**MaRELATÓRIO DEMONSTRATIVO DO ANO-BASE 2018**

**RECURSOS HUMANOS DIRETOS**

**Nome:** Francisco Fechine Borges

**CPF:** 373.715.764-20

**Formação:** Doutorado em Engenharia Elétrica

**Cargo/Função:** Pesquisador

**Total de Horas:** 80

**1. Atividade:** Planejamento e Preparação

**Período de Execução:** 01/12/2017 a 31/12/2017

**Horas na Atividade:** 6

**Justificativa de participação na atividade:** Nesta etapa, o colaborador atuou na gestão da equipe para execução das atividades iniciais. Coordenou reunião técnica para explicação e discussão sobre o escopo do projeto, uma vez que trata-se de um projeto muito específico, de desenvolvimento de software embarcado para um conjunto de módulos eletrônicos da empresa HI-MIX-Telit. Após entendimento conjunto do escopo, o colaborador coordenou estudo sobre as tecnologias e metodologias a serem adotadas, com a definição do que seria necessário para a execução das atividades da melhor maneira possível.

**2. Atividade:** Testes dos Módulos e Montagem dos Ambientes de Desenvolvimento - Parte I

**Período de Execução:** 01/12/2017 a 31/12/2017

**Horas na Atividade:** 6

**Justificativa de participação na atividade:** Nesta etapa, o colaborador apoiou a preparação dos ambientes de desenvolvimento necessários para o desenvolvimento do software. Realizou gestão técnica da equipe e das atividades realizadas.

**3. Atividade:** Estudo e Criação dos Projetos e Documentações Iniciais

**Período de Execução:** 01/12/2017 a 31/12/2017

**Horas na Atividade:** 8

**Justificativa de participação na atividade:** Coordenou reunião de planejamento. Em conjunto com a equipe, definiu e dividiu as atividades a serem realizadas pelos desenvolvedores Mariana Lins, Pedro Granville e Roosevelt Vinícius.

**4. Atividade:** Implementação de Biblioteca de Comandos AT

**Período de Execução:**01/01/2018 a 31/01/2018

**Horas na Atividade:** 6

**Justificativa de participação na atividade:** Gestão técnica da equipe e das atividades de implementação das bibliotecas de comandos AT, que permitem a comunicação com as portas seriais do chip microcontrolador SAMD21 (do fabricante Microchip), utilizado na placa Arduíno M0 Pro, e gerenciamento dos comandos AT para a plataforma de software Atmel Studio (ASF+FreeRTOS), necessários para que as bibliotecas a serem implementadas pudessem se comunicar com cada um dos módulos Telit.

**5. Atividade:** Implementação de Firmware Base

**Período de Execução:**01/01/2018 a 31/01/2018

**Horas na Atividade:** 8

**Justificativa de participação na atividade:** Coordenou reunião de acompanhamento, para avaliação técnica das alternativas propostas pelos participantes para o desenvolvimento dos softwares embarcados. Decidiu-se que, antes do início do desenvolvimento das bibliotecas e exemplos definitivos, seria necessário a criação de um projeto de testes, que permitisse a realização de teste das primeiras rotinas de comunicação com os módulos. Em seguida, iniciou-se a criação propriamente dita das bibliotecas e exemplos definitivos.

**6. Atividade:** Testes dos Módulos e Montagem dos Ambientes de Desenvolvimento - Parte II

**Período de Execução:**01/01/2018 a 31/01/2018

**Horas na Atividade:** 6

**Justificativa de participação na atividade:** Nesta etapa o colaborador atuou na gestão da equipe e das atividades. Contribuiu para a organização e preparação dos ambientes, tanto de hardware quanto de software, mais adequados para o desenvolvimento das atividades. Foi realizada uma avaliação de confirmação da pertinência de cada atividade em função do perfil de cada desenvolvedor, decidindo-se pela manutenção da distribuição de atividades previamente realizada.

**7. Atividade:** Implementação da Aplicação Túnel TCP

**Período de Execução:**01/02/2018 a 28/02/2018

**Horas na Atividade:** 4

**Justificativa de participação na atividade:** Gestão da equipe e das atividades. Nesta etapa, houve avaliação técnica parcial dos produtos já desenvolvidos até aqui, incluindo a implementação da aplicação Túnel TCP.

**8. Atividade:** Implementação da Aplicação Porteiro Eletrônico

**Período de Execução:**01/02/2018 a 28/02/2018

**Horas na Atividade:** 6

**Justificativa de participação na atividade:** Coordenou reunião de acompanhamento e avaliação. Realizou avaliação técnica dos produtos desenvolvidos até aquele momento, incluindo a aplicação Porteiro Eletrônico. Participou de decisão conjunta sobre a melhor alternativa para o desenvolvimento dos próximos exemplos.

**9. Atividade:** Implementação da Aplicação SMS Commands

**Período de Execução:**01/02/2018 a 28/02/2018

**Horas na Atividade:** 5

**Justificativa de participação na atividade:** Gestão técnica da equipe e das atividades. Coordenou reunião de acompanhamento e avaliação. Realizou avaliação técnica dos produtos desenvolvidos até aquele momento, incluindo a aplicação SMS Commands Participou de decisão conjunta sobre a melhor alternativa para o desenvolvimento dos próximos exemplos.

**10. Atividade:** Implementação das bibliotecas dos módulos

**Período de Execução:**01/02/2018 a 28/02/2018

**Horas na Atividade:** 5

**Justificativa de participação na atividade:** Gestão técnica da equipe e das atividades. Coordenou reunião de acompanhamento e avaliação. Realizou avaliação técnica dos produtos desenvolvidos até aquele momento, incluindo a implementação das bibliotecas dos módulos. Participou de decisão conjunta sobre a melhor alternativa para o desenvolvimento dos próximos exemplos.

**11. Atividade:** Implementação da Aplicação GPS Geofence

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:** 4

**Justificativa de participação na atividade:** Realizou validação de produtos e de documentação produzida. Coordenou reunião de acompanhamento e avaliação. Realizou avaliação técnica dos produtos desenvolvidos até aquele momento, incluindo a aplicação GPS Geofence. Participou de decisão conjunta sobre a melhor alternativa para o desenvolvimento dos próximos exemplos.

**12. Atividade:** Implementação da Aplicação LBS Geofence

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:** 4

**Justificativa de participação na atividade:** Realizou validação de produtos e de documentação produzida. Coordenou reunião de acompanhamento e avaliação. Realizou avaliação técnica dos produtos desenvolvidos até aquele momento, incluindo a aplicação LBS Geofence. Participou de decisão conjunta sobre a melhor alternativa para o desenvolvimento dos próximos exemplos.

**13. Atividade:** Implementação da Aplicação LBS Geofence SMS

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:** 2

**Justificativa de participação na atividade:** Realizou validação de produtos e de documentação produzida. Coordenou reunião de acompanhamento e avaliação. Realizou avaliação técnica dos produtos desenvolvidos até aquele momento, incluindo a aplicação LBS Geofence SMS. Participou de decisão conjunta sobre a melhor alternativa para o desenvolvimento dos próximos exemplos.

**14. Atividade:** Implementação da Aplicação GPS Geofence SMS

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:** 2

**Justificativa de participação na atividade:** Realizou validação de produtos e de documentação produzida. Coordenou reunião de acompanhamento e avaliação. Realizou avaliação técnica dos produtos desenvolvidos até aquele momento, incluindo a aplicação GPS Geofence SMS. Participou de decisão conjunta sobre a melhor alternativa para o desenvolvimento dos próximos exemplos.

**15. Atividade:** Implementação da Aplicação MQTT Hardware LBS

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:** 4

**Justificativa de participação na atividade:** Realizou validação de produtos e de documentação produzida. Coordenou reunião de acompanhamento e avaliação. Realizou avaliação técnica dos produtos desenvolvidos até aquele momento, incluindo a aplicação MQTT Hardware LBS. Participou de decisão conjunta sobre a melhor alternativa para o desenvolvimento dos próximos exemplos.

.

**16. Atividade:** Implementação da Aplicação MQTT Hardware

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:** 4

**Justificativa de participação na atividade:** Realizou validação de produtos e de documentação produzida. Coordenou reunião de acompanhamento e avaliação. Realizou avaliação técnica dos produtos desenvolvidos até aquele momento, incluindo a aplicação MQTT Hardware. Participou de decisão conjunta sobre a melhor alternativa para o desenvolvimento dos próximos exemplos.

**Nome:** Mariana Lins Urquiza

**CPF:**058.694.174-65

**Formação:** Especialização em Engenharia

**Cargo/Função:** Desenvolvedor

**Total de Horas:** 300

**1. Atividade:** Planejamento e Preparação

**Período de Execução:** 01/12/2017 a 31/12/2017

**Horas na Atividade:** 44

**Justificativa de participação na atividade:**

Nos dedicamos inicialmente ao entendimento do escopo do projeto, objetivos, metodologias, processos a serem utilizados durante o desenvolvimento e familiarização com as tecnologias e ferramentas a serem utilizadas nas atividades subsequentes.

1. **Entendimento do Escopo e Objetivos**

Inicialmente foi feita uma análise em equipe das solicitações e especificidades solicitadas para serem desenvolvidas. Ficou entendido que os objetivos a serem alcançados no final do projeto são:

* 1. Estudo e documentação no WIKI da placa Arduino M0 Pro
  2. Estudo e documentação no WIKI do módulo Telit GSM HE910
  3. Estudo e documentação no WIKI do módulo Telit Wifi GS2101M
  4. Estudo e documentação no WIKI do módulo Telit Bluetooth Bluemod+S42
  5. Estudo e documentação no WIKI da IDE Atmel Studio e Framework ASF (Advanced Software Framework)
  6. Estudo e documentação no WIKI da IDE e framework Arduino
  7. Desenvolvimento e documentação no WIKI de conjunto de bibliotecas em linguagem de programação C utilizando a IDE Atmel Studio, Framework ASF e sistema operacional de tempo real FreeRTOS para os módulos da Telit HE910, GS2101M e BlueMod+S42
  8. Desenvolvimento e documentação no WIKI de bibliotecas em linguagem de programação C++ utilizando a IDE/Framwork Arduino para os módulos da Telit HE910, GS2101M e BlueMod+S42
  9. Desenvolvimento e documentação no WIKI de aplicações de exemplo para as plataformas Atmel Studio e Arduino:
     1. Sistema logger baseado em modem: Tasks contínuas (RTOS) / Loop infinito (no Arduino) que repassa o tráfego de uma das portas seriais para um socket aberto com um servidor remoto, através dos comandos AT para sockets. Com a possibilidade de bufferizar os bytes que foram recebidos enquanto a conexão não estava ativa. As configurações de servidor, baudrate e afins devem ser manuais, através de constantes no código. O sistema deverá verificar que a conexão caiu (contexto com a APN ou o socket caíram) e tentar continuamente uma reconexão. Permitir também o fluxo de dados servidor→dispositivo, refletido na porta serial também. Basicamente um tunel de porta serial por rede celular e TCP.
     2. O mesmo sistema logger acima, baseado em WiFi, através dos comandos AT para sockets.
     3. O mesmo sistema logger acima, baseado em Bluetooth / comandos AT Terminal IO do módulo.
     4. Sistema logger de Bluetooth através de modem: coloca o módulo Bluetooth para ouvir beacons e broadcasts / advertisement próximos e repassa eles diretamente para um socket aberto no modem.
     5. Sistema logger de Bluetooth, baseado em Wifi.
     6. Sistema geofence para GPS: Determinar o pertencimento de uma latitude / longitude recebida através dos códigos NMEA emitidos automaticamente pelo módulo GPS, pela porta serial, com o interior polígono definido através de um array de coordenadas. Quando o fix, representado pela mensagem $GPGGA, estiver fora do polígono, enviar onda quadrada para um buzzer / falante e acender um LED.
     7. Sistema de geofence por LBS (#MONI). Utilização dos comandos AT#MONI e AT#AGPSSND para obtenção de posição por triangulação por rede celular, com análise de polígono considerando a grande imprecisão da posição obtida. Os comandos são simples de utilizar: AT#MONI=7 (Aguarda), AT#MONI, AT#AGPSSND (receberá lat/long). Devido à imprecisão, o status de ³dentro e fora do polígono² pode ser gradual, ao invés de binário / tudo ou nada.
     8. Publicação de variável em sistema MQTT, comandos AT de MQTT (método fácil) para modem: Criar task ou rotina (arduino) que lê uma fila de dados a serem enviados para o portal MQTT da Telit, em pares ³nome da da variável² e ³valor², e para cada item executa os comandos AT de post de variáveis por MQTT embutido nos módulos.
     9. Publicação de variável em sistema MQTT, comandos AT de Socket (método direto) para modem: Criar task ou rotina (arduino) que lê uma fila de dados a serem enviados para o portal MQTT da Telit, em pares ³nome da da variável² e ³valor², e para cada item efetua chamadas a uma biblioteca para o protocolo MQTT, se conectando ao servidor através dos comandos AT de socket do módulo.
     10. Publicação de variável em sistema MQTT, comandos AT de MQTT para o módulo WiFi.
     11. Publicação de variável em sistema MQTT, comandos AT de Socket para o módulo WiFi.
     12. Comunicação por SMS: task ou rotina que lê continuamente as mensagens SMS recebidas pelo módulo, lendo seu conteúdo e executando comandos conforme o texto enviado pela mensagem, com resposta por SMS.
     13. Sistema de atendimento de ligações com recebimento de comandos por DTMF e resposta/menu através de sons pré-gravados. Os comandos de detecção de DTMF possuem utilização simples e para tocar os sons basta chamar AT#APLAY=1,<diretório>,<nome\_do\_arquivo>. Ou AT#SAMR. Implementar menus de exemplo e em alguns deles o modem deverá enviar mensagens de SMS conforme o código apertado, enviar variáveis para o servidor MQTT conforme o código apertado.
     14. Sistema porteiro eletrônico: se um botão for pressionado, liga para um telefone predefinido.

1. **Definição da Metodologia de Desenvolvimento**

Utilizaremos o processo ágil Scrum como referência principal em relação a processo, utilizando a ferramenta de gestão de tarefas Trello utilizando como ferramenta de Kanban contendo os quadros “A Fazer”, “Fazendo”, “Bloqueadas”, “A Testar”, “A Documentar” e “Prontas”.

Colocaremos sempre como prioridades as atividades que possuem riscos técnicos, com o objetivo de sanamos eventuais problemas ou dificuldades no início da execução do projeto, antecipando a detecção do problema e sua resolução.

Os desenvolvedores farão os estudos das tecnologias em conjunto, de maneira horizontal, entretanto cada desenvolvedor será responsável por desenvolver e documentar um determinado escopo do projeto:

**Mariana Lins:** Desenvolvimento e documentação das bibliotecas na Plataforma Atmel Studio

**Pedro Granville:** Desenvolvimento e documentação das bibliotecas na Plataforma Arduino

**Roosevelt Vinicius:** Desenvolvimento e documentação das aplicações nas plataformas Atmel Studio e Arduino

E quando um dos desenvolvedores finalizar as suas atividades, irá ajudar nas atividades do outro que estiver com a maior quantidade de atividades a serem finalizadas.

