

# EEN 3D ACCELEROMETER VOOR IEDEREEN

*Ivo Van den Mooter*

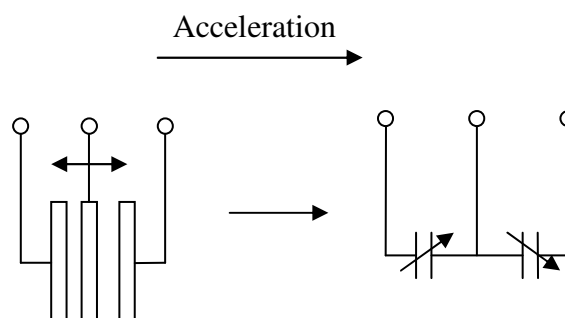
## **Inleiding**

Op enkele dure auto's, op race games, onderaan het televisiescherm tijdens formule 1 wedstrijden ziet men het soms: een grafische voorstelling van de versnellingen die het voertuig ondervindt tijdens het rijden. Het geeft aan hoe snel het voertuig accelereert, remt of een bocht maakt. Dankzij het volgende project, gebaseerd op het R8C13 experimenteerbord en een een accelerometer, is dit interessant gadget nu beschikbaar voor iedereen, meer nog, daar dit project uitgevoerd is met een 3-D accelerometer (X, Y en Z) is het ook mogelijk om tegelijkertijd de versnelling in de verticale richting te meten, m.a.w. de verandering van verticale snelheid. Het gebruik is dus niet enkel beperkt tot een voertuig, maar tot alle voorwerpen waarvan we een idee van hun beweging, in de ruimte willen krijgen. Met de versnelling van een voorwerp kan men ook de snelheid bepalen door de versnelling te integreren over de tijd. Integreert men de snelheid naar de tijd bekomt men de positie.

Ik wilde altijd al experimenteren met microcontrollers, maar vond tot voor enkele maanden de drempel te hoog: programmeerapparaat aanschaffen, ingewikkelde programmertalen aanleren,... enz. Het Elektuur R8C13 project leek me daarom, vanwege de eenvoud en kracht, een kans die ik moest grijpen, en terecht. Daar ik eerder met mechanische zaken bezig ben leek mij het boeiend om "beweging" te gaan meten m.b.v. een accelerometer. En zo is dit project ontstaan.

## **De sensor: 3D-Accelerometer**

De gekozen accelerometer: Freescale MMA 7260 Q meet de versnellingen (verandering van snelheid) die hij ondergaat volgens de 3 orthogonale assen X, Y en Z. Dit type sensor wordt vaak gebruikt in laptops, om te detecteren of ze plots bewegen of kantelen, in camera's voor de beeldstabilisatie, als controle voor de air bags in wagens, enz..., kortom, de sensor is een werkpaard onder de sensoren. Het werkingsprincipe van de sensor is uitgebreid uitgelegd in de datasheet (<http://www.freescale.com>) maar het komt erop neer dat men de verandering van capaciteiten meet van een g-cel. Een g-cel is een microgemachineerd stukje halfgeleider waarvan een middendeel beweegt onder invloed van g-krachten tussen 2 vaste oppervlakken (Figure 1). Met de verandering van de 2 capaciteiten ontstaan tussen het bewegend gedeelte en het vaste gedeelte wordt d.m.v een asic de acceleratie berekend, geconditioneerd en op de pennen gezet.



**Figure 1**

De ISO eenheid voor versnelling is  $\text{m/s}^2$  maar er wordt in veel gevallen gerelateerd naar de valversnelling “g”. ( $1\text{g} = 9.81\text{m/s}^2$ ) ook de documentatie van de accelerometer relateert de versnelling ook naar g. De valversnelling is verantwoordelijk voor de zwaartekracht die wij allemaal ondervinden, dus ook de accelerometer. Vandaar dat, wanneer we de sensor zo plaatsen dat X - en Y –as horizontaal liggen en de Z-as verticaal naar beneden, de uitleeswaarden dan resp 0g, 0g en 1g bedragen. Door deze eigenschap is het ook mogelijk om de (3D - rotatie) positie van een voorwerp “in rust” te bepalen t.o.v. de zwaartekracht m.b.v. deze sensor door enkele trigonometrische berekeningen te doen op de 3 versnellingswaarden.

