

MATLAB Example

(INITIALIZATION)

```
ai_object = analoginput('keithley',0);
```



(CONFIGURATION)

```
set(ai_object, 'SampleRate', samples_per_sec);  
set(ai_object, 'SamplesPerTrigger', aq_duration*samples_per_sec);  
  
channels_objects = addchannel(ai_object, channels);  
for i=1:length(channels_objects)  
    %podawa se nomer na red (i-tiq red), i se wzima celiq (dwete stoinosti)  
    set(channels_objects(i), 'InputRange', ranges(i,:));  
end
```



(MEASUREMENT)

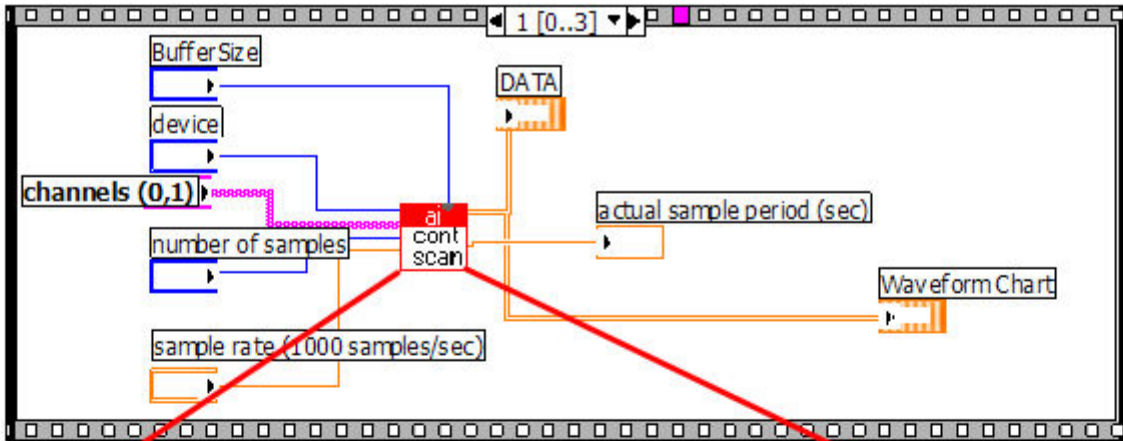
```
start(ai_object)  
trigger(ai_object)  
data = getdata(ai_object);
```



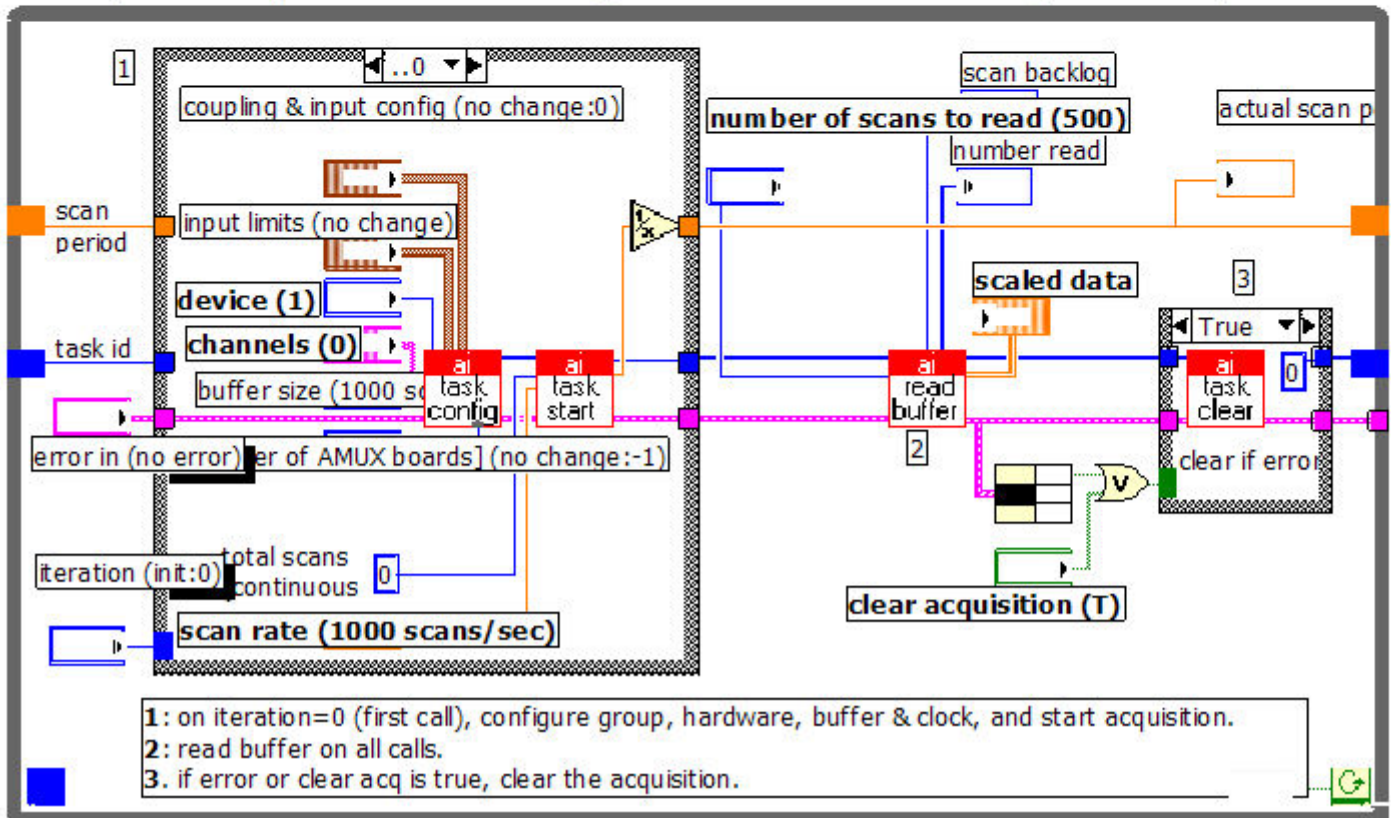
(CLOSE THE DEVICE)

```
delete(ai_object)  
clear ai_object
```

