2000/04
Sonar Ranging

Holger Zahnleiter
holger@zahnleiter.org

Date: 07.04.03

Abstract

This project describes how to reuse ultrasonic transducers and electronics taken from a Polaroid 670AF instant camera as sonar distance ranging sensors. The documentation also describes the difference between the board taken from the camera and the bord which one may by at robot supply stores (Polaroid series 6500 module). In this example the ranging module is controlled by a MCU compatible to the MCS-51 series. The circuit described herein could be connected to other MCUs/CPUs as well.

1 Project characteristics

Project code	2000/04
Project name	Sonar Ranging
Started	16.11.00
Ended	not yet
Used tools and libraries	EAGLE Layout Editor 3.55r3.
Used for other projects or applications	#1999/01

2 Differences between 670AF and 6500 Modules

As mentioned above this interface circuit drives a module taken from a Polaroid 670AF instant camera. One may find lots of pages on the internet dedicated to the series 6500 module. Both modules look very similar. But the electrical behaviour is different. The difference lies in the ECHO output. Once raised to a logic high it never goes down even though if one resets INIT to logic low again. This may result from the fact that an instant camera is always switched off. It is switched on only if you pull the trigger to take a picture. After that the trigger is released and the camera switched off again. So the ranging module is reset, too.

The solution was to switch off the module after the messurement was done. Because the ECHO output has to be pulled up the module is always supplied with power. So the ground (GND) is disconnected. While fireing the module consumes a current of 2 to 2.5[A]. So I took a BUZ71 MOSFET for the switching. The board has an extra power supply input for the sensors. My robot (see project #1999/01) is equipped with two power supplys (5V/2A). The additional power input is connected to the second power supply to separately supply microcontroller electronics and sensor electronics. As to be read on documents I found on the internet controller resets may result from the fireing of the sensors when not applying an additional capacitor as a buffer. In my robot no resets do occur due to fireing. This may result from the capacitor and the separate power supply, too.

Another difference is that both modules do have different connectors. The series 6500 module has nine pins where the 670AF module has only eight pins. This is illustrated in the picture below (fig. 1).

Common to both modules is that the ECHO output is an open collector which has to be pulled up by applying a $4k7\Omega$ resistor.

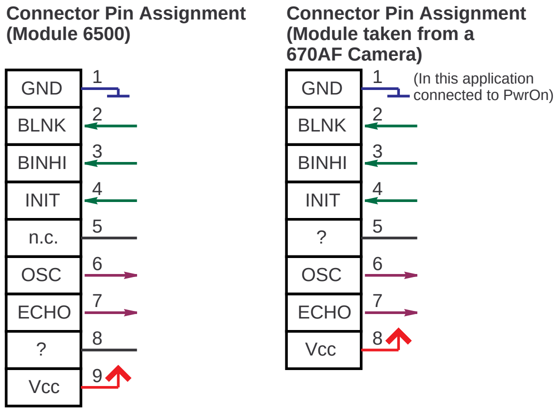

Figure 1: Connectors of the 670AF and Series 6500 Modules

3 Signals and Timing Diagram

Everything described below was taken from the Polaroid series 6500 datasheet or from homepages dedicated to that module respectively. Please download the document from my page and see the other pages for additional and more detailed information.

As mentioned above the Polaroid module taken from the 670AF camera has eight pins. With the `INIT` signal one triggers the messurement procedure. The module then sends out 16 pulses of 50kHz. When the reflected wave returns it is detected and the `ECHO` signal goes high.

To avoid detecting the transmission itself the internal blanking is aktivated for a short period of time (2.38ms) after the `INIT` signal went high. The echo detection is supressed while the `INTERNAL BLANKING` is high. This sets the minimum distance which could be messured by the module to 40cm approximately. One could inhibit the internal blanking by setting the `BINHI` signal (blanking inhibit). Doing so decreases the minimum distance which could be messured. It also increases the danger of false readings. The internal blanking should take 0.9ms at least. That means that `BINHI` should go high at least 0.9ms after `INIT` went high.