De sensor, een SMD type, is reeds op een breakout bordje gezet door Spark Fun, met de nodige weerstanden en condensatoren om een geschikt signaal te verkrijgen, zodat die gemakkelijk is in te zetten (Figure 2 ). Het schema is terug te vinden op de Spark Fun website:

[http://www.sparkfun.com/commerce/product\\_info.php?products\\_id=252](http://www.sparkfun.com/commerce/product_info.php?products_id=252))

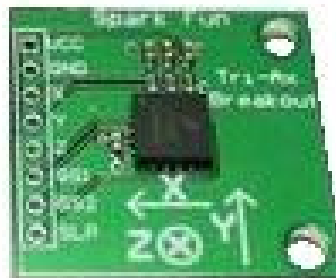


Figure 2

Penbezetting:

1	Vcc:	Spanning 3.3V	5	Z:	Z-as
2	GND:	Spanning 0V	6	GS1:	G-select 1
3	X:	X-as	7	GS2:	G-select 2
4	Y:	Y-as	8	SLP:	Sleep

De X, Y en Z as geven een spanning proportioneel met de versnelling. Met een voedingsspanning van 3.3V geven ze 1,65V bij 0g. De maxima – en + worden bereikt bij resp 0.25V en 2,85V.

De maxima, of het meetbereik kan ingesteld worden d.m.v. de pennen: GS1 en GS2. Er zijn 4 meetbereiken mogelijk: Table 1

GS1	GS2	G Range	Sensitiviteit
0	0	$\pm 1.5\text{g}$	800mV/g
0	1	$\pm 2.0\text{g}$	600mV/g
1	0	$\pm 4.0\text{g}$	300mV/g
1	1	$\pm 6.0\text{g}$	200mV/g

Table 1

Verder is er nog een Sleep pin aanwezig om, indien relevant, batterijcapaciteit te sparen.

## Het project

De vereisten van het project zijn als volgt gedefinieerd:

- De 3 actuele versnellingswaarden X, Y en Z visualiseren op het LCD als getal, maar ook grafisch (bar graph) om snel variërende waarden beter te kunnen waarnemen.
- De maximale waarden (negatieve en positieve) over een bepaalde meetperiode bewaren, visualiseren, met de mogelijkheid tot resetten.
- Visualisering van het actuele meetbereik
- Na de montage van de sensor in het voertuig of op een voorwerp: calibreren of “nullen” van de waarden
- De 3 gemeten waarden serieel uitsturen zodat via de PC kan gelogd en gevisualiseerd worden