Utilizaremos um repositório para controle de versão e não será necessário a utilização de branches, uma vez que cada desenvolvedor trabalhará em partes distintas dos códigos fontes e documentações. Os commits deverão ser feitos sempre que uma versão parcial ou final for concluída, sendo um intervalo máximo de 7 dias entre cada commit por um desenvolvedor.

1. **Definição, Instalação e Configuração das Ferramentas de Gestão**

Decidimos utilizar as seguintes ferramentas:

**Gestão de Conhecimento (WIKI):** Mkdocs + plugin Mkdocs-material

**Gestão de Tarefas:** Trello

**Controle de Versões:** Git utilizando a plataforma Bitbucket

Ficou decidido a seguinte divisão para configuração dessas ferramentas:

**Mariana Lins:** Criação e configuração do Mkdocs

**Pedro Granville:** Criação e configuração do Trello

**Roosevelt Vinicius:** Criação e configuração do Git no Bitbucket

1. **Análise das Tecnologias Envolvidas**
   1. Módulo Telit GSM HE910
      1. Conjunto de comandos ATs do módulo
      2. Conexão TCP cliente
      3. Sistema de Localização GPS
      4. Sistema de Localização LBS
      5. Protocolo MQTT
      6. Envio e Recebimento de SMS
      7. Lligação e Atendimento de Chamadas de Voz
      8. Detecção de tons DTMF
      9. Gravação e Reprodução de Arquivos de Audio
   2. Módulo Telit Wifi GS2101M
      1. Conjunto de comandos ATs do módulo
      2. Conexão TCP cliente
      3. MQTT
   3. Módulo Telit Bluetooth BlueMod+S42
      1. Bluetooth LE (Low Energy)
      2. Beacons e broadcasts / adverting
      3. RFCOMM
2. **Análise das Ferramentas de Desenvolvimento a serem Utilizadas**

Análise do framework ASF e sistema operacional de tempo real FreeRTOS

Foi feita a análise inicial do framework ASF e do sistema operacional de tempo real FreeRTOS onde cheguei nas seguintes conclusões:

O framework ASF é integrado com a IDE Atmel Studio e possui boa documentação e implementação de diversos drivers para a arquitetura de microcontroladores ATSAMD21, utilizada na placa Arduino M0 Pro. Os principais drivers que iremos utilizar serão os drivers de comunicação pela porta serial, driver de utilização de timers e GPIOs. Foi constatado ainda que o framework ASF já inclui também a integração com o sistema operacional FreeRTOS, o que facilitará a sua integração no projeto.

Um ponto negativo, é que não existe o projeto (chamado de board) pré configurado para a placa Arduino M0, de modo que será necessário fazer uma engenharia reversa e análise de diagrama esquemático da placa Arduino M0 para identificar quais pinos do microcontrolador são utilizados para cada porta serial e ainda suas definições de frequência de cristal, entre outras. O resultado desse trabalho será um arquivo de definição da placa Arduino M0 para o Atmel Studio / ASF.

**2. Atividade:** Estudo e Criação dos Projetos e Documentações Iniciais

**Período de Execução:** 01/12/2017 a 31/12/2017

**Horas na Atividade:** 11

**Justificativa de participação na atividade:**

O objetivo desta etapa é definir a estrutura de projeto para a plataforma Atmel Studio contendo os principais componentes de software que serão utilizados na implementação das bibliotecas e exemplos a serem desenvolvidos para esta plataforma. Para isso foram feitas as seguintes atividades:

**Proposta inicial de estrutura de projeto:**

Foi feita a especificação da estrutura de arquivos a ser adotada, organizando as bibliotecas, que serão comuns para todos os projetos e os projetos propriamente ditos, que conterão os arquivos específicos para cada um dos exemplos a serem desenvolvidos. Chegamos a seguinte estrutura:

* **asf:** Neste subdiretório conterão todos os projetos, bibliotecas e exemplos de aplicações para a plataforma Atmel Studio
  + **libs:** Neste subdiretório serão armazenados todos os códigos fontes compartilhados para todos os projetos de exemplos. Nele ficarão as bibliotecas dos módulos e implementações genéricas.
    - **inc:** Neste subdiretório ficarão os arquivos de cabeçalho (headers ou includes) compartilhados para todos os projetos de exemplos.
    - **src:** Neste subdiretório ficarão as implementações das bibliotecas dos módulos e implementações genéricas, comuns a todos os projetos de exemplos.
  + **examples:** Neste subdiretório ficarão as implementações de cada exemplo. E cada exemplo será composto por um projeto do Atmel Studio e seus códigos fontes específicos.
    - **he910:** Neste subdiretório estarão todos os projetos de exemplos para o módulo Telit GSM HE910
      * **tunnelTcp:** Estrutura interna do projeto e códigos fontes específicos
      * **geofenceGps:** Estrutura interna do projeto e códigos fontes específicos
      * **geofenceLbs:** Estrutura interna do projeto e códigos fontes específicos
      * **mqttHard:** Estrutura interna do projeto e códigos fontes específicos
      * **geofenceMqtt:** Estrutura interna do projeto e códigos fontes específicos
      * **comandosSms:** Estrutura interna do projeto e códigos fontes específicos
      * **geofenceSms:** Estrutura interna do projeto e códigos fontes específicos
      * **voiceMenuSms:** Estrutura interna do projeto e códigos fontes específicos
      * **EletronicPorter:** Estrutura interna do projeto e códigos fontes específicos
    - **gs2101m:** Neste subdiretório estarão todos os projetos de exemplos para o módulo Telit Wifi GS2101M
      * **tunnelTcp:** Estrutura interna do projeto e códigos fontes específicos
      * **mqttSoft:** Estrutura interna do projeto e códigos fontes específicos
    - **bluemod-s42:** Neste subdiretório estarão todos os projetos de exemplos para o módulo Telit Bluetooth Bluemod+S42
      * **tunnelTcp:** Estrutura interna do projeto e códigos fontes específicos
      * **beaconsTcp:** Estrutura interna do projeto e códigos fontes específicos

**Estudo e testes para criação do projeto Atmel Studio**

Foi criado com sucesso um projeto na IDE Atmel Studio para o mcu SAMD21G18 (o mcu utilizado na placa Arduino M0 Pro) com uma board customizada pelo usuário e inserido a versão mais atual do framework ASF.

**Estudo e testes para criação da board Arduino M0 Pro**

No Atmel Studio o termo “board” se refere às definições específicas de uma placa como, por exemplo, pinos de GPIO que estão ligados os LEDs ou quais portas seriais do microcontrolador serão utilizadas para cada finalidade.

Para a definição da board para a placa a ser utilizada, a Arduino M0 Pro, será necessário criar um header chamado conf\_ArduinoM0Pro\_board.h contendo essas definições da placa.

**Estudo e testes de utilização dos drivers ASF e integração com o FreeRTOS**

Em seguida foi feito um estudo inicial sobre o FreeRTOS e como integrá-lo a um projeto no Atmel Studio. Foi localizado uma documentação no próprio site do FreeRTOS com o passo a passo para isso no endereço: <http://ww1.microchip.com/downloads/en/AppNotes/Atmel-42138-Getting-Started-with-FreeRTOS-on-SAM-D20-D21-R21-L21-L22_ApplicationNote_AT03664.pdf>

**Estudo do sistema operacional de tempo real FreeRTOS**

Um microcontrolador (MCU) é um único chip com um processador simples que pode ser encontrado em diversos dispositivos, como aparelhos, sensores, controladores de aptidão física, automação industrial e automóveis. Muitos desses pequenos dispositivos podem se beneficiar com a conexão à nuvem ou com a conexão local a outros dispositivos. Por exemplo, medidores de consumo elétrico inteligentes precisam se conectar à nuvem para relatar o uso e sistemas de segurança predial precisam se comunicar localmente para destrancar uma porta quando você usa o crachá para entrar. Os microcontroladores têm poder computacional e capacidade de memória limitados e normalmente executam tarefas funcionais simples. Muitas vezes, os microcontroladores executam sistemas operacionais que não têm funcionalidades incorporadas para conexão a redes locais ou à nuvem, o que é um desafio para os aplicativos de IoT. O FreeRTOS ajuda a resolver esse problema oferecendo o sistema operacional central (para execução no dispositivo) e as bibliotecas de software que facilitam a conexão segura com a nuvem (ou com outros dispositivos) para que você possa coletar dados deles para os aplicativos de IoT e executar ações.

Por se tratar de um sistema operacional específico para microcontroladores, contendo características e metodologias de desenvolvimento específicas, foi necessário fazer um estudo profundo sobre o FreeRTOS para aprender os seus conceitos e mecanismos de aplicação.

O FreeRTOS oferece recursos como gerenciamento de memória, podendo utilizar alocação dinâmica ou estática, definido pelo usuário, implementações de filas, multi tarefas, controle de comunicação entre tarefas, utilizando mecanismos de mutexes e semáforos, além de muitos outros recursos e bibliotecas.

Os principais desafios identificados inicialmente serão:

1. Configuração do FreeRTOS para utilização no projeto
   1. Configuração de clocks e base de tempo
   2. Ajustes de tamanho de memória de Heap e Stack
   3. Definição das temporizações para cada task
2. Gerenciamento de troca de dados entre as tasks
   1. Como teremos pelo menos duas tasks em cada projeto, uma responsável pela aplicação principal, e a outra pelo gerenciamento da comunicação com o módulo (biblioteca), será necessário utilizar tanto filas quanto mutexes para fazer esse gerenciamento.

Para testes foi criado um projeto de exemplo utilizando o FreeRTOS com uma task básica.

**Documentação**

Por fim foi documentado no WIKI os conhecimentos adquiridos nesses testes e estudos.

**3. Atividade:** Testes dos Módulos e Montagem dos Ambientes de Desenvolvimento - Parte I

**Período de Execução:** 01/12/2017 a 31/12/2017

**Horas na Atividade:** 20

**Justificativa de participação na atividade:**

Recebemos 4 placas Arduino M0 Pro, 2 kits de desenvolvimento EVK2, do módulo HE910, 2 kits de desenvolvimento GainSpan GS2100M EVB3, do módulo GS2101M e 2 kits de desenvolvimento BlueEva, do módulo Bluemod+S42. Em comum acordo, decidimos que Pedro Granville iria testar o kit EVK2, eu iria testar o kit GainSpan GS2100M EVB3 e Roosevelt Vinicius o kit BlueEva.

**Testes do kit GainSpan GS2100M EVB3:**

Após os testes chegamos ao seguinte procedimento:

**Configuração do Ambiente de Hardware:**

Os procedimentos de setup inicial serão listados a seguir:

1 - Conectar a placa EVB3 através de seu conector microUSB (USB0) na porta USB do computador (a alimentação da placa ocorre pela USB do computador).

2 - Instalar o drive FTDI.

3 - Instalar o software serial TeraTerm VT termial.

4 - Executar o TeraTerm e configurar a porta serial como nas imagens abaixo:

**Obs. 1:** Os primeiros testes realizados com o kit EVB3 só funcionaram com o baudrate 460800 para a versão 3.5.0 do firmware de avaliação (AT+VER=?? -> comando que retorna a versão de firmware instalado na placa).

**Obs. 2:** O envio de comandos AT também podem ser realizados através do software AT Controller disponível na área de downloads no site da Telit.

Atualização de Firmware

O procedimento de atualização de firmware foi realizado seguindo a documentação técnica: GS2K Module Programming User Guide.

1 - Acessar o site http://www.gainspan.com.br e realizar o cadastro para ter acesso a portal de suporte.

2 - Clique em SDK Buikder e selecione o módulo GS2000.

Obs. 1: Clique em next na próxima tela.

3 - Na tela apresentada abaixo, foi selecionada opção para criação do firmware de avaliação da EVB3.

Obs: 1: Após clicar em next nesta tela, uma versão do firmware, com as configurações selecionadas, será criada. Em seguida ao clicar na aba Build History, o projeto do firmware, com a ferramenta de atualização e documentações, estará disponível para download em alguns minutos.

4 - Configure a placa para o modo de program na chave (vermelha), conecte a placa pela usb0 no computador e ligue a chave de alimentação da placa.

5 - Na pasta Tools está disponível a ferramenta de atualização de firmware GS\_programming\_tool.

Obs. 1: Selecione os parâmetros de conexão e clique em check connection para verificar se existe comunicação com a placa. Caso o software não consiga se comunicar com a placa basta selecionar o baudrate adequado, com o jumper J13 presente o baudrate é igua a 921600 e sem o jumper o baudrate é 115200.

Obs. 2: Em seguida, nas opções Super Block e Current FW Version, selecione os dois arquivos binários de atualização dentro da pasta Embedded\s2w e clique em Erase and Program.

Integração com Arduino M0 Pro

A Conexão da EVB3 com o arduino deve ser:

**Pino Arduino M0 Pro Pino EVB3**

GND GND

3.3V VIN\_3V3

10 GPIO0/UART0\_RX\_M

11 GPIO1/UART0\_TX\_M

09 EXT\_RTC\_RESET\_N

**4. Atividade:** Implementação de Biblioteca de Comandos AT

**Período de Execução:** 01/01/2018 a 31/01/2018

**Horas na Atividade:** 38

**Justificativa de participação na atividade:**

Implementação de Biblioteca de Comandos AT para a plaforma Atmel Studio

Foi feita a implementação da biblioteca de comunicação com as portas seriais do MCU Microchip SAMD21 (utilizado na placa Arduino M0 Pro) e de gerenciamento dos comandos AT para a plataforma de software Atmel Studio (ASF+FreeRTOS), necessários para que as bibliotecas a serem implementadas pudessem se comunicar com cada um dos módulos Telit.

Foram criados os códigos fontes:

**asf/libs/src/at.c:** Arquivo contendo a implementação do tratamento genérico dos comandos AT

**asf/libs/src/serial.c:** Arquivo contendo a implementação da porta serial utilizada pelas aplicações.

**asf/libs/src/serial2.c:** Arquivo contendo a implementação da porta serial auxiliar, utilizada para comunicação com os módulos.

**asf/libs/inc/at.h:** Arquivo de cabeçalho da implementação do tratamento genérico dos comandos AT.

**asf/libs/inc/serial.h:** Arquivo de cabeçalho d a implementação da porta serial utilizada pelas aplicações.

**asf/libs/inc/serial2.h:** Arquivo de cabeçalho da implementação da porta serial auxiliar, utilizada para comunicação com os módulos.

Foi criado ainda um projeto de teste para validação das 3 bibliotecas criadas. Esse projeto de testes foi evoluído e se transformou posteriormente na implementação da primeira aplicação de exemplo (tunnelTcp).