In my application either `BINHI` nor `BLNK` is used. They are pulled low internally. So I left the pins unconnected. The only signals used by my interface circuit are the `INIT` input and the `ECHO` output. $\overline{\texttt{PwrOn}}$ and $\overline{\texttt{Enable}}$ were chosen active low because this pins are high after MCU reset. I do not want the sensors to be activated without intend.

Figure 2 does not show the Polaroid signals only. The signals introduced by my interface circuit are included, too.

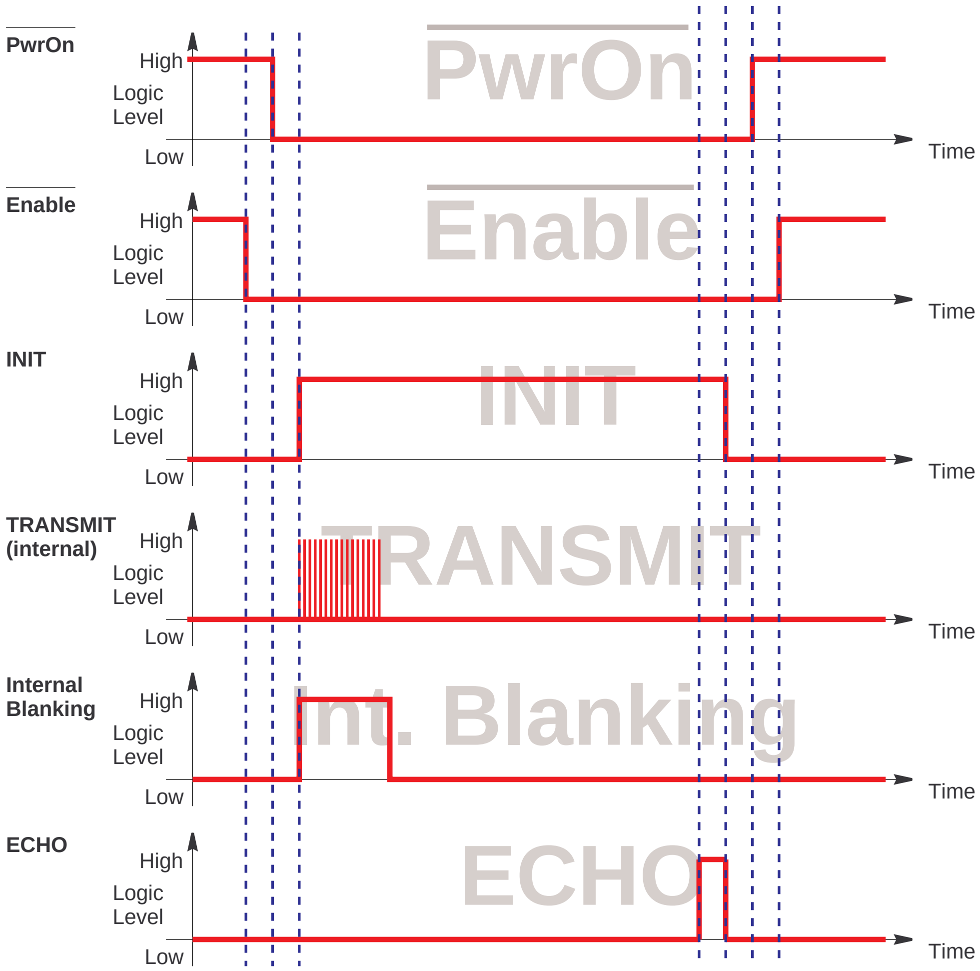

Figure 2: Timing Diagram

4 AT89S8252 Controller Board

I reuse an old microcontroller board[1] equipped with an ATMEL AT89S8252 MCU and an Intel 82527 CAN bus controller. The interface circuit and the controller board are mounted as a sandwich. The bord is connected with the interface via three ribbon cables. One supplies the board with power. The second connects the CAN bus pins with the VG [2] connector located on the interface board. The third cable connects the additional digital I/O ports of the '527 with the interface. The interface is controlled by the ports of the '527.

