Quadravox

Playback Modules: QV306M4-P and QV306M4

QV306m4-P: pre-programmed RS232 playback module for ISD4003-4MP series ChipCorders™

The QV306m4-P module is functionally identical to the QV306m4 module (see data sheet for QV306m4). The only difference between the two modules is that QV306m4-P comes pre-programmed with 240 professionally recorded words and phrases. It can be used "directly out of the box" for numerous "playback only" applications, such as:

- Talking clock
- Talking calendar
- Talking thermometer
- Talking calculator
- Talking speller
- Talking instruments
- .....

The complete list of pre-programmed words and phrases is given in the annex to this document. The vocabulary list is arranged in such an order as to make algorithmic access easier to certain parts of the list. For example, all number words are arranged in order at the beginning of the list. The days of the weeks, months, letters and military alphabet letters are also sequentially organized. Most of the rest of the vocabulary is arranged in alphabetical order.

A sample Basic Stamp II™ application is supplied, to illustrate the interface between QV306m4-P and a popular software development system. This basic program contains several subroutines to:

- ✓ Speak a number from 0 to 65535
- ✓ Say a date between 1/1/1900 and 12/31/2099
- ✓ Say the time of day
- ✓ Change the output volume
- ✓ Get the module's revision number and type

Since the QV306m4-P is functionally identical to the QV306m4, it can be reprogrammed using a Quadravox QV430P programmer, if the vocabulary is not suitable for a certain application.

QV306m4 RS232 playback module for ISD33000-4000 series ChipCorders®

Features:

- delivered with 4 minute ISD4003-04
- up to 240 messages
- three addressing modes
- low power dissipation: <2mA at 5V, less than 1uA in power down mode
- single 5V supply (3V is generated on-board)
- separate amplifier power control and busy functions
- digitally controlled analog volume control
- 300mW amplifier on-board, compatible with QVamp3 12W amplifier module
- built-in sequential record mode for initial ChipCorder® setup
- automatic sensing of number of phrases when used with our recording software
- free recording software from our website
- PC interface hardware available, or build your own from our schematics
- for use with simple RXD/TXD RS232 controls

General description:

The QV306m4 module provides a simple method of adding voice and sound to your system. It comprises a controller based on a Microchip PIC processor, an ISD4003-04 ChipCorder® analog recording chips with four minutes capacity at 4kHz bandwidth, a digital potentiometer for volume control, and a 300mW audio amplifier. A line level, volume controlled output is also available. It is controlled with a simple two-wire RS232 connection (the return path is optional). Baud rate is selectable.

By taking advantage of the ChipCorder's® cueing mode, the QV306 can manage up to 240 separate messages without explicit reference to physical addresses. Three phrase selection modes are provided and are selected via a mode setting command. The addressing mode can be changed at any time.

A single n,8,1 byte transfer selects one of 240 messages (subject to the constraints explained later in this document). A code of 0fxh is interpreted as a mode- or volume- set command. A byte in the range 0-239 selects a single message in direct addressing mode, or determines the upper bound of a linear sequence or random selection. Upon power up or after a recording, the system will automatically detect the number of messages and define its own maximum upper bound.

The ChipCorder® and amplifier are powered only during operations. The QV306 is powered from start-up until the sleep mode bit is set with a set mode command. The mute/power pin can be used to switch power to an external amplifier. RESET_ wakes the QV306 from sleep mode.

Simple programming tools for creating properly formatted ISD ChipCorders® are available in software and schematic form free from <www.quadravox.com>. These tools eliminate the need to purchase a sound development system. If required, the Quadravox QV401P gang programmer can be used to mass produce formatted ChipCorder® devices.

In order to make the module as small as possible, the RS232 level conversion, audio connectors, power jack and 5V regulator are on a separate PCB, the QV430P. The QV430P makes it convenient to use our free QV300S2 programming software, but it not required for operation.

QV306m4 pin out and control lines:

Control lines	pin	level	Function
RXD (5V)	**1**	**--**	RS232 receive line
TXD (5V)	**2**	**--**	RS232 transmit line
BUSY_	**3**	**low**	indicates system is busy
BR1	**4**	**--**	msb of baud rate selection; must be +5V or GND
BR0	**5**	**--**	lsb of baud rate selection; must be +5V or GND
RECLED_	**6**	**low**	can be used to sink current for record LED. The external system must provide the current-limiting resistor.
+5V	**7**	**+5**	system power
GND	**8**	**0**	system ground
ANA_IN	**9**	**--**	line level analog input for recording
ANA_OUT	**10**	**--**	line level, volume controlled, analog output
GND	**11**	**0**	system ground
PWR_	**12**	**low**	can be used to control an external power amplifier such as the QVamp3.
AUX	**13**	**--**	not used in this version.
RESET_	**14**	**low**	system reset
SP-	**15**	**--**	speaker output; bridge tied load
SP+	**16**	**--**	speaker output; bridge tied load

Baud rate selection:

BR1	BR0	Rate
0	0	2400b
0	1	4800b
1	0	9600b
1	1	19200b

Commands:

Value	Record mode?	Action
0-239	No	Play phrase <value> according to mode settings
240-255	No	Set mode: see mode table for functions
1	Yes	Start recording individual phrase
0	Yes	Stop recording individual phrase

Set mode functions:

Value (hex)	Function
0f0h	set play mode to direct addressing
0f1h	set play mode to linear sequence
0f3h	set play mode to random sequence
0f4h	record mode; subsequent 1 and 0 commands start and stop recording
0f6h	stop play
0f8h	sleep
0fch	set volume with next byte transmitted. Only values 0-31 are valid
0fdh	software reset
0feh	return version byte (00ch)
0ffh	return type byte (036h)

Setting the addressing mode:

The QV306m4 offers three modes of playback addressing. The modes are selected using the two low-order bits of the set mode command. Bit 0 selects direct or sequential addressing. In direct addressing (bit 0 = 0), the value transferred in the command byte is used to select the phrase of the same number. If a value exceeding the maximum number of recorded phrases is sent, the last phrase in the memory will be played. If bit 0 is set to one, the phrase played will lie within the bounds of zero and the number transferred in the command byte. If bit 1 of the mode value is 0, the phrases in this range will be played in sequence, rolling back to zero after the top phrase (the one corresponding to the command byte) has been played. If bit 1 is 1, a randomly-selected one of the same range will be played.

The mode can be changed at any time, so system operation can combine the different addressing options. For example, to have ten phrases triggered by specific events and a different set of ten randomly selected by a fifth event, do the following:

- Record the phrases in order with the ten randomly selected ones as the first ten in the list.
- To randomly select one of the bottom group, first issue a set mode command specifying "random sequence" (0f3h). Repeatedly sending a command byte of "9" will produce a random selection of phrases 0-9.
- To select one of the fixed phrases, issue a set mode command for "direct" (0f0h), the send the number of the phrase to be played.
- If you desire that the phrases in the first group be played in sequence, issue a set mode "linear sequence" (0f1h) instead of 0f3h.

Recording messages:

The primary function of the QV306m4 is to manage message playback. The recording function is intended to be used only to set up the system initially. Messages must be recorded in the order they are to be selected, that is, from number 0 to n, and must all be recorded in the same session. Resetting record mode places a very short file on the ChipCorder to serve as the "last" marker. Quadravox provides the QV300S2 software and optional QV430P hardware to perform this operation, but it is simple enough to be easily integrated into your application.

To record a new set of messages:

- issue a set mode "record" command (0f4h)
- send a command byte of "1" (01h) and at the same time start applying the analog signal to be recorded to the analog-in+ pin of module. The recording level may have to be adjusted to achieve the best overall recording quality, as explained in ISD's documentation.
- When the analog segment is complete, send a command byte of "0" to stop the recording and place an EOM marker in the ChipCorder.
- When all phrases have been recorded, issue a set mode command of 0f0h.(or 0f1h or 0f3h for different playback modes)

Playing messages:

To play a message, first ensure that the mode (as explained as above) matches your intentions. Then, issue a single byte in a contiguous range from 0 to the number of recorded messages – 1 (maximum 239). When the phrase has completed, a byte with the number of the phrase spoken will be returned.

Message play can be stopped by issuing a stop command (0f6h). There is no return code for a stop command. Be aware that the ISD ChipCorder may require up to 50ms to effect the stop after it is received. Commands sent during this time may fail to execute properly.

The BUSY_ line can also be used to determine when the message is complete.

Power management:
The QV306m4 sends a power up command to the ChipCorder on startup. The QV306 and the ChipCorder will remain powered up until a set mode “sleep” command is issued. The amplifier will be muted or powered down except during playback. This muting is essential to cover address scan noise. The PWR_ pin can be used for the same function with an external amplifier. The system can be awakened by use of the RESET_ pin.

Applications:

The QV306m4 is intended for message playback applications where the user requires the large capacity of ISD’s serial interface ChipCorders®, but does not want to develop custom microprocessor algorithms to manage the devices.
The QV306m4 can also be used in conjunction with a host microprocessor in lieu of developing custom real-time code on the host platform. When used in cueing mode, the ISD chips in this series require either constant polling or an interrupt driven service routine for advancing to the proper address. If used in address mode, they require that the microprocessor maintain an address table with entries for each of the independently accessed phrases. The QV306m4 can offload these functions as well as system power management from the main processor.

Limitations:

The QV306m4 is limited to 240 phrase selections.

The minimum allowable phrase length is 200ms, which in practical terms means that the minimum valid phrase length is more than a single segment for most ISD part types. This restriction is due to the necessity of using a particularly short phrase as an end-of-project marker.

