<b>4.3.1</b> (4.4.1)
N()	next valid instruction position in the code	9.4.3, 9.5
V		,

$\begin{array}{c} O() & \text{iterator function which defines the result of a} \\ & \text{single cycle of the state machine} \\ O, O_{number} & \text{log entry created through execution of the} \\ & \text{transaction (see $R_1$)} \\ O_{a} & \text{logger's address } (\mathbb{B}_{20}) & 4.3.1(4.4.1) \\ O_{d} & \text{number of bytes of data } (\mathbb{B}) & 4.3.1(4.4.1) \\ O_{d} & \text{series of log topics} & 4.3.1(4.4.1) \\ O_{d} & \text{number of bytes of data } (\mathbb{B}) & 4.3.1(4.4.1) \\ O_{d} & \text{number of bytes of data } (\mathbb{B}) & 4.3.1(4.4.1) \\ O_{d} & \text{output data of a message call or created contract's code (the resultant byte sequence from the execution of the initialisation code) \\ O & \text{(contract creation or transaction) originator } & 7, 8 \\ \text{(see also $T_0$)} \\ \mathbb{P}, \mathbb{P}_{number} & \text{positive integer, smaller than $2^{number}$} & (3, 4.1, 4.3, 4.4.1, 4.4.3) \\ O & \text{(contract creation or message call) gas price} \\ P(B_H), P(H) & \text{parent block of $B$ (see $H_{\rm P}$)} & 4.3.2(4.4.2) \\ O & \text{(contract creation or message call) gas price} \\ O() & \text{per-account collapse function (see $L_{S}$())} & 4.1 \\ O() & \text{pairwise RLP transformation} & 4.3.2(4.4.2) \\ O() & \text{proof-of-work function} & 4.3.4(4.4.4) \\ O() & \text{proof-of-work function} & 4.3.4(4.4.4) \\ O() & \text{proof-of-work function} & 4.3.4(4.4.4) \\ O() & \text{proof-of-work function} & 4.3.1(4.4.1) \\ O() & \text{proof-of-work function} & 4.3.1(4.4.1)$	$\mathbb{N}, \mathbb{N}_{number}$	positive integer, smaller than $2^{number}$	3(), 4.1() <b>4.2</b> (),
stead phases $(1,150,000)$ 4.3.4(4.4), (6.2), (7)  Transaction nonce (see $T_n$ ) 3  D() iterator function which defines the result of a single cycle of the state machine  D, Onumber log entry created through execution of the transaction (see $R_1$ )  Da logger's address ( $\mathbb{B}_{20}$ ) 4.3.1(4.4.1)  Da logger's address ( $\mathbb{B}_{20}$ ) 4.3.1(4.4.1)  Da series of log topics 4.3.1(4.4.1)  Dtumber log topic ( $\mathbb{B}_{32}$ ) 4.3.1(4.4.1)  D toutput data of a message call or created contract's code (the resultant byte sequence from the execution of the initialisation code)  C(contract creation or transaction) originator (see also $T_0$ )  P, P, number positive integer, smaller than $2^{number}$ (3, 4.1, 4.3, 4.4.1, 4.4.3)  D (contract creation or message call) gas price 7, 8  P(BH), P(H) parent block of $B$ (see $H_p$ ) 4.3.2(4.4.2), 4.3.4(4.4.4), 10, 10, 11, 11.4  D (contract creation or message call) gas price 7, 8  D (contract creation or message call) gas price 7, 8  D (contract creation or message call) gas price 7, 8  D (contract creation or message call) gas price 7, 8  D (contract creation or message call) gas price 7, 8  D (contract creation or message call) gas price 7, 8  D (contract creation or message call) gas price 7, 8  D (contract creation or message call) gas price 7, 8  D (contract creation or message call) gas price 7, 8  D (contract creation or message call) gas price 7, 8  D (contract creation or message call) gas price 7, 8  D (contract creation or message call) gas price 7, 8  D (contract creation or message call) gas price 7, 8  D (contract creation or message call) gas price 7, 8  D (contract creation or message call) gas price 7, 8  D (contract creation or message call) gas price 7, 8  D (contract creation or message call) gas price 7, 8  D (contract creation or message call or created contraction 7, 8  D (contract creation 0 message call or created 1, 11, 11, 11, 11, 11, 11, 11, 11, 11,	Nir	block number between the Frontier and Home-	
transaction nonce (see $T_n$ ) 3 $O(1)$ iterator function which defines the result of a single cycle of the state machine $O(1)$ output created through execution of the transaction (see $R_1$ ) $O(1)$ anumber of bytes of data ( $\mathbb{B}$ ) 4.3.1(4.4.1) $O(1)$ series of log topics 4.3.1(4.4.1) $O(1)$ series of log topics 4.3.1(4.4.1) $O(1)$ transaction (see $O(1)$ output data of a message call or created contract scode (the resultant byte sequence from the execution of the initialisation code) $O(1)$ (contract creation or transaction) originator (see also $O(1)$ ) $O(1)$ parent block of $O(1)$ (see $O(1)$ ) 4.3.2(4.4.2) $O(1)$ parent block of $O(1)$ (contract creation or message call) gas price (contract creation (contract creation (contract (contract creation (contract (contract creation (contract (contract creation (contract (cont	1 <b>H</b>		
