

Abaixo segue exemplo de código de como o programador irá repassar os dados as funções do componente para gerar o arquivo do sped. No exemplo atual estamos utilizando dados fixos “fictícios”, ou seja, não estamos puxando os dados de nenhum banco de dados.

```
//IDENTIFICAÇÃO DO CONTRIBUINTE
```

```
Retorno := SpedPisCofins1.Registro0000('002', //Código da versão do leiaute conforme a tabela 3.1.1.
```

```
    '0', //Tipo de escrituração: 0 - Original; 1 - Retificadora.
```

```
    '1', //Indicador de situação especial
```

```
    '', //Número do Recibo da Escrituração Anterior, a ser retificada. Utilizado quando  
TIPO_ESCRIT for igual a 1
```

```
    '01/02/2011', //Data inicial das informações contidas no arquivo.
```

```
    '31/02/2011', //Data final das informações contidas no arquivo.
```

```
    'Pessoa Juridica', //Nome empresarial da pessoa juridica
```

```
    CNPJ, //Número de inscrição do estabelecimento matriz da pessoa jurídica no CNPJ:
```

```
    'MG', //Sigla da unidade da federação da pessoa jurídica.:
```

```
    '3134202', //Código do município do domicílio fiscal da pessoa jurídica, conforme a tabela  
IBGE:
```

```
    '550309012', //Inscrição da pessoa jurídica na SUFRAMA:
```

```
    '01', //Indicador da natureza da pessoa jurídica: 00 - Sociedade empresária em geral; 01 -  
Sociedade cooperativa
```

```
    '0' //Indicador de tipo de atividade preponderante: 0 – Industrial ou equiparado a  
industrial; 1 – Outros.
```

```
);
```

```
Retorno_SPED(sender, Retorno);
```

**Retorno** é a variável do tipo inteiro onde será armazenado o retorno da função, 0 para execução com sucesso, -1 para erro, nesse caso os detalhes do erro serão gravados no arquivo temperro.txt.

A função **Retorno\_SPED** acima faz a checagem do conteúdo da variável **Retorno** e exibe uma mensagem em caso de erro.

Segue mais alguns exemplos:

```
// Registro F990 - ENCERRAMENTO DO BLOCO F
```

```
Retorno := SpedPisCofins1.RegistroF990;
```

```
Retorno_SPED(sender, Retorno);
```

```
// Registro M100 - CRÉDITO DE PIS/PASEP RELATIVO AO PERÍODO
```

```
Retorno := SpedPisCofins1.RegistroM100('399', // Código de Tipo de Crédito apurado no período,  
conforme a Tabela 4.3.6.
```

```
    '0', // Indicador de Crédito Oriundo
```

```
    '00', // Valor da Base de Cálculo do Crédito
```

```
    '', // Alíquota do PIS/PASEP (em percentual)
```

```
    '00', // Quantidade - Base de cálculo PIS
```

```
    '15', // Alíquota do PIS (em reais)
```

```
    '00', // Valor total do crédito apurado no período
```

```
    '00', // Valor total dos ajustes de acréscimo
```

```
    '00', // Valor total dos ajustes de redução
```

```
    '00', // Valor total do crédito diferido no período
```

```
    '00', // Valor Total do Crédito Disponível relativo ao Período (08 + 09 - 10 - 11)
```

```
    '0', // Indicador de opção de utilização do crédito disponível no período
```

```
    '00', // Valor do Crédito disponível, a descontar, da contribuição apurada no próprio  
período.
```

```
    '00' // Saldo de créditos a utilizar em períodos futuros (12 - 14)
```

```
);
```

```
Retorno_SPED(sender, Retorno);
```

//RegistroM200 - CONSOLIDAÇÃO DA CONTRIBUIÇÃO PARA O PIS/PASEP DO PERÍODO

Retorno := SpedPisCofins1.RegistroM200('00', // Valor Total da Contribuição Não Cumulativa do Período  
(campo 12 do Registro M210)

'00', // Valor do Crédito Descontado, Apurado no Próprio Período da Escrituração (campo 14 do  
Registro M100)

'00', // Valor do Crédito Descontado, Apurado em Período de Apuração Anterior (campo 13 do  
Registro 1100)

'00', // Valor Total da Contribuição Não Cumulativa Devida (02 - 03 - 04)

'00', // Valor Retido na Fonte Deduzido no Período

'00', // Outras Deduções no Período

'00', // Valor da Contribuição Não Cumulativa a Recolher/Pagar (05 - 06 - 07)

'00', // Valor Total da Contribuição Cumulativa do Período (campo 12 do Registro M210)

'00', // Valor Retido na Fonte Deduzido no Período

'00', // Outras Deduções no Período

'00', // Valor da Contribuição Cumulativa a Recolher/Pagar (09 - 10 - 11)

'00' // Valor Total da Contribuição a Recolher/Pagar no Período (08 + 12)

);

Retorno\_SPED(sender, Retorno);

// RegistroM600 - CONSOLIDAÇÃO DA CONTRIBUIÇÃO PARA A SEGURIDADE SOCIAL - COFINS DO PERÍODO

```
Retorno := SpedPisCofins1.RegistroM600('00',          // Valor Total da Contribuição Não Cumulativa do Período
(campo 12 do Registro M610)

      '00',          // Valor do Crédito Descontado, Apurado no Próprio Período da Escrituração (campo 14
do Registro M500)

      '00',          // Valor do Crédito Descontado, Apurado em Período de Apuração Anterior (campo 13
do Registro 1500)

      '00',          // Valor Total da Contribuição Não Cumulativa Devida (02 - 03 - 04)

      '00',          // Valor Retido na Fonte Deduzido no Período

      '00',          // Outras Deduções no Período

      '00',          // Valor da Contribuição Não Cumulativa a Recolher/Pagar (05 - 06 - 07)

      '00',          // Valor Total da Contribuição Cumulativa do Período (campo 12 do Registro M610)

      '00',          // Valor Retido na Fonte Deduzido no Período

      '00',          // Outras Deduções no Período

      '00',          // Valor da Contribuição Cumulativa a Recolher/Pagar (09 - 10 - 11)

      '00'          // Valor Total da Contribuição a Recolher/Pagar no Período (08 + 12)

);
```

Retorno\_SPED(sender, Retorno);

Para repassar os dados das informações existentes em algum banco de dados basta informar os campos da sua tabela do banco de dados nos parametros das funções, exemplo:

Tendo:

dmSPED como Datamodule

qRegistroM600 a tabela / query de onde você esta “puxando” os dados.

Seguido de seu campo respectivamente:

TotalContribuicaoPeriodo.asstring // campo

Então passamos:

SpedPisCofins1.RegistroM600(dmSPED.

qRegistroM600TotalContribuicaoPeriodo.asstring,...etc

E assim por diante nos demais parâmetros que compõem a função de geração do registro M600.

Só chamamos a função e passamos os dados nos parâmetros da mesma. Bem simples o funcionamento.

Outro ponto importante a se observar é que o programador não precisa se preocupar com a ordem em que será chamada as funções, podem ser chamadas em qualquer ordem, pois o componente se encarregará de criar os registros na ordem correta na hora de gravar fisicamente o arquivo `sped.txt`

Todas essas características apresentadas neste texto se aplicam aos componentes Sped Fiscal, Sped Contábil e Pis/Cofins. Todos funcionam sob as mesmas regras.