

# PORTA27

## Display



(HP-41CX, Hewlett Packard 1983 and DM41X, [SwissMicros](#) 2020)

## Overview<sup>1</sup>

Program PORTA27 implements an encryption variation by [Giambattista della Porta](#), known as [polyalphabetic cipher](#), so called bigraphic substitution. [Porta](#) published De Furtivis Literarum Notis in 1563. This program is written for the HP-41CX as minimisation of the program in chapter 7 of the book: [Kryptologie \(HP-41 C/CV\): Chiffrierung, Textverschlüsselung, Datenschutz, Morsealphabet; Alt, Helmut; Schumny, Harald \[Hrsg.\], Braunschweig \[u.a.\], Vieweg, 1983. Vieweg-Programm-bibliothek Taschenrechner Band 7. ISBN 3-528-04256-7.](#)

## Formula

Coding according to Porta is done by looking up the value of two consecutive characters in a table (see Encryption Table). For example the word "PO" can be coded as 420. If a word has an odd number of characters, a space is appended for completion of the Porta coding. The table lookup can be computed by following below algorithms:

FORWARD	Taking the two consecutive characters in a word $C_1$ and $C_2$ , the Porta code $c_p$ can be determined as follows:	
	$c_p = 27 \cdot (C_1 - 1) + C_2$	$C_1, C_2$ are A..Z, space
	$c_p = 27 \cdot C_1 + C_2 - 1819$	Substituting $C_1 = C_1 + (65-1)$ , where 65 is char value of A
BACKWARD	Taking the Porta code $c_p$ , the two consecutive characters in a word $C_1$ and $C_2$ can be determined as follows:	
	$C_1 = \text{INT}(0,99 + c_p/27)$ $C_2 = \text{INT}(0,99 + 27 \cdot \text{FRC}(c_p/27))$	$c_p$ is 1..729

The lookup table shows a base of 27. More bases are possible if more characters need to be encrypted. An example of coding the word HOOXIES is shown in the encryption table which translates the word into a series of  $c_p$  values: 204, 402, 221 and 513.

<sup>1</sup> This program is copyright and is supplied without representation or warranty of any kind. The author assumes no responsibility and shall have no liability, consequential or otherwise, of any kind arising from the use of this program material or any part thereof

## Encryption Table

Cross first character in the left column with the second character in the top row. The character sequence is the encrypted code. For example, the word HOOXIES can be coded as follows:

1. "H0", in table is: 204
2. "OX", in table is: 402
3. "IE", in table is: 221
4. "S ", in table is: 513

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	
A	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
B	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54
C	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81
D	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100	101	102	103	104	105	106	107	108
E	109	110	111	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	128	129	130	131	132	133	134	135
F	136	137	138	139	140	141	142	143	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	160	161	162
G	163	164	165	166	167	168	169	170	171	172	173	174	175	176	177	178	179	180	181	182	183	184	185	186	187	188	189
H	190	191	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216
I	217	218	219	220	221	222	223	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240	241	242	243
J	244	245	246	247	248	249	250	251	252	253	254	255	256	257	258	259	260	261	262	263	264	265	266	267	268	269	270
K	271	272	273	274	275	276	277	278	279	280	281	282	283	284	285	286	287	288	289	290	291	292	293	294	295	296	297
L	298	299	300	301	302	303	304	305	306	307	308	309	310	311	312	313	314	315	316	317	318	319	320	321	322	323	324
M	325	326	327	328	329	330	331	332	333	334	335	336	337	338	339	340	341	342	343	344	345	346	347	348	349	350	351
N	352	353	354	355	356	357	358	359	360	361	362	363	364	365	366	367	368	369	370	371	372	373	374	375	376	377	378
O	379	380	381	382	383	384	385	386	387	388	389	390	391	392	393	394	395	396	397	398	399	400	401	402	403	404	405
P	406	407	408	409	410	411	412	413	414	415	416	417	418	419	420	421	422	423	424	425	426	427	428	429	430	431	432
Q	433	434	435	436	437	438	439	440	441	442	443	444	445	446	447	448	449	450	451	452	453	454	455	456	457	458	459
R	460	461	462	463	464	465	466	467	468	469	470	471	472	473	474	475	476	477	478	479	480	481	482	483	484	485	486
S	487	488	489	490	491	492	493	494	495	496	497	498	499	500	501	502	503	504	505	506	507	508	509	510	511	512	513
T	514	515	516	517	518	519	520	521	522	523	524	525	526	527	528	529	530	531	532	533	534	535	536	537	538	539	540
U	541	542	543	544	545	546	547	548	549	550	551	552	553	554	555	556	557	558	559	560	561	562	563	564	565	566	567
V	568	569	570	571	572	573	574	575	576	577	578	579	580	581	582	583	584	585	586	587	588	589	590	591	592	593	594
W	595	596	597	598	599	600	601	602	603	604	605	606	607	608	609	610	611	612	613	614	615	616	617	618	619	620	621
X	622	623	624	625	626	627	628	629	630	631	632	633	634	635	636	637	638	639	640	641	642	643	644	645	646	647	648
Y	649	650	651	652	653	654	655	656	657	658	659	660	661	662	663	664	665	666	667	668	669	670	671	672	673	674	675
Z	676	677	678	679	680	681	682	683	684	685	686	687	688	689	690	691	692	693	694	695	696	697	698	699	700	701	702
	703	704	705	706	707	708	709	710	711	712	713	714	715	716	717	718	719	720	721	722	723	724	725	726	727	728	729

## Example (1)

Please note my default FIX 5 setting in de examples below for the codes.

