

[Home](#)  
[Information](#)

[News](#)

Products

[CANDIP/AVR1](#)  
[CANDIP/AVR1M](#)

[CANDIP/M162](#)

[Features](#)  
[Schematics](#)  
[Activity board](#)  
[Starter kits](#)  
Software

[CANDIP/505](#)

[Download](#)

[Projects](#)

[Order](#)

[Dealers](#)

[Links](#)

## CANDIP Software

### WiCAN Layer 2 CANlib for ImageCraft ICCAVR / CodeVision AVR and SJA1000

ICCAVR version works with **CANDIP/AVR1, CANDIP/AVR1M & CANDIP/M162**

CodeVision AVR version works only sofar with **CANDIP/AVR1M**

**NEW** Current version is 1.0.7, dated the 1:st of July 2003. See [History Log](#).

**WiCAN** CANlib is an easy way of getting up running quickly and you do not need a great knowledge in how CAN works. WiCAN is 2.0B compatible and supports both the Standard 11bit Identifiers as well as the Extended 29bit Identifiers in Philips PeliCAN mode, RTR frames, Filtering, easy initialization of the CAN controller with standard bitrates as well as user defined, FIFO queues for receiving and transmitting CAN frames trough interrupt driven control, status information etc. The LIB is optimized to consume less and save the resources in the AVR, so there is plenty of room for the actual application both in RAM and FLASH. Currently the WiCAN LIB takes less than 2k of the total 8K/16K code space (AT90S8515, ATmega161 or ATmega162) together with the demo application (tested in ICCAVR 6.28) and RAM depends on how many frames you have in the FIFO queues. So you have about 6K/14K code (FLASH) for your own application (depending on which MCU you have)

There are 3 versions of the WiCAN CANlib for ICCAVR and CV. A free version (demo, see below) with fixed 20kbit bus speed and FIFO queues are preset to 4 CAN frames. With this version it is possible to test the CANlib and to see if it fits your needs. Then next version (CLIB/1) is in Object form (LIB) and cannot be altered or adjusted, but there is no limits in the speed and FIFO's is set to handle 8 messages. The last version (CLIB/2) is with full documented source code and here is possibilities for you to modify or add functions to suit your need. The source code is treated as a site licence and you may not distribute or sell the source code further on. Contact us for more information regarding this.

**CLIB/1** is on Object form and can not be changed. Both FIFO queues are preset to 4 CAN frames. This is an inexpensive way of testing the CANlib with full functionality and speed. All you need to do is to include the LIB in the project and add a header file where all CANlib defenitions are done. Sample programs are included to show how it works. We can of course change the size of the FIFO queues to suit your needs.

**CLIB/2** is in fully documented C source code and be changed to suit your need. Both FIFO queues can be set to suit your needs but only in size of 1,2,4,8,16... to maximum RAM of the AVR. All you need to do is to include the WiCAN.C and WiCAN.H file to your project and you are up running. Sample prgrams are included to show how it works.

[Click here](#) for a simple C source code to demonstrate how to setup the CAN controller, send some frames and receive some frames (both standard 11bit ID's, 29bit ID's and RTR frames, detect the received frames and then transmit them back on the bus.

The WiCAN demo for AT90S8515 and ICCVAR can be [downloaded here](#).

The WiCAN demo for ATmega161 and ICCAVR can be [downloaded here](#).

The WiCAN demo for ATmega162 and ICCAVR can be [downloaded here](#). **NEW**

The WiCAN demo for ATmega161 and CVAVR can be [downloaded here](#).

The WiCAN [PDF manual](#) can be downloaded here.

The WiCAN CANlib works together with ImageCraft ICC compiler 6.28 or later, CVAVR compiler 1.23.6 or later together with Atmel AVR AT90S8515 (only ICCAVR), ATmega161, ATmega162, ATMega128 (only Source code version) and simular AVRs together with the Philips SJA1000 CAN

controller. Source version can also easily be changed to fit IAR.

---

Simple C code for getting started with ATmega161 & SJA1000 can be [downloaded here](#).

**Note!** This code has nothing to do with WiCAN, this is just a simple quick and dirty code for those who want to write their own drivers and just a quick start on how to read/write to the SJA1000 registers using the BASIC CAN mode of SJA1000. Full source code is included, but no support is given on this code.

---

## CANDIP/M162

| [Features](#) | [Schematics](#) | [Activity board](#) | [Starter kits](#) | Software |

---

| [Home](#) | [Information](#) | [News](#) | [Products](#) | [Download](#) | [Projects](#) | [Order](#) | [Dealers](#) | [Links](#) |

---

© Copyright 2000-2003 [LAWICEL](#). All rights reserved. Images and text used on these pages may not be copied without written approval from LAWICEL. Information in this document is subject to change without notice. Other products and companies referred to herein are trademarks or registered trademarks of their respective companies or mark holders.

---

Contact [webmaster@candip.com](mailto:webmaster@candip.com) for reporting errors on these pages.

These pages where last updated 28/08/2003