The QV306m4 uses ISD’s cueing mode for message management. Playback starting delay depends on both the position of the data in the chip and the number of messages in the project. Although the cueing mode scan runs 1600 time faster than normal chip speed, this still can amount to 150ms for a four-minute chip. To this is added the overhead of restarting the scan after each message is located. This delay complicates use of the 306 for phrase concatenation, but with care in the data preparation, some limited use is possible. Place all the frequently concatenated sounds (e.g. numbers) at the beginning of the list, followed by ending phrase segments, and finally by beginning phrase sections. This will optimize the allocation of the unavoidable delays

Simple development platform:

The QV300S2 software and QV430P programming hardware provide a low-cost method of programming the 306m4 modules. The QV300S2 software is available free from the Quadravox website. The software makes use of your PC’s sound card and serial port to control the QV306m4, which in turn invokes a cueing-mode record routine. The connections are as follows:

Serial port pins:
Ground (DB9 pin 5) connects to system ground.
TXD (DB9 pin 2) connects via the MAX232 level converter to QV305 RXD.
RXD (DB9 pin 3) connects via the MAX232 level converter to QV305 TXD.
Sound card line out, left channel, is capacitively coupled to one of the analog-in pins of the ISD chip.

Electrical Characteristics:

Absolute Maximum Ratings:

Stresses above these limits may cause permanent damage to the PIC16C505 controller device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may effect device reliability.

Ambient temperature under bias	-40°C to +125°C
Storage temperature	-65°C to +150°C
Voltage on V_{DD} with respect to Vss	0 to +7 V
Voltage on all other pins with respect to Vss	-0.6 v to (VDD + 0.6V)
Total power dissipation	700 mW
Maximum current out of Vss pin	150 mA
Maximum current into V_{DD} pin	125 mA
Input clamp current, I_{IK} ($V_I < 0$ or $V_I > V_{DD}$)	+/- 20 mA
Output clamp current, I_{OK} ($V_o < 0$ or $V_o > V_{DD}$)	+/- 20 mA
Maximum output current sunk by any output pin	25 mA
Maximum output current sourced by any input pin	25 mA

DC Characteristics: standard operating temperature $0°C \leq TA \leq +70°C$

Power supply pins:

Symbol	Description	Min	Typ(1)	Max	Uni	Conditions
V_{DD}	**Supply voltage**	3.5		5.5	V	
V_{POR}	**V_{DD} start voltage to ensure power-on reset**	--	Vss	--	V	See section on power-on reset for details
S_{VDD}	**V_{DD} rise rate to ensure power-on reset**	0.05	--	--	V	See section on power-on reset for details
I_{DD}	**Supply current(2)**	--	1.8	2.4	mA	V_{DD} = 5.0V
I_{PD}	**Power down current**	--	4	12	µA	V_{DD} = 3.5V
V_{IL}	**Input low voltage** RXD, BR0-1-	Vss	--	0.5	V	
V_{IH}	**Input high voltage** RXD, BR0-1	2.0	--	V_{DD}	V	$4.5 \leq V_{DD} \leq 5.5V$
I_{IL}	**Input leakage current(2)** RXD, BR0-1	--	--	+/- 1	µA	$Vss \leq V_{PIN} \leq V_{DD}$
V_{OL}	**Output low voltage** RECLED_, TXD	--	--	0.6	V	I_{OL}=8.5mA, I_{DD}=4.5V
V_{OH}	**Output high voltage** RECLED_, TXD	V_{DD}-0.7	--	--	V	
C_{IO}	**Capacitive loading specs on output pins**	--	--	50	pF	

Note 1: Data in the typical ("typ") column is based on characterization results at 25°C. This data is for design guidance only and is not tested.

2: Negative current is defined as coming out of the pin

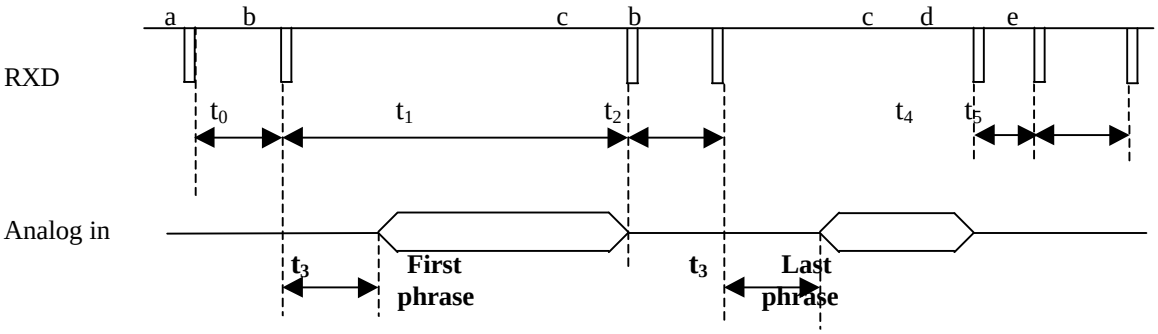
AC Characteristics: standard operating temperature 0°C ≤ TA ≤ +70°C

Symbol	Description	Min	Typ(1)	Max	Unit	Conditions
TioR	**Pin output rise time**	--	10	25(2)	ns	
TioF	**Pin output fall time**	--	10	25(2)	ns	

Note 1: Data in the typical ("typ") column is based on characterization results at 5V, 25°C. These parameters are for design guidance only and are not tested.

