A SELF ARBITRATION UNIVERSAL SIGNAL BUS COMMUNICATION

George Mahalu **Adrian Graur**

"Ştefan cel Mare" University of Suceava
Electrical Engineering and Computer Science Department

Abstract: *In this paper is treated a designing self arbitration bus communication used into the data transfer macro and micro-systems. The solution presented below is optimized like as minimum hardware linkage support and more efficiently data transferring between the bus lines and the other coupled devices. All these solutions covered both digital telecommunication applications area and digital micro-system interchange information applications area.*

Introduction

The modern data process control has like necessary a high informational traffic. This is valuable both in the data quantity and the data velocity transfer and in this sense are involved some functionality requests which put in front the efficient arbiter and control protocols. All these specificity can rise an architectural problem set which resolving both hardware and software level.
A data bus communication can constitutes a challenge for system designer. A same task imposes some features which use the compromise solutions. In this mode, from hardware viewpoint, a system bus request:

- minimize the number of physical lines – responding by the simplify of the field structure request, and implicit to the simplify of the all interfaces, of the minimum transfer time and of the international electromagnetic compatibility actual norms;
- using of the intelligent circuits which work into the master-slave tandem – request of the functional compatibility with the modern microprocessor/microcontroller system structures involved in the communication and processing data;
- assure data transfer through a channel set with different physical nature and redundant functionality, that warn the informational link fault and rise the reliability of the data transfer and security mechanisms.

From software viewpoint the system bus must respond to the actual access technologies and data security adapting requests, with implementing of the protocols and compatibilities between different management structure types.

The approach solution

In the try of identify the bus type which responds most of the above hardware and software requests, we focused about the I^{2}C (Inter - Integrated Circuits) bus type. This bus uses three physical lines:

- a data bidirectional line, noted SDA (Serial DAta);
- a clock bidirectional line, noted SCL (Serial CLock);
- a ground line, noted GND (GrouND).

Each bus connected component is recognized by unique address and can function to the distinct times

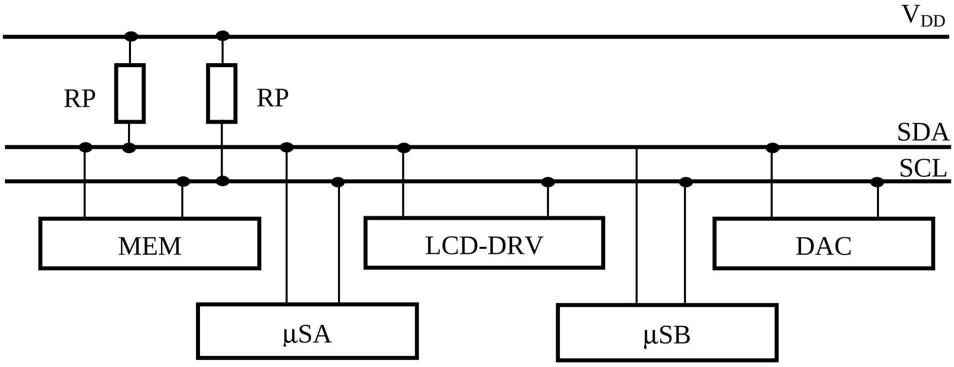

Figure 1. More devices connected by I^{2}C bus

like a data transmitter/receiver in relation with the role played into system. To the I^2C bus can be connected more master and slave devices. The initiative to execute one data bus transfer is of the master device. Each time, only one master device has the bus control and this will access only one slave device. The master-slave pair will make a data transfer from master to slave (writing regime) or from slave to master (reading regime). An example of bus connection by more devices is shown in the figure 1.

The data and clock are bidirectional lines. The free state of these lines is 1 logical. The traffic participant devices are connected to all these lines through logical access gates of open collector (or open drain) type which performing the AND-cabled logical operator.

The means of the notations are:

- RP – polarizing resistor
- MEM – memory block
- LCD-DRV – LCD driver
- DAC – digital-analog converter
- µSA, µSB – microsystems
- V_{DD} – supply voltage
- SDA – serial data signal
- SCL – serial clock signal

The bus control access is performed by one concurrent process. Gain of the control and set free of the bus are processes signaling on the following conditions (enabled by the 1 logical on the SCL):

- *START* – transition from 1 in 0 logical for SDA
- *STOP* – transition from 0 in 1 logical for SDA

The bus is considered busy after performed *START* condition and is considered set free after performed *STOP* condition. These signal combinations are generating by the master device (microsystem). In figure 2 is shown the above conditions