transaction nonce (see $T_n$ )  iterator function which defines the result of a single cycle of the state machine  O, $O_{number}$   log entry created through execution of the transaction (see $R_1$ )  Oa   logger's address ( $\mathbb{B}_{20}$ )   4.3.1(4.4.1)  Oa   logger's address ( $\mathbb{B}_{20}$ )   4.3.1(4.4.1)  Ob   series of log topics   4.3.1(4.4.1)  Ot   series of log topics   4.3.1(4.4.1)  Ot   series of log topics   4.3.1(4.4.1)  Ot   output data of a message call or created contract's code (the resultant byte sequence from the execution of the initialisation code)  O (contract creation or transaction) originator (see also $T_o$ )  P, $\mathbb{P}_{number}$   positive integer, smaller than $2^{number}$   (3, 4.1, 4.3, 4.4.1, 4.3)  Occordinate (contract creation or message call) gas price   7, 8  Occordinate (contract creation or message call) gas price   7, 8  Occordinate (contract creation or message call) gas price   7, 8  Occordinate (contract creation or message call) gas price   7, 8  Occordinate (contract creation or message call) gas price   7, 8  Occordinate (contract creation or message call) gas price   7, 8  Occordinate (contract creation or message call) gas price   7, 8  Occordinate (contract creation or message call) gas price   7, 8  Occordinate (contract creation or message call) gas price   7, 8  Occordinate (contract creation or message call) gas price   7, 8  Occordinate (contract creation or message call) gas price   7, 8  Occordinate (contract creation or message call) gas price   7, 8  Occordinate (contract creation or message call) gas price   7, 8  Occordinate (contract creation or message call) gas price   7, 8  Occordinate (contract creation or message call) gas price   7, 8  Occordinate (contract creation or message call) gas price   7, 8  Occordinate (contract creation or message call) gas price   7, 8  Occordinate (contract creation or message call) gas price   7, 8  Occordinate (contract creation or message call) gas price   7, 8  Occordinate (contract creation or message (call) gas price   7		blead phases (1, 150, 000)	, , , ,
$\begin{array}{c} O() & \text{iterator function which defines the result of a} \\ & \text{single cycle of the state machine} \\ O, O_{number} & \text{log entry created through execution of the} \\ & \text{transaction (see $R_1$)} \\ O_A & \text{logger's address } (\mathbb{B}_{20}) & 4.3.1(4.4.1) \\ O_A & \text{number of bytes of data } (\mathbb{B}) & 4.3.1(4.4.1) \\ O_A & \text{series of log topics} & 4.3.1(4.4.1) \\ O_A & \text{series of log topics} & 4.3.1(4.4.1) \\ O_{1 & \text{number}} & \text{log topic } (\mathbb{B}_{32}) & 4.3.1(4.4.1) \\ O_{2 & \text{number}} & \text{log topic } (\mathbb{B}_{32}) & 4.3.1(4.4.1) \\ O_{3 & \text{output data of a message call or created contract's code (the resultant byte sequence from the execution of the initialisation code)} \\ O & \text{(contract creation or transaction) originator } & 7, 8 \\ \text{(see also $T_0$)} \\ P, \mathbb{P}_{number} & \text{positive integer, smaller than } 2^{number} & (3, 4.1, 4.3, 4.4.1, 4.3) \\ O & \text{(contract creation or message call) gas price} & (3.4.4.4.4) \\ O & \text{(contract creation or message call) gas price} & (4.3.2(4.4.2), 4.3.4(4.4.4), 10, 11.1, 11.4 \\ O & \text{(contract creation or message call) gas price} & (4.3.4(4.4.4), 10, 11.1, 11.4 \\ O & \text{(contract creation or message call) gas price} & (4.3.4(4.4.2), 10, 11.1, 11.4 \\ O & \text{(contract creation or message call) gas price} & (4.3.4(4.4.2), 10, 11.1, 11.4 \\ O & \text{(contract creation or message call) gas price} & (4.3.4(4.4.2), 10, 11.1, 11.4 \\ O & \text{(contract creation or message call)} & \text{(a.3.4}(4.4.2), 10, 11.1, 11.4 \\ O & \text{(contract creation or message call)} & \text{(a.3.4}(4.4.2), 10, 11.1, 11.4 \\ O & \text{(contract creation or message call)} & \text{(a.3.4}(4.4.2), 10, 11.1, 11.4 \\ O & \text{(contract creation or message call)} & \text{(a.3.4}(4.4.2), 10, 11.1, 11.4 \\ O & \text{(contract creation or message call)} & \text{(a.3.4}(4.4.2), 10, 11.1, 11.4 \\ O & \text{(contract creation or message call)} & \text{(a.3.4}(4.4.2), 10, 11.1, 11.4 \\ O & \text{(contract creation or message call)} & \text{(a.3.4}(4.4.2), 10, 11.1, 11.4 \\ O & \text{(contract creation or message call)} & \text{(a.3.4}(4.4.2), 10, 11.1, 11.4 \\ O & \text{(a.3.4}(4.4.2), 10, 1$	$\overline{n}$	transaction nonce (see $T_n$ )	
