

Electronic Voucher

V1.1

References

[1] Reference Functional Specifications VIC Standard

Abbreviation

ECR	Electronic Cash Register
PDV	Purchase Device
VMC	Vending Machine Controller
V.I.A.	Voucher Issuers Association

1. Description of the integration

The principle of the solution consists of sending from the Electronic Cash Register (ECR) the *product code* (Eco Voucher) that will be used for the transaction as payment means. This *product code* will be transmitted through the VIC protocol to the payment terminal. In turn, the terminal will transmit the *product code* to the Transaction Host Processor. Based on the product code, the Transaction Host Processor will do the necessary to identify and debit the corresponding account of the cardholder.

The field to be used through the VIC interface in order to start a transaction for a specific product is the *vic_data* (field number 36) in the *vmc_debit_request* message. Its content is composed of a set of data where each data owns a specific typed code. The type code that must be used in this context is the type code value 05 "discretionary data". Its value must respect the one defined as Product Category. From now on, the product code used in this context will be called an « **Acquirer Discretionary Data** » field value (**ADD**).

If ADD is provided, the host will check the presence and order of the ADD fields. The host will also check if the selected product is allowed in the merchant contract and will refuse the transaction if it is not the case. Otherwise the host should process the transaction based on the information provided in the ADD field. If ADD is not provided, or missing the multi-product tag, the host should process the transaction using the product-id derived from the merchant contract. In the case where the host receives a request without ADD and multi-product is linked to the account, the transaction will be refused.

Contact person for technical questions: kenzai.abderrahmane@sodexo.com
+32 484 71 82 77

3. VIC messages

How to modify the VIC protocol in your cash register application to be able to accept the Lunch Voucher and/or Eco Voucher (or other electronic product in the future)

3.1. VMC_debit_request of electronic voucher

FIELD		FORMAT		COMMENTS
index	Name	# bytes	type	
-	vic_protocol_id	2	4x	Always 01 07
-	vic_msg_code	2	2a	Always 56 44 (=VD)
-	vic_bit_map	24	192b	
3	vic_tx_amt	3	6x	Amount in eurocent (hexadecimal!). Example : 10,00 EUR -> 00 03 E8 Range : [00 00 01..FF FF FF]
12	tx_type	1	2i	Always 04 (single transaction)
16	vic_card_ind	1	2i	Always 01 (remove card after transaction)
20	vic_to	2	4x	Time given for card insertion. Always 00 1E (30")
21	vic_tx_id	4	8x	Transaction identifier issued by VMC. Must be different for every transaction. Range : [00 00 00 01..FF FF FF FF]
23	Curcy	2	4i	Always 09 78 (EURO)
25	vic_cust_ind	1	2i	Always 01 (ask for cardholder validation)
36	vic_data		llllvar	All values for the discretionary_data (Type code value: 05): 200001# (lunch-Voucher) 200002# (eco-Voucher) 200003# (cadeau-Voucher) 200004# (sport culture-Voucher) 200005# (book-Voucher) 200006# (transport-Voucher) 200007# (Consumption Pass)
143	operator_nr	2	4i	Identifies the operator of the VMC. Range : [00 01 .. 99 99]
179	vic_version	1	2i	Always 13 (vic_protocol_subversion 1.07/13)

3.2. PDV_debit_result of Meal or Eco voucher

FIELD		FORMAT		COMMENTS
index	name	# bytes	length	
-	vic_protocol_id	2	4i	Always 01 07
-	vic_msg_code	2	2a	Always 50 44 (=PD)
-	vic_bit_map	24	192b	
2	term_id	8	8a	Terminal identifier
3	vic_tx_amt	3	6x	Transaction amount as asked by the VMC. range: [00 00 00 .. FF FF FF]
4	iep_tx_inc	2	4i	Indicates the result of the transaction range: [00 00 .. 99 99]
5	lg_cust	1	2i	Indicates the language of the card used range : [00 .. 99]
9	pdv_state	4	32b	Indicates the 'state' of the PDV. Irrelevant.
11	card_id_disp	24	24a	Contains the card nr, sent in "ready to print" ASCII-format; some digits of the number may be scrambled (depends on acquirer's security rules)
21	vic_tx_id	4	8x	Echo of transaction identifier issued by the VMC.