2: These parameters are design targets and are not tested. No Characterization data available.

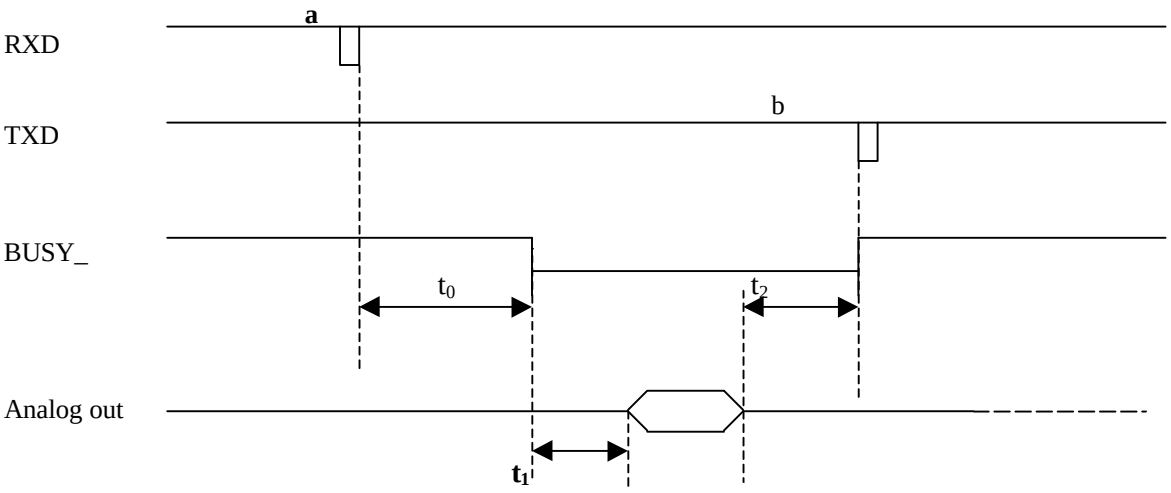
Record timing:



Timing values:

Symbol	event	min	typ	max	unit
t0	record mode start to first recording start	200			ms
t1	recording length	200		chip-400	ms
t2	spacing between subsequent recordings	50			ms
t3	start recording to analog in	15			ms
t4	end of last recording to end of recording mode	50			ms
t5	end of recording mode to next command	500			ms
a	set recording mode (F4h)				
b	start recording command (01h)				
c	stop recording command (00h)				
d	clear recording mode (F0h)				
e	any other command				

Play Timing:

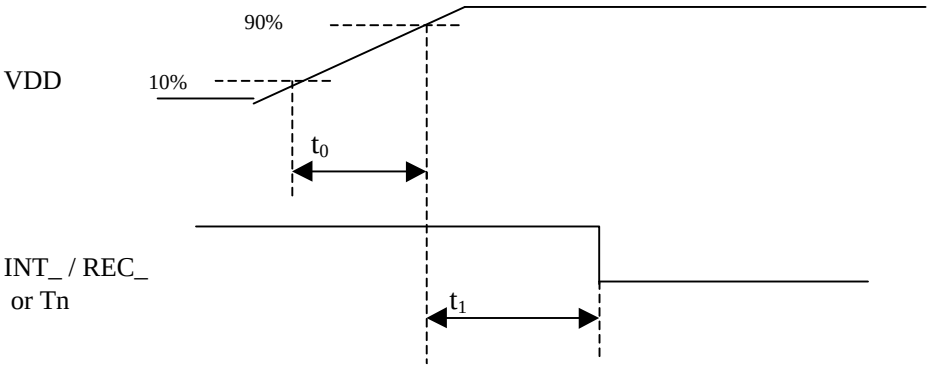


Timing values:

Symbol	event	min	typ	max	unit
t0	play command to BUSY_ active		10		us
t1	BUSY_ active to audio output		15	*	ms
t2	end of sound to BUSY_ inactive		1		ms
a	play command (00h – Efh)				
b	“done” response from QV306m4 (word value)				

*Note: the delay depends on the position of the data in the ISD chip and the number of messages in the project. For the ISD4003-04 and 20 messages, the maximum possible delay is approximately 150ms.

Power up timing:



Timing values:

Symbol	Event	min	typ	max	unit
t0	Vcc rise time			9	ms
t1	Delay to first command	500			ms

Online Support:

Quadravox maintains the latest specifications, schematic diagrams, and support software in the support section of our website, www.quadravox.com/support.htm.

For questions not answered there or for other inquiries, please write us at support@quadravox.com, or call 1-800-779-1909 from the U.S. and Canada, or 1-972-669-4002 from other countries.