The figure 3 shows the connector where all four sonar ranging modules are to be connected to. There is a 20 wire ribbon cable with a plug which fits into the connector. The other end of the cable is split. Four connectors are soldered there to connect to the 670AF board (see figure above).

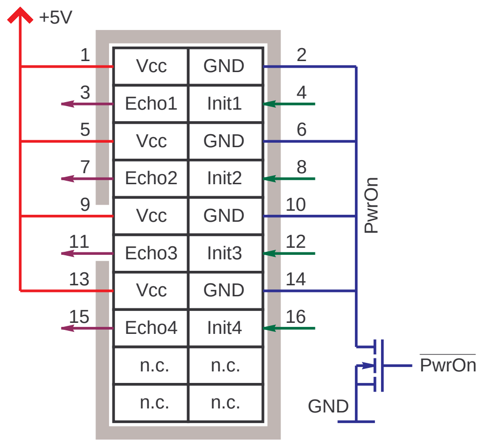

Figure 3: Polaroid Sensor Phalanx Connector

[1]A big thanks goes out to Alexander Wiedekind who gave me that board.
[2]See project #2000/01.

5 The Interface Board and it's Environment

The following picture (fig. 4) shows a schematic image of the board and it's connectors. It also shows four sensors connected to the interface.

Even though this project is about interfacing 670AF modules to any MCU/CPU in general the adaptions made to allow reusage of the old microcontroller board are part of the wiring diagram which one will find on the download section of my homepage.

