

RELATÓRIO DE ATIVIDADES PRÁTICAS EM LABORATÓRIO APRESENTAÇÃO DE PROJETOS

Curso: Engenharia da Computação

Termo: 9º Noturno

Disciplina: Projeto de Sistemas Embarcados

Projeto: Controle de Senhas de Atendimento

Acadêmico(s):

RA	Nome	Assinatura
207180	Lucas Fernando Costa dos Santos	
207540	Pedro César Vagner Nogueira	

Professor(es):
Amadeu Zanon Neto

**Araçatuba-SP
2020**

RELATÓRIO DE ATIVIDADES PRÁTICAS EM LABORATÓRIO – APRESENTAÇÃO DE PROJETOS

CURSO	TERMO	PERÍODO	DISCIPLINA	
ENGENHARIA DA COMPUTAÇÃO	9	NOTURNO	PROJETO DE SISTEMAS EMBARCADOS	
PROFESSOR(A)			LABORATÓRIO / SALA	ATIVIDADE
AMADEU ZANON NETO			LAB. AUTOMAÇÃO	02/2020

Aluno(s):

#	RA	Nome	#	RA	Nome
1	207180	Lucas Fernando Costa dos Santos	6		
2	207540	Pedro César Vagner Nogueira	7		
3			8		
4			9		
5			10		

Função ou tarefa atribuída a cada aluno da equipe:

#	Função/atribuição	#	Função/atribuição
1	Programação/Circuito	6	
2	Programação/Circuito	7	
3		8	
4		9	
5		10	

Projeto ou atividade a ser desenvolvida – Resumo, enunciado ou descrição: Exercício () - Trabalho (X) - Avaliação ()

As equipes deverão desenvolver um sistema de senhas de atendimento.

O projeto deverá ser composto pelos itens a seguir:

- Um painel de numeração com displays de 7 segmentos para chamado do cliente com um sinal sonoro, este painel deve ser composto por 3 dígitos (000 a 999) e uma sinalização de atendimento prioritário.
- Um sistema de controle de contagem de ordem e envio de chamado ao painel luminoso, repetir chamado e reset.
- Um software desenvolvido em Visual Basic (obrigatório) que permite verificar quantas senhas foram chamadas a posição atual e controles de reset, incremento de contagem, atualização do valor de forma manual e chamada de clientes.
- Obrigatório o uso de microcontroladores da linha PIC.
- Obrigatório a programação em mikroBasic.
- Fazer banner para apresentação conforme modelo fornecido pelo professor.

Desenhos, esquemáticos ou croqui de modelo proposto para o trabalho ou projeto:



Datas e prazos:

Etapa	Prevista		Efetiva	
Apresentação do projeto pelo professor:	Data: 13/03/2020	Hora: 21:00	Data: 13/03/2020	Hora: 21:00
Início do projeto:	Data: 13/03/2020	Hora: 21:00	Data: 13/03/2020	Hora: 21:00
Encerramento, entrega ou apresentação:	Data: 10/06/2020	Hora: 21:00	Data: ____/____/____	Hora: __:__

Forma de apresentação:

Prevista / Solicitada	Efetiva	Nota, ciência e avaliação do professor	
(X) Montagem prática	() Montagem prática	Professor: Amadeu Zanon Neto	
(X) Seminário / Apresentação / Banner	() Seminário / Apresentação / Banner	Nota/Pontuação	Assinatura/Visto/Carimbo
(X) Relatório	() Relatório		
(x) Simulação	() Simulação		
() Visita	() Visita		
(X) Mostra / Evento	() Mostra / Evento		
() Artigo	() Artigo	Data	
(X) Desenvolvimento de software	() Desenvolvimento de software		
() Monografia	() Monografia		
() Outro	() Outro		

Observações/anotações:

Componentes e materiais utilizados:

#	Qte	Descrição	#	Qte	Descrição
1	2	MÓDULO SERIAL	21		
2	2	MICROCONTROLADOR PIC 16F690	22		
3	2	PROTOBOARD	23		
4	8	RESISTOR 220R	24		
5	14	RESISTOR 10K	25		
6	1	LED	26		
7	6	BOTÃO	27		
8	4	DISPLAY 7 SEGMENTOS	28		
9	1	DISPLAY 7 SEGMENTOS MPX4	29		
10	5	RESISTOR 1K	30		
11	5	TRANSISTORES	31		
12	7	PORTA NOT	32		
13	2	RX8	33		
14	2	CI 4511	34		
15			35		
16			36		
17			37		
18			38		
19			39		
20			40		

