G29
Programmable Reference Voltage

1. FUNCTION

1.1. Datasheet

1.1.1. Application

Sequential generation of reference voltages

1.1.2. Data

Parameter	Wert
Memory	32kWords
Amplitude	0..10V
Resolution	12bit
Clockrate	..1Mhz
Input Signal Pulswidth	>30ns

1.1.3. Features

Internal and external clock.

Single- and continous function.

1.1.4. Setup

2/12 NIM-Case.

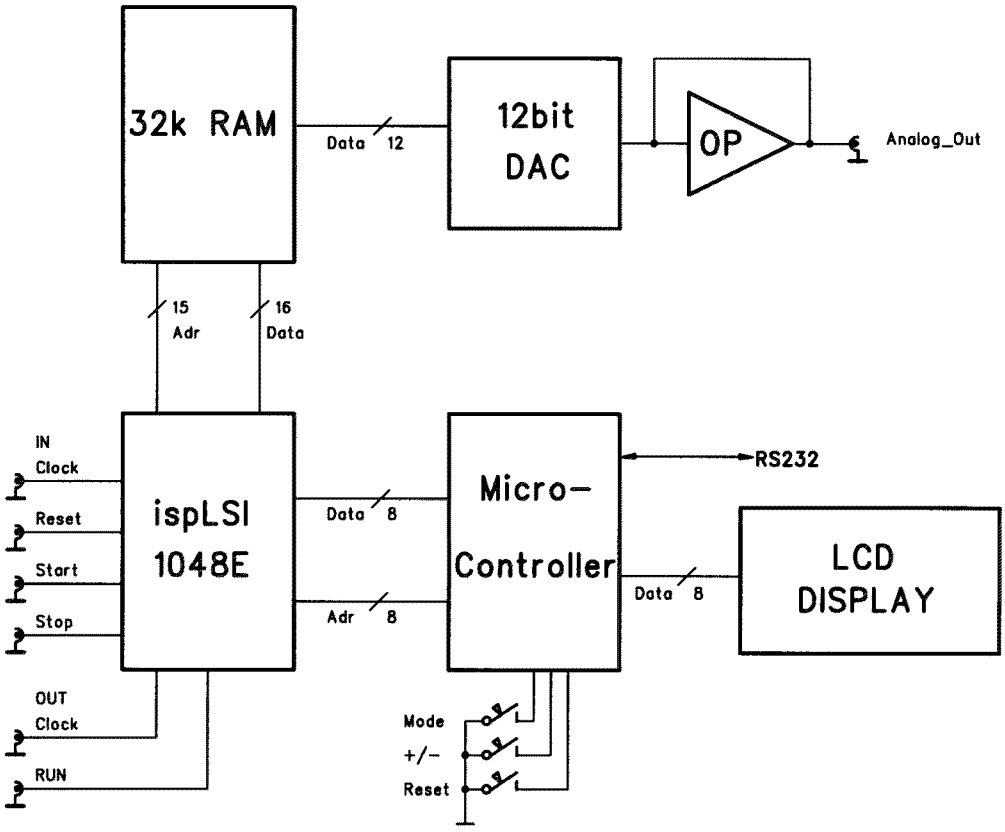
2-Line LCD – Display.

MP35 Controller.

1.1.5. Power

Voltage	Current	Power
+6V	500 mA	3,0 W
-6V	250 mA	1,5 W
+24V	30 mA	0,72 W
-24V	30 mA	0,72 W
Total		5,94W

1.2. Blockdiagram



1.3. Function

The application in G-2 requires the following operation:

The memory (address= 0..) has to be filled with appropriate values to generate different voltages in a sequencial manner. The last memory address has to be marked (R) for automatic restart at the sequence at address 0. You can do this either with the serial commands or (somehow cumbersome) also with the front buttons.

An external clock at input at **IN1 CLOCK** determines how the sequence will be stepped through. Please note that the clock mode has to be set for external clock (front buttons or RS232 command) and that the module has to be started once (input **IN3 START** or front buttons or RS232 command!).

The external signal **IN2 RESET** can be used to synchronize the sequence to a given start condition (address counter will be simply resetted to zero).

The different reference voltages appear stepwise according to the clock at the output connector ANALOG-OUT.