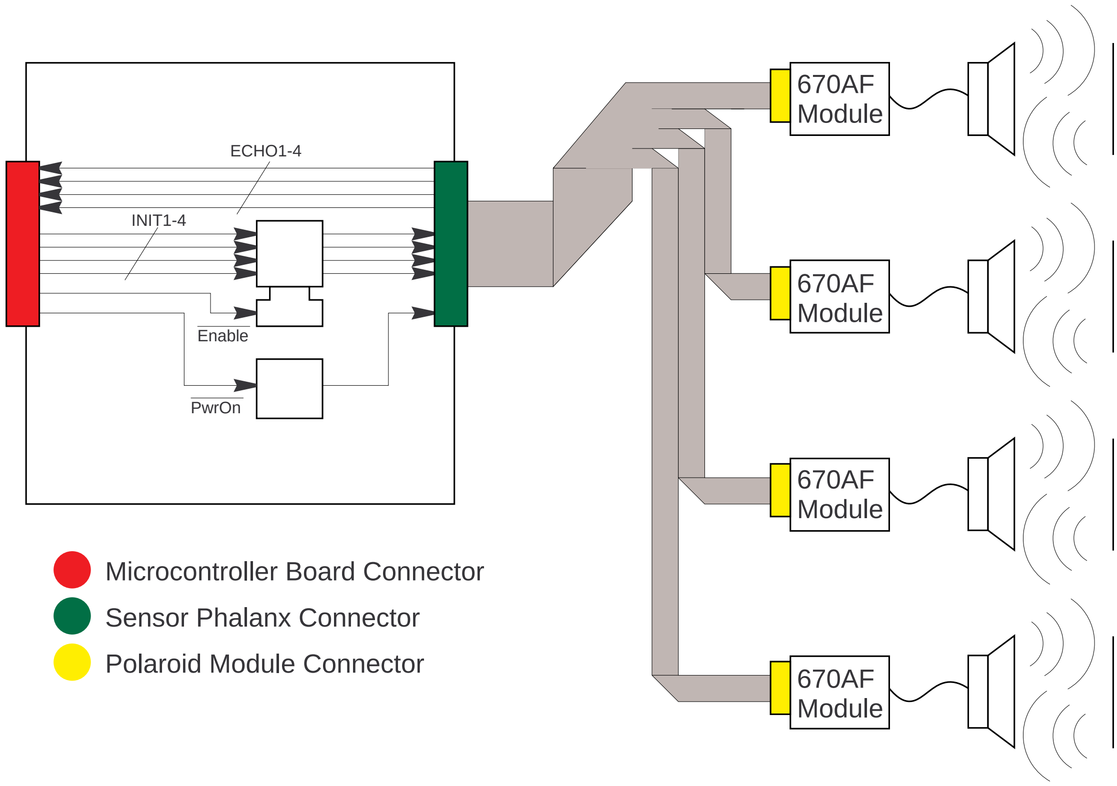

Figure 4: Sensors connected to the Interface

6 Required Hardware Ressources

The following AT89S8252 and i82527 ressources are used.

Ressource	**Purpose**
Timer #0	Generate a general delay time scalable in steps og 1ms form 1ms up to 65.335s.
Timer #1	Messurement of the running length of the sonar ping.
Timer #2	Baud rate generator. Debug info might be printed to the RS232 interface if desired.
RS232	Debug info output.
Interrupt #1	Overflow interrupt of Timer #0.
Interrupt #3	Overflow of Timer #1. Setzt the time out flag if running length of sonar ping succeeds the limit.
P2.0	Indicator output. Connected to a LED which blinks if program is running fine.

Table 2: AT89S8252 Ressources

Ressource	Purpose
Port #1	Used to switch on/off the sonar ranging phalanx and to trigger messurements.
Port #2	Used to detect the echo output from the Polaroid module.

Table 3: i82527 Ressources

6.1 AT89S8252 Port Assignment

Port P2.0 is connected to a LED (D2). The program lets the LED blink when everything is fine. A LED switched on/off permanently might indicate an error or instable condition.

6.2 i82527 Port Assignment

Port	'527 Pin	Connector Pin	Purpose	Signal Name
`P1.0`	32	`SV1.20`	When set to one (high) switches the power off for the sensor modules. Switches power on when set to zero (low).	$\overline{\texttt{PwrOn}}$
`P1.1`	31	`SV1.19`	Disconnects the `INIT` inputs from the modules from the '527 when set to port P1.2 up to P1.5 when set to one (high). Connects them when set to zero (low).	$\overline{\texttt{Enable}}$
`P1.2`	30	`SV1.18`	Tells module #1 to start it's messurement procedure when set to one (high)	`Init1`
`P1.3`	29	`SV1.17`	Tells module #2 to start it's messurement procedure when set to one (high)	`Init2`
`P1.4`	28	`SV1.16`	Tells module #3 to start it's messurement procedure when set to one (high)	`Init3`
`P1.5`	27	`SV1.15`	Tells module #4 to start it's messurement procedure when set to one (high)	`Init4`
`P2.0`	11	`SV1.12`	Input for module #1 echo.	`Echo1`
`P2.1`	10	`SV1.11`	Input for module #2 echo.	`Echo2`
`P2.2`	9	`SV1.10`	Input for module #3 echo.	`Echo3`
`P2.3`	8	`SV1.9`	Input for module #4 echo.	`Echo4`

Table 4: i82527 Port Pin Assignment

7 Ulrasonic Test - a first Test Program

In the download section one will find a first test program (`ustest.c` and `ustest.hex`). It was written for a 80C517/537 microcontroller. This controller (see projects #1999/04 and #1999/05) is integrated into my robot (#1999/01) already and is able to control the robot's motors. Like mentioned above it is easy to connect the circuit described herein to any MCU/CPU. I used a ribbon cable to connect the circuit with port P4 of the 80C537.

The testprogram implements the behavior described below. It is a small test which has to show if the sensors and the interface circuit are working as expected.

Behaviour:

1. Initialize the hardware and switch off all motors. Wait 4 seconds so that the user is able to use the security switch to supply the motors with power.