**5. Atividade:** Implementação de Firmware Base

**Período de Execução:** 01/01/2018 a 31/01/2018

**Horas na Atividade:** 7

**Justificativa de participação na atividade:**

A partir do projeto de testes implementado na etapa de desenvolvimento das bibliotecas de comandos AT, foi criado o projeto do primeiro exemplo de aplicação. Esse firmware deverá ser utilizado como base para o desenvolvimento dos projetos seguintes.

A partir da estrutura previamente definida, e das implementações de testes o FreeRTOS e comandos AT, foi criado o firmware. Entretanto ainda houveram diversos desafios, como a correta utilização e melhoria das implementações do FreeRTOS e ajustes e correções de bugs nas implementações de porta serial e comandos AT.

Por fim se chegou a uma versão satisfatória para termos como base para o desenvolvimento dos exemplos de aplicações.

**6. Atividade:** Testes dos Módulos e Montagem dos Ambientes de Desenvolvimento - Parte II

**Período de Execução:** 01/01/2018 a 31/01/2018

**Horas na Atividade:** 30

**Justificativa de participação na atividade:**

Utilizando o firmware base feito para a plataforma Atmel Studio, foram feitos novos testes com os kits de desenvolvimento EVK2 e GainSpan GS2100M EVB3, e feito um firmware de testes implementando os comandos AT básicos de inicialização do dois módulos (HE910 e GS2101M).

Esse firmware de testes servirá como base para implementação das bibliotecas dos módulos.

**7. Atividade:** Implementação das bibliotecas dos módulos

**Período de Execução:** 01/02/2018 a 28/02/2018

**Horas na Atividade:** 8

**Justificativa de participação na atividade:**

Criação inicial das bibliotecas de cada módulo, que implementam a *task* de gerenciamento de comunicação e funções que proveem as funcionalidades dos módulos para a camada de aplicação, utilizando a plataforma de software Atmel Studio (ASF+FreeRTOS).

Foram criados os seguintes códigos fontes:

**asf/libs/src/gs2101m.c:** Implementação das funções que disponibilizam as funcionalidades do módulo WIFI GS2101M.

**asf/libs/src/he910.c:** Implementação das funções que disponibilizam as funcionalidades do módulo GSM HE910.

**asf/libs/inc/gs2101m.h:** Arquivo de cabeçalho das funções que disponibilizam as funcionalidades do módulo WIFI GS2101M.

**asf/libs/inc/he910.h:** Arquivo de cabeçalho das funções que disponibilizam as funcionalidades do módulo GSM HE910.

**8. Atividade:** Implementação da Aplicação SMS Commands

**Período de Execução:** 01/02/2018 a 28/02/2018

**Horas na Atividade:** 8

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo HE910 lê continuamente as mensagens SMS recebidas pelo módulo, interpretando o seu conteúdo e executando comandos conforme o texto enviado pela mensagem, com resposta por SMS.

Foi criado o projeto de exemplo para a plataforma Atmel Studio chamado SMSCommands e uma task do FreeRTOS que implementa a aplicação.

Foram implementadas também as funções de envio e recebimento de SMS na biblioteca he910.c.

**9. Atividade:** Implementação da Aplicação Túnel TCP

**Período de Execução:** 01/02/2018 a 28/02/2018

**Horas na Atividade:** 39

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo HE910 é um sistema de logger baseado em modem que repassa o tráfego de uma das portas seriais da placa Arduino para um socket aberto com um servidor remoto, através dos comandos AT para sockets. Bufferiza os bytes que são recebidos enquanto a conexão não estava ativa. As configurações de servidor, baudrate e afins são manuais, através de constantes no código. O sistema verifica que a conexão caiu (contexto com a APN ou o socket) e tenta continuamente uma reconexão. Permite também o fluxo de dados servidor->dispositivo, refletido na porta serial. Um túnel de porta serial por TCP.

No desenvolvimento dessa aplicação, foram consolidados a utilização efetiva do FreeRTOS e das implementações de acesso às portas seriais na plataforma Atmel Studio. A aplicação foi desenvolvida para ser o modelo para as próximas.

Foram criados o projeto Atmel Studio da aplicação de exemplo chamado tunnelTcp e a task FreeRTOS contendo a lógica da aplicação do túnel, chamado tunnelTcp.c.

Foram implementadas também as funções de conexão, escrita, leitura e desconexão TCP na biblioteca he910.c.

**10. Atividade:** Implementação da Aplicação Porteiro Eletrônico

**Período de Execução:**01/02/2018 a 28/02/2018

**Horas na Atividade:**20

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo HE910 implementa um sistema de porteiro eletrônico que, se um botão for pressionado, liga para um número de telefone predefinido e estabelece uma comunicação de voz entre eles.

Foram criados o projeto Atmel Studio da aplicação de exemplo chamado intercomGsm e a task FreeRTOS contendo a lógica da aplicação do túnel, chamado intercomGsm.c.

Foram implementadas também as funções de chamadas de voz e atendimento de chamadas na biblioteca he910.c.

**11. Atividade:** Implementação da Aplicação LBS Geofence SMS

**Período de Execução:** 01/03/2018 a 31/03/2018

**Horas na Atividade:** 8

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo HE910 implementa uma associação entre o sistema de geofence com envio de posições lidas a partir do LBS através do sistema de comunicação por SMS.

Foram criados o projeto Atmel Studio da aplicação de exemplo chamado lbsGeofenceSms e a task FreeRTOS contendo a lógica da aplicação, chamado lbsGeofenceSms.c.

Foram implementadas também as funções de localização através de LBS na biblioteca he910.c.

**12. Atividade:** Implementação da Aplicação GPS Geofence

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:**14

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo HE910 determina o pertencimento de uma latitude/longitude recebida através dos códigos NMEA emitidos automaticamente pelo módulo GPS, pela porta serial, com o interior polígono definido através de um array de coordenadas. Quando o fix, representado pela mensagem $GPGGA, está fora do polígono, envia onda quadrada para um buzzer/falante e acende um LED.

Foram criados o projeto Atmel Studio da aplicação de exemplo chamado gpsGeofence e a task FreeRTOS contendo a lógica da aplicação, chamado gpsGeofence.c.

Foram implementadas também as funções de localização através da tecnologia GPS na biblioteca he910.c.

**13. Atividade:** Implementação da Aplicação MQTT LBS

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:**11

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo HE910 é uma associação do sistema de geofence com publicação de posições lidas a partir do LBS em um servidor MQTT.

Foram criados o projeto Atmel Studio da aplicação de exemplo chamado lbsMqtt e a task FreeRTOS contendo a lógica da aplicação, chamado lbsMqtt.c.

Foram implementadas também as funções de conexão, escrita, leitura e desconexão MQTT na biblioteca he910.c.

**14. Atividade:** Implementação da Aplicação GPS Geofence SMS

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:**9

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo HE910 é uma associação entre o sistema de geofence com envio de posições lidas a partir do GPS através do sistema de comunicação por SMS.

Foram criados o projeto Atmel Studio da aplicação de exemplo chamado gpsGeofenceSms e a task FreeRTOS contendo a lógica da aplicação, chamado gpsGeofenceSms.c.

**15. Atividade:** Implementação da Aplicação MQTT

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:**14

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo GS2101M publica variáveis em sistema MQTT que lê uma fila de dados a serem enviados para o portal MQTT da Telit, em pares "nome da variável" e "valor", e para cada item executa os comandos de post de variáveis por MQTT.

Foram criados o projeto Atmel Studio da aplicação de exemplo chamado mqtt e a task FreeRTOS contendo a lógica da aplicação, chamado mqtt.c.

Foi feita a integração da biblioteca Paho Embedeed MQTT C/C++ (http://www.eclipse.org/paho/clients/c/embedded/) com o projeto, para a implementação de protocolo MQTT por software.

**16. Atividade:** Implementação da Aplicação Túnel TCP

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:**19

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo GS2101M é um sistema de logger baseado em modem que repassa o tráfego de uma das portas seriais da placa Arduino para um socket aberto com um servidor remoto, através dos comandos AT para sockets. Bufferiza os bytes que são recebidos enquanto a conexão não estava ativa. As configurações de servidor, baudrate e afins são manuais, através de constantes no código. O sistema verifica que a conexão caiu e tenta continuamente uma reconexão. Permite também o fluxo de dados servidor->dispositivo, refletido na porta serial. Um túnel de porta serial por TCP.

Foram criados o projeto Atmel Studio da aplicação de exemplo chamado tunnelTcp e a task FreeRTOS contendo a lógica da aplicação do túnel, chamado tunnelTcp.c.

**Nome:** Mateus Assis Máximo de Lima

**CPF:**046.479.104-90

**Formação:** Mestrado em Engenharia Elétrica

**Cargo/Função:** Coordenador do Projeto

**Total de Horas:** 40

**1. Atividade:** Estudo e Criação dos Projetos e Documentações Iniciais

**Período de Execução:** 01/12/2017 a 31/12/2017

**Horas na Atividade:**4

**Justificativa de participação na atividade:** Coordenação do projeto.

**2. Atividade:** Testes dos Módulos e Montagem dos Ambientes de Desenvolvimento - Parte I

**Período de Execução:** 01/12/2017 a 31/12/2017

**Horas na Atividade:** 2

**Justificativa de participação na atividade:** Coordenação do projeto.

**3. Atividade:** Planejamento e Preparação

**Período de Execução:** 01/12/2017 a 31/12/2017

**Horas na Atividade:**4

**Justificativa de participação na atividade:** Coordenação do projeto.

**4. Atividade:** Implementação de Firmware Base

**Período de Execução:** 01/01/2018 a 31/01/2018

**Horas na Atividade:** 3

**Justificativa de participação na atividade:** Reunião de acompanhamento. Coordenação do projeto.

**5. Atividade:** Testes dos Módulos e Montagem dos Ambientes de Desenvolvimento - Parte II

**Período de Execução:** 01/01/2018 a 31/01/2018

**Horas na Atividade:** 3

**Justificativa de participação na atividade:** Coordenação do projeto.

**6. Atividade:** Implementação de Biblioteca de Comandos AT

**Período de Execução:** 01/01/2018 a 31/01/2018

**Horas na Atividade:** 4

**Justificativa de participação na atividade:** Coordenação do projeto.

**7. Atividade:** Implementação da Aplicação Túnel TCP

**Período de Execução:** 01/02/2018 a 28/02/2018

**Horas na Atividade:** 2

**Justificativa de participação na atividade:** Coordenação do projeto.

**8. Atividade:** Implementação da Aplicação SMS Commands

**Período de Execução:** 01/02/2018 a 28/02/2018

**Horas na Atividade:** 2

**Justificativa de participação na atividade:** Coordenação do projeto.

**9. Atividade:** Implementação da Aplicação Porteiro Eletrônico

**Período de Execução:** 01/02/2018 a 28/02/2018

**Horas na Atividade:** 4

**Justificativa de participação na atividade:** Reunião de acompanhamento. Coordenação do projeto.

**10. Atividade:** Implementação das bibliotecas dos módulos

**Período de Execução:** 01/02/2018 a 28/02/2018

**Horas na Atividade:**2

**Justificativa de participação na atividade:** Coordenação do projeto.

**11. Atividade:** Implementação da Aplicação MQTT Hardware LBS

**Período de Execução:** 01/03/2018 a 31/03/2018

**Horas na Atividade:**2

**Justificativa de participação na atividade:** Coordenação do projeto.

**12. Atividade:** Implementação da Aplicação GPS Geofence

**Período de Execução:** 01/03/2018 a 31/03/2018

**Horas na Atividade:**1

**Justificativa de participação na atividade:** Coordenação do projeto. Encerramento.

**13. Atividade:** Implementação da Aplicação LBS Geofence SMS

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:**2

**Justificativa de participação na atividade:** Coordenação do projeto.

**14. Atividade:** Implementação da Aplicação LBS Geofence

**Período de Execução:** 01/03/2018 a 31/03/2018

**Horas na Atividade:** 1

**Justificativa de participação na atividade:** Coordenação do projeto.

**15. Atividade:** Implementação da Aplicação GPS Geofence SMS

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:** 2

**Justificativa de participação na atividade:** Coordenação do projeto.

**16. Atividade:**I mplementação da Aplicação MQTT Hardware

**Período de Execução:** 01/03/2018 a 31/03/2018

**Horas na Atividade:** 2

**Justificativa de participação na atividade:** Coordenação do projeto.

**Nome:** Pedro Granville Gonçalves

**CPF:** 039.104.924-01

**Formação:** Mestrado em Engenharia

**Cargo/Função:** Desenvolvedor

**Total de Horas:** 464

**1. Atividade:** Planejamento e Preparação

**Período de Execução:** 01/12/2017 a 31/12/2017

**Horas na Atividade:** 44

**Justificativa de participação na atividade:**

Nos dedicamos inicialmente ao entendimento do escopo do projeto, objetivos, metodologias, processos a serem utilizados durante o desenvolvimento e familiarização com as tecnologias e ferramentas a serem utilizadas nas atividades subsequentes.

