

# Control iD

---

**REP iDX**



## Manual do desenvolvedor

Instalação da SDK para integração com o  
REP iDX Control iD

# 1. Introdução

Este documento tem por objetivo apresentar o procedimento de instalação do kit de desenvolvimento (SDK) oferecido pela Control iD para desenvolvimento de soluções que envolvam comunicação com o REP iDX da Control iD.

A SDK de comunicação é uma interface para troca de dados com os Registradores Eletrônicos de Ponto e permite manipular informações dos usuários cadastrados nos REP's e suas impressões digitais, bem como obter os registros de marcação de ponto mantidos nos equipamentos.

## Histórico de revisões:

Data	Revisão	Descrição
01/03/2013	1	Versão inicial
12/06/2013	2	Novo capítulo “4”, explicando conversão de cartões.
13/08/2013	3	Nova seção 4.2 sobre transferência de usuários via Pen Drive.

## 2. Instalação

A SDK é oferecida como uma DLL da plataforma Microsoft .NET Framework 4.0 e pode ser utilizada diretamente em linguagens como Microsoft C# e Visual Basic .NET, ou a partir de praticamente qualquer outro ambiente de desenvolvimento através da interface COM oferecida. O mesmo binário é utilizado nos dois casos e é compatível com Windows XP e versões mais recentes do Windows, nas variantes 32 e 64 bits. As próximas subseções explicitam os procedimentos necessários para utilização da SDK em cada um dos casos mencionados.

### 2.1. Utilização em ambiente Microsoft .NET (Visual Studio 2010 ou superior)

A utilização da SDK com o Microsoft Visual Studio 2010 ou superior nas linguagens C# e Visual Basic .NET é trivial. Basta incluir uma referência em seu Projeto para a biblioteca RepCid.dll fornecida nesta SDK. Lembre-se que seu projeto deverá utilizar o .NET Framework na versão 4.0 ou superior para poder utilizar a SDK.

### 2.2. Utilização em outros ambientes (objeto COM)

Como qualquer objeto COM, é necessário **registrar** o objeto no sistema antes que possa ser utilizado. Entretanto, por se tratar de um objeto COM que utiliza a plataforma Microsoft .NET, o .NET Framework 4.0 (ou superior) deverá estar instalado no computador. Além disso, o processo de registro do objeto COM é diferenciado.

1. Copie o arquivo RepCid.dll para um local em seu computador. É importante escolher um local que não precisará ser alterado pois, caso seja alterado, o processo de registro precisará ser refeito.
2. Encontre o programa RegAsm.exe (este é instalado como parte do .NET Framework SDK). Em geral, ele fica localizado em uma das localidades abaixo, escolha a mais adequada para seu sistema. Os passos seguintes utilizarão como exemplo o caminho da versão de 32 bits.

**Versão 32 bits:** C:\Windows\Microsoft.NET\Framework\v4.0.30319\

**Versão 64 bits:** C:\Windows\Microsoft.NET\Framework64\v4.0.30319\

3. Registre o objeto COM no Windows:

a. Abra um terminal MS-DOS como administrador.

b. Mude para a pasta do RegAsm.exe:

```
cd C:\Windows\Microsoft.NET\Framework\v4.0.30319\
```

c. Registre os tipos exportados pela biblioteca:

```
RegAsm.exe "C:\Caminho\Da\Pasta\RepCid.dll" /tlb
```

d. Registre o objeto da biblioteca:

```
RegAsm.exe "C:\Caminho\Da\Pasta\RepCid.dll" /codebase
```

Note que nesse ponto deve surgir um alerta, que é esperado e não implica em nenhum problema:

```
RegAsm : warning RA0000 : Registering an  
unsigned assembly with /codebase can cause  
your assembly (...)
```

4. Adicione uma referência ao objeto COM em seu ambiente de desenvolvimento.

a. **Microsoft® Visual Studio 2010, C++:**

Use o projeto de demonstração TesteRepCidCpp como exemplo. Ele mostra como utilizar a biblioteca RepCid através do Visual C++. O mais importante a ser notado é o uso das diretivas `#import` no arquivo TesteRepCidCpp.cpp daquele exemplo.

b. **Borland® Delphi 2005 – 2006:**

Com um projeto aberto, abra o Menu **Project** e selecione o item **Import Type Library**. Na tela que se abre, selecione o item **RepCid**. (se não encontrar esse item, clique no botão Abrir e encontre o arquivo RepCid.tlb). Pressione Create Unit e o componente deve estar pronto para utilização.

c. **Outros:** siga as instruções oferecidas no ambiente de desenvolvimento para incluir uma referência ao objeto **RepCid**.