## De hardware:

R8C13 experimenteerbord met standaard 2 x 16 characters LCD

Een 3 dimensionale accelerometer MMA7260 Q

Een druktoets

Een buzzer

2 jumpers

een 3.3 V spanningsregelaar LF33CV

enkele condensatoren en weerstanden

Een regelbare weerstand 10K voor de contrastregeling van het display

Het elektrisch schema: Figure 3

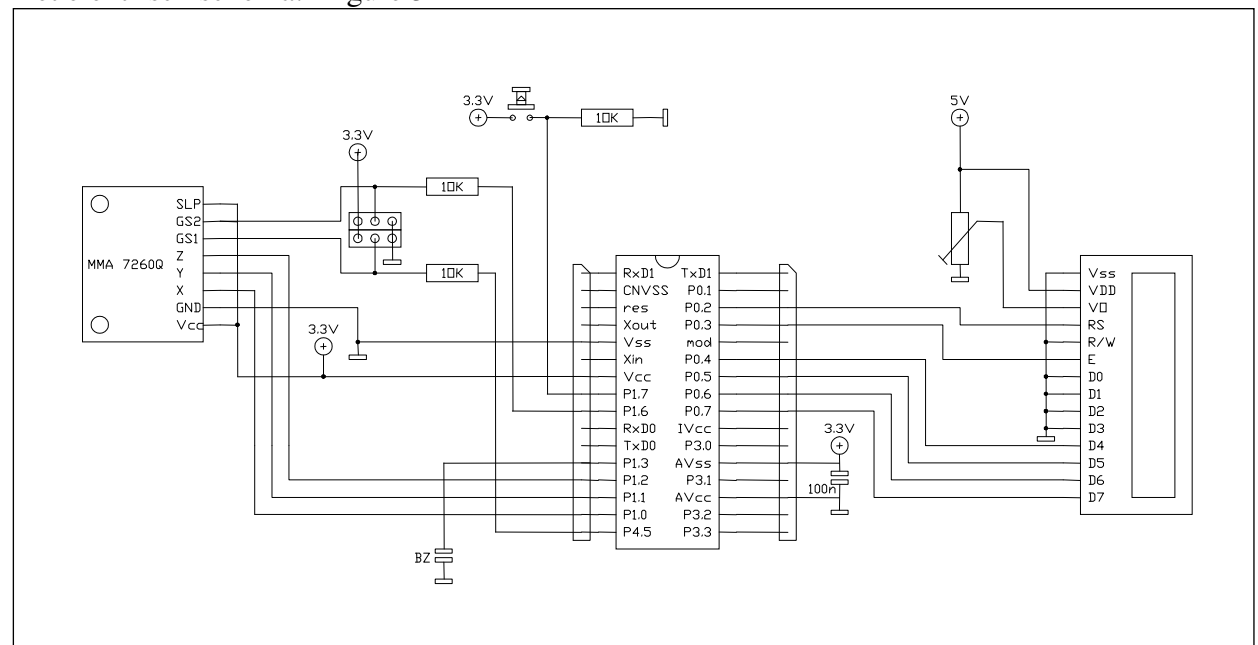


Figure 3

Om het experimenteerbord nog lang te kunnen gebruiken werd een gaatjesbord verbonden met een flat kabeltje om de nodige zaken op te solderen. Het is niet echt een toonbeeld van netheid en

orde maar als experimenteeromgeving is dit een handige oplossing. Zie Figure 4. Later is de sensor in een handige behuizing gemonteerd en met een kabel verbonden met het experimenteerbord zodat bij testen niet het hele bord bewogen moet worden (

Figure 7). In deze behuizing kan de sensor in bv een auto bevestigd worden.

De Sensor wordt gevoed met 3.3V, het R8C13 experimenteerbord werd ook op dezelfde spanning gezet om het volle meetbereik over 10 bits te kunnen benutten. Hiervoor werd, zoals reeds eerder beschreven in Elektuur, de jumper JP11 aangepast zodat de 5V wordt omgezet naar 3.3V over de spanningsregelaar LF33CV. Enkel de contrastpin van het LCD moet ong 4.5V hebben. Hiervoor werd pen van de LCD omgeplooid en apart gevoed met 5V over een regelbare weerstand.

- X, Y en Z werden resp. op p1\_0, p1\_1 en p1\_2 gezet
- De GSEL1 en GSEL2 jumper status wordt ingelezen via resp. p1\_6 en p4\_5
- Een druktoets wordt ingelezen op p1\_7
- De Buzzer staat op p1\_3

