IRp-6 industrial robot control panel

Bogdan Fabianski
Poznan University of Technology
60-965 Poznan, ul. Piotrowo 3a, e-mail: bogdan.fabianski@put.poznan.pl

In the article, an IRp-6 originally manufactured robot control panel adaptation was presented. Modernized robot programming panel was adjusted to the new control system which replaced factory made steering case. The main goal for the project was to increase overall system value as an industrial solution. Original mechanical construction was left untouched, so the default HMI. CPU (8081) was replaced with Cortex core based micro-controller provided hardware Ethernet interface (which was main interface of the new robot control system).

1. Introduction

A. IRP-6 UNIT

Detailed information about factory-made construction of IRp-6 was included in its technical documentation [1]. There are several units in scope of robot system: control case, manipulator with 5th freedom degree (handling 6 kilos) and programming panel which is the main research target described in the article.

In the work [2] an historical overview first and single (until now) Polish robot unit was introduced. Market competition in face of lack of modernization, development backup bring production to close. However, thanks to the presence at polish technical universities, IRp-6 was became object research in field of electrical drives, visual or high-end, overriding movement trajectory generating systems in such as academic units as: AGH University of Science and Technology, Wroclaw University of Technology, Warsaw University of Technology and Poznan University of Technology.

B. FACTORY-MADE CONTROL PANEL

A standard, factory-made control (programing) panel system was well suited for its destination s a part of robotic stand. It have had a solid and ergonomic case for handling as presented on Fig. 1, below.

Panel HMI consist of [3]:

- keyboard in configuration of matrix, divided in three segments,
- analog joystick for manipulator movement,
- kinematics switch (external, internal),
- “not aus” push button,
- security switch,
- LEDs built in keyboard panel,
- alphanumeric VFD: 2 rows/40 chars.

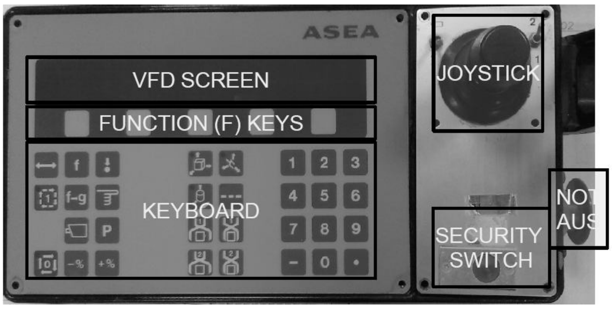

Fig. 1. IRp-6 programing panel

Keys form F group could take a function defined by the lower row display. A multifunction keys are very popular in modern HMIs (***H**uman **M**achine **I**nterfaces*) – also the robotic ones.

Inside the panel could be divide into two PCBs (***P**rinted **C**ircuit **B**oards*). The upper one includes VFD (***V**acuum **F**luorescent **D**isplay*) and its driver board, when the second central processing unit (CPU) based on 8081 micro-processor.

2. A new concept

A. PROJECT GENESIS

Works [2] and [4] that were expended for main aspects of the IRp-6 new control system (NCS) weren't enclosed important part of the industrial robotic unit which is mobile programing panel.

Factory-made construction of the considered object couldn't be directly used in the NCS. The main barrier was low-tech communication interface based on serial RS-232 standard, where the main communication bus for worked out system (NCS) was Ethernet (functional describe [5]). According to the present, available technology, decision was made to remove panel main board and compile the new one served with Ethernet.

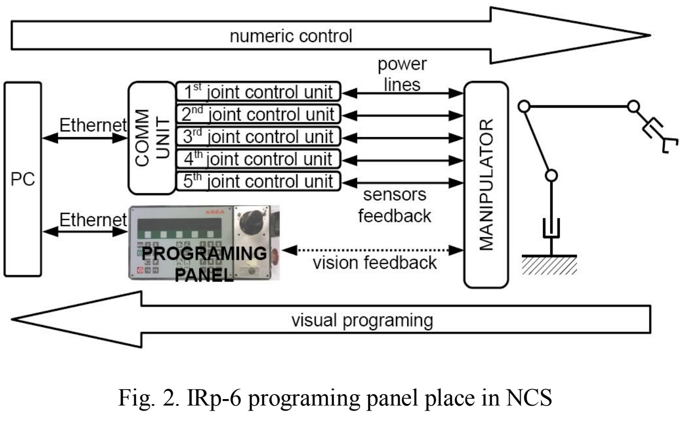

Fig. 2. IRp-6 programing panel place in NCS

B. MAIN GOALS

According to the factory-made panel analysis, that pointed on economical justification panel case leaving untouched (original system meets safety requirements), the main goal was to change communication interface for new control system. On the functional level, mobile control panel (MCP) is alternative for numeric programing from remote PC application, the alternative that in industrial environment rapidly increasing comfort and flexibility programing process. MCP makes very popular "*teaching by doing*" trajectory generation method possible.

