

从QR文字列中抽出型号的方法

ini.csv的Qrmode为0时
用2次元扫描枪读取的QR文字列的先头开始、把BarCodeHeaderStrNum的文字数删除后剩下的文字列认识为型号
QR文字列的数式「PO@型式」的时候，就用这个模式就OK。

ini.csv的Qrmode为1的时候
通过2次元扫描枪读取的QR文字列中，把QRsplTStr定义的文字进行分割，把MODELplc指定的地方的文字列作为型号进行识别的模式
QR文字列书式为「PO@型式@数量」的时候，使用这个模式。

1 本系统中，读取的型号情报以「-」键进行分割。

当读取了C-CCC10-20A-P5-30这个型号的时候，在内部

「C」	B
「CCC10」	CCC10
「20A」	
「p5」	B-CCC10
「30」	

文字列是被分割成这样的。（以下称为**分割文字列**）

通过登录header.csv、可以作为如下记一样的「C-」为型号来被认识

「C-CCC10」		
「20A」	header.csv登録例	
「p5」	DEFINE	KIND
「30」	C	add
	B	add

※把KIND以del来定义的话，读取型号的C-会被判断为不要，最后被判断为「CCC10-20A-P5-30」这个型号。

另外分割文字列中，第一个字以数字开始「20A」、「30」等、在内部被定义为size1、size2。（第一个是size1、第二个是size2・・・）
想利用P5等文字开头的文字列中的数字的时候，需要另行定义。

分割文字列	内部定义
「C-CCC10」	-
「20A」	size1
「p5」	-
「30」	size2

2 型号读取→宏变量情报显示出来为止的参考master的构造
（例）读取C-CCC10-20A-P5S-30的时候

① 把读取的型号从后开始一个字一个字的删除，搜索type_define.csv的DEFINE项目。取得匹配的行的「NO」。

検索文字列	type_define.csv検索結果	type_define.csvのNO
C-CCC10-20A-P5S-30	×	—
C-CCC10-20A-P5S-3	×	—
C-CCC10-20A-P5S-	×	—
...	×	—
C-CCC10	×	—
C-CCC1	×	—
C-CCC	○	3

type_define.csv登録内容

NO	TYPE	DEFINE1	DEFINE2
1	AAA	define1-1	define1-2
2	AAA	define2-1	define2-2
3	C-CCC		

N03的时候，因为DEFINE1,2没有定义，所以不参照type_relation.csv、type_chngv1.csv。
N01,2的时候，参照type_relation.csv、type_chngv1.csv。

- ② 在type_prg.csvの「NO」列里，搜索①取得的值。搜索匹配行，取得程序表示顺序する。
※在prg.csv时、有必要对使用的程序进行定义。
这个例子的时候，N03的表示顺序为prg3→prg1→prg2。

type_prg.csv登録内容

NO	prg1	prg2	prg3
1	1	2	3
2	3	2	1
3	2	3	1

prg.csv登録内容

PRGNO	PRGNAME	PRGTEXT
1	prg1	test1
2	prg2	test2
3	prg3	test3

- ③ 初次表示prg文件夹内，搜索load.csv。以NO为Key、取得对象行。
这个例子中，初次表示的是prg3文件夹里的load.csv。「NO」取得了3这一行的情报取得。

prg3/load.csv登録内容

NO	TYPE	DRAWING	DISPFLG	#500	#501	#502	#503	#504	#505	#506	#507	#508	#509
1	AAA	test1.jpg	1	10	size1	define1-1	define1-2	5	measure1	select1	relation1	switch1	correct1
2	AAA	test1.jpg	1	20	size1	define2-1	define2-2	5	measure1	select2	relation2	switch1	correct1
3	C-CCC	test1.jpg	1	30	size1	define3-1	define3-2	5	measure1	select3	relation3-3	switch1	correct3