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$			
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$			3.4.0, 3.4.1, 3.0
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	$O, O_{number}$		4.3.1(4.4.1)
$\begin{array}{c} \mathcal{O}_{\mathbf{d}} & \text{number of bytes of data } (\mathbb{B}) & 4.3.1(4.4.1) \\ \mathcal{O}_{\mathbf{t}} & \text{series of log topics} & 4.3.1(4.4.1) \\ \mathcal{O}_{\mathbf{t}number} & \text{log topic } (\mathbb{B}_{32}) & 4.3.1(4.4.1) \\ \mathcal{O}_{\mathbf{t}number} & \text{log topic } (\mathbb{B}_{32}) & 4.3.1(4.4.1) \\ \mathcal{O}_{\mathbf{t}number} & \text{output data of a message call or created contract's code (the resultant byte sequence from the execution of the initialisation code) \\ \mathcal{O}_{\mathbf{c}} & (\text{contract creation or transaction)} & 7,8 \\ (\text{see also } T_o) & (\text{contract creation or transaction)} & 7,8 \\ (\text{see also } T_o) & (3,4.1,4.3,4.4.1,4.3) \\ \mathcal{O}_{\mathbf{c}} & (\text{contract creation or message call)} & (3,4.1,4.3,4.4.1,1.1.1) \\ \mathcal{O}_{\mathbf{c}} & (\text{contract creation or message call)} & (3,4.1,4.3,4.4.1,1.1.1) \\ \mathcal{O}_{\mathbf{c}} & (\text{contract creation or message call)} & (3,4.1,4.3,4.4.1,1.1.1) \\ \mathcal{O}_{\mathbf{c}} & (\text{contract creation or message call)} & (3,4.1,4.3,4.4.1,1.1.1) \\ \mathcal{O}_{\mathbf{c}} & (\text{contract creation or message call)} & (3,4.1,4.3,4.4.1,1.1.1) \\ \mathcal{O}_{\mathbf{c}} & (\text{contract creation or message call)} & (3,4.1,4.3,4.4.1,1.1.1) \\ \mathcal{O}_{\mathbf{c}} & (\text{contract creation or message call)} & (3,4.1,4.3,4.4.1,1.1.1) \\ \mathcal{O}_{\mathbf{c}} & (\text{contract creation or message call)} & (3,4.1,4.3,4.4.1,1.1.1) \\ \mathcal{O}_{\mathbf{c}} & (\text{contract creation or message call)} & (3,4.1,4.3,4.4.1,1.1.1) \\ \mathcal{O}_{\mathbf{c}} & (\text{contract creation or message call)} & (3,4.1,4.1,3.1,1.1.1) \\ \mathcal{O}_{\mathbf{c}} & (\text{contract creation or message call)} & (3,4.1,4.1,4.1,1.1.1) \\ \mathcal{O}_{\mathbf{c}} & (\text{contract creation or message call)} & (3,4.1,4.1,4.1,1.1.1) \\ \mathcal{O}_{\mathbf{c}} & (\text{contract creation or message call)} & (3,4.1,4.1,4.1,1.1.1) \\ \mathcal{O}_{\mathbf{c}} & (\text{contract creation or message call)} & (3,4.1,4.1,4.1,4.1,4.1,4.1,4.1,4.1,4.1,4.1$	$O_{\rm a}$		4.3.1(4.4.1)
$\begin{array}{c} \mathcal{O}_{\mathbf{t}} & \text{series of log topics} \\ \mathcal{O}_{\mathbf{t}number} & \text{log topic}  (\mathbb{B}_{32}) \\ \text{output data of a message call or created contract's code (the resultant byte sequence from the execution of the initialisation code)} \\ \mathcal{O} & \text{(contract creation or transaction) originator the execution of the initialisation code)} \\ \mathcal{O} & \text{(see also } T_{\circ}) \\ \mathbb{P}, \mathbb{P}_{number} & \text{positive integer, smaller than } 2^{number} \\ \text{(3, 4.1, 4.3, 4.4.1, 4.3)} \\ \text{(4.4.3)} \\ \text{(4.4.3)} \\ \text{(4.3.2(4.4.2), 4.3.4(4.4.4), 10, 11.1, 11.4} \\ \text{(contract creation or message call) gas price} \\ \mathcal{O} & \text{(contract creation or message call) gas price} \\ \mathcal{O} & \text{(per-account collapse function (see } L_S())} \\ \mathcal{O} & \text{(per-account collapse function (see } L_S())} \\ \mathcal{O} & \text{(per-account collapse function (see } L_S())} \\ \mathcal{O} & \text{(per-account collapse function (see } L_S())} \\ \mathcal{O} & \text{(per-account collapse function (see } L_S())} \\ \mathcal{O} & \text{(per-account collapse function (see } L_S())} \\ \mathcal{O} & \text{(per-account collapse function (see } L_S())} \\ \mathcal{O} & \text{(per-account collapse function (see } L_S())} \\ \mathcal{O} & \text{(per-account collapse function (see } L_S())} \\ \mathcal{O} & \text{(per-account collapse function (see } L_S())} \\ \mathcal{O} & \text{(per-account collapse function (see } L_S())} \\ \mathcal{O} & \text{(per-account collapse function (see } L_S())} \\ \mathcal{O} & \text{(per-account collapse function (see } L_S())} \\ \mathcal{O} & \text{(per-account collapse function (see } L_S())} \\ \mathcal{O} & \text{(per-account collapse function (see } L_S())} \\ \mathcal{O} & \text{(per-account collapse function (see } L_S())} \\ \mathcal{O} & \text{(per-account collapse function (see } L_S())} \\ \mathcal{O} & \text{(per-account collapse function (see } L_S())} \\ \mathcal{O} & \text{(per-account collapse function (see } L_S())} \\ \mathcal{O} & \text{(per-account collapse function (see } L_S())} \\ \mathcal{O} & \text{(per-account collapse function (see } L_S())} \\ \mathcal{O} & \text{(per-account collapse function (see } L_S()) \\ \mathcal{O} & \text{(per-account collapse function (see } L_S()) \\ \mathcal{O} & \text{(per-account collapse function (see } L_S()) \\ \mathcal{O} & (per-account collapse f$	$O_{\mathbf{d}}$		
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	$O_{\mathbf{t}}$		
output data of a message call or created contract's code (the resultant byte sequence from the execution of the initialisation code)  (contract creation or transaction) originator (see also $T_o$ )  positive integer, smaller than $2^{number}$ (3, 4.1, 4.3, 4.4.1, 4.3)  P(BH), P(H)  parent block of B (see $H_p$ )  (3, 4.4, 4.3, 4.4.1, 10, 11.1, 11.4)  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call gas price 7, 8  (contract creation or message call gas price 7, 8  (d.4.4.1), 11.4  (d.4.3, 44.4.1), 11.4  (d.4.4.4), 10, 11.4  (d.4.3, 44.4.1), 11.4  (d.4.3, 44.4.1), 11.4  (d.4.3, 44.4.1), 11.4  (d.4.4.4), 10, 11.4  (d.4.4.4)			
tract's code (the resultant byte sequence from the execution of the initialisation code)  (contract creation or transaction) originator 7, 8  (see also $T_o$ )  P. P. $number$ positive integer, smaller than $2^{number}$ (3, 4.1, 4.3, 4.4.1, 4.4.3)  P(BH), P(H) parent block of $B$ (see $H_p$ )  (contract creation or message call) gas price 7, 8  P(C) per-account collapse function (see $L_S()$ )  (contract transaction or message call) gas price 7, 8  P(C) pairwise RLP transformation 4.3.2(4.4.2)  P(C) pairwise RLP transformation 4.3.4(4.4.4), 11.5.0  P(C) pairwise RLP transformation 4.3.4(4.4.4), 11.5.0  P(C) proof-of-work function 4.3.4(4.4.4), 11.5.0  P(C) P(C) P(C) P(C) P(C) P(C) P(C) P(C)	<b>O</b>		
the execution of the initialisation code)  (contract creation or transaction) originator (see also $T_{\rm o}$ )  (see also $T_{\rm o}$ )  positive integer, smaller than $2^{number}$ (3, 4.1, 4.3, 4.4.1, 4.4.3)  (BH), $P(H)$ parent block of $B$ (see $H_{\rm p}$ )  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call gas price 7, 8  (contract creation or message call gas price 7, 8  (contract creation or message call gas price 7, 8  (contract creation or message call gas price 7, 8  (contract creation or message call gas price 7, 8  (contract creation or message call gas price 7, 8  (contract creation or message call gas price 7, 8  (contract creation or message call gas price 7, 8  (contract creation or message call gas price 7, 8  (contract creation or message call gas price 7, 8  (contract creation or message call gas price 7, 8  (contract creation or message call gas price 7, 8  (contract creation or message call gas price 7, 8  (contract creation or message call gas price 7, 8  (contract creation or message call gas price 6, 9  (contract creation or message call recipient 9, 9, 4, 4  (contract creation or message call gas price 6, 9  (contract creation or message call gas price 7, 8  (contract creation or message call gas price 6, 9  (contract creation or message call gas price 6, 9  (contract creation or message call gas price 6, 9  (contract creation or message call			3, 1(), 3, 0.0