				Range: [00 00 00 01 .. FF FF FF FF]
22	vic_msg_type	1	8b	Default 00. If 10, indicates that a message has to be displayed by the VMC, as given in the field display_message or referenced by the field display_text_ref (depends on the presence of the fields).
23	curcy	2	4i	Always 09 78 (EURO)
36	vic_data	2 + 1..999	4x + 1..1998x	May contain different information from the acquirer. See full spec for format and details. e.g.: acquirer-issued authorisation code for the transaction and the used discretionary data of the debit_request message. Example: 00170706363135373234080532303030303223
125	<tx>vic_tx_amt	3	6x	Transaction amount as debited by PDV ¹ . Range: [00 00 01 .. FF FF FF]
126	<tx>curcy	2	4i	Always 09 78 (EURO)
145	ticket_data	2 + 1..999	4i + 1..999a	The first two bytes announce the length of the field (only digits), these two bytes not included . Contains data (sent in "ready-to-print" ASCII-format) which must be printed by the VMC.
159	display_data	2 + 1..999	4i + 1..999a	This field has to be taken into account only if indicated by the vic_msg_type field. The first two bytes announce the length of the field (only digits), these two bytes not included . Contains data (sent in "ready-to-print" ASCII-format) that must be displayed by the VMC.
170	transaction_protocol	2	4i	Info about communication between PDV and host. Range : [00 00 .. 00 34] (1 st byte always 00)
171	vic_bit_map_application_id	16	128b	Identifies the card application used for the transaction.
172	transaction_identifier	4	8i	Acquirer-issued identifier of the transaction. Range : [00 00 00 00 .. 99 99 99 99]
173	date_and_time	7	14i	Time of the transaction ² Format : YYYY MM DD hh mm ss
174	brand_id	2	4i	Brand_id of the card used. Range : [00 00 .. 99 99]
176	cvm	2	4i	Info about how cardholder validates the transaction. range : [00 00 .. 00 22] (1 st byte always 00) Cardholder must sign the merchant copy of the transaction ticket only if cvm = xx1x.
178	brand_name	1 + 1..24	2i + 1..24a	The first byte indicates the length of the field, this byte not included. The following bytes contain the brand name. This field identifies the means of payment used by the cardholder.
179	vic_version	1	2i	Always 13 (vic_protocol_subversion 1.07/13)
182	display_text_ref	2	4x	This field has to be taken into account only if indicated by the vic_msg_type field. Contains a reference to a message to be displayed by the VMC.

See **Annex 1: VIC logging of an electronic Voucher transaction**

¹ In the case of Partial Approval (indicated by an iep_tx_inc at 00 01), this amount can be different than the one specified in the field vic_tx_amt. Partial Approval is only available in VIC 1.07/13 (a re-certification is required to support P.A.)

² As logged by the acquirer, or by the terminal in case of emergency fall-back

4. Available configurations

4.1. Mono-product contract

Accepted:

- Transaction Processing Host receives request without ADD multi-product tag 20 (product selection is done based on contract)
- Transaction Processing host receives request with ADD multi-product tag 20 and selected product is allowed in the contract (well-formatted and correct value)

Refused:

- Transaction Processing host receives request with ADD multi-product tag 20 but not well-formatted and/or with an incorrect value
- Transaction Processing host receives request with ADD multi-product tag 20 and selected product is not allowed in contract

4.2. Multi-product contract

Accepted:

- Transaction Processing Host receives request with ADD multi-product tag 20 (well-formatted and correct value)

Refused:

- Transaction Processing Host receives request with ADD multi-product tag 20 not well-formatted and/or with an incorrect value
- Transaction Processing Host receives request without ADD multi-product tag 20

This integration is supported by all the VIC 1.07/xx versions. No additional VIC re-certification is required for this integration. These electronic voucher cards are supported by the attended terminals XENTA and YOMANI.