LABVIEW Example



While Loop executes only once, but uses its shift registers to remember **task id** & **scan period** from a previous call to this VI.



C/C++ Program Example

(Open the DriverLINX driver)

```
HINSTANCE m_driverInstance;  
m_driverInstance = OpenDriverLINX(m_hWnd, "driver");  
m_pSR = (DL_ServiceRequest*) new DL_ServiceRequest;  
//allocate memory for the service request  
memset(m_pSR,0,sizeof(DL_ServiceRequest));  
//reset the members of the service request to defaults  
DL_SetServiceRequestSize(*m_pSR);
```

(Initializing a device)

```
m_pSR->device=m_LogicalDevice;  
// Device = integer number assigned in DriverLINX configuration  
m_pSR->operation=INITIALIZE;  
m_pSR->mode=OTHER;  
m_pSR->subsystem=DEVICE;  
m_pSR->hWnd=m_hWnd;  
DriverLINX(m_pSR);  
showMessage(m_pSR); //Show DriverLINX errors, if any
```

(Measurement)

```
m_pSR->operation=START; //Start the acquisition  
m_pSR->subsystem=AI; //using the AI subsystem  
m_pSR->mode=POLLED; //use polled mode  
m_pSR->start.typeEvent=COMMAND; //Start on command  
m_pSR->timing.typeEvent=NULLEVENT;  
m_pSR->stop.typeEvent=TCEVENT;  
m_pSR->channels.nChannels=2;  
m_pSR->channels.chanGain[0].channel=0; //start on channel 0  
m_pSR->channels.chanGain[0].gainOrRange=  
Gain2Code(m_logicalDevice,AI,-1.0); //Use bipolar unity gain  
m_pSR->channels.chanGain[1].channel=7; // stop on channel 7  
m_pSR->channels.chanGain[1].gainOrRange=  
Gain2Code(m_logicalDevice,AI,-1.0);  
m_pSR->channels.numberFormat=tNATIVE;  
//use the native format (integer counts)  
m_pSR->lpBuffers=(DL_BUFFERLIST*) new BYTE[DL_BufferListBytes(1)];  
//create a buffer list pointer for one buffer  
m_pSR->lpBuffers->notify=NOTIFY; //enable the buffer filled message  
m_pSR->lpBuffers->nBuffers=1; //use only one buffer  
m_pSR->lpBuffers->bufferSize=Samples2Bytes  
(m_logicalDevice,AI,m_logicalChannel,m_samples); //set the size of the  
buffer (in bytes) to hold the number of samples  
m_pSR->lpBuffers->BufferAddr[0]=BufAlloc(GBUF_INT,m_pSR->lpBuffers->  
bufferSize);  
//Allocate Buffer 0 based on the size we just specified  
DriverLINX(m_pSR);  
//Execute the service request to start the acquisition  
showMessage(m_pSR); //show any errors  
done();
```

(Closing the DriverLINX driver)

```
CloseDriverLINX(m_DriverInstance);
```