### 3. Informações da API

Para obter informações detalhadas sobre a utilização da SDK, por favor utilize os manuais identificados abaixo:

**DocumentacaoRepCid\_.NET.chm:** Documentação da SDK para utilização em linguagens Microsoft .NET.

**DocumentacaoRepCid\_COM.chm:** Documentação das funções disponíveis como objeto COM.

## 4. Informações adicionais

### 4.1. Números de cartões de proximidade

Os números de cartão de proximidade lidos pelo REP são gravados no formato padrão Wiegand 26 bits.

No Wiegand 26 bits, existem 2 bits de paridade que não são gravados no REP, então o número lido do REP traz 24 bits de dados (26 menos os 2 de paridade) contendo o número do cartão.

Para analisar a conversão dos números de cartão, vamos utilizar um cartão hipotético com os seguintes dados:

**Número impresso no cartão:** 168, 52892

**Número lido no REP através da DLL RepCid:** 11062940

Passando o número lido através da DLL para binário, com 24 bits:

101010001100111010011100

Destacando as partes que compõe esse número:

10101000 1100111010011100

A parte em **vermelho** corresponde ao “Site Code” (também chamado “Facility Code”); a parte em **azul** corresponde ao número do cartão.

10101000 (binário) = 168 (decimal)

1100111010011100 (binário) = 52892 (decimal)

A seguir são apresentados exemplos de rotinas simples para converter os números de cartões.

**Para converter o número lido no número impresso:**

```
int cartao = 11062940; // Lido do REP com a DLL RepCid
int sitecode = cartao / 65536; // Resulta 168

int num = cartao % 65536; // Resulta 52892
```

**Para converter o número impresso no número lido:**

```
int sitecode = 168; // sitecode deve ser entre 0 e 255
int num = 52892; // num deve ser entre 0 e 65535

int cartao = sitecode * 65536 + num; // Resulta 11062940
```

## 4.2. Transferência de usuários via Pen Drive

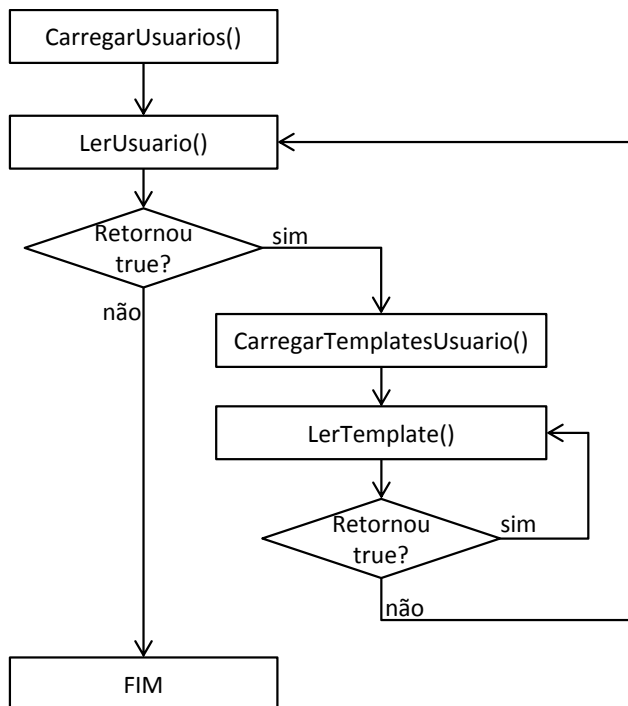
Para ler e graver dados de usuários em Pen Drive, utilize uma das opções a seguir:

**A) Utilizando a SDK de linguagens Microsoft .NET (C# / VB.NET / etc.)**

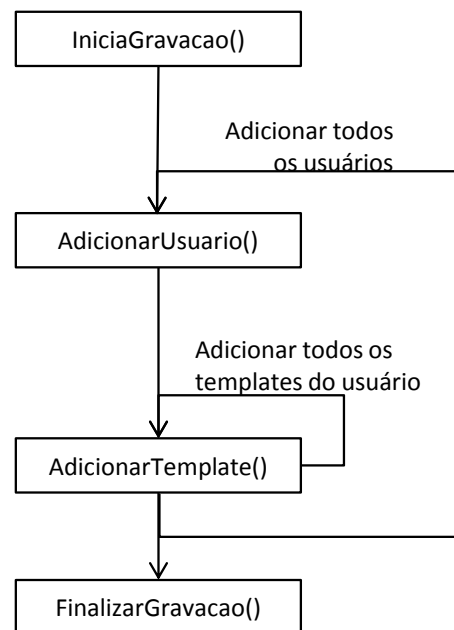
Utilize a classe RepCidUsbUserManagement.

**B) Utilizando a SDK como objeto COM:**

Utilize a interface IRepCidUsb, respeitando os fluxogramas mostrados abaixo.



Fluxograma para leitura de usuários de Pen Drive



Fluxograma para gravação de dados de usuários em Pen Drive