For technical support about the VIC integration, please contact the ECR-Vending certifications mailbox: ecrvendingcertifications-belgium@worldline.com

Annex 1

Event Number	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	ASCII	
1	02	05	01	07	56	44	a0	11	1a	80	10	10	00	00	00	80	%020501075644a0111a80101000000080	
17	00	00	00	00	00	00	00	00	10	03	00	00	00	00	00	20	UUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUU	
33	00	00	10	03	08	00	00	1e	f1	d3	b5	11	09	78	00	00	UUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUU	
49	09	08	05	32	30	30	30	30	32	23	12	34	20	20	13	03	%0908053230303030303223123420201303	
65	75	d6	06	05	02	1a	01	07	50	44	f8	a0	0e	00	10	10	u%75d60605021a01075044f8a00e001010	
81	00	00	00	80	00	00	00	00	00	00	00	0c	00	00	80	00	UUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUU	
97	02	7d	60	00	30	31	36	36	33	30	31	34	00	00	10	03	%} `027d6000303136363330313400001003	
113	00	00	04	40	00	00	00	78	78	78	78	78	78	78	78	78	UUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUU	
129	78	78	78	36	34	37	32	20	20	20	20	20	20	20	20	f1	xxx6472%78787836343732202020202020202020f1	
145	d3	b5	11	00	09	78	00	17	07	06	36	31	35	37	32	34	%d3b51100097800170706363135373234	
161	08	05	32	30	30	30	30	32	23	00	00	10	03	09	78	07	%08053230303030322300001003097807	
177	19	43	4c	49	45	4e	54	20	54	49	43	4b	45	54	0d	0a	%19434c49454e54205449434b45540d0a	
193	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	-----	
209	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	-----	
225	2d	2d	2d	2d	2d	2d	2d	0d	0a	04	0e	20	20	20	20	20	----- %2d2d2d2d2d2d2d0d0a040e2020202020	
241	20	20	20	20	20	20	20	20	20	20	20	20	20	0d	0a	04	%20202020202020202020202020200d0a04	
257	0a	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	%0a20202020202020202020202020202020	
273	20	20	20	20	0d	0a	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	%202020200d0a2d2d2d2d2d2d2d2d2d2d2d	
289	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	-----	
305	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	0d	0a	04	54	----- %2d2d2d2d2d2d2d2d2d2d2d2d0d0a0454	
321	65	72	6d	69	6e	61	6c	3a	20	30	31	36	36	33	30	31	erminal:%65726d696e616c3a2030313636333031	
337	34	0d	0a	04	4d	65	72	63	68	61	6e	74	3a	20	30	30	4%340d0a044d65726368616e743a203030	
353	30	30	30	31	30	34	20	20	20	20	20	20	20	0d	0a	04	000104%30303031303420202020202020200d0a04	
369	50	65	72	69	6f	64	3a	20	30	30	33	31	20	20	20	20	Period:%506572696f643a203030333120202020	
385	20	54	72	61	6e	73	61	63	74	69	6f	6e	3a	20	30	30	%205472616e73616374696f6e3a203030	
401	30	30	30	31	34	35	0d	0a	0d	0a	04	0e	20	20	20	53	000145%3030303134350d0a0d0a040e20202053	
417	6f	64	65	78	6f	20	65	50	61	73	73	20	20	20	20	0d	odexo%6f6465786f206550617373202020200d	
433	0a	0d	0a	20	20	20	20	20	20	20	20	20	20	20	20	28	%0a0d0a2020202020202020202020202028	
449	41	30	30	30	30	30	30	32	36	36	30	31	29	20	20	20	A00000026601)%4130303030303030323636303129202020	