$\begin{array}{c} (\text{contract creation or transaction}) \text{ originator } \\ (\text{see also } T_o) \\ (\text{positive integer, smaller than } 2^{number}) \\ (3, 4.1, 4.3, 4.4.1, 4.4.3) \\ (4.4.3) \\ (2, 4.3.4(4.4.4), 10, 10. 11.1, 11.4) \\ (2, 4.3.4(4.4.4), 10, 11.1, 11.4) \\ (3, 4.3.4(4.4.4), 10, 11.1, 11.4) \\ (4, 3.4(4.4.4), 10, 11.1, 11.4) \\ (5, 6) (\text{per-account collapse function (see } L_S())) \\ (5, 6) (\text{per-account collapse function (see } L_S())) \\ (5, 6) (\text{per-account collapse function (see } L_S())) \\ (5, 6) (\text{per-account collapse function (see } L_S())) \\ (5, 6) (\text{per-account collapse function (see } L_S())) \\ (5, 6) (\text{per-account collapse function (see } L_S())) \\ (5, 6) (\text{per-account collapse function (see } L_S())) \\ (5, 7) (\text{per-account collapse function (see } L_S())) \\ (5, 7) (\text{per-account collapse function (see } L_S())) \\ (5, 7) (\text{per-account collapse function (see } L_S())) \\ (5, 7) (\text{per-account collapse function (see } L_S())) \\ (5, 7) (\text{per-account collapse function (see } L_S())) \\ (5, 7) (\text{per-account collapse function (see } L_S())) \\ (5, 7) (\text{per-account collapse function (see } L_S())) \\ (5, 7) (\text{per-account collapse function (see } L_S())) \\ (1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1$		*	
$ \begin{array}{c} (\text{see also $T_o$}) \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\$	0	,	7. 8
$ \begin{array}{c} \mathbb{P}_{r} \mathbb{P}_{number} & \text{positive integer, smaller than } 2^{number} & (3, 4.1, 4.3, 4.4.1, 4.4.3) \\ & 4.4.3) \\ & 4.3.2(4.4.2), \\ & 4.3.4(4.4.4), & 10, \\ & 11.1, & 11.4 \\ & 2.0 & (\text{contract creation or message call}) \text{ gas price} \\ & 7, & 8 \\ & 90 & \text{per-account collapse function } (\text{see } L_{S}()) & 4.1 \\ & 90 & \text{poirwise RLP transformation} & 4.3.2(4.4.2) \\ & 90 & \text{prof-of-work function} & 4.3.2(4.4.2) \\ & 11.5.0 \\ & 1$	-	, –	., •
$P(B_H), P(H) \qquad \text{parent block of } B \text{ (see } H_{\text{P}}) \qquad \begin{array}{c} 4.4.3 \\ 4.3.2 (4.4.2), \\ 4.3.4 (4.4.4), & 10, \\ 11.1, & 11.4 \end{array}$ $P(B_H), P(H) \qquad \text{parent block of } B \text{ (see } H_{\text{P}}) \qquad \begin{array}{c} 4.3.2 (4.4.2), \\ 4.3.4 (4.4.4), & 10, \\ 11.1, & 11.4 \end{array}$ $P(B_H), P(H) \qquad \text{parent block of } B \text{ (see } H_{\text{P}}) \qquad \begin{array}{c} 4.3.4 (4.4.4), & 10, \\ 11.1, & 11.4 \end{array}$ $P(B_H), P(H) \qquad \text{proof core count collapse function (see } L_S()) \qquad 4.1 \end{array}$ $P(B_H) \qquad \text{proof cof-work function} \qquad 4.3.2 (4.4.2) \qquad 4.3.2 (4.4.2) \qquad 4.3.4 (4.4.4), \\ P(B_H) \qquad P(B_H)$	P. Pnumber	positive integer, smaller than $2^{number}$	(3. 4.1. <b>4.3</b> . 4.4.1.
$\begin{array}{c} P(B_H), P(H) & \text{parent block of } B \text{ (see $H_{\rm p}$)} \\ & 4.3.2(4.4.2), \\ & 4.3.4(4.4.4), & 10, \\ & 11.1, & 11.4 \\ Poly & (\text{contract creation or message call) gas price} & 7, 8 \\ Poly & per-account collapse function (see $L_S()$) & 4.1 \\ Poly & pairwise RLP transformation & 4.3.2(4.4.2) \\ Poly & proof-of-work function & 4.3.4(4.4.4), \\ Poly & Proof-of-work function & 4.3.4(4.4.4), \\ Poly & Poly$	= , = namoer	positive integer, emailer email 2	
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	$P(B_H), P(H)$	parent block of $B$ (see $H_p$ )	· · · · · · · · · · · · · · · · · · ·
(contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation or message call) gas price 7, 8  (contract creation (see $L_S()$ ) 4.1  (contract creation (see $L_S()$ ) 4.2  (contract creation (see $L_S()$ ) 4.3  (contract creation (see	- (-11),- ()	P == === (=============================	, , , , ,
$\begin{array}{c} P & \text{(contract creation or message call) gas price} & 7, 8 \\ P & P & P & P & P \\ P & P & P & P & P$			
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	$\overline{p}$	(contract creation or message call) gas price	
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	p()		
Formula       !!! $(\mu_{pc})$ 9.4.1         POW       proof-of-work function       4.3.4(4.4.4),         PUSH1       !!!       9.4.3         PUSH32       !!!       9.4.3         R       !!!       11.2         R       !!!       11.3()         Ro       post-transaction receipt       4.3.1(4.4.1)         Ro       !!!       (11.3)         R[n] $\sigma$ !!!       (5[n])       (11.4)         Rb       Bloom filter composed from information in those logs ( $\mathbb{B}_{256}$ )       4.3.1(4.4.1), 11.4         Rb       !!!       (11.3)         Rblock       block's beneficiary account       11.3()         Rc       !!!       (11.3)         Rblock       block's beneficiary account       11.3()         Rc       cution of the transaction       11.3()         Rc       4.3.1(4.4.1), 11.4         Rc       cution of the transaction       4.3.1(4.4.1), 11.2, 11.2, 11.2         Rc       status code of the transaction ( $\mathbb{N}$ )       4.3.1(4.4.1), 11.4()         Rc       status code of the transaction ( $\mathbb{N}$ )       4.3.1(), 11.4()         Rc       refunds ( $A_r$ )       9.3         C()       !!!       11.4	()		