C. REQUIREMENTS

Before the exact phase of design, requirements identification in range of power supply and internal panel parts interfacing was done. The task was as difficult as the lack of original technical documentation. After hard stage of signal decrypting all was made clear. The keyboard keys were connected into matrix configuration of a-rows, b-columns where an interface was a+b wire tape cable. Joystick position x-y was obtained from two potentiometers powered from voltage source. Rest of switches was simple line shorting devices. It was important to pay attention on "not aus" button. Safety requirements defines, that this device must guarantee taking off supply from joints motor drives thus it mustn't be connected as an internal CPU signal, but traced out as control signal for power relay. As mentioned before, there was also an VFD as part of the programing panel. Because of PCB integrity (see Fig. 3), the factory-made display control board was used. The PCB had 40-pins interface connector and all of those pins were functionally recognized.

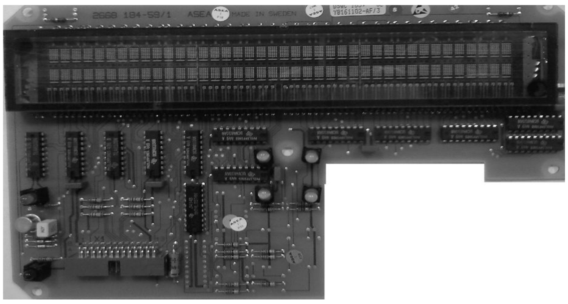

Fig. 3. VFD control board

As Fig. 3 shows, VFD control board consist of thirteen chips, were twelve of them were serial-to-parallel interface logic (STP) and one 8-bit buffer. The buffer was used for driving LEDs (lighting controls are visible from front of the MCP). When VFD considering as an dots matrix, one character field has 7(8)x5 points (one row for underlining). Eight STPs were used for control all dots in single column characters grid (one STP has 10 parallel output – five for upper character and five for other). All of those eight serial-to-parallel chips are connected to all of the grids in display, so there's need to have one signal to enable exactly one column which is valid for appropriate characters. This was done by rest four chips whose having summary 40 parallel outputs could provide "grid enable" signals for all display consist of forty columns (Fig. 4, below).

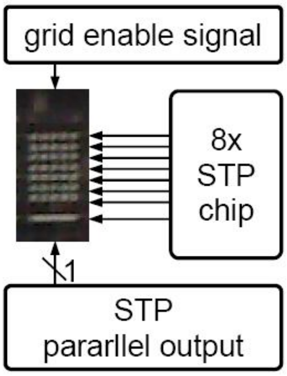

Fig. 4. Single character dot matrix control schema

Based on researches and theory of Vacuum Fluorescent Displays [6], construction of power supply section which isn't as simple as standard LCD was proposed. VFD driver require three levels of voltage: 3.3 for CPU, 5 for STP logic and 24 [V] anode supply. It's important to know that there are two ways to get electron from cathode to be emitted and so – the brighter displayed text, more clear for user: increasing cathode current or anode voltage. The first solution should be use very carefully. The higher cathode line current – the faster oxidizing process leading to cathode line breakdown. On the other hand we've anode voltage which is limited to power supply possibilities and serial-to-parallel chip maximum driving voltage.

D. CENTRAL PROCESSING UNIT

STM32 was selected for panel CPU. The reasons were as follow: high speed up to 72 [MHz], computing power up to 1.25 [MIPS/MHz]. As the follower of ARM7, Cortex M3 (which is STM32 core) was designed to increase computing power/energy consumption coefficient. As the main project requires, STM32 Connectivity Line includes hardware 100 [Mbps] Ethernet MAC (***M**edia **A**ccess* ***C**ontroller)* process unit with dedicated DMA (***D**irect **M**emory **A**ccess)* what has a key influence on communication system performance. STM32 require physical layer Ethernet controller in cooperation which gives enough power to drive Ethernet UTP (***U**niversal **T**wisted **P**air*) cable wires. The MII (***M**edia **I**ndependent* ***I**nterface*) provides data interface between units. Ethernet was described in [4].

Except of STM32 performance, it's the unit widely available in local market, relatively cheap and large, free to use with STM code libraries[7][8].

E. VFD UNIT

As mentioned, display was an array, an array of 3200 dots (8 rows x 5 columns x 40 characters x 2 lines). To display a text, human eye abilities were used. Consequences of the VFD hardware construction, only one character can be displayed at same time. All screen must be refreshed in frequencies above 50 [Hz] (like TVs). An advantages of VFD are high contrast and lack of back-lightning (points of VFD array light itself). Control algorithm was presented above, at Fig. 5. All variables begins with "grid_" are signal for serial-to-parallel group chips generating signal "grid enable", variables begins with "dots_" are signal for STP chips driving dots in single grid. TB means text buffer array 2 lines x 40 characters in line. It represents text to display. FM – is an byte two dimensional array where single character is encoded into five parts (as there are five columns in single character grid). FM was initialized in CPU memory. It defines characters font. To make font defining process easier, an computer application was created.