Annex 1: Vocabulary list

zero
one
two
three
four
five
six
seven
eight
nine
ten
eleven
twelve
thirteen
fourteen
fifteen
sixteen
seventeen
eighteen
nineteen
twenty
thirty
forty
fifty
sixty
seventy
eighty
ninety
hundred
thousand
million
billion
two thousand
a.m.
p.m.
a
alpha
b
bravo
c
Charlie
d
delta
e
echo
f
fox
g
golf
h
hotel
i
India
j
Juliet
k
kilo
l
lima
m
mike
n
november0
o
Oscar
p
papa
q
Quebec
r
Romeo
s
sierra
t
tango
u
uniform
v
victor
w
whiskey
x
Xray
y
Yankee
z
Zulu
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
Sunday
January
February
March
April
May
June
July
August
September
October
November
December
amp
amps
Bars
bit
bits
black
blue
brown
byte
bytes
cable
Celsius
centimeter
centimeters
Cents
chip
Cubic
degrees
divided by
Dollars
down
equals
Fahrenheit
farads
feet
foot
Gallons
gigahertz
go
gold
good afternoon
good morning
goodbye
gram
grams
gray
green
hello
hertz
hour
hours
inch
inches
indoor temperature
is
key
kilobit
kilobits
kilobyte
kilohm
kilometer
kilometers
kiloohms
light
Liters
mega
megabytes
megaohms
megohms
meter
meters
micro
micron
microns
midnight
mile
miles
milli
minus
minute
minutes
noon
o clock
Of mercury
Of water
ohms
orange
Pascals
per hour
per
percent
pico
pink
plus
point
pound
pounds
p.s.i
purple
r.p.m
red
second
seconds
silver
square
start
stop
switch
tan
temperature is
the current time is
the current
the outside
the speed is
the
thee

times
up
volt
volts
white
wire
yard
yards
yellow
your speed is
the date
is more
please wait
please
is less
thank you
than
and
are closed
are down
are off
are on
are open
are up
is closed
is down
is off