1. **Entendimento do Escopo e Objetivos**

Inicialmente foi feita uma análise em equipe das solicitações e especificidades solicitadas para serem desenvolvidas. Ficou entendido que os objetivos a serem alcançados no final do projeto são:

* 1. Estudo e documentação no WIKI da placa Arduino M0 Pro
  2. Estudo e documentação no WIKI do módulo Telit GSM HE910
  3. Estudo e documentação no WIKI do módulo Telit Wifi GS2101M
  4. Estudo e documentação no WIKI do módulo Telit Bluetooth Bluemod+S42
  5. Estudo e documentação no WIKI da IDE Atmel Studio e Framework ASF (Advanced Software Framework)
  6. Estudo e documentação no WIKI da IDE e framework Arduino
  7. Desenvolvimento e documentação no WIKI de conjunto de bibliotecas em linguagem de programação C utilizando a IDE Atmel Studio, Framework ASF e sistema operacional de tempo real FreeRTOS para os módulos da Telit HE910, GS2101M e BlueMod+S42
  8. Desenvolvimento e documentação no WIKI de bibliotecas em linguagem de programação C++ utilizando a IDE/Framwork Arduino para os módulos da Telit HE910, GS2101M e BlueMod+S42
  9. Desenvolvimento e documentação no WIKI de aplicações de exemplo para as plataformas Atmel Studio e Arduino:
     1. Sistema logger baseado em modem: Tasks contínuas (RTOS) / Loop infinito (no Arduino) que repassa o tráfego de uma das portas seriais para um socket aberto com um servidor remoto, através dos comandos AT para sockets. Com a possibilidade de bufferizar os bytes que foram recebidos enquanto a conexão não estava ativa. As configurações de servidor, baudrate e afins devem ser manuais, através de constantes no código. O sistema deverá verificar que a conexão caiu (contexto com a APN ou o socket caíram) e tentar continuamente uma reconexão. Permitir também o fluxo de dados servidor→dispositivo, refletido na porta serial também. Basicamente um tunel de porta serial por rede celular e TCP.
     2. O mesmo sistema logger acima, baseado em WiFi, através dos comandos AT para sockets.
     3. O mesmo sistema logger acima, baseado em Bluetooth / comandos AT Terminal IO do módulo.
     4. Sistema logger de Bluetooth através de modem: coloca o módulo Bluetooth para ouvir beacons e broadcasts / advertisement próximos e repassa eles diretamente para um socket aberto no modem.
     5. Sistema logger de Bluetooth, baseado em Wifi.
     6. Sistema geofence para GPS: Determinar o pertencimento de uma latitude / longitude recebida através dos códigos NMEA emitidos automaticamente pelo módulo GPS, pela porta serial, com o interior polígono definido através de um array de coordenadas. Quando o fix, representado pela mensagem $GPGGA, estiver fora do polígono, enviar onda quadrada para um buzzer / falante e acender um LED.
     7. Sistema de geofence por LBS (#MONI). Utilização dos comandos AT#MONI e AT#AGPSSND para obtenção de posição por triangulação por rede celular, com análise de polígono considerando a grande imprecisão da posição obtida. Os comandos são simples de utilizar: AT#MONI=7 (Aguarda), AT#MONI, AT#AGPSSND (receberá lat/long). Devido à imprecisão, o status de ³dentro e fora do polígono² pode ser gradual, ao invés de binário / tudo ou nada.
     8. Publicação de variável em sistema MQTT, comandos AT de MQTT (método fácil) para modem: Criar task ou rotina (arduino) que lê uma fila de dados a serem enviados para o portal MQTT da Telit, em pares ³nome da da variável² e ³valor², e para cada item executa os comandos AT de post de variáveis por MQTT embutido nos módulos.
     9. Publicação de variável em sistema MQTT, comandos AT de Socket (método direto) para modem: Criar task ou rotina (arduino) que lê uma fila de dados a serem enviados para o portal MQTT da Telit, em pares ³nome da da variável² e ³valor², e para cada item efetua chamadas a uma biblioteca para o protocolo MQTT, se conectando ao servidor através dos comandos AT de socket do módulo.
     10. Publicação de variável em sistema MQTT, comandos AT de MQTT para o módulo WiFi.
     11. Publicação de variável em sistema MQTT, comandos AT de Socket para o módulo WiFi.
     12. Comunicação por SMS: task ou rotina que lê continuamente as mensagens SMS recebidas pelo módulo, lendo seu conteúdo e executando comandos conforme o texto enviado pela mensagem, com resposta por SMS.
     13. Sistema de atendimento de ligações com recebimento de comandos por DTMF e resposta/menu através de sons pré-gravados. Os comandos de detecção de DTMF possuem utilização simples e para tocar os sons basta chamar AT#APLAY=1,<diretório>,<nome\_do\_arquivo>. Ou AT#SAMR. Implementar menus de exemplo e em alguns deles o modem deverá enviar mensagens de SMS conforme o código apertado, enviar variáveis para o servidor MQTT conforme o código apertado.
     14. Sistema porteiro eletrônico: se um botão for pressionado, liga para um telefone predefinido.

1. **Definição da Metodologia de Desenvolvimento**

Utilizaremos o processo ágil Scrum como referência principal em relação a processo, utilizando a ferramenta de gestão de tarefas Trello utilizando como ferramenta de Kanban contendo os quadros “A Fazer”, “Fazendo”, “Bloqueadas”, “A Testar”, “A Documentar” e “Prontas”.

Colocaremos sempre como prioridades as atividades que possuem riscos técnicos, com o objetivo de sanamos eventuais problemas ou dificuldades no início da execução do projeto, antecipando a detecção do problema e sua resolução.

Os desenvolvedores farão os estudos das tecnologias em conjunto, de maneira horizontal, entretanto cada desenvolvedor será responsável por desenvolver e documentar um determinado escopo do projeto:

**Mariana Lins:** Desenvolvimento e documentação das bibliotecas na Plataforma Atmel Studio

**Pedro Granville:** Desenvolvimento e documentação das bibliotecas na Plataforma Arduino

**Roosevelt Vinicius:** Desenvolvimento e documentação das aplicações nas plataformas Atmel Studio e Arduino

E quando um dos desenvolvedores finalizar as suas atividades, irá ajudar nas atividades do outro que estiver com a maior quantidade de atividades a serem finalizadas.

Utilizaremos um repositório para controle de versão e não será necessário a utilização de branches, uma vez que cada desenvolvedor trabalhará em partes distintas dos códigos fontes e documentações. Os commits deverão ser feitos sempre que uma versão parcial ou final for concluída, sendo um intervalo máximo de 7 dias entre cada commit por um desenvolvedor.

1. **Definição, Instalação e Configuração das Ferramentas de Gestão**

Decidimos utilizar as seguintes ferramentas:

**Gestão de Conhecimento (WIKI):** Mkdocs + plugin Mkdocs-material

**Gestão de Tarefas:** Trello

**Controle de Versões:** Git utilizando a plataforma Bitbucket

Ficou decidido a seguinte divisão para configuração dessas ferramentas:

**Mariana Lins:** Criação e configuração do Mkdocs

**Pedro Granville:** Criação e configuração do Trello

**Roosevelt Vinicius:** Criação e configuração do Git no Bitbucket

1. **Análise das Tecnologias Envolvidas**
   1. Módulo Telit GSM HE910
      1. Conjunto de comandos ATs do módulo
      2. Conexão TCP cliente
      3. Sistema de Localização GPS
      4. Sistema de Localização LBS
      5. Protocolo MQTT
      6. Envio e Recebimento de SMS
      7. Lligação e Atendimento de Chamadas de Voz
      8. Detecção de tons DTMF
      9. Gravação e Reprodução de Arquivos de Audio
   2. Módulo Telit Wifi GS2101M
      1. Conjunto de comandos ATs do módulo
      2. Conexão TCP cliente
      3. MQTT
   3. Módulo Telit Bluetooth BlueMod+S42
      1. Bluetooth LE (Low Energy)
      2. Beacons e broadcasts / adverting
      3. RFCOMM
2. **Análise das Ferramentas de Desenvolvimento a serem Utilizadas**

Foi feita a análise inicial do framework Arduino onde cheguei nas seguintes conclusões:

Para a utilização da placa Arduino M0 Pro na plataforma Arduino é necessário a instalação adicional da placa na IDE. Para isso basta ir na opção Tools/Board/Board Manager e instalar a placa “Arduino SAMD boards”.

Também é necessária a instalação do driver EDBG (mecanismos de gravação e debug utilizado na placa Arduino M0 Pro) entretanto no Windows essa instalação é feita automaticamente, no momento que se conecta a placa no computador.

**2. Atividade:** Estudo e Criação dos Projetos e Documentações Iniciais

**Período de Execução:** 01/12/2017 a 31/12/2017

**Horas na Atividade:** 11

**Justificativa de participação na atividade:**

**Estudo de Procedimento para Criação de Bibliotecas e Exemplos na IDE Arduino**

Foi feito uma pesquisa de algumas bibliotecas contendo exemplos integrados com a IDE arduino e, a partir de suas estruturas, foi identificado o procedimento a ser feito para essa integração.

A IDE Arduino possibilita integrar em sua estrutura interna tanto a biblioteca quanto os exemplos desenvolvidos, criando uma estrutura pré definida pela IDE Arduino, da seguinte forma:

* diretório base da biblioteca
  + **doc:** subdiretório contendo arquivos textos contendo as documentações da biblioteca.
  + **examples:** subdiretório que contém as estruturas de projetos e arquivos .ino que implementam cada exemplo
  + **arquivos .h e .cpp**: implementações da biblioteca

**Criação de Estrutura Inicial**

Foi definida e criada a estrutura inicial para as bibliotecas e exemplos da plataforma Arduino, de forma a serem integradas com a IDE.

**Documentação**

Esse procedimento foi documentado no WIKI, com textos descritivos e imagens ilustrando os procedimentos de configuração.

**3. Atividade:** Testes dos Módulos e Montagem dos Ambientes de Desenvolvimento - Parte I

**Período de Execução:** 01/12/2017 a 31/12/2017

**Horas na Atividade:** 20

**Justificativa de participação na atividade:**

Recebemos 4 placas Arduino M0 Pro, 2 kits de desenvolvimento EVK2, do módulo HE910, 2 kits de desenvolvimento GainSpan GS2100M EVB3, do módulo GS2101M e 2 kits de desenvolvimento BlueEva, do módulo Bluemod+S42. Em comum acordo, decidimos que eu iria testar o kit EVK2, Mariana Lins iria testar o kit GainSpan GS2100M EVB3 e Roosevelt Vinicius o kit BlueEva.

**Testes do kit GainSpan GS2100M EVB3:**

Após os testes chegamos ao seguinte procedimento:

**Configuração do Ambiente de Hardware:**

O ambiente de hardware utilizado para os exemplos consiste na integração da placa Arduino M0 Pro com o Kit de Desenvolvimento EVK2.

Os procedimentos de setup inicial serão listados a seguir:

1 - Baixar e instalar os drives USB's do link: Drivers..

2 - Conectar a placa CS1467C através de seu conector miniUSB na porta USB do computador, conectar a fonte de alimentação na placa base (EVK2) e pressionar o botão de on por 5s.

3 - Para testar a comunicação com o módulo através de comandos AT, deve-se instalar o software serial Telit AT Controller, disponível no mesmo link usado anteriormente pra baixar os drivers.

Configurada a conexão, basta clicar então no botão de "Connect", que é o botão da esquerda no meio da tela. Dando tudo certo e com o módulo conectado, são exibidas uma série de informações tais como IMEI, Fabricante, Modelo e versão de Firmware no centro da tela do programa, indicando que o mesmo está comunicando e recebendo dados do módulo.

Integração com Arduino M0 Pro

A Conexão da EVK2 com o arduino deve ser de acordo com a seguinte pinagem:

Pino Arduino M0 Pro Pino EVK2

GND GND

10 C104/RXD

11 C103/TXD

09 RESET

As descrições dos pinos utilizados na placa EVK2 podem ser observadas documento: EVK2 User Guide

**4. Atividade:** Implementação de Biblioteca de Comandos AT

**Período de Execução:** 01/01/2018 a 31/01/2018

**Horas na Atividade:** 38

**Justificativa de participação na atividade:**

Foi criada a biblioteca de comandos AT genérica para a plataforma Arduino já dentro da estrutura do projeto para o módulo HE910. Foi criado também um projeto de teste para a primeira implementação que evoluiu posteriormente para se transformar no exemplo tunnelTcp.

Os arquivos desenvolvidos foram:

**arduino/Telit\_GSM/AT.cpp:** Implementação dos médodos da biblioteca de comandos AT para arduino.

**arduino/Telit\_GSM/AT.h:** Definição da classe da biblioteca de comandos AT para arduino.

**5. Atividade:** Implementação de Firmware Base

**Período de Execução:** 01/01/2018 a 31/01/2018

**Horas na Atividade:** 7

**Justificativa de participação na atividade:**

O firmware base foi desenvolvido a partir do firmware de testes utilizado no desenvolvimento da biblioteca de comandos AT. Poucos ajustes precisaram ser feitos já que a plataforma Arduino é simplista e de fácil implementação.

**6. Atividade:** Testes dos Módulos e Montagem dos Ambientes de Desenvolvimento - Parte II

**Período de Execução:** 01/01/2018 a 31/01/2018

**Horas na Atividade:** 30

**Justificativa de participação na atividade:**

Com o firmware base feito para a plataforma Arduino, foram efetuados testes com os kits de desenvolvimento EVK2 e GainSpan GS2100M EVB3, e feito um firmware de testes implementando os comandos AT básicos de inicialização do dois módulos (HE910 e GS2101M).

Esse firmware de testes servirá como base para implementação das bibliotecas dos módulos.

**7. Atividade:** Implementação das bibliotecas dos módulos

**Período de Execução:** 01/02/2018 a 28/02/2018

**Horas na Atividade:** 8

**Justificativa de participação na atividade:**

Foram implementadas as classes das bibliotecas para os módulos HE910 e GS2101M. Por enquanto foram feitas apenas os métodos de inicialização e conexão nas redes. O restante das funcionalidades serão implementadas de acordo com a necessidade, durante o desenvolvimento das aplicações de exemplo.

Foram criados os seguintes códigos fontes:

arduino/Telit\_GSM/HE910G.cpp

arduino/Telit\_GSM/HE910G.h

arduino/Telit\_Wifi/GS2101M.cpp

arduino/Telit\_Wifi/GS2101M.h

**8. Atividade:** Implementação da Aplicação SMS Commands para o Módulo HE910

**Período de Execução:** 01/02/2018 a 28/02/2018

**Horas na Atividade:** 8

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo HE910 lê continuamente as mensagens SMS recebidas pelo módulo, interpretando o seu conteúdo e executando comandos conforme o texto enviado pela mensagem, com resposta por SMS.

Foi criado o exemplo para a plataforma Arduino que implementa a aplicação chamado Telit\_GSM\_HE910G\_SMSCommands.ino.

Foram implementadas também as funções de envio e recebimento de SMS na biblioteca HE910.cpp.

**9. Atividade:** Implementação da Aplicação Túnel TCP para o Módulo HE910

**Período de Execução:** 01/02/2018 a 28/02/2018

**Horas na Atividade:** 39

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo HE910 é um sistema de logger baseado em modem que repassa o tráfego de uma das portas seriais da placa Arduino para um socket aberto com um servidor remoto, através dos comandos AT para sockets. Bufferiza os bytes que são recebidos enquanto a conexão não estava ativa. As configurações de servidor, baudrate e afins são manuais, através de constantes no código. O sistema verifica que a conexão caiu (contexto com a APN ou o socket) e tenta continuamente uma reconexão. Permite também o fluxo de dados servido<-->dispositivo, refletido na porta serial. Um tunel de porta serial por TCP.

Foi criado o exemplo para a plataforma Arduino que implementa a aplicação chamado Telit\_GSM\_HE910G\_UartSocketTunnel.ino.

Foram implementadas também as funções de conexão, escrita, leitura e desconexão TCP na biblioteca HE910.cpp.

**10. Atividade:** Implementação da Aplicação Porteiro Eletrônico para o Módulo HE910

**Período de Execução:**01/02/2018 a 28/02/2018

**Horas na Atividade:**20

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo HE910 implementa um sistema de porteiro eletrônico que, se um botão for pressionado, liga para um número de telefone predefinido e estabelece uma comunicação de voz entre eles.

Foi criado o exemplo para a plataforma Arduino que implementa a aplicação chamado Telit\_GSM\_HE910G\_Porteiro\_Eletronico.ino.

Foram implementadas também as funções de chamadas de voz e atendimento de chamadas na biblioteca HE910.cpp.

**11. Atividade:** Implementação da Aplicação LBS Geofence SMS para o Módulo HE910

**Período de Execução:** 01/03/2018 a 31/03/2018

**Horas na Atividade:** 8

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo HE910 implementa uma associação entre o sistema de geofence com envio de posições lidas a partir do LBS através do sistema de comunicação por SMS.

