UNIFIED COMMUNICATION PROTOCOLS INTERFACE
OVER INMARSAT NETWORK AND INTERNET

Sorin-Robertino SINTEA, Nicolae BORDEA, Vasilica BORDEA

Maritime University of Constanta, sinteasorin@yahoo.com

Abstract − **This paper describes the modality to unify two communication ways through INMARSAT in one communication layer. The authors' intention is to create one communication layer which can be connected with INMARSAT data network or with the Internet network (over BGAN network). This layer can be accessed by software using unified methods. The high level software layer can easily access this layer and can select the communication support easily using a simple flag. The layer is responsible for data packaging, data correction, synchronization, securing and compressing of the sent data, and on the other side the same layer is responsible with unpacking data, data correction, synchronizing, checking data security and decompressing of received data. The layer also is responsible for the connection with INMARSAT physical level and also with Internet TCP/IP communication stack.**

Keywords: *satellite communications; maritime data communications; communication software; network solutions.*

1. INTRODUCTION

INMARSAT satellites network is the most used digital data transfer network for maritime messaging systems. All maritime ships are equipped with INMARSAT radio stations for communication between them or with fixed Earth stations. This service ensures about 80% coverage from Earth surface, that means the INMARSAT service ensures communication on all navigable oceans and internal continental waters. In conformity with SOLAS requests all ships which navigate in A2/A3 areas must be equipped with INMARSAT-C radio stations. The US Government requests all ships that navigate in US waters to be equipped with at least one satellite communication channel.

It is possible to see a development of maritime satellite communication in the following years. The use of new LRIT (Long Range Identification and Tracking Systems) and the AIS (Automated Identification System) in the new international political order will enforce communication over the satellite channels.

Until now INMARSAT satellite network is developed until IV stage of satellites. Last satellites version ensures more services than the older ones and also ensures a good coverage than the older system, by using multi-beam technology.

Since December 2006 HNS (Hughes Network System LLC) together with Inmarsat LLC developed a new service over Inmarsat data package network. This service is named BGAN (Broadcast Gateway Area Network) and the service is based on IPoS (Internet Protocol over Satellites) technology, introduced by HNS since 2005. Also HNS developed BGAN routers for Internet connections over INMARSAT satellites data network.

It is important to observe that since 2007 we have the older digital services over Inmarsat network (Inmarsat A, Inmarsat D+, Inmarsat C, Inmarsat mini-C, Inmarsat B, Inmarsat M, Inmarsat mini-M, Fleet F33, Fleet-F55, Fleet-F77) and the new Internet connection through Inmarsat BGAN service. Data connection between a terminal and digital services ensured by Inmarsat services is NMEA-2000 standard ports or serial port. Connections between digital data terminal/controller with Inmarsat network is made through Inmarsat station. Depending on Inmarsat radio station type (producer) we have different protocol implementation.

On the other side, the newer BGAN router, provides a simple connection between a data terminal/controller to Internet network using a simple IEEE 802.3 (Ethernet) cable connection. This connection is simple and solves a lot of connection problems. BGAN routers have more advantages than the older Inmarsat stations:

- the router provides a standard interface for connecting to worldwide network (Internet), irrespective to the equipments manufacturer;
- the router is simple to be installed;
- the router can switch data connection automatically from a satellite to the other without the operator intervention;
- no extra pay or taxes for the data packages transfer are involved; all costs are included in the basic license.

The main disadvantage is that all older equipment are designed to work with older Inmarsat radio stations.

We develop a controller which is able to connect on the both communication supports and creates a communication layer between external data network

Figure 1: RCM 4010 CPU

and Inmarsat Satellite Data Network. This controller works as a communication protocol adapter.

2. HARDWARE CONTROLLER

For our solution we adopted a RCM4010 CPU unit. This controller has 6 RS232 interfaces on board; it has also one IEEE 802.3 (Ethernet) connection port. The controller is 8 bits CISC CPU @ 56 MHz, with 256kB RAM and 512kB flash program memory. This architecture provides us with the computing power for the protocol adapting layer.

The controller unit can be programmed using Direct-C compiler. The compiler can generate a multi-tasking code. The tasks can run in a cooperative-preemptive runtime environment. This advantage creates an ideal working environment for us and for our protocol adapter. The compiler is completed with all libraries required to access the RCM 4010 devices.

This controller is plugged into an interface adapter board to be interfaced with ship equipment. This board contains signal level adapters and interfaces used to connect the controller to maritime equipment and to Inmarsat radio station (Fig.2).