Annex 2: Basic Stamp 2 sample program

```
' Quadravox Talking Module for Basic Stamp 2 (QV306M4-P)
'
'=========================================
'QV306M4-P pinout and control lines:
'
'Control lines     pin     level   Function
'RXD (5V)                  1       --              RS232 receive line
'TXD (5V)                  2       --              RS232 transmit line
'BUSY_             3       low             indicates system is busy
'BR1                       4       --              msb of baud rate selection; must be +5V or GND
'BR0                       5       --              lsb of baud rate selection; must be +5V or GND
'RECLED_                   6       low             can be used to sink current for record LED. The external
'                                                  system must provide the current-limiting resistor.
'+5V                       7       +5              system power
'GND                       8       0               system ground
'ANA_IN                    9       --              line level analog input for recording
'ANA_OUT                   10      --              line level, volume controlled, analog output
'GND                       11      0               system ground
'PWR_                      12      low             can be used to control an external power amplifier such
'                                                  as the QVamp3.
'AUX                       13      --              not used in this version.
'RESET_                    14      low             system reset
'SP-                       15      --              speaker output; bridge tied load
'SP+                       16      --              speaker output; bridge tied load
'Baud rate selection:
'BR1 BR0 Rate
'0 0 2400b                 'this is the one implemented here
'0 1 4800b
'1 0 9600b
'1 1 19200b
'Commands:
'Value             Action
'-------------------------
'0-239             Play phrase <value> according to mode settings
'240-255                   Set mode: see mode table below for functions
'
'Set mode functions:
'Value (hex)       Function
'-------------------------
'0f0h              set play mode to direct addressing
'0f6h              stop play
'0f8h              sleep
'0fch              set volume with next byte transmitted. Only values 0-31 are valid
'0fdh              software reset
'0feh              return version byte (00dh)
'0ffh              return type byte (036h)
'-------------------------
'
' Board of Education Connections
'
'  We call Q1, Q2, ... pins 1,2 ... of the QV306M4-P module:
```

```
'
' 1) Insert the QV306M4-P module at the far end of the Board of Education, component side
'          facing away from the Basic Stamp 2. Align the left most pin (Q16) of the connector with
'          the left most row of the board (P15). This way, Pin 1 (Q1) of QV306M4-P corresponds to the row
'          of P1 for the Basic Stamp.
' 2) Wire connections:
'          On the QV306M4-P side of the "Board of Education":
'          - VDD to Q7 (QV306M4-P Pin 7: Power)
'          - VSS (Ground) to Q8  (QV306M4-P  Pin 8: Ground) . Then from there to Q5 (QV306M4-P pin 5:
BR0) and Q4 (QV306M4-P  Pin 4: BR1)
'                              (to set the baudrate at 2400 bps)
'          - connect a wire to Q15 and one Q14: these two wires will go to an 8 Ohm speaker
'
'          From the Px connections, on the Stamp side:
'
'          -P1 to Q1 on the QV306M4-P side (QV306M4-P receive)
'          -P2 to Q2 on the QV306M4-P side (QV306M4-P transmit)
'          -P3 to Q3 on the QV306M4-P side (QV306M4-P busy_)
'          -P14 to Q14 on the QV306M4-P side (QV306M4-P  reset)
'
'                    Q Q Q Q Q Q Q Q Q Q Q Q Q Q Q
'                    1 1 1 1 1 1 1 9 8 7 6 5 4 3 2 1
'                    6 5 4 3 2 1 0
'       ======================================== QV306M4-P
'                    . . . . . . . . . . . . . . . .
'          V         . . . . . . . . . . . . . . . .
'          S         . . . . . . . . . . . . . . . .
'          S         . . . . . . . . . . . . . . . .
'                    . . . . . . . . . . . . . . . .
'
'
'                    . . . . . . . . . . . . . . . .
'          V         . . . . . . . . . . . . . . . .
'          D         . . . . . . . . . . . . . . . .
'          D         . . . . . . . . . . . . . . . .
'                    . . . . . . . . . . . . . . . .
'
'                    = = = = = = = = = = = = = = = = =
'                    P P P P P P P P P P P P P P P P
'                    1 1 1 1 1 1 9 8 7 6 5 4 3 2 1 0
'                    5 4 3 2 2 0
'
'
'
'                      _____________________
'          | Basic Stamp 2 |
'                     ------------------
'
'
'
'
' Vocabulary list
'
NUM0            con         0
NUM1            con         1
NUM2            con         2
```

```
NUM3            con     3
NUM4            con     4
NUM5            con     5
NUM6            con     6
NUM7            con     7
NUM8            con     8
NUM9            con     9
NUM10           con     10
NUM11           con     11
NUM12           con     12
NUM13           con     13
NUM14           con     14
NUM15           con     15
NUM16           con     16
NUM17           con     17
NUM18           con     18
NUM19           con     19
NUM20           con     20
NUM30           con     21
NUM40           con     22
NUM50           con     23
NUM60           con     24
NUM70           con     25
NUM80           con     26
NUM90           con     27
NUM100          con     28
NUM1000         con     29
million         con     30
billion  con    31
NUM2000         con     32
a_m             con     33
p_m             con     34
let_a           con     35      'this is the letter "a"
alpha           con     36
let_b           con     37
bravo           con     38
let_c           con     39
charlie  con    40
let_d           con     41
delta           con     42
let_e           con     43
echo            con     44
let_f           con     45
fox             con     46
let_g           con     47
golf            con     48
let_h           con     49
hotel           con     50
let_i           con     51
india           con     52
let_j           con     53
juliet   con    54
let_k           con     55
kilo            con     56
let_l           con     57
lima            con     58
```

```
let_m           con     59
mike            con     60
let_n           con     61
november0       con     62
let_o           con     63
oscar           con     64
let_p           con     65
papa            con     66
let_q           con     67
quebec  con     68
let_r           con     69
romeo           con     70
let_s           con     71
sierra  con     72
let_t           con     73
tango           con     74
let_u           con     75
uniform con     76
let_v           con     77
victor  con     78
let_w           con     79
whiskey con     80
let_x           con     81
xray            con     82
let_y           con     83
yankee  con     84
let_z           con     85
zulu            con     86
monday  con     87
tuesday con     88
wednesday       con     89
thursday con    90
friday  con     91
saturday con    92
sunday  con     93
january con     94
february con    95
march           con     96
april           con     97
may             con     98
june            con     99
july            con     100
august  con     101
september       con     102
october con     103
november        con     104
december        con     105
amp             con     106
amps            con     107
Bars            con     108
bit_w           con     109     'this is the word "bit"
bits            con     110
black           con     111
blue            con     112
brown           con     113
byte_w  con     114     'this is the word "byte"
```