2. OPERATION

2.1. Configuration

2.1.1. Jumper

Input and Output signals can be choosen with jumpers for NIM- or TTL levels. The analog output voltage can be set to the following ranges:

Range	Jumper	State	Jumper	State
0..10V	J20	short	J21	open
+10V..-10V	J20	open	J21	short
+5V..-5V	J20	short	J21	short

2.2. Connectors

2.2.1. IN1 CLOCK (NIM or TTL, positive edge triggered)

Input for external clock.

2.2.2. IN2 RESET (NIM or TTL high active)

Adresscounter will be set to zero (0). Sequence restarts at address 0!

2.2.3. IN3 START (NIM or TTL high active)

Module is set to Run mode. Clock is active.

2.2.4. IN4 STOP (NIM or TTL high active)

Module is set to Stop mode. Clock is not active.

2.2.5. OUT1 CLOCK (NIM or TTL)

Output of selected internal clock.

2.2.6. OUT2 RUN (NIM or TTL)

Indicates running state of module.

2.2.7. ANALOG-OUT (analog, 500 Ohm impedance)

Digital to analog converter output in the selected range.

2.3. Front

After powerup (or Reset) the display shows the module type (G29) and the version of software. After a few seconds the serial number and modul number (RS232-bus) will come up and a little later it shows the general meaning of the push buttons on each side. Finally the module will stay in the RUN/RESET mode.

The button **MODE** (on the left side of the LCD) toggles through the following modi:

CLOCK	The buttons + and - (on the right side) increase or decrease the internal clock frequency or select **external** clock mode.
FORMAT	The buttons + and – switch to binary or mV display.
DAC	The buttons + and – increase or decrease the voltage at the analog output (This does not affect the current memory value).

A:	The buttons + and – increase or decrease the data value in the current address.
	The button MODE and simultanously the buttons + and – increase or decrease the current address.
	Press both buttons + and – simultanously to set the RUN markers in the current memory address to either „R=Restart“, „S=Stop, „Z=Stop & Reset“ or „ =no marker“ (see also command „O“!).
RUN…	The button + starts the sequence at the current address.
	The button – resets to address 0 and enters mode A:.
	In case of STOP the button + stops any sequence!

An asterisk (*) in the display shows that the module is running in a sequence!

2.4. Backside RS232

2.4.1. Overview

Communication is 9600 baud, 8 bits, no parity.

At any time the command „?“ sends an overview of all available commands:

```
?               Help (this screen!)
! n             Attention Module
# n             Set Module Nr
D p,text<cr>    Display text at postion p (0=unlock)
d               Get Keys
K/k             Key LOCK/UNLOCK
F/f             Format mV/bin
A n/a           Address (n=0..32767) set/get
M d/m           Memory set/get
N d/n           Memory set/get Autoincrement
O d/o           Overrun set/get (1:Restart; 2:Stop; 3:both)
C n/c           Clock (0=Ext; 1MHz/n) set/get
R a/r           Run with start at a/Stop
L s,d/l s       Load/list Memory size (0=all) with d
P s,d           Ramp period per size with amplitude d
S s,d           Sine period per size with amplitude d
X d             DAC set
^ code          Save setup in flash
```

2.4.2. Commands

?	Short list of all available commands.
!n	If using the module at the RS323-Bus, a specific module has to be selected with this command and a module number (e.g. „!9“).
#n	Set number (selected module) for communication.
Dp,txt	show text on display.
d	send status of front buttons.

	(0 = none, 1 = Mode, 2 = +, 4 = -)
K	lock front buttons.
k	unlock front buttons.
F/f	select binary or „mV“ format.
A n	set current address counter to n
a	send current address counter value.
M d	set memory value at current address.
m	send memory value at current address.
N d	set memory value and autoincrement address counter.
n	send memory value and autoincrement address counter.
O d	set marker at current address for sequence control:
	0: none
	1:Restart: continues with next clock at address 0.
	2:Stop: put module into stop mode
	3:Stop & Reset: module stops and the addresscounter=0.
o	send marker at current address. (R=1, S=2, Z=3)
C n	select clock frequency; n=divider for 1MHz
	e.g. „C100“ = 1Mhz/100 = 10kHz) ; „C0“ selects external clock.
c	send divider value.
Ra	start sequence at address a
	e.g.: „R return“ starts at address=0.
	„R100 return“ starts at address=100.
r	stops sequence.
L s,d	set memory value to d in the following s addresses
	e.g. „L100,23“ sets memory value 23 into the next 100 addresses.
l s	send values of the next s addresses.
P s,d	put a ramp with amplitude d into next s addresses.
S s,d	put a sine wave with amplitude d into the next s addresses.
X d	load DAC with the value d.