N03の登録内容

項目	設定値	form形状	説明	値搜索处file名
DRAWING	test1.jpg	—	在load.csv file配置文件夹内的「DRAWING」文件夹内，搜索左边的file后显示。	—
DISPFLG	1	—	1的时候，表示这个程序。0的时候不表示，而表示下面的程序（这个例子的时候是prg1）	—
#500	30	load	从cntrl.csv上搜索#500的定义，取得form形式。设定「30」的值。	—
#501	size2	load	从cntrl.csv上搜索#501的定义，取得form形式。设定「size1」的值。	—
#502	define3-1	load	从cntrl.csv上搜索#502的定义，取得form形式。设定「define3-1」上被定义了的值。	define.csv
#503	define3-2	load	从cntrl.csv上搜索#503的定义，取得form形式。设定「define3-2」上被定义了的值。	define.csv
#504	5	input	从cntrl.csv上搜索#504的定义，取得form形式。设定「5」的值。	—
#505	measure1	measure	从cntrl.csv上搜索#505的定义，取得form形式。设定「measure1」上被定义了的值。	measure.csv
#506	select3	select	从cntrl.csv上搜索#506的定义，取得form形式。设定「select3」上被定义了的值。	select.csv
#507	relation3-3	relation	从cntrl.csv上搜索#507的定义，取得form形式。设定「relation3-3」上被定义了的值。	relation.csv
#508	switch1	switch	从cntrl.csv上搜索#508的定义，取得form形式。设定「switch1」上被定义了的值。	switch.csv
#509	correct3	correct	从cntrl.csv上搜索#509的定义，取得form形式。设定「correct3」上被定义了的值。	correct.csv

cntrl.csv登录内容

MACRO	KIND	DISPFLG	SENDFLG	AUTOSENDFLG	BTNNNAME	CHANGEPRG	LABELTXT	MIN	MAX	CMNT B	CMNT R
#500	load	1	1	0							
#501	load	1	1	0							
#502	load	1	1	0							
#503	load	1	1	0							
#504	input	1	1	1	角A						
#505	measure	1	1	1							
#506	select	1	1	0							
#507	relation	1	1	0							
#508	switch	1	1	1							
#509	correct	1	1	0							

④ 以③上取得了的macro情报为基础(带#的项目名)、制作显示内容。

读入型式: C-CCC10-20A-P5S-30

項目	KIND	値
#500	load	30

load的格式(form)内、设定30的值后显示

读入型式: C-CCC10-20A-P5S-30

項目	KIND	値
#501	load	size2

在load的格式(form)内设定size2的值,表示。(此例来看, size2: 30)

項目	KIND	値
#502	load	define3-1

load的格式(form)内设定define3-1的值,显示。

define.csv登录内容 读入型式: C-CCC10-20A-P5S-30

DEFINE	STR	BEFORE	AFTER	CHNGVL	CALC
define3-1	C-CCC	C-CCC			

在分割文字列中、取得以STR的定义开始的文字列。针对于此值
将BEFORE上被定义了的值变换为AFTER后的值作为define3-1的值。
此例来看、C-CCC10的C-CCC会取得变换为"10"后的值10。

分割文字列	内部定义
「C-CCC10」	-
「20A」	size1
「p5S」	-
「30」	size2

項目	KIND	値
#503	load	define3-2

load形式内设定define3-2的值,显示。

define.csv登录内容 读入型式: C-CCC10-20A-P5S-30

DEFINE	STR	BEFORE	AFTER	CHNGVL	CALC
define3-2	P	P0		0 chngS	
define3-2	P	P1		1 chngS	
define3-2	P	P2		2 chngS	
define3-2	P	P3		3 chngS	
define3-2	P	P4		4 chngS	
define3-2	P	P5		5 chngS	calc2-2
define3-2	P	P6		6 chngS	
define3-2	P	P7		7 chngS	
define3-2	P	P8		8 chngS	
define3-2	P	P9		9 chngS	
define3-2	P	-		0	