Foi criado o exemplo para a plataforma Arduino que implementa a aplicação chamado Telit\_GPS\_HE910G\_LBS\_Geofence\_SMS.ino.

Foram implementadas também as funções de localização através de LBS na biblioteca HE910.cpp.

**12. Atividade:** Implementação da Aplicação GPS Geofence para o Módulo HE910

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:**14

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo HE910 determina o pertencimento de uma latitude/longitude recebida através dos códigos NMEA emitidos automaticamente pelo módulo GPS, pela porta serial, com o interior polígono definido através de um array de coordenadas. Quando o fix, representado pela mensagem $GPGGA, está fora do polígono, envia onda quadrada para um buzzer/falante e acende um LED.

Foi criado o exemplo para a plataforma Arduino que implementa a aplicação chamado Telit\_GPS\_HE910G\_GPS\_Geofence.ino.

Foram implementadas também as funções de localização através da tecnologia GPS na biblioteca HE910.cpp.

**13. Atividade:** Implementação da Aplicação MQTT LBS para o Módulo HE910

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:**11

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo HE910 é uma associação do sistema de geofence com publicação de posições lidas a partir do LBS em um servidor MQTT.

Foi criado o exemplo para a plataforma Arduino que implementa a aplicação chamado Telit\_GSM\_UE910GL\_HARDWARE\_MQTT\_LBS.ino.

Foram implementadas também as funções de conexão, escrita, leitura e desconexão MQTT na biblioteca HE910.cpp.

**14. Atividade:** Implementação da Aplicação GPS Geofence SMS para o Módulo HE910

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:**9

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo HE910 é uma associação entre o sistema de geofence com envio de posições lidas a partir do GPS através do sistema de comunicação por SMS.

Foi criado o exemplo para a plataforma Arduino que implementa a aplicação chamado Telit\_GPS\_HE910G\_GPS\_Geofence\_SMS.ino.

**15. Atividade:** Implementação da Aplicação MQTT para o Módulo GS2101M

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:**14

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo GS2101M publica variáveis em sistema MQTT que lê uma fila de dados a serem enviados para o portal MQTT da Telit, em pares "nome da variável" e "valor", e para cada item executa os comandos de post de variáveis por MQTT.

Foi criado o exemplo para a plataforma Arduino que implementa a aplicação chamado Telit\_GSM\_GS2101M\_MQTT.ino.

**16. Atividade:** Implementação da Aplicação Túnel TCP para o Módulo GS2101M

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:**19

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo GS2101M é um sistema de logger baseado em modem que repassa o tráfego de uma das portas seriais da placa Arduino para um socket aberto com um servidor remoto, através dos comandos AT para sockets. Bufferiza os bytes que são recebidos enquanto a conexão não estava ativa. As configurações de servidor, baudrate e afins são manuais, através de constantes no código. O sistema verifica que a conexão caiu e tenta continuamente uma reconexão. Permite também o fluxo de dados servidor->dispositivo, refletido na porta serial. Um túnel de porta serial por TCP.

Foi criado o exemplo para a plataforma Arduino que implementa a aplicação chamado Telit\_Wifi\_GS2101M\_UartTcpTunnel.ino.

**Nome:** Roosevelt Vinícius Chaves de Souza

**CPF:**072.296.004-21

**Formação:** Especialização em Desenvolvimento de Sistemas Móveis

**Cargo/Função:** Desenvolvedor

**Total de Horas:** 464

**1. Atividade:** Planejamento e Preparação

**Período de Execução:** 01/12/2017 a 31/12/2017

**Horas na Atividade:** 44

**Justificativa de participação na atividade:**

Nos dedicamos inicialmente ao entendimento do escopo do projeto, objetivos, metodologias, processos a serem utilizados durante o desenvolvimento e familiarização com as tecnologias e ferramentas a serem utilizadas nas atividades subsequentes.

1. **Entendimento do Escopo e Objetivos**

Inicialmente foi feita uma análise em equipe das solicitações e especificidades solicitadas para serem desenvolvidas. Ficou entendido que os objetivos a serem alcançados no final do projeto são:

* 1. Estudo e documentação no WIKI da placa Arduino M0 Pro
  2. Estudo e documentação no WIKI do módulo Telit GSM HE910
  3. Estudo e documentação no WIKI do módulo Telit Wifi GS2101M
  4. Estudo e documentação no WIKI do módulo Telit Bluetooth Bluemod+S42
  5. Estudo e documentação no WIKI da IDE Atmel Studio e Framework ASF (Advanced Software Framework)
  6. Estudo e documentação no WIKI da IDE e framework Arduino
  7. Desenvolvimento e documentação no WIKI de conjunto de bibliotecas em linguagem de programação C utilizando a IDE Atmel Studio, Framework ASF e sistema operacional de tempo real FreeRTOS para os módulos da Telit HE910, GS2101M e BlueMod+S42
  8. Desenvolvimento e documentação no WIKI de bibliotecas em linguagem de programação C++ utilizando a IDE/Framwork Arduino para os módulos da Telit HE910, GS2101M e BlueMod+S42
  9. Desenvolvimento e documentação no WIKI de aplicações de exemplo para as plataformas Atmel Studio e Arduino:
     1. Sistema logger baseado em modem: Tasks contínuas (RTOS) / Loop infinito (no Arduino) que repassa o tráfego de uma das portas seriais para um socket aberto com um servidor remoto, através dos comandos AT para sockets. Com a possibilidade de bufferizar os bytes que foram recebidos enquanto a conexão não estava ativa. As configurações de servidor, baudrate e afins devem ser manuais, através de constantes no código. O sistema deverá verificar que a conexão caiu (contexto com a APN ou o socket caíram) e tentar continuamente uma reconexão. Permitir também o fluxo de dados servidor→dispositivo, refletido na porta serial também. Basicamente um tunel de porta serial por rede celular e TCP.
     2. O mesmo sistema logger acima, baseado em WiFi, através dos comandos AT para sockets.
     3. O mesmo sistema logger acima, baseado em Bluetooth / comandos AT Terminal IO do módulo.
     4. Sistema logger de Bluetooth através de modem: coloca o módulo Bluetooth para ouvir beacons e broadcasts / advertisement próximos e repassa eles diretamente para um socket aberto no modem.
     5. Sistema logger de Bluetooth, baseado em Wifi.
     6. Sistema geofence para GPS: Determinar o pertencimento de uma latitude / longitude recebida através dos códigos NMEA emitidos automaticamente pelo módulo GPS, pela porta serial, com o interior polígono definido através de um array de coordenadas. Quando o fix, representado pela mensagem $GPGGA, estiver fora do polígono, enviar onda quadrada para um buzzer / falante e acender um LED.
     7. Sistema de geofence por LBS (#MONI). Utilização dos comandos AT#MONI e AT#AGPSSND para obtenção de posição por triangulação por rede celular, com análise de polígono considerando a grande imprecisão da posição obtida. Os comandos são simples de utilizar: AT#MONI=7 (Aguarda), AT#MONI, AT#AGPSSND (receberá lat/long). Devido à imprecisão, o status de ³dentro e fora do polígono² pode ser gradual, ao invés de binário / tudo ou nada.
     8. Publicação de variável em sistema MQTT, comandos AT de MQTT (método fácil) para modem: Criar task ou rotina (arduino) que lê uma fila de dados a serem enviados para o portal MQTT da Telit, em pares ³nome da da variável² e ³valor², e para cada item executa os comandos AT de post de variáveis por MQTT embutido nos módulos.
     9. Publicação de variável em sistema MQTT, comandos AT de Socket (método direto) para modem: Criar task ou rotina (arduino) que lê uma fila de dados a serem enviados para o portal MQTT da Telit, em pares ³nome da da variável² e ³valor², e para cada item efetua chamadas a uma biblioteca para o protocolo MQTT, se conectando ao servidor através dos comandos AT de socket do módulo.
     10. Publicação de variável em sistema MQTT, comandos AT de MQTT para o módulo WiFi.
     11. Publicação de variável em sistema MQTT, comandos AT de Socket para o módulo WiFi.
     12. Comunicação por SMS: task ou rotina que lê continuamente as mensagens SMS recebidas pelo módulo, lendo seu conteúdo e executando comandos conforme o texto enviado pela mensagem, com resposta por SMS.
     13. Sistema de atendimento de ligações com recebimento de comandos por DTMF e resposta/menu através de sons pré-gravados. Os comandos de detecção de DTMF possuem utilização simples e para tocar os sons basta chamar AT#APLAY=1,<diretório>,<nome\_do\_arquivo>. Ou AT#SAMR. Implementar menus de exemplo e em alguns deles o modem deverá enviar mensagens de SMS conforme o código apertado, enviar variáveis para o servidor MQTT conforme o código apertado.
     14. Sistema porteiro eletrônico: se um botão for pressionado, liga para um telefone predefinido.

1. **Definição da Metodologia de Desenvolvimento**

Utilizaremos o processo ágil Scrum como referência principal em relação a processo, utilizando a ferramenta de gestão de tarefas Trello utilizando como ferramenta de Kanban contendo os quadros “A Fazer”, “Fazendo”, “Bloqueadas”, “A Testar”, “A Documentar” e “Prontas”.

Colocaremos sempre como prioridades as atividades que possuem riscos técnicos, com o objetivo de sanamos eventuais problemas ou dificuldades no início da execução do projeto, antecipando a detecção do problema e sua resolução.

Os desenvolvedores farão os estudos das tecnologias em conjunto, de maneira horizontal, entretanto cada desenvolvedor será responsável por desenvolver e documentar um determinado escopo do projeto:

**Mariana Lins:** Desenvolvimento e documentação das bibliotecas na Plataforma Atmel Studio

**Pedro Granville:** Desenvolvimento e documentação das bibliotecas na Plataforma Arduino

**Roosevelt Vinicius:** Desenvolvimento e documentação das aplicações nas plataformas Atmel Studio e Arduino

E quando um dos desenvolvedores finalizar as suas atividades, irá ajudar nas atividades do outro que estiver com a maior quantidade de atividades a serem finalizadas.

Utilizaremos um repositório para controle de versão e não será necessário a utilização de branches, uma vez que cada desenvolvedor trabalhará em partes distintas dos códigos fontes e documentações. Os commits deverão ser feitos sempre que uma versão parcial ou final for concluída, sendo um intervalo máximo de 7 dias entre cada commit por um desenvolvedor.

1. **Definição, Instalação e Configuração das Ferramentas de Gestão**

Decidimos utilizar as seguintes ferramentas:

**Gestão de Conhecimento (WIKI):** Mkdocs + plugin Mkdocs-material

**Gestão de Tarefas:** Trello

**Controle de Versões:** Git utilizando a plataforma Bitbucket

Ficou decidido a seguinte divisão para configuração dessas ferramentas:

**Mariana Lins:** Criação e configuração do Mkdocs

**Pedro Granville:** Criação e configuração do Trello

**Roosevelt Vinicius:** Criação e configuração do Git no Bitbucket

1. **Análise das Tecnologias Envolvidas**
   1. Módulo Telit GSM HE910
      1. Conjunto de comandos ATs do módulo
      2. Conexão TCP cliente
      3. Sistema de Localização GPS
      4. Sistema de Localização LBS
      5. Protocolo MQTT
      6. Envio e Recebimento de SMS
      7. Lligação e Atendimento de Chamadas de Voz
      8. Detecção de tons DTMF
      9. Gravação e Reprodução de Arquivos de Audio
   2. Módulo Telit Wifi GS2101M
      1. Conjunto de comandos ATs do módulo
      2. Conexão TCP cliente
      3. MQTT
   3. Módulo Telit Bluetooth BlueMod+S42
      1. Bluetooth LE (Low Energy)
      2. Beacons e broadcasts / adverting
      3. RFCOMM
2. **Análise das Ferramentas de Desenvolvimento a serem Utilizadas**

Foi feita a análise inicial das ferramentas de software e kits de desenvolvimento de cada módulo da Telit onde cheguei nas seguintes conclusões:

Os três módulos da Telit utilizam uma mesma ferramenta de testes chamada “Telit AT Controller”. Essa ferramenta disponibiliza um terminal de comunicação e diversos exemplos de sequências de comandos AT para teste das principais funcionalidades de cada módulo. Esses exemplos poderão ser utilizados como base para a criação das bibliotecas, além de possibilitarem os testes iniciais dos módulos.

Em seguida foi feito uma pesquisa dos kits de desenvolvimento de cada módulo. Nessa pesquisa foram identificadas as documentações e conhecidas as suas características. Os kits de desenvolvimento dos módulos HE910, GS2101M e BlueMod+S42 são respectivamente: Kit de desenvolvimento EVK2, Kit de Desenvolvimento GainSpan GS2100M EVB3 e Kit de desenvolvimento BlueEva+S42. As suas referências e documentações foram inseridas no WIKI.

**2. Atividade:** Estudo e Criação dos Projetos e Documentações Iniciais

**Período de Execução:** 01/12/2017 a 31/12/2017

**Horas na Atividade:** 11

**Justificativa de participação na atividade:**

Para entendimento com mais detalhes das funcionalidades e características de cada módulo, foi feito um estudo sobre cada um dos módulos a serem utilizados no projeto:

1. **HE910**

O módulo HE910 faz parte da família de módulos GSM da Telit xE910 destinados para aplicações automotivas e telemáticas 3G e 2G. Disponibiliza recursos avançados como:

* 1. Efetua e recebe chamadas de voz
  2. Envia e recebe mensagens de texto SMS
  3. Envia e recebe tons DTMF
  4. Localização GPS
  5. Localização LBS
  6. Pilha TCP/IP com conexões Socket TCP e UDP, podendo ter até 16 conexões simultaneas
  7. Protocolos HTTP, MQTT e outros.

1. **GS2101M**

A série GS2101M é um módulo Wi-Fi (IEEE 802.11 b / g / n) totalmente certificado e seguro que oferece integração rápida e econômica da conectividade Wi-Fi aos projetos de produtos. A série está disponível com uma antena de rastreio PCB opcional ou conector U.FL.

**Características:**

* 3 ADCs sigma-delta de 16 bits, ideais para dispositivos inteligentes de medição e energia
* 18 x 25 x 2,7 mm 40 pinos
* Medição de alta resolução e medição de energia
* Modo sem host (sem MCU externo), executando Wi-Fi completo e pilha de rede TCP/IP
* Modo host, a biblioteca de comandos AT mais abrangente conectando-se a qualquer microcontrolador host de 8/16/32 bits usando interfaces UART, SPI ou SDIO
* Fácil provisionamento de smartphones com AP limitado ou Wi-Fi Protected Set-up (WPS)
* Baixo consumo com recursos de gerenciamento de energia: 2,7V - 3,6V, 3,3V nominal

1. **BlueMod+S42**

O BlueMod+S42 é um módulo plug-and-play Bluetooth 4.2 Low Energy, de modo único, qualificado, de alta performance, com Bluetooth 4.2. Ele apresenta baixo consumo de energia e flexibilidade para projetistas como o perfil I/O de terminal para casos de uso de substituição de cabo serial, bem como a maioria dos perfis Bluetooth baseados em GATT por meio de uma implementação genérica do GATT. Parte da família BlueMod + Sx, o BlueMod + S42 é intercambiável com as gerações anteriores sem precisar de novos spins de hardware ou software. Embora seja um módulo de baixo consumo de energia de modo único, ele oferece uma alta taxa de transferência de até 470kbit/s usando o SPP exclusivo da Telit, como o serviço Terminal I/O.

