

Final Project – Micro Controllers 31226

Smart Home

Submitted by:

Dvir Kanfi

Oren Shefer

Reuven Petroschensky

Reshef Finklestein

Moderator:

Dr. Fadel Tarif

Table of contents:

Abstract & Theoretical Background:	3
Technical Specifications:	3
Labor division:	3
Brick chart:	4
Flowchart:	4
Considerations in platform selection:.....	6
Resource allocation:.....	6
System operation instructions:	7
Description of project's functions:.....	8
Problems & Solutions:.....	9
Summary & Conclusions	9
Project Files.....	9
Sources	10

Abstract & Theoretical Background:

The system is designed to control a smart home. It will manage the front entrance gate for the house perimeter and will allow activating and deactivating electronic equipment both from a distance and from proximity.

The main control system utilizes a screen with a keypad as a user interface and can also be accessed remotely.

The purpose of the main system is to select an action to be executed from a menu and then send it to the secondary system.

The purpose of the secondary system is to communicate with the main system, receive opcodes from it, and execute the selected action.

Technical Specifications:

Component	Description	Execution parameters
An 8051 controller	Sending activation and deactivation commands to equipment inside the house	Activating ports and serial communications, performing measurements in the house.
An extended 8051 controller	Allowing a user interface	Displaying a user menu and allowing to control the house's parameters through serial communication with the first controller, or through Bluetooth commands.
Bluetooth component	Allowing wireless communication for the user	Sending commands from the user to the controller
Servo	Servo engine	Simulating gate operation
Relay	-	Simulating electronic device operation
LEDs	Built-in LEDs in the controller	Activating electronic equipment

Labor division:

Algorithm & idea development – Everyone

Main controller programming – Reshef, Dvir, and Reuven

Secondary controller programming – Oren, Reuven, and Reshef

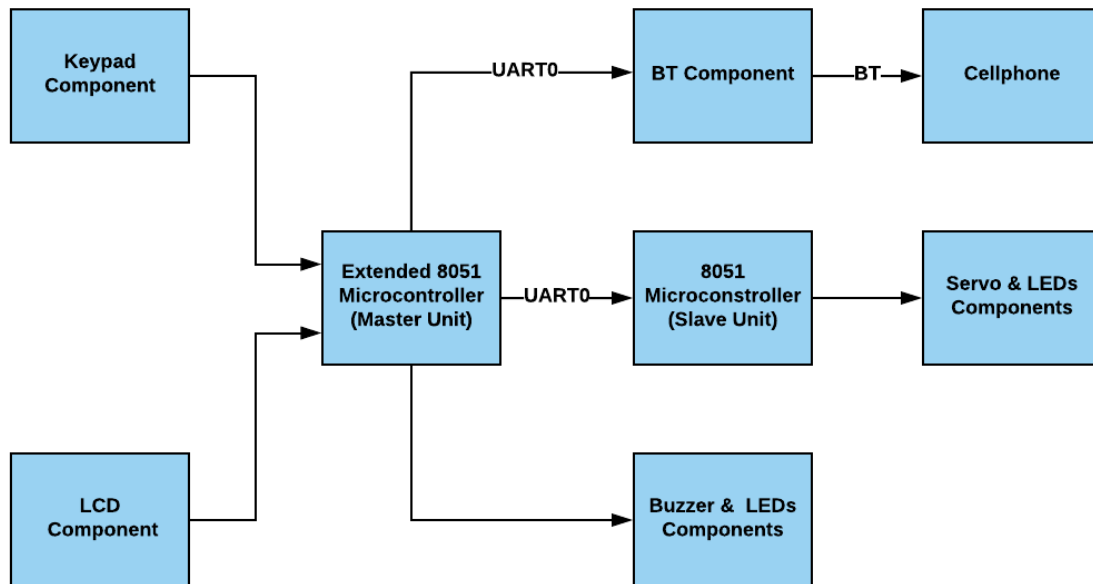
Presentation preparation – Oren, Dvir, and Reshef