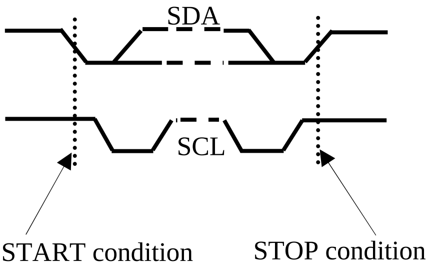

Figure 2. START and STOP specification

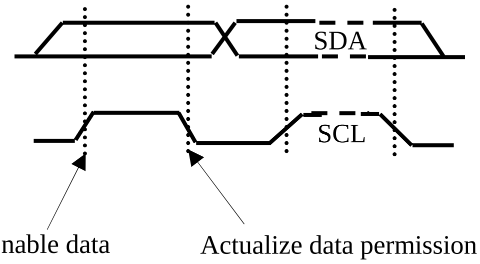

Figura 3. Transfer data on the I^2C bus example

The data is transferred on the bus like packets of bits which called *words*. Most usual example of *word* is the byte. Each transferred word between transmitter and receiver is accompanied by one confirmation bit. In this mode transferring data on the word constitutes an easy controllable operation. More that, can be possible finding a way to speed transfer adapting between transmitter and receiver. For this, after word complete reception, just to next word reception time, receiver forcing in 0 logical the SCL line. Thus, the transmitter is switched into the wait state and data transfer is resumption after the SCL line clearance. The transfer confirmation is performed by the receiver device, using a clock pulse which is generated by the master, called confirmation bit. After the last bit transmitted on line, the transmitter will clearance the data line (SDA=1). If the receiver has correct received data, it will force the SDA line in 0 logical on during of the confirmation bit (figure 3). The master has duty to confirm this bit transmitting and operate consequently. Thus, if detected confirmation, it continues with a new transfer command (for next word). Contrary, and in non-existent data transferring case, the master device forcing the STOP condition and release the SDA line.

Bus synchronization

During data transferring bus, each master device must generating clock signal. After logical synthesis on all these signals is performed the active clock signal. This act consists into synchronization mechanism for all master devices function. Consequently, is performing finally bus access arbitration.

The arbitration mechanism can be useful understand on the figure 4 chronograms base. The clock signal that was forced in 0 logical by one master device, release a measuring process for a t_{low} length on each device, parallel with forcing SCL line in the low state. After this length exhaustion, the master devices will release one by one the SCL line. When the last device it released this line switching on high state. The switching SCL line from 0 in 1 logical conducts to releasing a new measuring process for the t_{high} length when the SCL line is released (let in high state by the master devices). The first master device which will exhausts t_{high} length measuring will command the SCL line in low state so that will be take again generating for a new clock period on the SCL line.

In figure 4 is shown the way for access synchronization performing on the data bus. We are considered the clock signal regeneration by two master devices.

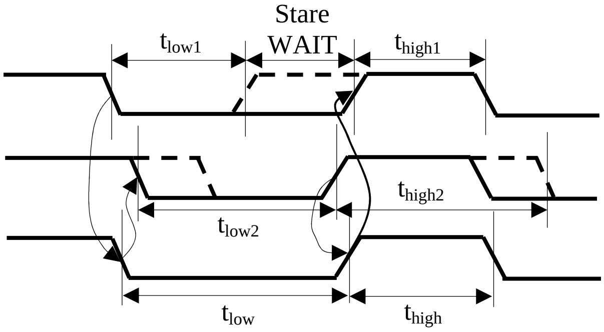

Figure 4. Tact synchronization into the arbitration procedure.

performing will contribute all bus connected master devices. The clock signal obtained is characterized by:

- t_{low} – the SCL forced in low state length – is referred to most big t_{low} duration which is generated by the bus connected devices;
- t_{high} – the SCL forced in high state length – is referred to most small t_{high} duration which is generated by the bus connected devices.

Bus access arbitration

After detecting STOP condition, two or more connected bus master devices trying -sometimes simultaneous - to gain the bus control. In these situations is necessary to act an arbitration protocol which must solution all confliction states. The major advantages of approach solution of the I^{2}C bus consist in arbitration management parallel with data transferring.
The initiating arbitration protocol premises are:

- synchronization performing at bus level;
- performing AND-cabled logical operator by the access structures of the transmitter devices on SDA line.

In the above example, the two master devices, synchronized, will provide the first bit on the data line. Another hand, the master devices will compare the transmitted logical value with the SDA attached state logical value. If the two values are identically the devices will perform the next bit transmitting, following the same procedure. Results that in the limit case when more master devices will transmit simultaneously same information is not necessary inhibition in access to bus line, for some those. Unlike above case, when one or more devices transmitting a logical different value that all those from other master devices, the mechanisms are different.