**Características:**

* Plataforma IOT autônoma ou aplicativos hospedados por Bluetooth
* Módulo qualificado Bluetooth v4.2
* RED, FCC, IC, KC, MIC, RCM e Certificação Anatel
* Poderoso Cortex M4F MCU
* Projetado para aplicações de baixa potência
* Melhor do seu desempenho de intervalo de classe
* Melhor desempenho de rendimento da sua classe
* Cliente genérico GATT ou implementação de servidor
* Comunicações ponto a ponto ou ponto a multiponto
* Perfil de E/S de terminal (semelhante a SPP) para transferência de dados transparente rápida e fácil
* Perfil de E/S de automação para transferência de informações de status digital e analógico
* I/O de terminal e código de amostra de E/S de automação para iOS e Android
* Suporte NFC para emparelhamento simples seguro
* LE Secure Connection com criptografia AES de 128 bits

**3. Atividade:** Testes dos Módulos e Montagem dos Ambientes de Desenvolvimento - Parte I

**Período de Execução:** 01/12/2017 a 31/12/2017

**Horas na Atividade:** 20

**Justificativa de participação na atividade:**

Recebemos 4 placas Arduino M0 Pro, 2 kits de desenvolvimento EVK2, do módulo HE910, 2 kits de desenvolvimento GainSpan GS2100M EVB3, do módulo GS2101M e 2 kits de desenvolvimento BlueEva, do módulo Bluemod+S42. Em comum acordo, decidimos que Pedro Granville iria testar o kit EVK2, Mariana Lins iria testar o kit GainSpan GS2100M EVB3 e eu o kit BlueEva.

**Testes do Kit BlueEva+S42:**

Não tivemos sucesso com os testes com este kit. Após várias tentativas decidimos fazer contato com a equipe da Telit através do seguinte email:

***Subject:*** *Suporte técnico para o módulo BluDev+S42*

*Oi, pessoal.*

*Boa tarde.*

*Vocês poderiam nos indicar um contato para suporte técnico relativo ao módulo BluDev+S42?*

*Segue abaixo uma breve descrição do problema. Em resumo, não estamos conseguindo nenhuma resposta aos comandos AT enviados, mesmo seguindo toda a documentação encontrada. Conseguimos nos comunicar com comandos AT com os outros módulos, menos esse. Testamos em todos os baudrates e configurações possíveis, sem sucesso. O estranho é que é o mesmo comportamento nos dois módulos. Então ainda pode ser que seja algum procedimento errado nosso.*

*Precisamos também liberar os chips GSM que vieram com o kit GSM. Contudo, o link que é passado na documentação para desbloquear não existe. Fizemos contato com o canal de suporte da Telit e estamos aguardando resposta. Terias algum canal mais rápido para nos ajudar?*

*Problema*

*As duas placas recebidas modelo: BlueDev+S42, não respondem aos comandos AT’s enviados pelo software AT controller, disponibilizado na área de download do site da Telit, nem pelo software terminal TeraTerm. Também não respondem ao procedimento de atualização de firmware.*

*Procedimento adotado*

*Foram seguidos detalhadamente os procedimentos listados no documento disponibilizado no site do fabricante do módulo (Telit), encontrado no link abaixo. Foram observados, desde a posição dos jumpers, como também a instalação dos drivers, softwares e configuração da porta serial para o devido funcionamento da placa.*

*https://www.telit.com/wp-content/uploads/2018/01/BlueEvaS42\_Central\_User\_Guide\_r1.pdf*

*https://www.telit.com/wp-content/uploads/2016/08/BlueMod\_S42\_Software\_User\_Guide\_r1.pdf*

Após esse contato tivemos a seguinte resposta da equipe da Telit:

*Subject: RE: Suporte técnico para o módulo BluDev+S42 | Case Number: 00027852*

*Boa tarde,*

*Verifiquei esse P/N e este é um kit de desenvolvimento voltado para a programação em C interna do módulo. Desta forma ele não vem com um firmware próprio que contempla os comandos AT. Estou verificando internamente o que pode ser feito nesse caso e já te retorno na sequência.*

*Quanto a ativação do simcard que vêm junto com o kit estou montando algumas instruções de ativação e já passo para você. Você poderia me passar o ICCID dos simcards? Este número geralmente esta impresso no simcard ou pode ser obtido pelo comando AT: AT+CCID*

*Att,*

*Jonatas Pazin*

*AMERICAS Technical Support*

*Telit Application Engineering*

Dessa forma ficou constatado que de fato havia um problema nos kits BlueEva enviado para nós. O problema estava no tipo de firmware carregados nos módulso dos kits enviados para nós e ficamos no aguardo do envio dos kits com o firmware correto.

**4. Atividade:** Implementação de Biblioteca de Comandos AT

**Período de Execução:** 01/01/2018 a 31/01/2018

**Horas na Atividade:** 38

**Justificativa de participação na atividade:**

Foi feito o levantamento e testes das sequências de comandos AT necessários para a conexão com a rede GSM/GPRS para o módulo HE910 e para conexão com a rede WIFI para o módulo GS2101M. Esses comandos serão utilizados em seguida para iniciar as implementações dos primeiros exemplos de aplicações.

**5. Atividade:** Implementação de Firmware Base

**Período de Execução:** 01/01/2018 a 31/01/2018

**Horas na Atividade:** 7

**Justificativa de participação na atividade:**

Foi feito o levantamento dos comandos AT dos módulos HE910 e GS2101M para cada função:

* Inicialização do módulo
* Conexão na rede
* Conexão TCP
* Envio de dados TCP
* Recebimento de dados TCP

Após esse levantamento foram feitos testes para validação dos comandos utilizando a ferramenta de software Telit AT Controller.

**6. Atividade:** Testes dos Módulos e Montagem dos Ambientes de Desenvolvimento - Parte II

**Período de Execução:** 01/01/2018 a 31/01/2018

**Horas na Atividade:** 30

**Justificativa de participação na atividade:**

Foi dado o suporte para Mariana Lins e Pedro Granville para configuração de hardware, configuração e utilização dos kits de desenvolvimento EVK2 e GainSpan GS2100M EVB3.

A maior parte do esforço dedicado nesta atividade foi o envio correto e tratamento das respostas dos comandos AT de inicialização dos módulos e conexão com as redes GSM e Wifi.

**7. Atividade:** Implementação das bibliotecas dos módulos

**Período de Execução:** 01/02/2018 a 28/02/2018

**Horas na Atividade:** 8

**Justificativa de participação na atividade:**

Foi feito testes e validações em todos os comandos AT dos módulos HE910 e GS2101M utilizados nas implementações feitas por Mariana Lins e Pedro Granville. As sequência de comandos para cada função foram sendo validadas uma a uma antes de passar para eles.

**8. Atividade:** Implementação da Aplicação SMS Commands para o Módulo HE910

**Período de Execução:** 01/02/2018 a 28/02/2018

**Horas na Atividade:** 8

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo HE910 lê continuamente as mensagens SMS recebidas pelo módulo, interpretando o seu conteúdo e executando comandos conforme o texto enviado pela mensagem, com resposta por SMS.

Foi feito testes, validações e acompanhamento da implementação dos comandos de envio e recebimento de SMS para o módulo HE910. Foi feito ainda a especificação do funcionamento da aplicação, testes e documentação no Wiki.

**9. Atividade:** Implementação da Aplicação Túnel TCP para o Módulo HE910

**Período de Execução:** 01/02/2018 a 28/02/2018

**Horas na Atividade:** 39

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo HE910 é um sistema de logger baseado em modem que repassa o tráfego de uma das portas seriais da placa Arduino para um socket aberto com um servidor remoto, através dos comandos AT para sockets. Bufferiza os bytes que são recebidos enquanto a conexão não estava ativa. As configurações de servidor, baudrate e afins são manuais, através de constantes no código. O sistema verifica que a conexão caiu (contexto com a APN ou o socket) e tenta continuamente uma reconexão. Permite também o fluxo de dados servidor->dispositivo, refletido na porta serial. Um tunel de porta serial por TCP.

Foram feitos testes, validações e acompanhamento da implementação dos comandos de abertura de conexão, escrita, leitura e desconexão do protocolo TCP para o módulo HE910. Foi feito ainda a especificação do funcionamento da aplicação, testes e documentação no Wiki.

**10. Atividade:** Implementação da Aplicação Porteiro Eletrônico para o Módulo HE910

**Período de Execução:**01/02/2018 a 28/02/2018

**Horas na Atividade:**20

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo HE910 implementa um sistema de porteiro eletrônico que, se um botão for pressionado, liga para um número de telefone predefinido e estabelece uma comunicação de voz entre eles.

Foi feito testes, validações e acompanhamento da implementação dos comandos de chamada de voz para o módulo HE910. Foi feito ainda a especificação do funcionamento da aplicação, testes e documentação no Wiki.

**11. Atividade:** Implementação da Aplicação LBS Geofence SMS para o Módulo HE910

**Período de Execução:** 01/03/2018 a 31/03/2018

**Horas na Atividade:** 8

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo HE910 implementa uma associação entre o sistema de geofence com envio de posições lidas a partir do LBS através do sistema de comunicação por SMS.

Foi feito testes, validações e acompanhamento da implementação dos comandos de localização através de LBS para o módulo HE910. Foi feito ainda a especificação do funcionamento da aplicação, testes e documentação no Wiki.

**12. Atividade:** Implementação da Aplicação GPS Geofence para o Módulo HE910

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:**14

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo HE910 determina o pertencimento de uma latitude/longitude recebida através dos códigos NMEA emitidos automaticamente pelo módulo GPS, pela porta serial, com o interior polígono definido através de um array de coordenadas. Quando o fix, representado pela mensagem $GPGGA, está fora do polígono, envia onda quadrada para um buzzer/falante e acende um LED.

Foi feito testes, validações e acompanhamento da implementação dos comandos de localização através da tecnologia GPS para o módulo HE910. Foi feito ainda a especificação do funcionamento da aplicação, testes e documentação no Wiki.

**13. Atividade:** Implementação da Aplicação MQTT LBS para o Módulo HE910

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:**11

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo HE910 é uma associação do sistema de geofence com publicação de posições lidas a partir do LBS em um servidor MQTT.

Foi feito testes, validações e acompanhamento da implementação dos comandos de conexão, leitura, escrita e desconexão MQTT para o módulo HE910. Foi feito ainda a especificação do funcionamento da aplicação, testes e documentação no Wiki.

**14. Atividade:** Implementação da Aplicação GPS Geofence SMS para o Módulo HE910

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:**9

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo HE910 é uma associação entre o sistema de geofence com envio de posições lidas a partir do GPS através do sistema de comunicação por SMS.

Foi feita a especificação do funcionamento da aplicação, testes e documentação no Wiki.

**15. Atividade:** Implementação da Aplicação MQTT para o Módulo GS2101M

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:**14

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo GS2101M publica variáveis em sistema MQTT que lê uma fila de dados a serem enviados para o portal MQTT da Telit, em pares "nome da variável" e "valor", e para cada item executa os comandos de post de variáveis por MQTT.

Foi feita a especificação do funcionamento da aplicação, testes e documentação no Wiki.

**16. Atividade:** Implementação da Aplicação Túnel TCP para o Módulo GS2101M

**Período de Execução:**01/03/2018 a 31/03/2018

**Horas na Atividade:**19

**Justificativa de participação na atividade:**

Essa aplicação de exemplo para o módulo GS2101M é um sistema de logger baseado em modem que repassa o tráfego de uma das portas seriais da placa Arduino para um socket aberto com um servidor remoto, através dos comandos AT para sockets. Bufferiza os bytes que são recebidos enquanto a conexão não estava ativa. As configurações de servidor, baudrate e afins são manuais, através de constantes no código. O sistema verifica que a conexão caiu e tenta continuamente uma reconexão. Permite também o fluxo de dados servidor->dispositivo, refletido na porta serial. Um túnel de porta serial por TCP.

Foi feito testes, validações e acompanhamento da implementação dos comandos de conexão, leitura, escrita e desconexão TCP para o módulo GS2101M. Foi feito ainda a especificação do funcionamento da aplicação, testes e documentação no Wiki.

1. **RESULTADOS DO PROJETO**

**Atividades executadas:**

**Atividade:** Implementação da Aplicação LBS Geofence

Esta aplicação tem como finalidade demonstrar a aplicação de *Geofence*, que é a delimitação de uma área geográfica através de coordenadas de latitude e longitude e detecção quando a localização do dispositivo ultrapassar os limites dessa área. Mais informações sobre Geofence podem ser obtidas [aqui](http://www.decom.ufop.br/imobilis/tutorial-android-criando-e-monitorando-geofences/)). Dentre outras finalidades, a tecnologia de Geofence pode ser aplicada em áreas como segurança, como tornozeleiras eletrônicas, e de logísticas, como controle de rotas e frotas de caminhões. Nessa demonstração foi utilizada a tecnologia de localização chamada LBS, que é alternativa ao GPS, utilizando a triangulação das antenas de celular para obter a localização. A principal vantagem dessa tecnologia é a possibilidade de obter a localização em ambientes fechados, contanto que exista cobertura de celular.

**Período de Execução:** 28/02/2018 a 30/03/2018

**Justificativa:** Esta atividade está associada a(s) etapa(s) APLICAÇÕES DE EXEMPLOS do projeto, Implementação da Aplicação LBS Geofence.

**Participantes:** Francisco Fechine Borges

Mariana Lins Urquiza

Mateus Assis Maximo de Lima

Pedro Granville Gonçalves

Roosevelt Vinícius Chaves de Souza

**Atividade:** Implementação da Aplicação MQTT Hardware LBS

MQTT é uma tecnologia recente que surgiu a partir da popularização da Internet das Coisas (IoT). Se trata de um protocolo de comunicação, semelhante ao HTTP (protocolo utilizado para transmissão de páginas WEB). Entretanto o MQTT é destinado para que pequenos dispositivos conectados à internet como, por exemplo, câmeras IP ou rastreadores, possam se comunicar com o seu servidor na nuvem e assim termos acesso aos seus recursos pela WEB. Esta aplicação demonstra a utilização desse protocolo integrado com o sistema de localização LBS.

**Período de Execução:** 26/03/2018 a 30/03/2018

**Justificativa:** Esta atividade está associada a(s) etapa(s) APLICAÇÕES DE EXEMPLOS do projeto, Implementação da Aplicação MQTT Hardware LBS.