2. Drive forward (set direction forward, accelerate).

3. If the left sensor detects an obstacle closer than a certain limit, then the right wheel has to turn slower. Doing so causes the robot to turn right avoiding a collision on the left side. If there is no obstacle anymore go to 2.

4. If the right sensor detects an obstacle, then speed down the left motor for the same reason. If there is no obstacle anymore go to 2.

5. If both sensord detect an obstacle slow down. Turn around until no more obstacles are detected within the given perimeter. Go on with 2.

8 The Program

XXX Still to be written (cooming soon).

8.1 Initializing the Controller

8.2 Initializing the Interface Circuit

A Parts list

Name	Part/value	Comments
	Prototype Board	160 x 100 mm
C1	100μF	Capacitor
C2	100nF	Capacitor
C3	100nF	Capacitor
C4	100nF	Capacitor
D1	LED	red (power indicator)
D2	LED	green (is blinking when OK)
IC1	74HCT244	Octal Line Driver
IC2	74LS04	Hex Inverter
IC3	74HCT244	Octal Line Driver
OK1	CNY17	Optocoupler
Q1	BUZ71	MOS FET
R1	100Ω	Resistor
R2	100kΩ	Resistor
R3	150Ω	Resistor
R4	4k7Ω	Resistor
R5	4k7Ω	Resistor
R6	4k7Ω	Resistor
R7	4k7Ω	Resistor
R8	150Ω	Resistor
R9	150Ω	Resistor
ST1	VG-64 Jack	Backplane Connector (male)
SV1	Jack	2x10 Pins (82527 Ports, male)
SV2	Jack	2x10 Pins (Sensor Phalanx Connector, male)
SV3	Jack	2x5 Pins (ATMEL Power Connector, male)
SV4	Jack	1x3 Pins (ATMEL CAN Bus Connector, male)
SV5	Connector	1x3 Pins (ATMEL CAN Bus Connector, female)
X1	Jack	Sub-D 9 CAN BUS Connector

Table 5: Parts list

B List of project files

File Name	Comments
`src\ustest.C`	First Test Program (for 80C527 MCU).
`src\USTEST.HEX`	First Test Program (for 80C527 MCU).

Table 6: Project files

C Suppliers

[S1] **Acroname**

www.acroname.com/

Electronic parts, robot parts, sensors, microcontrollers, ...

[S2] **Arrick Roboticks**

www.robotics.com

Electronic parts, robot parts, sensors, microcontrollers, ...

[S3] **MESSRING Systembau MSG GmbH**
Am Haag 9
82166 Gräfeling
Germany
www.messring.de/

Automation products, Polaroid ultrasonic components.

D Related web pages

[W1] **Acroname:Polaroid Sonar Ranging Primer**
www.acroname.com/robotics/info/articles/sonar/sonar.html

Good page about the Polaroid Series 6500 Sonar Ranging Module and the Polaroid Transducers (Series 7000, Instrument Grade, Series 9000).

[W2] **Arrick Robotics: Polaroid Sonar Application Note**
www.robotics.com/arobot/sonar.html

A BASIC Stamp II project using the Polaroid Series 6500 Ranging Module and Instrument Grade Transducer.

[W3] **Notes on Polaroid 6500 System**
www.bigblackbook.uk.com/jim/usi_project/pol_note.htm

Brief description of the Series 6500 Sonar Ranging Module by Polaroid.

E Related literature

[B1] **Joseph L. Jones, Anita M. Flynn**:
Mobile Roboter: Von der Idee zur Implementierung
Addison-Wesley (1996), ISBN 3-89319-855-5

Tips and tricks for hobbyists.

[B2] **Texas Instruments**:
The TTL Data Book Volume 1
Texas Instruments (1985), ISBN 3-88078-078-1

Describes almost every Standard, Low-Power Schottky and Schottky TTL. Pinout, schematics, function tables, physical/electrical characteristics. INDISPENSABLE!

Disclaimer