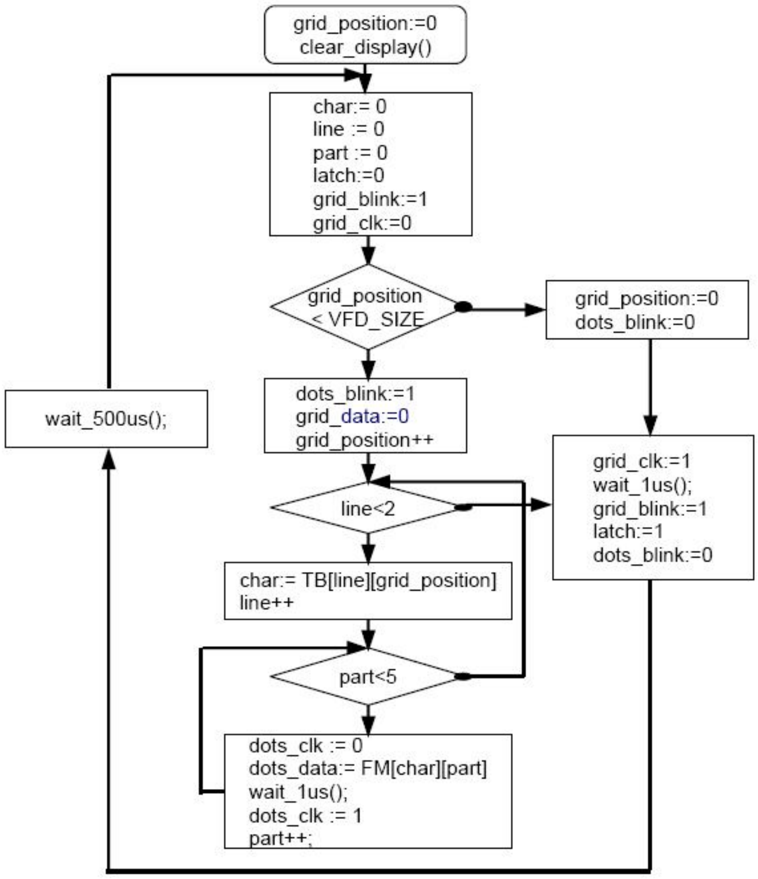

Fig. 5. VFD driving algorithm schematics

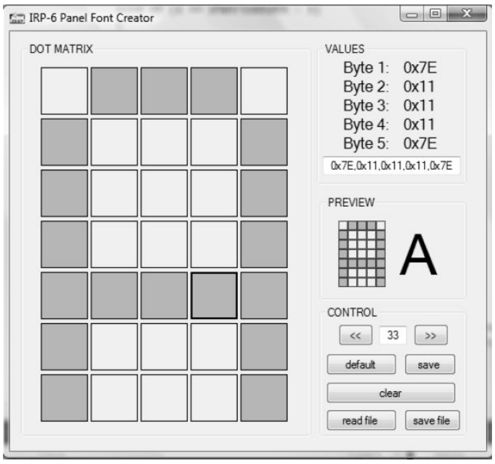

Fig. 6. IRp-6 panel font creator PC application

The application consist of main field with 5x7 buttons matrix, where single dot can be light on or off by click on field. On the right there are 5 bytes describing font for single character, a preview, where correct character is show and control panel form where it's possible to save work result to memory and all defined font to file. The file is text formatted and provide easily copy to embedded system code. The application makes possible to read and modify font created before.

3. Summary

At the beginning of the article, an IRp-6 robot unit was focused so the new control system was. Omitted before, factory-made programming panel was adapted to make it suitable in Ethernet network. New IRp-6 control system become more attractive and more applicable in industrial works. In the end, on Fig. 7, proposed VFD driving algorithm was presented as an real working panel photo.

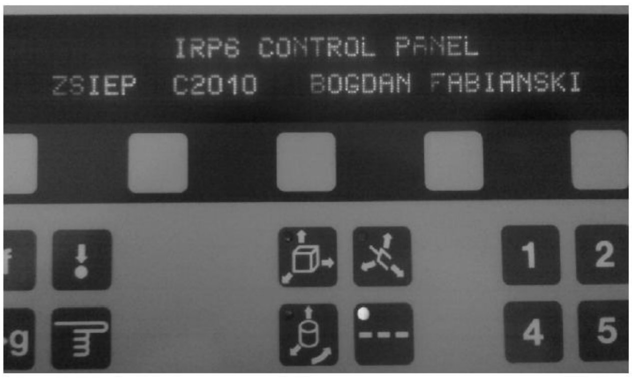

Fig. 7. Example of VFD driving with proposed algorithm

References

[1] Industrial Automation Division (ZAP): *DTR RMM-01-00*, Ostrow Wielkopolski, 1988.
[2] Fabianski B.: *Industrial robot drive*, SENE conference, Lodz, 2009.
[3] ZAP: *IRp robot programing guide* (RMMS-0466), 1988.
[4] Fabianski B.: *Communication interfaces in robot control system*, SENE conference, Lodz, 2009.
[5] Microchip Corp.: *Ethernet theory of operation* (AN1120), 2008.
[6] http://en.wikipedia.org/wiki/Vacuum_fluorescent_display, 2009.
[7] Yiu J.: *The definitive guide to the ARM Cortex-M3*, Elsevier, 2007.
[8] Paprocki K.: *Micro-controllers STM32 in practice*, BTC, 2009.