We'll considerate two master devices which begin transmitting simultaneously. On the *n*-th bit transmit the first master device (noted M1) puts 0 logical on the SDA line and the second master device (noted M2) puts 1 logical on the same line. Thanks the AND-cabled structures of the bus access circuits, the data line will set in 0 logical. Thus, the M2 master device will detect a difference between transmitted logical value and existent on line logical value. Consequently, the M2 master device will decouple from data line and the M1 master device continue to transmit without access arbitration process to affect the current transfer cycle. In figure 5 is shown time delivery of the arbitration process.

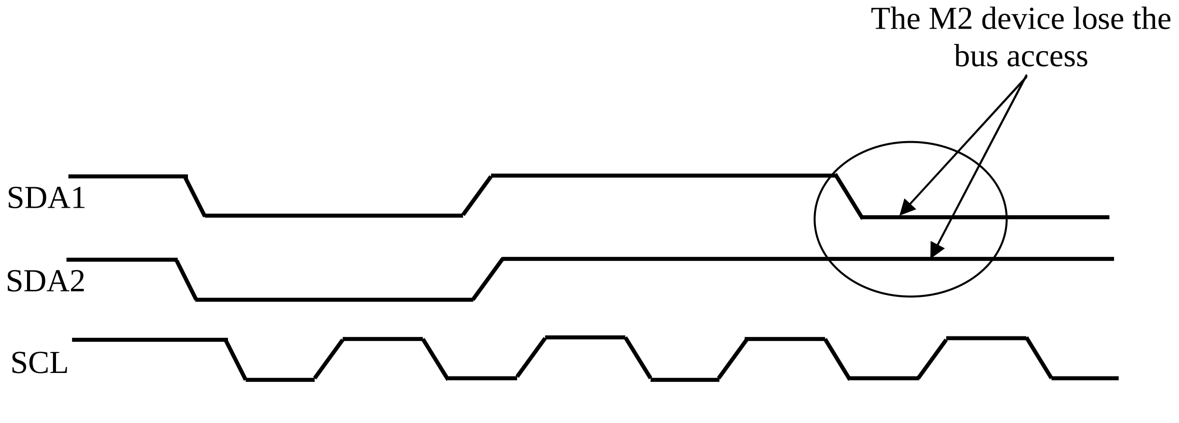

Figure 5. The bus access arbitration

Conclusions

The SAUB design imposes a series of functional considerations, at physical level and at logical level respectively, specifying the conditions when a same bus can be used with a multiple physical channels support. From possible solutions was be selected I^2C type, extended to situation that in addition to classical used channel (electrical channel) appear new channels, with different physical nature, like as supplementary radio-channel SSS-FH with binary FSK modulation.

The frequency synthesizer performing one from those $M=2^m$ allocated frequencies for all distinctly *m* bits combinations. One bit is performed by message and rest of *m-1* bits are generated by the PS code. If bit which is provided by message is the last significance bit then modification will products least frequency deviation, so that the output signal can be considered with binary FSK type. The rest of *m-1* bits provided by the PS code generator will determine the jump signal in all possible frequency domains.

Because selected solution is binary FSK modulation, must be fixed priority rule for transmitting signal in access of data bus line. For the extended bus which we design, the priority logical value is 0. In radio line case when transmitter working on the "all or nothing" principle, the priority signal is the active signal (transmitting) and not the passive signal (pause in transmitting).

References

1. A. Mihalcea, A. Şerbănescu, P. Tabarcea – *Sisteme moderne de comunicație*, Editura Militară, Bucureşti 1992.
2. I. Creangă, C. Pui – *Microprocesoare de comenzi din receptoare TV*, Editura Teora, Bucureşti 1997.
3. G. Mahalu, C. Ciufudean – *Radio-Transmitter System for PC Controlled Structures,* Proceedings of the Second European Conference on Intelligent Systems and Technologies ECIT 2002, Iaşi, July 17-20, ISBN 973-8075-20-3, CD recorded, 2002.
4. G. Mahalu, A. Graur, C. Ciufudean – *Virtual Interface in Control and Measurement Instrumentation,* XIII-th International Symposium on Electrical Apparatus and Technologies SIELA 2003, Plovdiv, May 29-30, vol.I, pg. 147-151, ISBN 954-90209-2-4, Plovdiv, Bulgaria 2003.
5. G. Mahalu, A. Graur, C. Ciufudean – *PC Signal Acquisition and Control Board for Technological Processes,* ELMA, IEEE Section Bulgaria, Tenth International Conference on Electrical Machines, Drives and Power Systems, Sofia, Bulgaria, ISBN 954-902-091-6, vol.1, pp.263-268, 2002.
6. G. Mahalu, A. Graur, C. Ciufudean, R. Pentiuc, P. Ioachim, M. Milici, D. Milici – *Intelligent Driver System for Technological Processes,* ELMA, IEEE section Bulgaria, Tenth International Conference on Electrical Machines, Drives and Power Systems, Sofia, Bulgaria, vol.1, ISBN 954-902-091-6, pp.281-285, 2002.