bytes	con	115	
cable	con	116	
celsius	con	117	
centimeter	con	118	
centimeters	con	119	
Cents	con	120	
chip	con	121	
connector	con	122	
Cubic	con	123	
degrees	con	124	
divided_by	con	125	
Dollars	con	126	
down	con	127	
equals	con	128	
fahrenheit	con	129	
farads	con	130	
feet	con	131	
foot	con	132	
Gallons	con	133	
gigahertz	con	134	
go	con	135	
gold	con	136	
good_aft	con	137	'good afternoon
good_morn	con	138	'good morning
goodbye	con	139	
gram	con	140	
grams	con	141	
gray	con	142	
green	con	143	
hello	con	144	
hertz	con	145	
hour	con	146	
hours	con	147	
inch	con	148	
inches	con	149	
indoor_temp	con	150	'indoor temperature
is	con	151	
key	con	152	
kilobit	con	153	
kilobits	con	154	
kilobyte	con	155	
kilohm	con	156	
kilometer	con	157	
kilometers	con	158	
kiloohms	con	159	
light	con	160	
Liters	con	161	
mega	con	162	
megabytes	con	163	
megaohms	con	164	
megohms	con	165	
meter	con	166	
meters	con	167	
micro	con	168	
micron	con	169	
microns	con	170	

```
midnight        con     171
mile            con     172
miles           con     173
milli           con     174
minus           con     175
minute  con     176
minutes con     177
noon            con     178
o_clock con     179
Ofmerc  con     180
Ofwater con     181
ohms            con     182
orange  con     183
Pascals con     184
per_hour        con     185
per             con     186
percent con     187
pico            con     188
pink            con     189
plus            con     190
point           con     191
pound           con     192
pounds  con     193
Psi             con     194
purple  con     195
r_p_m           con     196
red             con     197
second  con     198
seconds con     199
silver  con     200
Square  con     201
start           con     202
stop_w  con     203     'this is the word "stop"
switch  con     204
tan             con     205
temp_is con     206     'temperature is
t_curr_t_is     con     207     'the current time is
the_current     con     208
the_outside     con     209
t_speed_is      con     210     'the speed is
the             con     211
thee            con     212
times           con     213
up              con     214
volt            con     215
volts           con     216
white           con     217
wire            con     218
yard            con     219
yards           con     220
yellow  con     221
y_speed_is      con     222     'your speed is
the_date con    223
is_more con     224
please_wait     con     225
please  con     226
```

```
is_less   con      227
thank_you          con      228
than               con      229
and_w              con      230      'this is the word "and"
are_closed         con      231
are_down           con      232
are_off   con      233
are_on    con      234
are_open           con235
are_up    con      236
is_closed          con237
is_down con        238
is_off    con      239
'
' Aliases for I/O definitions
'
RECEIVE_LINE con      1
TRANSMIT_LINE      con      2
NBAUD2400          con 396
'=========================================
' I/O Definitions
'NBAUD2400
out14  = 0
output    14       'reset line
input     3        'busy line
out1      = 1                              'WARNING if RECEIVE_LINE is not 1, this line must be
changed accordingly!!!!
output    RECEIVE_LINE 'receive
input     TRANSMIT_LINE        'transmit
'=========================================
' Aliases for QV commands
'
QV_DIRECT      con      $F0
QV_STOP        con      $F6
QV_SLEEP       con      $F8
QV_VOLUME      con      $FC
QV_RESET       con      $FD
QV_REVISION    con      $FE
QV_TYPE        con      $FF
'=========================================
' variables for say_number program
'
znumber            var      word
zdigit    var      word
ztmp               var      word
ztmp1              var      word
'
' variable for say_date program
'
zcentury var       bit      ' 0 for 19xx, 1 for 20xx
zyear              var      byte     ' number from 0 to 99
zmonth    var      nib      ' 1 for January, 2 for February, etc...
zday               var      byte     ' day of the month: 1 through 31
zdayweek           var      nib      ' day of the week: 1 for Monday, 2 for Tuesday,...
'
' variables for say_time program
```

```
'
zam_pm var       bit       ' 0 for a.m. 1 for p.m.
zhour            var       nib       ' hour from 1 to 12
zmin             var       byte      ' minute: from 0 to 59
'=========================================
' Program start
'=========================================
gosub DoReset    'Reset the QV Module
'-------
' Test say_number
'-------
znumber =65530 'change value of number here to hear other numbers
gosub say_number  'module says" sixy five thousand five hundred thirty"
pause 2000
'-------
gosub getQvRevision
'debug HEX? ztmp          'uncomment this line to see Revision Number in Debug Window
gosub getQvType
'debug HEX? ztmp          'uncomment this line to see Type Number in Debug Window
'-------
ztmp=1
loopVolume:
if ztmp> 31 then endLoopVolume
gosub QVSetVolume