Pow       proof-of-work function $4.3.4(4.4.4)$ , $11.5.0$ PUSH1       !!! $9.4.3$ PUSH32       !!! $9.4.3$ R       !!! $11.2$ R       transaction receipt $4.3.1(4.4.1)$ R       !!! $11.3()$ Ro       post-transaction state $(4.4.1, 11.4)$ R[n] $\sigma$ !!! ( $\sigma$ [n]) $(11.4)$ Rb       Bloom filter composed from information in those logs ( $\mathbb{B}_{256}$ ) $4.3.1(4.4.1), 11.4$ Rb       !!! $(11.3)$ Rblock       block's beneficiary account $11.3()$ Rl       a series of log entries ( $O$ ) created through execution of the transaction $4.3.1(4.4.1), 11.4$ Ru       cumulative gas used immediately after the transaction has happended ( $\mathbb{N}$ ) $11.4$ Rz       status code of the transaction ( $\mathbb{N}$ ) $4.3.1(4.4.1), 11.4()$ Rz       status code of the transaction ( $\mathbb{N}$ ) $4.3.1(1.4.1), 11.4()$ Rb       Bloom filter of a receipt (see $O$ ) $4.3.2(4.4.2)$ RETURN       !!! $9.4.4$			
PUSH1 !!! 9.4.3  PUSH32 !!! 9.4.3  R. !!! 11.2  R. transaction receipt 4.3.1(4.4.1)  R. !!! 11.3()  R. post-transaction state (4.4.1, 11.4)  R[n] $_{\sigma}$ post-transaction state (4.4.1, 11.4)  R[n] $_{\sigma}$ !!! ( $\sigma$ [n]) (11.4)  R[b] Bloom filter composed from information in 4.3.1(4.4.1), 11.4  those logs ( $\mathbb{B}_{256}$ )  R[b] !!! (11.3)  R[b] block's beneficiary account 11.3()  R[a] a series of log entries (O) created through execution of the transaction  R[a] cumulative gas used immediately after the 4.3.1(4.4.1), 11.4  transaction has happended ( $\mathbb{N}$ ) 11.4  R[a] status code of the transaction ( $\mathbb{N}$ ) 4.3.1(), 11.4()  The message call recipient 8  The refunds ( $A_r$ ) 9.3			
PUSH1       !!!       9.4.3         PUSH32       !!!       9.4.3         R       !!!       11.2         R       transaction receipt       4.3.1(4.4.1)         R       !!!       11.3() $R_{\sigma}$ post-transaction state       (4.4.1, 11.4) $R_{[n]_{\sigma}}$ !!!       (5.1.4) $R_{[n]_{\sigma}}$ !!!       (11.4) $R_{b}$ Bloom filter composed from information in those logs ( $\mathbb{B}_{256}$ )       4.3.1(4.4.1), 11.4 $R_{b}$ !!!       (11.3) $R_{b}$ block's beneficiary account       11.3() $R_{b}$ cution of the transaction       4.3.1(4.4.1), 11.4 $R_{b}$ cumulative gas used immediately after the transaction in the transaction ( $\mathbb{N}$ )       4.3.1(4.4.1), 11.4() $R_{b}$ </td <td>l OW</td> <td>proof of work function</td> <td>* * * * * * * * * * * * * * * * * * * *</td>	l OW	proof of work function	* * * * * * * * * * * * * * * * * * * *
R       !!!       11.2         R       transaction receipt $4.3.1(4.4.1)$ R       !!! $11.3()$ Ro       post-transaction state $(4.4.1, 11.4)$ R[n] $\sigma$ !!! $(5n]$ Rb       Bloom filter composed from information in those logs ( $\mathbb{B}_{256}$ ) $4.3.1(4.4.1)$ , $11.4$ Rb       !!! $(11.3)$ Rc $(11.3)$ $(11.3)$ Rb $(11.3)$ $(11.3)$ Rc $(11.3)$ $(11.4)$ Rc <th< td=""><td>PUSH1</td><td>!!!</td><td></td></th<>	PUSH1	!!!	
R       !!!       11.2         R       transaction receipt $4.3.1(4.4.1)$ R       !!!       11.3()         Ro       post-transaction state $(4.4.1, 11.4)$ R[n] $\sigma$ !!! $(\sigma[n])$ $(11.4)$ Rb       Bloom filter composed from information in those logs ( $\mathbb{B}_{256}$ ) $4.3.1(4.4.1), 11.4$ Rb       !!! $(11.3)$ Rblock       block's beneficiary account $11.3()$ R1       a series of log entries ( $O$ ) created through execution of the transaction $4.3.1(4.4.1), 11.4$ Ru       cumulative gas used immediately after the transaction has happended ( $\mathbb{N}$ ) $11.4$ Rz       status code of the transaction ( $\mathbb{N}$ ) $4.3.1(4.4.1), 11.4()$ Rz       status code of the transaction ( $\mathbb{N}$ ) $4.3.1(1.4.1), 11.4()$ Rz       status code of the transaction ( $\mathbb{N}$ ) $4.3.1(1.4.1), 11.4()$ Rz       status code of the transaction ( $\mathbb{N}$ ) $4.3.1(1.4.1), 11.4()$ Rb       Bloom filter of a receipt (see $O$ ) $4.3.2(4.4.2)$ RETURN       !!! $9.4.4$	PUSH32	!!!	9.4.3
$\begin{array}{cccccccccccccccccccccccccccccccccccc$		!!!	