The controller connections with the communication layer are (Fig.3):

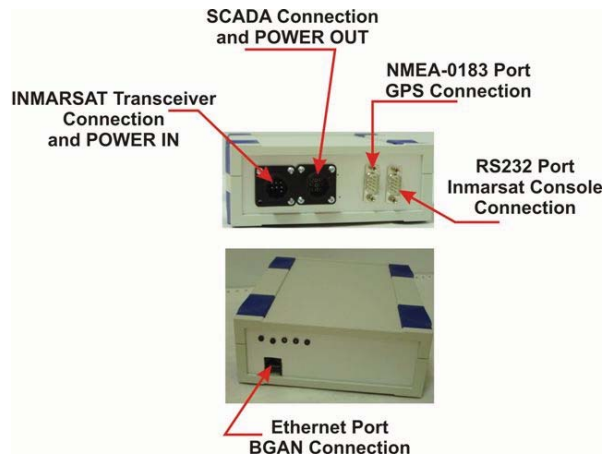Figure 2: Hardware controller unit

Figure 3: Hardware controller connections

- RS232 port with NMEA2000 adapter for the connection with INMARSAT-C transceiver (we recommend the use of Nera Saturn C transceiver);
- Ethernet Port for connecting with BGAN network. Connection with BGAN router can be made using a crossover cable or using a Ethernet Switch (or HUB).

The controller is also able to link with (see Figure 4):

- SCADA network over a special connector. The connection between controller and SCADA network is made using a RS-425 port. Also on this port the controller provides a 24V power supply @ 5 Amps for the powering of SCADA devices;
- Inmarsat Terminal Console (for example TT3020) using RS232 port;
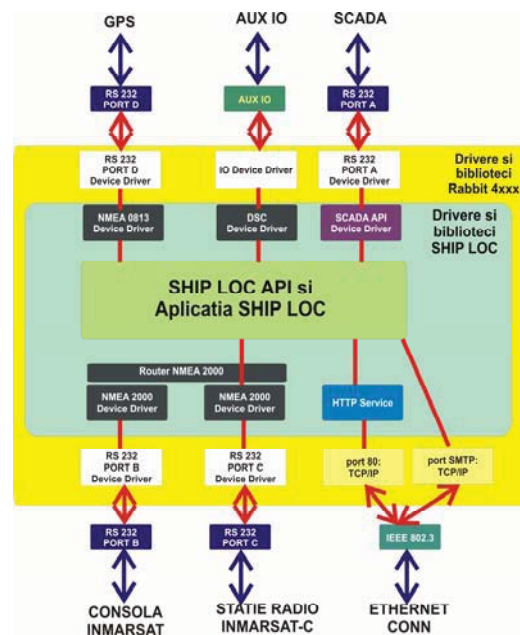- GPS connection over NMEA 0183 port.
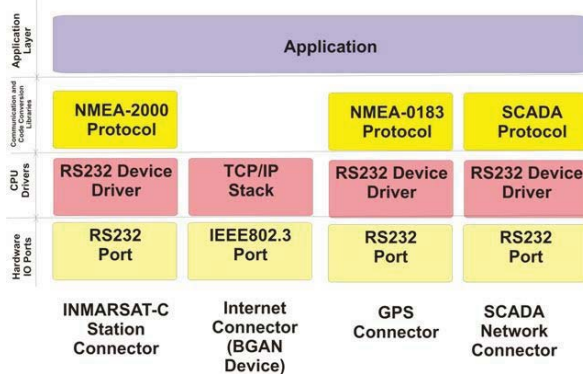

Figure 4: Controller's structure

Figure 5: Software Package

It is important to know that the controller powers over INMARSAT Transceiver Connector. The power request from transceiver is 24V @ 0.2Amps minimum (and if we power SCADA devices we recommend to provide 24V @ 5 Amps). The powering of these devices can be provided from radio station batteries. The power request by unit is less than 0.2 Amps.

3. SOFTWARE PACKAGE

Software package consists of three software levels:
- first level consists of CPU drivers with hardware devices;
- second level consists of communication and code conversion libraries;
- and the third level consists of application layer.

This layer is responsible for the application algorithms and data transfer between layers.

First level and second level are organized on communication layers stack.

For each external connection we have software communication connectors which are responsible for driving data through this connection.

Controller external links are (see Figure 5):
- communication connector with INMARSAT-C station;
- communication connector with Internet connection, using Ethernet port (for BGAN connectivity);
- communication connector with digital SCADA devices;
- communication connector with NME-0183 devices.

Each connector has a software stack representative image. A stack is composed of 3 entities:
- firstly hardware device
- secondly the software device driver attached on the hardware device
- and the third layer is software protocol layer, which is responsible of protocol and data link.

Data transfer from a layer to another one complies with the stack order.

The data received by a controller through hardware layer are transferred to software device driver.

Software device driver is responsible with receiving data from hardware layer, synchronizing and formatting of received data. On this level data are packaged in bytes or strings of bytes which are delivered to the next level protocol layer. The protocol layer is responsible with data decoding, data analysis and data packaging.