KEYSTROKES	DISPLAY	COMMENTS
		Run forward program
[XEQ] [ALPHA] P27F [ALPHA]	T X T = ?	Enter C <sub>i</sub> values, e.g. PORTA CODING ON HOOXIES
PORTA CODING ON HOOXIES [R/S]	420,00000	First value c <sub>p</sub>
[R/S]	479,00000	Second value c <sub>p</sub>
[R/S]	27,00000	Third value c <sub>p</sub>
[R/S]	69,00000	Fourth value c <sub>p</sub>
[R/S]	90,00000	Fifth value c <sub>p</sub>
[R/S]	358,00000	Sixth value c <sub>p</sub>
[R/S]	513,00000	Seventh value c <sub>p</sub>
[R/S]	392,00000	Eighth value c <sub>p</sub>
[R/S]	710,00000	Nineth value c <sub>p</sub>
[R/S]	393,00000	Tenth value c <sub>p</sub>
[R/S]	630,00000	Eleventh value c <sub>p</sub>
[R/S]	127,00000	Twelfth value c <sub>p</sub>
[R/S]	OK	Completed, code is: 420-479-27-69-90-358-513-392-710-393-630-127
[R/S]	T X T = ?	Enter C <sub>i</sub> values, e.g. HP
HP [R/S]	205,00000	Value c <sub>p</sub>
[R/S]	OK	Completed, code is: 205
[R/S]	T X T = ?	Enter other C <sub>i</sub> values

## Example (2)

Below example decrypts a numerical sequence back to a text.

KEYSTROKES	DISPLAY	COMMENTS
		Run backward program
[XEQ] [ALPHA] P27B [ALPHA]	CODE = ?	Enter first c <sub>p</sub> value, e.g. 204
204 [R/S]	HO	Enter second c <sub>p</sub> value, e.g. 402
402 [R/S]	HOOX	Enter third c <sub>p</sub> value, e.g. 221
221 [R/S]	HOOXIE	Enter fourth c <sub>p</sub> value, e.g. 513
513 [R/S]	HOOXIES	Stop new values, press R/S to continue
[R/S]	HOOXIES .	Completed indicated with dot at the end
[R/S]	CODE = ?	Start with new c <sub>p</sub> value(s)

## Program Listing

P27F makes use of an optimised (forward) algorithm that includes the character values of alphas A..Z (65..90). However, the space value (32) does not fit so nicely in this sequence, the formulae is non-linear. In LBL 02 this is intercepted by checking against 32. If the ATOX value exceeds 32, it can proceed according to the forward algorithm, otherwise it substitutes the value 91 representing the character value of a space in the optimised (forward) algorithm. P27B (backward) has to cope with the same non-linear problem of the space. In LBL 03 it checks whether the calculated character value is zero. This happens for character 27 (space) and results in cp/27 getting very close to zero. The program checks whether  $X=0?$  and appends a space to the text string, otherwise it converts XTOA (after first adding 64 to any of the possible values 1..26).

01 ■ LBL "P27F"	20 RCL 02	01 ■ LBL "P27B"	20 GTO 00
02 ■ LBL 03	21 +	02 "CODE=?"	21 ■ LBL 03
03 "TXT=?"	22 1819	03 AVIEW	22 .99
04 AON	23 -	04 CLA	23 +
05 PROMPT	24 STOP	05 ■ LBL 00	24 INT
06 AOFF	25 DSE 00	06 CF 22	25 X=0?
07 ALENG	26 GTO 00	07 STOP	26 >" "
08 STO 00	27 "OK"	08 FC? 22	27 X=0?
09 2	28 PROMPT	09 GTO 02	28 RTN
10 MOD	29 GTO 03	10 27	29 64
11 ST+ 00	30 ■ LBL 02	11 /	30 +
12 LASTX	31 32	12 STO 01	31 XTOA
13 ST/ 00	32 ATOX	13 XEQ 03	32 RTN
14 ■ LBL 00	33 X>Y?	14 RCL 01	33 ■ LBL 02
15 XEQ 02	34 RTN	15 FRC	34 >". "
16 27	35 91	16 27	35 AVIEW
17 *	36 END	17 *	36 END
18 STO 02		18 XEQ 03	
19 XEQ 02	(69 bytes)	19 AVIEW	(69 bytes)

## Registers, Labels and Flags

REGISTERS	COMMENTS	LABELS	COMMENTS
R00	Number of required lookups: length of the text divided by 2	LBL02	Present outcome
R01	C <sub>p</sub> divided by 27	LBL02	Loop around the given input text characters

  

FLAGS	COMMENTS
22	Check for keyboard input

## Downloads

The RAW/TXT format of the program is available via the website: [PORTA27](#) (in zip file).