Equipamentos e materiais de laboratório:

#	Qte	Descrição	#	Qte	Descrição
1	2	COMPUTADOR PC	13		FIOS
2		FONTE DE ALIMENTAÇÃO	14		SHIELD ARDUINO
3		CAIXA DE FERRAMENTAS	15		
4		MULTÍMETRO	16		
5		GRAVADOR DE MICROCONTROLADOR	17		
6		CLP – MODELO →	18		
7		SENSORES INDUSTRIAIS	19		
8		DISPLAY DE LCD	20		
9		IMPRESSORA 3D	21		
10		KIT ROBÓTICO LEGO	22		
11		KIT ROBÓTICO TAMIYA	23		
12		ESTANHO	24		

Softwares utilizados

#	Descrição /versão	#	Descrição/Versão
1	MikroBasic 7	6	
2	Proteus 7	7	
3	VSPE 0.9	8	
4	Visual Studio 2010	9	
5		10	

Conclusões e finalização de projeto

Problemas ou dificuldades encontradas e suas soluções (Se necessário anexar folhas suplementares):

Problemas e ou dificuldades	Soluções propostas e aplicadas.
Problemas com a comunicação serial em relação aos displays	Implementação dos cálculos de unidade, centena, dezena e
De 7 segmentos.	milhar no software e no mikroBasic, permitindo que as
	informações cheguem e retornem no padrão correto.
Demolimit no pic 1.	Não foi possível solucionar este, o pic 1 não pode receber a
	atualização dos dados feitos pelo software, pois não teve
	espaço suficiente para isso, por conta deste pic precisar de
	mais funções que o outro.

Conclusões e comentários finais

Este projeto proporcionou ótimos conhecimentos de forma prática e aplicada a problemas que podemos encontrar no dia a dia, como sendo o caso do problema das senhas e da questão da prioridade. Os conhecimentos obtidos são essenciais e podem ser aplicados em outros problemas. O uso de comunicação serial é um recurso muito interessante e promissor, pois como visto neste projeto, podemos ter vários módulos com um controlador gerenciando-os de forma que um módulo central consiga controlar todo o restante do software. Infelizmente o problema com o demolimit acaba limitando muito explorar recursos que os micro-controladores nos fornecem, e isso acaba de certa forma deixando a solução mais simples e cortada em algum ponto, ou maior. Um dos recursos fundamentais que a conexão serial nos proporcionou foi a possibilidade de integração com o Visual Studio, o software responsável pelo desenvolvimento do controle central, que permite reconhecer e integrar portas seriais e através da programação em Visual Basic, manipular todas as informações de acordo com a necessidade do problema encontrado. Com certeza essa integração é uma das mais interessantes, ficando possível também integração mobile, e isso permite aos utilizadores uma comodidade e facilidade bem maior em lidar com as soluções desenvolvidas desta forma.

ANEXO - I**Programação (Pode ser substituída por listagem impressa).**

program Pic1

```
dim recebe, txt as string[6]
dim b0, b1, b2, b3, b4, b5 as boolean
dim incP, incN, priorit as integer
dim uni, cen, dez, mil, valor as integer
dim valorN, valorP as integer
```

'Controle de prioridade e envio

sub procedure envia

```
    if priorit = 1 then
        valorN = 1000 + incN
        IntToStr(valorN, txt)
        Usart_Write_Text(txt)
        Usart_Write_Text("/")
    end if
    if priorit = 2 then
        valorP = 2000 + incP
        IntToStr(valorP, txt)
        Usart_Write_Text(txt)
        Usart_Write_Text("/")
    end if
end sub
```

sub procedure alerta

```
    IntToStr(65, txt)
    Usart_Write_Text(txt)
    Usart_Write_Text("/")
end sub
```

main:

```
TRISA=%11111111
TRISB=%01101111
TRISC=%00000000
INTCON=%00000000
ANSEL=%00000000
ANSELH=%00000000
PORTC=0
Usart_Init(9600)
```

b0=false

b1=false

b2=false

b3=false

b4=false

b5=false

priorit=1

incP=0

incN=0

executa:

'Recebe valores do controle central

if USART_Data_Ready() > 0 then

Usart_Read_Text(recebe, "/")

valor = StrToInt(recebe)

'if (valor < 2000) and (valor >= 1000) then

' priorit = 1

' Norm = valor - 1000

'end if

'if (valor < 3000) and (valor >= 2000) then

' priorit = 2

' Pref = valor - 2000

'end if

end if

```

Incrementa
if testbit(portA,0) = 1 then
    if b0 = false then
        b0 = true
        if priorit = 1 then
            incN = incN + 1
            if incN > 999 then
                incN = 999
            end if
            if incN < 0 then
                incN = 0
            end if
        else
            incP = incP + 1
            if incP > 999 then
                incP = 999
            end if
            if incP < 0 then
                incP = 0
            end if
        end if
    end if
end if
if testbit(portA,0) = 0 then
    if b0 = true then
        b0 = false
    end if
end if

```

```

Decrementa
if testbit(portA,1) = 1 then
    if b1 = false then
        b1 = true
        if priorit = 1 then
            incN = incN - 1
        else
            incP = incP - 1
        end if
    end if
end if
if testbit(portA,1) = 0 then
    if b1 = true then
        b1 = false
    end if
end if

```

```

Muda prioridade
if testbit(portA,3) = 1 then
    if b3 = false then
        b3 = true
        if priorit = 1 then
            priorit = 2
        else
            priorit = 1
        end if
    end if
end if
if testbit(portA,3) = 0 then
    if b3 = true then
        b3 = false
    end if
end if

```

```

'Emite o alerta
if testbit(portA,4) = 1 then
  if b4 = false then
    b4 = true
    alerta
  end if
end if
if testbit(portA,4) = 0 then
  if b4 = true then
    b4 = false
  end if
end if

'Envia dados
if testbit(portA,5) = 1 then
  if b5 = false then
    b5 = true
    envia
  end if
end if
if testbit(portA,5) = 0 then
  b5 = false
end if

'Controle de exibição
if priorit = 1 then
  valor = incN + 1000
else
  valor = incP + 2000
end if

mil = valor div 1000
cen = valor div 100 - (mil * 10)
dez = valor div 10 - (cen * 10) - (mil * 100)
uni = valor - (dez * 10) - (cen * 100) - (mil * 1000)

PORTC = uni + 16
delay_ms(5)
PORTC = dez + 32
delay_ms(5)
PORTC = cen + 64
delay_ms(5)
PORTC = mil + 128
delay_ms(5)

goto executa
end.

```

program MCU2

```

dim recebe as string[6]
dim aux as integer
dim uni as integer
dim dez as integer
dim cent as integer
dim mil as integer
dim valor as integer

```

inicio:

```

TRISA=%11011111
TRISB=%00101111
TRISC=%00000000
INTCON=%00000000

```



```
ANSEL=%00000000
ANSELH=%00000000
```

```
USART_INIT(9600)
portc = 0
porta = 0
```

```
PORTB.6=1
```

```
valor = 1000
aux = 0
```

```
executa:
```

```
if Usart_Data_Ready() > 0 then
  Usart_Read_Text(recebe, "/")
  aux = StrToInt(recebe)

  if (aux >= 1000) and (aux < 3000) then
    valor = aux
  end if
```

```
  gosub exhibe
```

```
  aux = 0
  recebe = ""
end if
```

```
  gosub exhibe
  delay_ms(10)
```

```
  goto executa
```

```
exibe:
```

```
  mil = valor div 1000
  cent = valor div 100 - (mil * 10)
  dez = valor div 10 - (cent * 10) - (mil * 100)
  uni = valor - (dez * 10) - (cent * 100) - (mil * 1000)
```

```
  if mil = 1 then
    portb.4 = 0
    portb.6 = 1
  end if
```

```
  if mil = 2 then
    portb.4 = 1
    portb.6 = 0
  end if
```

```
  PORTC = uni
  PORTC.4=1 PORTC.5=0 PORTC.6=0
  delay_ms(5)
```

```
  PORTC = dez
  PORTC.4=0 PORTC.5=1 PORTC.6=0
  delay_ms(5)
```

```
  PORTC = cent
  PORTC.4=0 PORTC.5=0 PORTC.6=1
  delay_ms(5)
```

```
  return
```

```
end.
```

ANEXO - II

Esquema elétrico - (Pode ser substituído por esquema impresso).