**Participantes:** Francisco Fechine Borges

Mariana Lins Urquiza

Mateus Assis Maximo de Lima

Pedro Granville Gonçalves

Roosevelt Vinícius Chaves de Souza

**Atividade:** Implementação da Aplicação GPS Geofence SMS

Esta aplicação tem como finalidade demonstrar a aplicação de *Geofence* (delimitação de uma área geográfica através de coordenadas de latitude e longitude e detecção quando a localização do dispositivo ultrapassar os limites dessa área. Mais informações sobre Geofence podem ser obtidas [aqui](http://www.decom.ufop.br/imobilis/tutorial-android-criando-e-monitorando-geofences/)). Dentre outras finalidades, a tecnologia de Geofence pode ser aplicada em áreas como segurança, como tornozeleiras eletrônicas, e de logísticas, como controle de rotas e frotas de caminhões. Nessa demonstração foi utilizada a tecnologia de localização GPS e SMS para envio das notificações, sempre que o dispositivo ultrapassar um dos limites da área previamente configurada. Este exemplo pode ser utilizado para implementar uma tornozeleira eletrônica, por exemplo.

**Período de Execução:** 14/03/2018 a 30/03/2018

**Justificativa:** Esta atividade está associada a(s) etapa(s) APLICAÇÕES DE EXEMPLOS do projeto, Implementação da Aplicação GPS Geofence SMS.

**Participantes:** Francisco Fechine Borges

Mariana Lins Urquiza

Mateus Assis Maximo de Lima

Pedro Granville Gonçalves

Roosevelt Vinícius Chaves de Souza

**Atividade:** Testes dos Módulos e Montagem dos Ambientes de Desenvolvimento - Parte I

Ao recebermos os módulos e placas de desenvolvimento, efetuamos testes funcionais e configuração dos softwares de testes e demonstrações, que acompanham os módulos. Nesse processo, foi detectado que um dos módulos (BlueMod+S42) estava com problema de fábrica. Ao detectarmos esse problema, comunicamos imediatamente aos fornecedores que constataram que, de fato, houve um erro no modelo de módulo enviado e se responsabilizaram pelos encaminhamentos para resolução.

**Período de Execução:**21/12/2017 a 30/12/2017

**Justificativa:** Esta atividade está associada a(s) etapa(s) ESTUDO DOS MÓDULOS E PLATAFORMAS DE MONTAGEM DOS AMBIENTES DE DESENVOLVIMENTO do projeto, Testagem inicial dos módulos e montagem dos ambientes de desenvolvimento.

**Participantes:** Francisco Fechine Borges

Mariana Lins Urquiza

Mateus Assis Maximo de Lima

Pedro Granville Gonçalves

Roosevelt Vinícius Chaves de Souza

**Atividade:** Implementação de Firmware Base

Foi desenvolvimento uma versão base do firmware para testar as plataformas de software e comunicação com cada um dos módulos para, a partir dessa versão base, desenvolvermos as versões definitivas de cada implementação.

**Período de Execução:** 14/01/2018 a 30/01/2018

**Justificativa:** Esta atividade está associada a(s) etapa(s) IMPLEMENTAÇÃO DAS BIBLIOTECAS DE COMUNICAÇÃO COM OS MÓDULOS do projeto, Atividades de implementação de firmware base, para suporte à comunicação com os módulos.

**Participantes:** Francisco Fechine Borges

Mariana Lins Urquiza

Mateus Assis Maximo de Lima

Pedro Granville GOnçalves

Roosevelt Vinícius Chaves de Souza

**Atividade:** Implementação de Biblioteca de Comandos AT

A biblioteca de Comandos AT é uma implementação de software genérica, utilizada em todas as implementações, que permite que a biblioteca de cada módulo se comunique com os dispositivos físicos, através do protocolo Hayes, conhecido com comandos AT. Essa implementação (para Arduino e Atmel Studio) poderá ser utilizada em qualquer outro projeto futuro que utilize esse tipo de protocolo, se tornando uma biblioteca de uso universal.

**Período de Execução:** 16/01/2018 a 30/01/2018

**Justificativa:** Esta atividade está associada a(s) etapa(s) IMPLEMENTAÇÃO DAS BIBLIOTECAS DE COMUNICAÇÃO COM OS MÓDULOS do projeto, Implementação da biblioteca que permite o envio de comandos AT para os módulos.

**Participantes:** Francisco Fechine Borges

Mariana Lins Urquiza

Mateus Assis Maximo de Lima

Pedro Granville Gonçalves

Roosevelt Vinícius Chaves de Souza

**Atividade:** Implementação da Aplicação MQTT Hardware

MQTT é uma tecnologia recente que surgiu a partir da popularização da Internet das Coisas (IoT). Se trata de um protocolo de comunicação, semelhante ao HTTP (protocolo utilizado para transmissão de páginas WEB). Entretanto o MQTT é destinado para que pequenos dispositivos conectados à internet como, por exemplo, câmeras IP ou rastreadores, possam se comunicar com o seu servidor na nuvem e assim termos acesso aos seus recursos pela WEB.

**Período de Execução:** 20/03/2018 a 30/03/2018

**Justificativa:** Esta atividade está associada a(s) etapa(s) APLICAÇÕES DE EXEMPLOS do projeto, Implementação da Aplicação MQTT Hardware.

**Participantes:** Francisco Fechine Borges

Mariana Lins Urquiza

Mateus Assis Maximo de Lima

Pedro Granville Gonçalves

Roosevelt Vinícius Chaves de Souza

**Atividade:** Implementação da Aplicação GPS Geofence

Esta aplicação tem como finalidade demonstrar a aplicação de Geofence (delimitação de uma área geográfica através de coordenadas de latitude e longitude e detecção quando a localização do dispositivo ultrapassar os limites dessa área. Mais informações sobre Geofence podem ser obtidas [aqui](http://www.decom.ufop.br/imobilis/tutorial-android-criando-e-monitorando-geofences/)). Dentre outras finalidades, a tecnologia de Geofence pode ser aplicada em áreas como segurança, como tornozeleiras eletrônicas, e de logísticas, como controle de rotas e frotas de caminhões. Nessa demonstração foi utilizada a tecnologia de localização GPS.

**Período de Execução:** 08/03/2018 a 30/03/2018

**Justificativa:** Esta atividade está associada a(s) etapa(s) APLICAÇÕES DE EXEMPLOS do projeto, Implementação da Aplicação GPS Geofence.

**Participantes:** Francisco Fechine Borges

Mariana Lins Urquiza

Mateus Assis Maximo de Lima

Pedro Granville Gonçalves

Roosevelt Vinícius Chaves de Souza

**Atividade:** Planejamento e Preparação

Para a sinergia e eficiência da equipe de desenvolvimento durante a execução do projeto, nos dedicamos ao entendimento do escopo do projeto, objetivos, metodologias, processos a serem utilizados durante o desenvolvimento e familiarização com as tecnologias e ferramentas a serem utilizadas nas atividades subsequentes. Estudo detalhado das funcionalidades e características de cada módulo da Telit.

**Período de Execução:** 03/12/2017 a 30/12/2017

**Justificativa:** Esta atividade está associada a(s) etapa(s) ESTUDO DOS MÓDULOS E PLATAFORMAS DE MONTAGEM DOS AMBIENTES DE DESENVOLVIMENTO do projeto, Planejamento e preparação das atividades.

**Participantes:** Francisco Fechine Borges

Mariana Lins Urquiza

Mateus Assis Maximo de Lima

Pedro Granville Gonçalves

Roosevelt Vinícius Chaves de Souza

**Atividade:** Implementação das bibliotecas dos módulos

Trata-se da construção das bibliotecas de software para as plataformas Arduíno e Atmel Studio + AFS + FreeRTOS, que disponibilizam todos os recursos existentes nos módulos da Telit (GS2101M e HE910), para serem utilizamos de maneira fácil e documentada pelos desenvolvedores que pretendem desenvolver aplicações com esses módulos.

**Período de Execução:** 31/01/2018 a 27/02/2018

**Justificativa:** Esta atividade está associada a(s) etapa(s) IMPLEMENTAÇÃO DAS BIBLIOTECAS DOS MÓDULOS do projeto, Implementação das bibliotecas dos módulos, baseadas nas bibliotecas de comunicação.

**Participantes:** Francisco Fechine Borges

Mariana Lins Urquiza

Mateus Assis Maximo de Lima

Pedro Granville GOnçalves

Roosevelt Vinícius Chaves de Souza

**Atividade:** Implementação da Aplicação LBS Geofence SMS

Esta aplicação tem como finalidade demonstrar a aplicação de Geofence (delimitação de uma área geográfica através de coordenadas de latitude e longitude e detecção quando a localização do dispositivo ultrapassar os limites dessa área. Mais informações sobre Geofence podem ser obtidas [aqui](http://www.decom.ufop.br/imobilis/tutorial-android-criando-e-monitorando-geofences/)). Dentre outras finalidades, a tecnologia de Geofence pode ser aplicada em áreas como segurança, como tornozeleiras eletrônicas, e de logísticas, como controle de rotas e frotas de caminhões. Nessa demonstração foi utilizada a tecnologia de localização LBBS e SMS para envio das notificações, sempre que o dispositivo ultrapassar um dos limites da área previamente configurada. Este exemplo pode ser utilizado para implementar uma tornozeleira eletrônica, entre outras aplicações.

**Período de Execução:**18/03/2018 a 30/03/2018

**Justificativa:** Esta atividade está associada a(s) etapa(s) APLICAÇÕES DE EXEMPLOS do projeto, Implementação da Aplicação LBS Geofence SMS.

**Participantes:** Francisco Fechine Borges

Mariana Lins Urquiza

Mateus Assis Maximo de Lima

Pedro Granville GOnçalves

Roosevelt Vinícius Chaves de Souza

**Atividade:** Testes dos Módulos e Montagem dos Ambientes de Desenvolvimento - Parte II

Ao efetuarmos o procedimento de atualização de firmware dos módulos, se fez necessário um novo ciclo de testes e atualização dos ambientes de desenvolvimento, verificando a resolução de bugs existentes nas versões anteriores dos módulos e softwares de testes.

**Período de Execução:** 31/12/2017 a 30/01/2018

**Justificativa:** Esta atividade está associada a(s) etapa(s) ESTUDO DOS MÓDULOS E PLATAFORMAS DE MONTAGEM DOS AMBIENTES DE DESENVOLVIMENTO do projeto, Testagem dos Módulos e Montagem dos Ambientes de Desenvolvimento - Parte II.

**Participantes:** Francisco Fechine Borges

Mariana Lins Urquiza

Mateus Assis Maximo de Lima

Pedro Granville Gonçalves

Roosevelt Vinícius Chaves de Souza

**Atividade :**Estudo e Criação dos Projetos e Documentações Iniciais

Esta etapa foi dedicada ao estudo das melhores alternativas para execução do projeto, com a elaboração de documentações iniciais de suporte.

**Período de Execução:** 06/12/2017 a 30/12/2017

**Justificativa:** Esta atividade está associada a(s) etapa(s) ESTUDO DOS MÓDULOS E PLATAFORMAS DE MONTAGEM DOS AMBIENTES DE DESENVOLVIMENTO do projeto, Atividades de estudo e estratégias de criação dos projetos, incluindo as documentações iniciais.

**Participantes:** Francisco Fechine Borges

Mariana Lins Urquiza

Mateus Assis Maximo de Lima

Pedro Granville Gonçalves

Roosevelt Vinícius Chaves de Souza

**Atividade:** Implementação da Aplicação Túnel TCP

Essa aplicação demonstra a conexão dos módulos a um servidor na internet, enviando e recebendo dados da porta serial da placa, conectada a um computador, e transmitindo para um sevidor TCP na nuvem. Esse exemplo é a base para desenvolver qualquer aplicação de IoT, para comunicação entre o dispositivo eletrônico e um servidor da internet.

**Período de Execução:** 18/02/2018 a 27/02/2018

**Justificativa:** Esta atividade está associada a(s) etapa(s) APLICAÇÕES DE EXEMPLOS do projeto, Implementação da Aplicação Tunel TCP.

**Participantes:** Francisco Fechine Borges

Mariana Lins Urquiza

Mateus Assis Maximo de Lima

Pedro Granville GOnçalves

Roosevelt Vinícius Chaves de Souza

**Atividade:** Implementação da Aplicação SMS Commands

Esse exemplo de aplicação possibilita executar comandos à distância, enviando mensagens de textos com comandos específicos que executam ações na placa de testes, como, ligar ou desligar um LED ou acionar um alarme sonoro.

**Período de Execução:** 25/02/2018 a 27/02/2018

**Justificativa:** Esta atividade está associada a(s) etapa(s) APLICAÇÕES DE EXEMPLOS do projeto, Implementação da Aplicação SMS Commands.

**Participantes:** Francisco Fechine Borges

Mariana Lins Urquiza

Mateus Assis Maximo de Lima

Pedro Granville Gonçalves

Roosevelt Vinícius Chaves de Souza

**Atividade:** Implementação da Aplicação Porteiro Eletrônico

Essa aplicação demonstra a utilização do serviço de voz do módulo GSM HE910, que, ao pressionar um botão, efetua uma ligação automaticamente para um número de telefone pré configurado e estabelece o canal de comunicação entre eles.

**Período de Execução:** 26/02/2018 a 27/02/2018

**Justificativa:** Esta atividade está associada a(s) etapa(s) APLICAÇÕES DE EXEMPLOS do projeto, Implementação da Aplicação Porteiro Eletrônico.

**Participantes:** Francisco Fechine Borges

Mariana Lins Urquiza

Mateus Assis Maximo de Lima

Pedro Granville Gonçalves

Roosevelt Vinícius Chaves de Souza

1. **GEROU OU IRÁ GERAR PATENTE?**

|  |  |
| --- | --- |
|  | Sim |
|  | Não |

1. **GEROU OU IRÁ GERAR PUBLICAÇÃO?**

|  |  |
| --- | --- |
|  | Sim |
|  | Não |

1. **DESCRIÇÃO DO PROJETO**

**Motivação**

A Telit é uma empresa multinacional, fabricante de componentes, módulos e produtos eletrônicos para as áreas de telecomunicações, telemetria e IoT (Internet of Things, ou Internet das Coisas). O mercado de módulos OEM (Original Equipment Manufacturer, ou Fabricante do Equipamento Original, é um termo usado quando uma empresa faz uma parte ou subsistema que é utilizado no produto final de outra empresa) tem crescido no mundo todo. No caso deste projeto, são módulos ou componentes eletrônicos que são feitos para serem comercializados como parte de um produto final, responsável por uma função específica como, por exemplo, um módulo Wifi ou módulo celular). O mercado de módulos eletrônicos OEM é bastante competitivo, com disputas por mercados específicos. Os módulos de outras empresas se tornaram populares, mundialmente, pelo fato de possuírem preços acessíveis e, principalmente, por possuírem placas Shield para Arduino e para Raspberry Pi, que dispõem de milhares de exemplos de códigos disponíveis na internet.