Interface & mobile app creation – Dvir

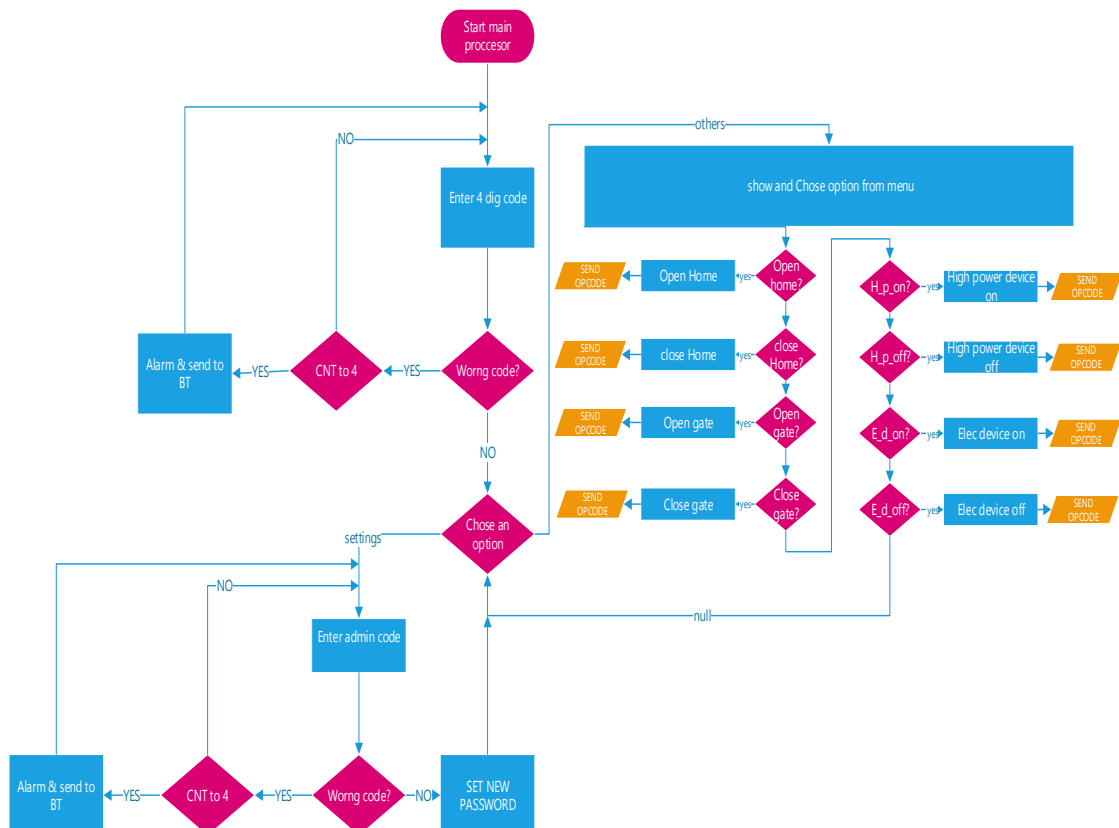
Integration with servo – Reuven and Oren