$R$ !!! $11.3()$ $R_{\sigma}$ post-transaction state $(4.4.1, 11.4)$ $R[n]_{\sigma}$ !!! $(\sigma[n])$ $(11.4)$ $R_b$ Bloom filter composed from information in those logs $(\mathbb{B}_{256})$ $4.3.1(4.4.1), 11.4$ $R_b$ !!! $(11.3)$ $R_{block}$ block's beneficiary account $11.3()$ $R_1$ a series of log entries $(O)$ created through execution of the transaction $4.3.1(4.4.1), 11.4$ $R_u$ cumulative gas used immediately after the transaction has happended $(\mathbb{N})$ $11.4$ $R_z$ status code of the transaction $(\mathbb{N})$ $4.3.1(), 11.4()$ $R_z$ status code of the transaction $(\mathbb{N})$ $4.3.1(), 11.4()$ $R_z$ status code of the transaction $(\mathbb{N})$ $4.3.1(), 11.4()$ $R_z$ status code of the transaction $(\mathbb{N})$ $4.3.1(), 11.4()$ $R_z$	$\overline{R}$	transaction receipt	
$R_{\sigma}$ post-transaction state (4.4.1, 11.4) $R[n]_{\sigma}$ !!! $(\sigma[n])$ (11.4) $R_{\rm b}$ Bloom filter composed from information in 4.3.1(4.4.1), 11.4 those logs ( $\mathbb{B}_{256}$ ) $R_{\rm b}$ !!! (11.3) $R_{\rm block}$ block's beneficiary account 11.3() $R_{\rm l}$ a series of log entries (O) created through execution of the transaction $R_{\rm l}$ cumulative gas used immediately after the 4.3.1(4.4.1), 11.2, transaction has happended ( $\mathbb{N}$ ) 11.4 $R_{\rm l}$ status code of the transaction ( $\mathbb{N}$ ) 4.3.1(), 11.4() message call recipient 8 $R_{\rm l}$ refunds ( $A_{\rm r}$ ) 9.3 $R_{\rm l}$ Bloom filter of a receipt (see O) 4.3.2(4.4.2) RETURN !!!		= = = = = = = = = = = = = = = = = = = =	
$\begin{array}{llllllllllllllllllllllllllllllllllll$		***	
Bloom filter composed from information in $4.3.1(4.4.1)$ , $11.4$ those logs ( $\mathbb{B}_{256}$ ) $R_b$ !!! (11.3) $R_{block}$ block's beneficiary account 11.3() $R_1$ a series of log entries ( $O$ ) created through execution of the transaction cumulative gas used immediately after the 4.3.1(4.4.1), 11.2, transaction has happended ( $\mathbb{N}$ ) 11.4 $R_2$ status code of the transaction ( $\mathbb{N}$ ) 4.3.1(), 11.4() message call recipient 8 $R_1$ refunds ( $A_r$ ) 9.3 $R_2$ Bloom filter of a receipt (see $O$ ) 4.3.2(4.4.2)  RETURN !!! 9.4.4			
those logs ( $\mathbb{B}_{256}$ ) $R_b$ !!!! (11.3) $R_{\mathrm{block}}$ block's beneficiary account 11.3() $R_{\mathrm{I}}$ a series of log entries ( $O$ ) created through execution of the transaction $R_{\mathrm{u}}$ cumulative gas used immediately after the 4.3.1(4.4.1), 11.2, transaction has happended ( $\mathbb{N}$ ) 11.4 $R_{\mathrm{z}}$ status code of the transaction ( $\mathbb{N}$ ) 4.3.1(), 11.4() message call recipient 8 $R_{\mathrm{u}}$ refunds ( $A_{\mathrm{r}}$ ) 9.3 $R_{\mathrm{u}}$ 11.4 $R_{\mathrm{d}}$ Bloom filter of a receipt (see $O$ ) 4.3.2(4.4.2) $R_{\mathrm{d}}$ 9.4.4			· /
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	πb		4.3.1(4.4.1), 11.4
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	R.	9 ( )	(11.3)
R <sub>I</sub> a series of log entries $(O)$ created through execution of the transaction cumulative gas used immediately after the 4.3.1(4.4.1), 11.2, transaction has happended $(\mathbb{N})$ 11.4  R <sub>z</sub> status code of the transaction $(\mathbb{N})$ 4.3.1(), 11.4()  message call recipient 8  refunds $(A_r)$ 9.3  r() !!! 11.4  Bloom filter of a receipt (see $O$ ) 4.3.2(4.4.2)  RETURN !!! 9.4.4			
$\begin{array}{c} \text{cution of the transaction} \\ R_{\text{u}} & \text{cumulative gas used immediately after the} \\ \text{cumulative gas used immediately after the} \\ \text{transaction has happended ($\mathbb{N}$)} & 11.4 \\ R_{\text{z}} & \text{status code of the transaction ($\mathbb{N}$)} & 4.3.1(), 11.4() \\ \text{message call recipient} & 8 \\ \text{refunds ($A_{\text{r}}$)} & 9.3 \\ \text{c()} & !!! & 11.4 \\ \text{cb} & \text{Bloom filter of a receipt (see $O$)} & 4.3.2(4.4.2) \\ \text{RETURN} & !!! & 9.4.4 \\ \end{array}$			