※没有以P开始的文字时、会被设定为0

在分割文字列中、取得以STR的定义开始的文字列。针对于此值
将BEFORE上被定义了的值变换为AFTER。
会得到5S的值

chngValue.csv的登录内容

DEFINE	BEFORE	AFTER
chngS	S	1

变换后51

将define.csv上变换了的值(5S)在chngValue.csv上再加变换。
5S的S变换为1、得到51的值

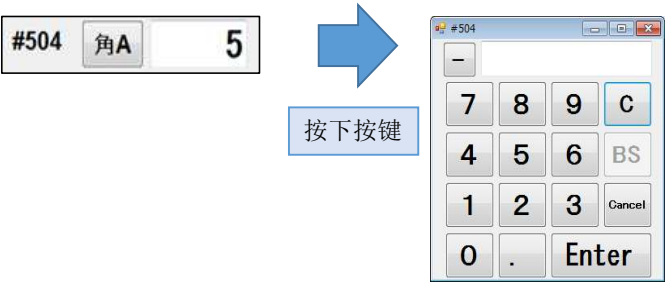
DEFINE	1	2	3	4
calc2-2	=	calc2-2	+	1

计算后: 52

因为是自己的定义+1、51+1、从而得到52的值。
此值为define3-2的值, 被设定到形式内。

項目	KIND	値	BTNNAME
#504	input	5	角A

input的形式内默认值设为5。
按键显示名为角A



項目	KIND	値
#505	measure	measure1

在measure的形式内设定在measure1上被定义了的值。

measure.csv登录内容

DEFINE	LABELVALUE	DEFVALUE	SENDVALUE
measure1	卡尺值	0	relation10

标签名: 卡尺值

默认值: 0

送信值: 设定relation10上被定义了的值

#505 ノギス値

蓝色涂鸦处为显示卡尺测定值处

上记的情况下, 此定义名为#505M (macro名+M)

relation.csv的登录内容

DEFINE	VALUE	1	2	3	4	5	6	7	8
relation10	1 and	#505M	>=		0 and	#505M	<=		1
relation10	#505M and	#505M	>		1				

卡尺测量值在0以上1以下时, 显示1。

卡尺测量值在1以上时, 显示卡尺测量值

項目	KIND	値
#506	select	select3

在select的form内设定select3上被定义了的值。

select.csv的登录内容

DEFINE	DISPVALUE	SENDVALUE
select3	test1-1	1
select3	test1-2	2

#506 test1-1

#506

test1-1
test1-1
test1-2

▼按下

列表上显示的值为test1-1, test1-2。

选择est1-1时为 1、选择test1-2时, 2会显示在旁边的文本框内2

項目	KIND	値
#507	relation	relation3-3

relation的form内设定relation3-3上定义了的值。

※relation form和load form一样、可以提前在load上定义好

define.csv的登录内容

DEFINE	VALUE	1	2	3	4
relation3-3	#506 and	#506	=		1
relation3-3	calcA and	#506	>=		2

#506的值在2以上时, calcA的计算结果会被显示。

#506的值为1时、#506的值会被显示。

calc.csv的登录内容

DEFINE	1	2	3	4	5	6	7	8	9	10
calcA	=	#513	-	Sqrt(calcTest)						
calcTest	=	#513	*	#513	-	#505	*	#505	/	4

calcA的计算结果、为#513-√ (#513^2-#505^2/4)。

項目	KIND	値
#508	switch	switch1

在switch的格式（form）内设定switch1上被定义了的值。

switch.csv的登录内容

DEFINE	DISPVALUE	SENDVALUE
switch1	test1	1
switch1	test2	2
switch1	test3	3

初期显示为按键表示値：test1 文本値：1
每按下按键时变化

項目	KIND	値
#509	correct	correct3

在correct的格式（form）内设定correct3上被定义了的值。

correct.csv的登录内容

DEFINE	DEFVALUE	UPDWN	UPPER	LOWER
correct3	1	0.5	10	0

默认表示：1
按下±时的变化値：0.5
上限：10
下限：0

#508

test1

1

→

#508

test2

2

←

按下按键

→

#508

test3

3

#509

+

1

-

3 按下Btn_chngprgm，变更了程序的时候，以已读取的型式情报为基础、参照下个prg文件夹内的master后，显示。