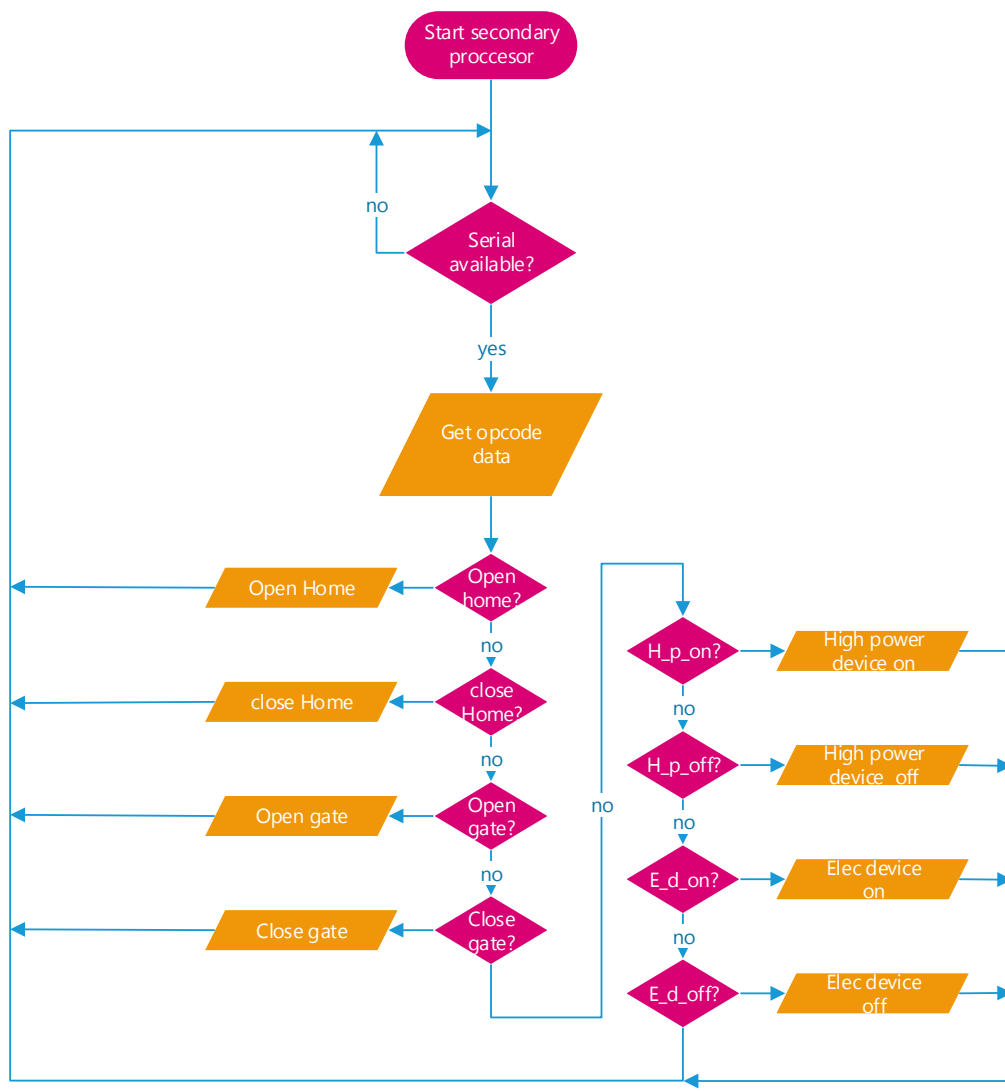
Report preparation – Everyone

Brick chart:



Flowchart:





Considerations in platform selection:

We had two platforms at our disposal:

- Micro controller toolstick – **For Slave Unit.**
- Micro controller c8051fx20 – **For Master Unit.**

To meet our system's requirements, we needed communication capabilities, therefore we used both platforms. Our choice was solely what to do and through which system. The first platform was the simpler one, so we chose it for processing the commands received from the extended controller, according to opcode. The purpose of the second platform (for the extended controller) is to identify the user as the true owner of the system, display the menu, allow choosing from the menu's options, and sending the correct opcode to the UART, through which the second controller will "know" which action to execute. The user does not have direct access to the main controller, all communications are done via the extended controller.

Resource allocation:

Main controllerⁱ:

Red LED – P1.7

Green LED – P1.6

Yellow LED – P1.5

Speaker – P1.3

Keypad – P3.0-P3.7

LCD – P1.0-P1.2, P2.0-P2.7

UART0 – P0.0-P0.1

Secondary controller:

Relay – P1.0-P1.7

LED – P5.4-P.7

UART1 – P0.2-P0.3

Servo – P0.4

Variable setting:

Initializing PCA variables:

PCA0CPH0=0xF9 for opening the gate

PCA0CPH0=0xE1 for closing the gate

Initializing timer variables:

TH1=B8 for BaudRate 9600

System operation instructions:

Initial system startup:



System code entry:

At the initial startup, the system's admin code is set to 0000

The user has 4 attempts to input a code. After a fourth failed attempt, an alarm will sound.

Interface & navigation:

The interface is built in a list format.

D is for scrolling down.

A is for scrolling up.

Users can choose the desired action by pressing the digit that's written beside the action's description.

is for going back to the main menu.



Changing password:

Option number 5 in the menu. In this mode, the user can choose between changing the admin password, or a regular user's password.

Both options will require inputting the old admin before allowing to change the code.

If a given code is wrong, the user will be returned to the initial screen.

After changing the code, the system will remember the code even after a restart.

Mobile interface:

The mobile interface updates the user on all actions performed in the house. Every command that's sent from the main controller to the secondary one can also be sent from the phone via Bluetooth.