To send a package data from controller, the data is transferred from the application layer to the protocol layer. On this layer data package is attached with destination address, synchronizing words and will be represented in strings of bytes. The formatted string of bytes is delivered to the device driver. Device driver will get the bytes and will deliver them to the hardware interface. The device driver is responsible for data delivery and validation of correct data transfer to the hardware device driver.

3.1. INMARSAT Station Connector

This connector implements a data connection stack with INMARSAT-C station. The main element of this connector is a NMEA-2000 protocol device driver. This device driver is a special component of the stack, because:

- it is responsible with INMARSAT-C station control. This component activates Inmarsat C station to link with Inmarsat satellite network and manage radio station power control to keep a high quality link.
- this component also links the application layer with RS232 device driver, in conformity with NMEA 2000 rev. A protocol recommendations. The data sent or received through this connector is packaged or unpacked from NMEA 2000 protocol format. Data transferred through this connector are in e-mail format.

To be able to transfer data, this layer is accessed through this function:

*inmsSendMail(char *eMailAddr,*
*   unsigned char waitReply, char *aText);*

The layer automatically forces Inmarsat-C station to connect with Inmarsat satellite network and afterwards sends *aText* message through Inmarsat data network at *eMailAddr* destination. The *waitReply* flag is set to specify if message requests a reply from destination.

On data receipt the device driver sends a message to the application layer with the following format:

*inmsOnReceiveMail(char *eMailAddr, char *aText);*

where

eMailAddress, specifies the sender's address, and

aText, specifies the text message received.

3.2. Internet Connector

This connector is based on a TCP/IP stack implemented
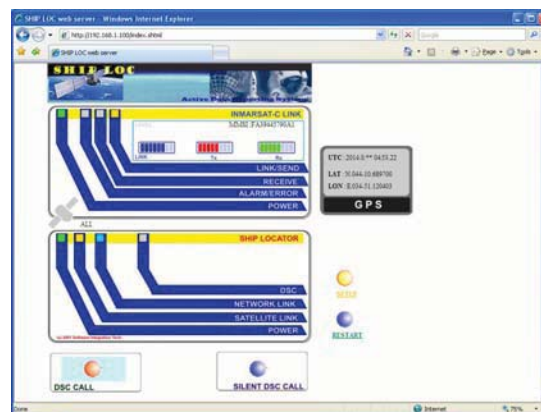
Figure 6: Controller's web page

Figure 7: Controllers status web page

in a RCM 4xxx TCP/IP device driver library. The connection with Internet layer is managed by the RCM 4xxx device driver library.

3.3. Controller Configuration

The controller can be configured using a local web page. The web page is accessed through a local Ethernet port using a web based computer.
Using web page we can configure (see Figure 6):
- the satellite connection;
- the TCP/IP parameters

3.4. Controller Status

The controller status is presented on the web page, where one can see the satellite link level and connection status (Fig. 7).

4. APPLICATION LAYER

This layer is mainly the layer responsible with high level routing data packets. This layer uses

*cof_SendMessage(char*eMailAddress,*
*char *aMessage)*

as the function to send data to destination. The function checks if the TCP/IP device driver is able to send the message, and if the answer is positive the controller will send the message through this layer. Otherwise the controller will send the message through the INMARSAT-C connector.
This function is also responsible with sending the message until it succeeds without user intervention. In case of error the function will return an error code. The function works in a multi-threading environment, and the function does not block the other application threads. The modes of action of this type of functions (named co functions) are described in Direct-C user manual.

5. CONCLUSION

This device easily connects through two mobile data networks a local area network of mobile system with a remote system. Also this equipment ensures link compression and correction for data tunnel between local network of mobile device and remote system.
In conclusion we develop this system to easily connect the local data network of mobile device with standard communication applications.

6. REFERENCES

[1] Rabbit Semiconductor. *Embedded Control. Rabbit Core RCM4000*. USA, Rabbit Semiconductor, a Digi International Company, 2006

[2] Rabbit Semiconductor. *Dynamic C, ver. 10.9*. USA, Rabbit Semiconductor, a Digi International Company, 2007

[3] Petru Eles, Horia Ciocârlie. *Programarea concurentă în limbaje de nivel înalt*. Bucuresti. Editura Stiintifică, 1991

[4] Sorin Tudor, *Tehnici de programare*, Bucuresti, Teora, 1995

[5] Donald E. Knuth, *Arta programării calculatoarelor. Vol. I, Algoritmi fundamentali*. Bucuresti, Addison Wesley and Teora, 2002

[6] Sorin-R. Sintea, *Maritime Communication Embedded Controller,* Szopol, Bulgaria, Electronics 2006, Book3, ISBN 954-438-566-5

[7] ***, *Saturn-C. Technical Manual*, ASEA Brown Boveri, 2001