serout RECEIVE_LINE,NBAUD2400,[switch]        ' Module says "switch" 32 times from lowest to
highest volume
gosub WaitNotBusy
ztmp =ztmp + 1
goto loopVolume
endLoopVolume:
'-------
' Test say_date
'-------
zcentury=1
zyear =1
zmonth=1
zday=19
zdayweek=5
'
' the date defined here is Friday ,January 19, 2001
'
gosub say_date    'Module says "Friday ,January 19, 2001"
pause 2000
'-------
' Test say_time
'-------
zhour=11
zmin = 52
zam_pm= 1
'
' the time defined here is 11:00 pm
'
gosub say_time    'Module says" the current time is eleven fifty two, p.m."

end_program:      'Program stops here
end
```

```
'
'
'
'****************
' subroutines  *
'****************
say_number:
'
' subroutine to speak the number contained in the variable znumber
' the maximum  value of this variable is 65535
'
            ztmp = znumber
            if ztmp >0 then strictly_pos
            serout RECEIVE_LINE,NBAUD2400,[num0]
            gosub WaitNotBusy
            return
strictly_pos:
            zdigit = ztmp /1000
            if(zdigit = 0) then next_test100
            gosub say_2dnumber
            serout RECEIVE_LINE,NBAUD2400,[NUM1000]    'say "thousand"
            gosub WaitNotBusy
next_test100:
            ztmp=ztmp - (zdigit * 1000)
            zdigit =ztmp /100' the hundreds
            if zdigit = 0 then next_test_10
            serout RECEIVE_LINE,NBAUD2400,[zdigit]
            gosub WaitNotBusy
            serout RECEIVE_LINE,NBAUD2400,[NUM100]     'say "hundred"
            gosub WaitNotBusy
next_test_10:
            zdigit = ztmp - (zdigit * 100)
            gosub say_2dnumber        'only a 2 digit number left
            return
'***********
say_2dnumber
'
' subroutine to speak a non zero 2 digit number contained in zdigit
'
            if zdigit >20 then more_than_20
            if zdigit =0 then ret_say_2dnumber
            serout RECEIVE_LINE,NBAUD2400,[zdigit]
            gosub WaitNotBusy
            return
more_than_20:
            ztmp1= zdigit / 10            'tens
            serout RECEIVE_LINE,NBAUD2400,[ztmp1 +num18]
            gosub WaitNotBusy
            ztmp1 = zdigit - (ztmp1 * 10)
            if(ztmp1 = 0) then ret_say_2dnumber
            serout RECEIVE_LINE,NBAUD2400,[ztmp1]
            gosub WaitNotBusy
ret_say_2dnumber:
            return
'***********
say_date:
```

```
        if zdayweek > 7 then skip_day_week
        serout RECEIVE_LINE,NBAUD2400,[monday + zdayweek -1]
        gosub WaitNotBusy
skip_day_week
        if zmonth > 12 then skip_month
        serout RECEIVE_LINE,NBAUD2400,[January + zmonth -1]
        gosub WaitNotBusy
skip_month:
        if(zday > 31) then skip_day
        if( zday =0) then skip_day
skip_day:
        zdigit=zday
        gosub say_2dnumber
if(zcentury = 1) then say_2000
        serout RECEIVE_LINE,NBAUD2400,[NUM19]
        gosub WaitNotBusy
        goto say_year_digits
say_2000:
        serout RECEIVE_LINE,NBAUD2400,[NUM2000]
        gosub WaitNotBusy

say_year_digits:
        if(zyear =0) then skip_year
        zdigit=zyear
        gosub say_2dnumber
skip_year:
return

'**********
say_time:
'
' subroutine to say time in 12 hour format (plus a.m. or p.m)
'
' the hour is contained in variable zhour (1-12)
' the minutes are in variable zmin (0-59)
' the variable zam_pm (1 bit) contains a.m. (0) or p.m. (1)
'
if(zhour =0) then skip_time         'hour invalid, don't talk
if(zhour > 12) then skip_time       'hour invalid, don't talk
if(zmin > 59) then skip_time        'minutes invalid, don't talk
serout RECEIVE_LINE,NBAUD2400,[t_curr_t_is]   'say: the current time is
gosub WaitNotBusy
zdigit=zhour
gosub say_2dnumber
zdigit=zmin
'
' several cases
'
if(zmin=0) then say_am_pm           ' don't say "zero" for minutes
if(zmin>=10) then say_minutes
say_oh:
'
' here, single digit minutes, we say "oh" first
'
serout RECEIVE_LINE,NBAUD2400,[let_o]
gosub WaitNotBusy
```

```
say_minutes:
gosub say_2dnumber
say_am_pm:
if(zam_pm =0) then say_am
say_pm:
        serout RECEIVE_LINE,NBAUD2400,[p_m]
        gosub WaitNotBusy
        goto skip_time
say_am:
        serout RECEIVE_LINE,NBAUD2400,[a_m]
        gosub WaitNotBusy
        goto skip_time
skip_time:
return
'**********
WaitNotBusy:
'
' wait till the QV Module is finished talking
' at which point IN3 = 1
'
hold:   if IN3 = 0 then hold
        return
'**********
DoReset:
out14 = 0           'do a reset
pause 100           '
out14 = 1
pause 2000          'wait while the QV module counts its phrases
return
'**********
GetQVRevision:
'
' queries revision number of QV Module
' stores it in variable ztmp
'
serout RECEIVE_LINE,NBAUD2400,[QV_REVISION]
serin TRANSMIT_LINE,NBAUD2400,[ztmp]
return
'**********
GetQVType:
'
' queries type of QV Module
' stores it in variable ztmp
'
serout RECEIVE_LINE,NBAUD2400,[QV_TYPE]
serin TRANSMIT_LINE,NBAUD2400,[ztmp]
return
'**********
QVSetVolume:
'
' sets volume of QV module with value (0--31) stored in ztmp
'
if ztmp <32 then volume_OK
ztmp=31
volume_OK
serout RECEIVE_LINE,NBAUD2400,[QV_VOLUME]
```

```
serout RECEIVE_LINE,NBAUD2400,[ztmp]
return
```