Description of project's functions:

Main controller's functions:

- MENU1, OPEN MENU – Functions responsible for displaying user menu.
- DISPLAY – During the system's initial status, displays the menu and allows navigating the system's functions, and displays an updated menu on the screen based on keypad presses.
- GetAndVerifyCode – Receives the password typed by the user from the keypad and compares it with the one stored by the system. If there are over 4 failed attempts, the system blocks access.
- Settings – Change of password. To reach this function, correct admin code must be entered. If the code is correct, the user may choose between changing admin or user password change. At the end, this function returns the user to the main menu.
- InterfaceIndication – Activation and deactivation of the buzzer and LED's, as well as control of their pauses.
- ISR UART0 – Transmits communications to secondary controller and BT component for information broadcasting.
- Admin lock – Locking of the controller by activating a LED and the buzzer and turning of all equipment in the house.
- SENDUARTDATA – Auxiliary function for sending data to secondary controller & BT component.
- GATECON, RELATCON, ELECCON BT DEVIDE – Functions that generate the opcode sent via the UART per the user's selection.

Secondary controller functions:

- UART0 ISR – Receives data from main controllers in string format and saves it in the KEYS array, through which the necessary commands may be executed:
 - HPCON – Turns electronic equipment on or off. If the string received from the main controller contains RELAY_ON, the secondary control turns the respective device on, depending on the next character in the string. A similar action is performed in the deactivation process, with the string RELAY_OFF instead. In practice, voltage enters at port 1.
 - ELECON – A function for turning electronic devices on or off. If the string send by the main controller contains ELECTRONIC_ON, the secondary controller turns the devices (LEDs) on based on the next digit following the string. A similar action is performed when turning devices off, with the string ELECTRONIC_OFF instead. The digits that can be sent after the string are 4 through 7, and they activate LEDs 5.4 through 5.7 respectively.
 - GATECON – This function accepts the string OPEN_GATE or CLOSE_GATE and opens or closes the gate respectively.
 - SERVO – This function receives the corresponding values from the function GATECON and opens or closes the gate accordingly.
 - LOCKDOWN – This function locks and shuts down all electronic devices and gate.

Problems & Solutions:

- The GUI has only one interfacing relay. A possible solution would be to expand the interface based on future designation.
- UART1 on both controllers allows only receiving. To solve this, we've used UART0 for broadcasting to the secondary controller and to the BT component.
- Bluetooth – there was a problem matching transmitting and receiving speeds with the BT component, therefore we switched to using an external isolator so that a Baud Rate of 9600 BPS could be achieved.^{ii,iii}
- Servo – Initially, a timer1 was used to activate the servo, however, that interfered with generating the UART's Baud Rate. Therefore, we've switched the PWM generation to PCA, which works with timer0 and allows reducing the rate.^{iv}

Summary & Conclusions

We've learned that to plan a complex system that includes several controllers and external devices, multiple factors must be considered, such as timers, resources, pins, etc. We've discovered that not all works as smoothly as it would've in theory, and sometimes extended adjustments are necessary to make things work properly in practice.

Project Files

- For Main controller project files:
 - **Electric_Gate.wsp** – Main controller project code file (**in C**).
 - **Gate_GUI.c** – Main controller code file (**in C**).
 - **Init_Device_Master.cwg** – Main controller initialization project code file (**in C**).
 - **Init_Device_Master.c** – Main controller initialization code file (**in C**).
 - **User_Interface_def.h** – Main controller support function header file (**in C**).
 - **User_Interface_Functions.c** – Main controller support function code file (**in C**).
- For Secondary controller project files:
 - **Slavepro.wsp** – Secondary controller project code file (**in C**).
 - **Gate_slave.c** – Secondary controller code file (**in C**).
 - **Init_DeviceB.cwg** – Secondary controller initialization project code file (**in C**).
 - **Init_DeviceB.c** – Secondary controller initialization code file (**in C**).
 - **User_Interface_def.h** – Secondary controller support function header file (**in C**).
 - **User_Interface_Functions.c** – Secondary controller support function code file (**in C**).
 - **String.h** – Secondary controller support string library functions header file (**in C**).
 - **String.c** – Secondary controller support string library functions code file (**in C**).
 - **compiler_defs.h** – compiler constants & definitions header file (**in C**).
 - **C8051F020_defs.h** – 8051F020 micro controller constant & definitions header file (**in C**).

Sources

ⁱ Controller data sheets on manufacturer's website:

www.silabs.com/documents/public/presentations/8051_Instruction_Set.pdf

ⁱⁱ Dr. Fadel Tarif's lectures on micro controllers: <http://moodle.braude.ac.il/course/view.php?id=16965>

ⁱⁱⁱ BT module data sheets: http://www.fecegypt.com/uploads/dataSheet/1480849570_hc06.pdf

^{iv} Servo engine data sheets: <http://moodle.braude.ac.il/course/view.php?id=1696>