**Problema Técnico-Científico**

Os módulos da Telit possuem diferenciais tecnológicos e preços competitivos, entretanto, não são amplamente conhecidos no mercado mundial e possuem poucas documentações e exemplos de utilização disponíveis na internet.

**Solução Proposta**

Contribuindo para popularizar os módulos OEM da Telit, este projeto visa fornecer documentações facilitadoras, implementações de bibliotecas de software e exemplos de aplicações para fins didáticos e de engenharia, facilitando tanto o primeiro contato de desenvolvedores e *hobistas* com esses módulos, quanto auxiliando as equipes de P&D das empresas fabricantes de produtos eletrônicos, para utilização dos módulos GSM, Wifi e Bluetooth da Telit.

**Objetivo do Projeto**

Desenvolver um Wiki (portal de informações na internet) contendo documentações explicativas, bibliotecas e exemplos de softwares que utilizam os módulos Telit HE910 (GSM), GS2101M (Wifi) e BlueMod+S42 (Bluetooth), para serem utilizados na placa de desenvolvimento e prototipagem Arduino M0 (plataforma amplamente difundida mundialmente para desenvolvimento de protótipos e projetos eletrônicos microcontrolados). Nessa mesma plataforma de hardware, serão desenvolvidas as bibliotecas e exemplos de firmware tanto para a plataforma Arduino (destinado para *hobistas* e iniciantes), quanto para a plataforma Atmel Studio, utilizando o framework ASF e o sistema operacional de tempo real FreeRTOS.

**Escopo do Projeto**

* Documentação em Wiki da utilização dos módulos Telit HE910 (GSM), GS2101M (Wifi) e BlueMod+S42 (Bluetooth);
* Desenvolvimento de biblioteca para cada módulo na plataforma software Arduino;
* Desenvolvimento de biblioteca para cada módulo na plataforma software Atmel Studio;
* Desenvolvimento de exemplos de aplicações para cada módulo na plataforma software Arduino;
* Desenvolvimento de exemplos de aplicações para cada módulo na plataforma software Atmel Studio.

**Detalhamento do Escopo**

* **Documentação em Wiki da utilização dos módulos Telit HE910 (GSM), GS2101M (Wifi) e BlueMod+S42 (Bluetooth)**
  + Wiki utilizando a plataforma Mkdocs, contendo todas as informações e códigos fontes gerados como resultado deste projeto.
* **Desenvolvimento de biblioteca para cada módulo na plataforma software Arduíno**
  + Desenvolvimento da biblioteca do módulo HE910, implementando o acesso a todos os recursos do módulo que será utilizado nas aplicações de exemplo;
  + Desenvolvimento da biblioteca do módulo GS2101M, implementando o acesso a todos os recursos do módulo que será utilizado nas aplicações de exemplo;
  + Desenvolvimento da biblioteca do módulo BlueMod+S42, implementando o acesso a todos os recursos do módulo que será utilizado nas aplicações de exemplo.
* **Desenvolvimento de biblioteca para cada módulo na plataforma software Atmel Studio+AF+FreeRTOS**
  + Desenvolvimento da biblioteca do módulo HE910, implementando o acesso a todos os recursos do módulo que será utilizado nas aplicações de exemplo;
  + Desenvolvimento da biblioteca do módulo GS2101M, implementando o acesso a todos os recursos do módulo que será utilizado nas aplicações de exemplo;
  + Desenvolvimento da biblioteca do módulo BlueMod+S42, implementando o acesso a todos os recursos do módulo que será utilizado nas aplicações de exemplo.
* **Desenvolvimento de Exemplos de aplicações para cada módulo na plataforma software Arduíno**
* **Exemplos GSM HE910**
  + **Túnel TCP:** Sistema de *logger* baseado em modem: loop infinito que repassa o tráfego de uma das portas seriais para um *socket* aberto com um servidor remoto, através dos comandos AT para *sockets*. *Bufferiza* os bytes que são recebidos enquanto a conexão não estava ativa. As configurações de servidor, *baudrate* e afins são manuais, através de constantes no código. O sistema verifica que a conexão caiu (contexto com a APN ou o *socket*) e tenta continuamente uma reconexão. Permite também o fluxo de dados servidor->dispositivo, refletido na porta serial. Ou seja, um túnel de porta serial por TCP.
  + **Geofence GPS:** Determina o pertencimento de uma latitude/longitude recebida através dos códigos NMEA emitidos automaticamente pelo módulo GPS, pela porta serial, com o interior do polígono definido através de um *array* de coordenadas. Quando o *fix*, representado pela mensagem $GPGGA, está fora do polígono, envia onda quadrada para um buzzer/falante e acende um LED.
  + **Geofence LBS:** Utiliza os comandos AT#MONI e AT#AGPSSND para obter a posição por triangulação das ERBs (antenas da rede celular), com análise de polígono considerando a grande imprecisão da posição obtida. Os comandos são simples, utilizam: AT#MONI=7 (Aguarda), AT#MONI, AT#AGPSSND (recebe lat/long). Devido à imprecisão, o status de dentro e fora do polígono pode ser gradual, ao invés de binário / tudo ou nada.
  + **MQTT**: Publica variáveis em sistema MQTT: rotina (Arduino) que lê uma fila de dados a serem enviados para o portal MQTT da Telit, em pares "nome da variável" e "valor".
  + **Geofence MQTT**: Associação do sistema de Geofence com publicação de posições lidas a partir do GPS/LBS em um servidor MQTT.
  + **Comandos SMS**: Comunicação por SMS: rotina que lê continuamente as mensagens SMS recebidas pelo módulo, lendo seu conteúdo e executando comandos conforme o texto enviado pela mensagem, com resposta por SMS.
  + **Geofence SMS**: Associação entre o sistema de Geofence com envio de posições lidas a partir do GPS/LBS, através do sistema de comunicação por SMS.
  + **Menu de Voz DTMF**: Sistema de atendimento de ligações com recebimento de comandos por DTMF e resposta/menu através de sons pré-gravados. Os comandos de detecção de DTMF possuem utilização simples e, para tocar os sons, basta chamar AT#APLAY=1,"diretorio","nome\_do\_arquivo". Ou AT#SAMR. Implementa menus de exemplo e, em alguns deles, o modem envia mensagens de SMS conforme o código digitado e envia variáveis para o servidor MQTT conforme o código digitado.
  + **Porteiro Eletrônico:** Sistema porteiro eletrônico: se um botão for pressionado, liga para um telefone predefinido.
* **Exemplos Wifi GS2101M**
  + **Túnel TCP:** Sistema de *logger* baseado em modem: loop infinito (Arduino) que repassa o tráfego de uma das portas seriais para um socket aberto com um servidor remoto, através dos comandos AT para sockets. *Bufferiza* os bytes que são recebidos enquanto a conexão não estava ativa. As configurações de servidor, *baudrate* e afins são manuais, através de constantes no código. O sistema verifica que a conexão caiu (contexto com a APN ou o *socket*) e tenta continuamente uma reconexão. Permite também o fluxo de dados servidor->dispositivo, refletido na porta serial. Ou seja, um túnel de porta serial por TCP.
  + **MQTT:** Publica variáveis em sistema MQTT: rotina que lê uma fila de dados a serem enviados para o portal MQTT da Telit, em pares "nome da variável" e "valor".
* **Desenvolvimento de Exemplos de aplicações para cada módulo na plataforma software Atmel Studio+ASF+FreeRTOS**
* **Exemplos GSM HE910**
  + **Túnel TCP:** Sistema de *logger* baseado em modem: *tasks* contínuas que repassam o tráfego de uma das portas seriais para um *socket* aberto com um servidor remoto, através dos comandos AT para *sockets*. *Bufferiza* os bytes que são recebidos enquanto a conexão não estava ativa. As configurações de servidor, *baudrate* e afins são manuais, através de constantes no código. O sistema verifica que a conexão caiu (contexto com a APN ou o *socket*) e tenta continuamente uma reconexão. Permite também o fluxo de dados servidor->dispositivo, refletido na porta serial. Ou seja, um túnel de porta serial por TCP.
  + **Geofence GPS:** Determina o pertencimento de uma latitude/longitude recebida através dos códigos NMEA emitidos automaticamente pelo módulo GPS, pela porta serial, com o interior polígono definido através de um array de coordenadas. Quando o fix, representado pela mensagem $GPGGA, está fora do polígono, envia onda quadrada para um buzzer/falante e acende um LED.
  + **Geofence LBS:** Utiliza os comandos AT#MONI e AT#AGPSSND para obter a posição por triangulação as ERBs (antenas da rede celular), com análise de polígono considerando a grande imprecisão da posição obtida. Os comandos são simples, utilizam: AT#MONI=7 (Aguarda), AT#MONI, AT#AGPSSND (recebe latitude/longitude). Devido à imprecisão, o status de dentro e fora do polígono pode ser gradual, ao invés de binário / tudo ou nada.
  + **MQTT:** Publica variáveis em sistema MQTT, comandos AT de MQTT (método nativo) para modem: Cria *task* que lê uma fila de dados a serem enviados para o portal MQTT da Telit, em pares "nome da variável" e "valor".

* + **Geofence MQTT:** Associação do sistema de Geofence com publicação de posições lidas a partir do GPS/LBS em um servidor MQTT.
  + **Comandos SMS:** Comunicação por SMS: *task* que lê continuamente as mensagens SMS recebidas pelo módulo, lendo seu conteúdo e executando comandos conforme o texto enviado pela mensagem, com resposta por SMS.
  + **Geofence SMS:** Associação entre o sistema de Geofence com envio de posições lidas a partir do GPS/LBS através do sistema de comunicação por SMS.
  + **Menu de Voz DTMF:** Sistema de atendimento de ligações com recebimento de comandos por DTMF e resposta/menu através de sons pré-gravados. Os comandos de detecção de DTMF possuem utilização simples e para tocar os sons basta chamar AT#APLAY=1,"diretorio","nome\_do\_arquivo". Ou AT#SAMR. Implementa menus de exemplo e em alguns deles o modem envia mensagens de SMS conforme o código digitado e envia variáveis para o servidor MQTT conforme o código digitado.
  + **Porteiro Eletrônico:** Sistema porteiro eletrônico: se um botão for pressionado, liga para um telefone predefinido.
* **Exemplos Wifi GS2101M**
  + **Túnel TCP:** Sistema de *logger* baseado em modem: *tasks* contínuas (ASF/FreeRTOS) ou Loop Infinito (Arduino) que repassa o tráfego de uma das portas seriais para um *socket* aberto com um servidor remoto, através dos comandos AT para *sockets*. *Bufferiza* os bytes que são recebidos enquanto a conexão não estava ativa. As configurações de servidor, *baudrate* e afins são manuais, através de constantes no código. O sistema verifica que a conexão caiu (contexto com a APN ou o socket) e tenta continuamente uma reconexão. Permite também o fluxo de dados servidor-→dispositivo, refletido na porta serial. Ou seja, um túnel de porta serial por TCP.
  + **MQTT:** Publica variáveis em sistema MQTT: Cria *task* que lê uma fila de dados a serem enviados para o portal MQTT da Telit, em pares "nome da variável" e "valor".

**Etapas do Projeto**

O projeto foi executado em etapas, descritas a seguir:

**1. ESTUDO DOS MÓDULOS E PLATAFORMAS DE MONTAGEM DOS AMBIENTES DE DESENVOLVIMENTO:** Etapa inicial, onde a equipe estudou as documentações existentes de cada módulo da Telit, com a placa Arduino M0 Pro e com as plataformas de software Arduino (integração de bibliotecas e exemplos com o framework); em com o Atmel Studio, framework ASF e sistema operacional de tempo real FreeRTOS. Com a chegada dos módulos e placas Arduino M0 Pro disponibilizadas para a equipe, foram efetuados testes e primeiras implementações com os mesmos.

**2. IMPLEMENTAÇÃO DAS BIBLIOTECAS DE COMUNICAÇÃO COM OS MÓDULOS:** Nesta etapa, foram implementadas as bibliotecas auxiliares para comunicação com os módulos, tanto para a plataforma Arduino quanto para a plataforma Atmel Studio + ASF + FreeRTOS. Essas são as bibliotecas que implementam a comunicação através das portas seriais UART e interpretação dos comandos AT, utilizados para interagir com todos os módulos utilizados neste projeto.

**3. IMPLEMENTAÇÃO DAS BIBLIOTECAS DOS MÓDULOS:** Nesta etapa, foram implementadas as bibliotecas de software que disponibilizam, de forma fácil e intuitiva, os recursos dos módulos que fazem parte do escopo deste projeto, para as plataformas Arduino e Atmel Studio+ASF+FreeRTOS.

**4. APLICAÇÕES DE EXEMPLOS:** Nesta última etapa, foram implementados diversos exemplos, conforme descritos no escopo detalhado deste documento, e suas respectivas documentações, disponibilizadas no corpo dos códigos fonte.

1. **ATIVIDADE INVESTIGATIVA (Validações executadas, conforme Critério C3):**
2. **CARACTERISTICA INOVATIVA**

No mercado de módulos OEM, os módulos mais bem documentados e difundidos são tecnologicamente mais simples, com poucos recursos. Além disso, suas documentações, bibliotecas e exemplos de códigos são voltados para hobistas, não podendo ser aplicados diretamente no desenvolvimento de produtos comerciais, por serem instáveis e limitados.

Este projeto visa produzir e disponibilizar documentações claras e validadas, para que pessoas com pouco conhecimento em eletrônica e microcontroladores possam utilizá-las para o desenvolvimento de projetos que incorporam alguns dos módulos mais modernos do mercado. E, ao mesmo tempo, disponibilizar bibliotecas e exemplos de firmwares que também possam ser aplicados diretamente no desenvolvimento de produtos por empresas de tecnologia, facilitando e reduzindo o tempo e o custo de desenvolvimento.

1. **APLICABILIDADE**

Os produtos deste projeto podem ser aplicados por qualquer hobista, desenvolvedor ou engenheiro eletrônico que esteja à procura de módulos de telecomunicações, telemetria ou IoT, de fácil utilização e acesso a informações claras e validadas e que necessitem desenvolver protótipos de produtos de maneira rápida e com custos acessíveis.

1. **PERSPECTIVAS MERCADOLÓGICAS**

A perspectiva é que esse Portal de Informações, bibliotecas e exemplos para os módulos da Telit, auxilie na popularização desses produtos da empresa e, consequentemente, aumente significativamente sua utilização componentes por hobistas, estudantes, engenheiros e outras empresas, em todo o mundo.

1. **ALCANCE DA INOVAÇÃO**

|  |  |
| --- | --- |
| **X** | Inovador para a empresa |
|  | Inovador no Mercado Interno |
|  | Inovador Internacionalmente |

1. **NÍVEL TÉCNICO**

|  |  |
| --- | --- |
|  | Pequeno |
|  | Médio |
|  | Grande |

1. **SITUAÇÃO ATUAL:**

|  |  |
| --- | --- |
|  | Iniciado |
|  | Em andamento |
|  | Interrompido |
|  | Cancelado |
|  | Concluído |