465	20	20	20	20	20	20	20	20	20	0d	0a	0d	0a	04	0e	43	%202020202020202020200d0a0d0a040e43	
481	61	72	64	3a	20	78	78	78	78	78	78	78	78	78	78	78	ard:%6172643a207878787878787878787878	
497	78	36	34	37	32	20	20	20	0a	03	0a	0d	0a	43	61	72	64	x6472%7836343732202020200a030a0d0a43617264
513	20	73	65	71	75	65	6e	63	65	20	4e	72	3a	20	31	0d	%2073657175656e6365204e723a20310d	
529	0a	56	61	6c	69	64	20	74	68	72	75	3a	20	33	30	2f	%0a56616c696420746872753a2033302f	
545	30	34	2f	32	30	0d	0a	0d	0a	54	45	53	54	20	45	56	04/20%30342f32300d0a0d0a54455354204556	
561	43	33	33	31	0d	0a	0d	0a	04	50	41	59	4d	45	4e	54	C331%433333310d0a0d0a045041594d454e54	
577	20	20	20	20	20	20	20	20	20	20	20	20	0d	0a	04	44	%20202020202020202020202020200d0a0444	
593	61	74	65	3a	20	32	31	2f	30	31	2f	32	30	31	36	20	ate:%6174653a2032312f30312f3230313620	
609	31	34	3a	32	34	20	20	20	20	20	20	20	20	20	20	20	14:24%31343a32342020202020202020202020	
625	20	20	20	20	20	0d	0a	0d	0a	04	41	75	74	68	2e	20	%2020202020200d0a0d0a04417574682e20	
641	63	6f	64	65	3a	20	36	31	35	37	32	34	20	20	0d	0a	code:%636f64653a2036313537323420200d0a	
657	0d	0a	0e	22	53	6f	64	65	78	6f	20	43	61	72	64	20	%0d0a0e22536f6465786f204361726420	
673	2f	20	45	63	6f	20	50	61	73	73	22	0d	0a	0d	0a	0d	/%2f2045636f2050617373220d0a0d0a0d	
689	0a	04	0e	54	6f	74	61	6c	3a	20	20	20	20	20	30	2c	%0a040e546f74616c3a20202020202020302c	
705	30	33	20	45	55	52	0d	0a	0d	0a	52	65	61	64	2d	6d	03%3033204555520d0a0d0a526561642d6d	
721	6f	64	65	3a	20	20	20	43	68	69	70	0d	0a	0e	0e	44	ode:%6f64653a202020436869700d0a0e0e44	
737	69	73	70	6f	6e	69	62	6c	65	20	31	34	39	2c	38	34	isponible%6973706f6e69626c65203134392c3834	
753	20	45	55	52	0d	0a	0e	64	6f	6e	74	20	31	34	39	2c	%204555520d0a0e646f6e74203134392c	
769	38	34	20	45	55	52	0d	0a	0e	65	78	70	69	72	65	20	84%3834204555520d0a0e65787069726520	
785	6c	65	20	31	37	2f	30	31	2f	32	30	31	37	0d	0a	20	le%6c652031372f30312f323031370d0a20	
801	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	%20202020202020202020202020202020	
817	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	%20202020202020202020202020202020	
833	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	%20202020202020202020202020202020	
849	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	%20202020202020202020202020202020	
865	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	%20202020202020202020202020202020	
881	20	20	20	20	20	20	20	20	20	20	20	20	20	0d	0a	0d	%20202020202020202020202020200d0a0d	
897	00	00	62	00	00	12	00	00	00	00	00	00	00	00	04	00	UUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUU	
913	00	00	00	00	00	00	00	00	01	45	20	16	01	21	14	24	UUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUU	
929	47	30	10	03	00	22	12	53	6f	64	65	78	6f	20	65	50	G0%47301003002212536f6465786f206550	
945	61	73	73	13	03	c2	d6	06	1a	@							ass%6173731303c2d6061a@	