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	$n_{ m l}$	- , , ,	4.3.1(4.4.1), 11.4
$R_z$ status code of the transaction (N) $4.3.1(), 11.4()$ $r$ message call recipient $8$ $r$ refunds $(A_r)$ $9.3$ $r()$ !!! $11.4$ $r_b$ Bloom filter of a receipt (see $O$ ) $4.3.2(4.4.2)$ RETURN       !!! $9.4.4$	$R_{ m u}$	cumulative gas used immediately after the	4.3.1(4.4.1), 11.2,
message call recipient       8         refunds $(A_r)$ 9.3         ()       !!!       11.4 $C_b$ Bloom filter of a receipt (see $O$ )       4.3.2(4.4.2)         RETURN       !!!       9.4.4			
refunds $(A_r)$ 9.3         r()       !!!       11.4         rb       Bloom filter of a receipt (see $O$ )       4.3.2(4.4.2)         RETURN       !!!       9.4.4	$R_{\mathbf{z}}$		4.3.1(), 11.4()
C)       !!!       11.4         Cb       Bloom filter of a receipt (see O)       4.3.2(4.4.2)         RETURN       !!!       9.4.4	r	message call recipient	8
Bloom filter of a receipt (see $O$ )  4.3.2(4.4.2)  RETURN  9.4.4	r	refunds $(A_{\rm r})$	9.3
Bloom filter of a receipt (see $O$ ) 4.3.2(4.4.2) RETURN !!! 9.4.4	r()	!!!	11.4
RETURN !!! 9.4.4	$\mathbf{r}_{\mathrm{b}}$	Bloom filter of a receipt (see O)	
			9.4.2()

REVERT	!!!	9.4.0(), 9.4.4()
RLP()	!!! RLP transformation	4.1, 4.3.2(4.4.2),
		4.3.3(4.4.3),
		4.3.4(4.4.4), 7
S(T)	transaction sender	4.2(4.3), 6.2
8	suicides set $(A_s)$	9.3
8	!!! $(\mu_{s})$	9.4.1
s	(contract creation or message call) sender	7, 8
s()	check "is-sibling" property	11.1
SELFDESTRUCT	!!!	6.1(), 9.4.2(),
		9.4.4()
SSTORE	!!!	6.1, 9.4.2()
STOP	!!!	6.1, 9.4.1, 9.4.4
SUICIDE	!!!	(6.1, 9.4.4)
$\overline{\mathbf{T}}$	transactions in the given block	4.3.0(4.4.0)
$T, T_{number}$	a transaction	2.0, 4.2(4.3), 6.0,
· 		6.2
$T_{\mathbf{d}}$	data, input data of the message call $(\mathbb{B})$	4.2(4.3), 6.2
$T_{ m g}$	gasLimit, maximum amount of gas used in this	4.2(4.3), 6.2
	transaction $(\mathbb{N}_{256})$	
$T_{\mathbf{i}}$	init, EVM-code for the account initialisation	4.2(4.3), 6.2
	procedure, executed only once at account cre-	
	ation $(\mathbb{B})$	
$T_{ m n}$	transaction nonce, number of transactions sent	3, 4.2(4.3)
	by the sender $(\mathbb{N}_{256})$	
$T_{\rm o}$	original transactor, which can differ from the	6.2
_	sender $(S(T))$	
$T_{\rm p}$	gasPrice, number of Wei to be paid per unit of	4.2(4.3), 6.2
	$\operatorname{gas}\left(\mathbb{N}_{256}\right)$	
$T_{ m r}$	one of transaction signature values $(\mathbb{N}_{256})$	4.2(4.3)
$T_{ m s}$	one of transaction signature values $(\mathbb{N}_{256})$	4.2(4.3)
$T_{ m t}$	to, message call' recipient address $(\mathbb{B}_{20})$ or (con-	4.2(4.3)
	tract creation transaction) $\emptyset$ ( $\mathbb{B}_0$ )	
$T_{ m v}$	value, number of Wei to be transferred $(\mathbb{N}_{256})$	4.2(4.3), 6.2
$T_{ m w}$	one of transaction signature values $(\mathbb{N}_5)$	4.2(4.3)
t	touched accounts $(A_{\mathbf{t}})$	8(), 9.3()
TRIE()	!!!	4.1, (4.4.1),
		4.3.2(4.4.2), 7, 8,
		11.4
U	set of uncle block headers	4.3.0(4.4.0)
V()	block header validity function	4.3.4(4.4.4), 11.1
v()	account validity function	4.1
v	(contract creation) endowment or (message	7, 8
	call) value	
$ ilde{v}$	value to be transferred for the DELEGATECALL	8
	instruction	-
<u>v'</u>	account's pre-existing value	7
$v_0$	up-front cost	6.2
W	!!!	9.4.2()
w	!!!	7(), 8(), 9.4.0(),
V()	!!!	9.4.1, 9.4.2, 9.5
X()	!!!	9.4.0

$X_{0,1,2,4}()$	!!!	(9.4.0)
$\overline{x}$	difficulty adjustment factor	<b>4.3.4</b> (4.4.4), 11.4
Z()	function which check an exceptional halting	9.4.0, 9.4.1, 9.4.2
	state of the machine (cf. $H()$ )	
z	status code !!!	6.2(), 7(), 8()