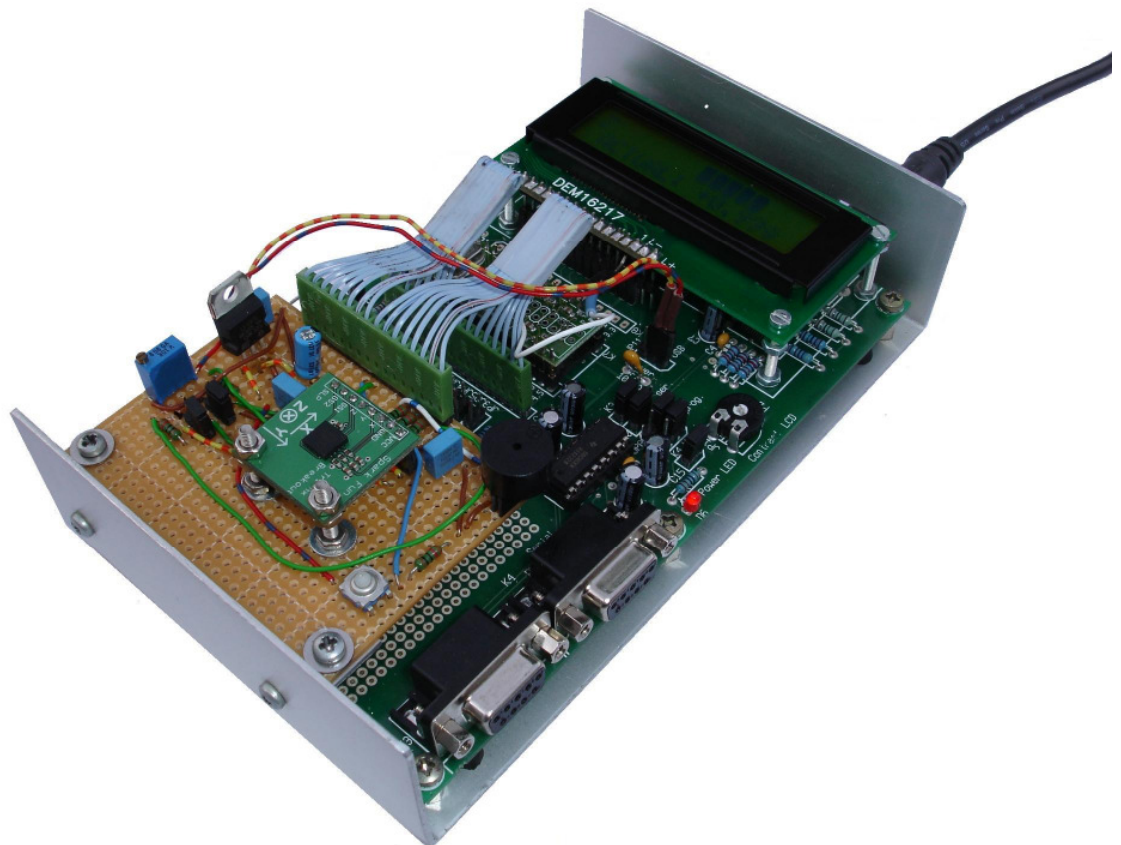


Figure 4

## De werking

Om zoveel informatie te visualiseren op een LCD met slechts 2 x 16 characters werden 8 “schermen” gedefinieerd:

- Een algemeen beginscherm: met naam en softwareversie

- Een scherm met actuele meetbereik met een bargraph (“SELECTED RANGE”)
- 3 schermen met actuele waarden X, Y en Z (“ACTUAL:”)
- 3 schermen met de maxima X, Y en Z (“MAXIMA”)

Figure 5 toont enkele schermen als voorbeeld.

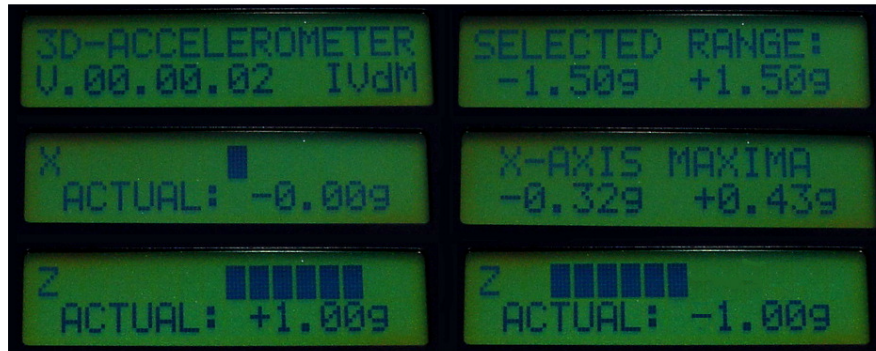


Figure 5

Het gebruik is heel eenvoudig gehouden, met 1 druktoets kan men alle nodige acties uitvoeren:

- Door kort op de druktoets te drukken “bladert” men door de 8 genoemde schermen.
- Blijft men echter drukken, tot een eerste korte beep van de buzzer, zullen de maxima gereset worden.
- Houdt men de druktoets na deze eerste korte beep nog ingedrukt zal een langere beep de calibratie aangeven. Hierbij worden X en Y op 0.00g gezet en Z op 1.00g.
- Bij elke samplename worden de 3 waarden op de seriele poort gezet.

## De software

De tijdbasis van het programma is een 15ms X-timer (“*settimerX*”). Als deze afloopt worden de 3 analoge kanalen 10 maal na elkaar uitgelezen (10 bit) waarvan vervolgens het gemiddeld wordt gemaakt om een stabiel waarde te krijgen. Er wordt meteen gekeken of het een maxima betreft en zoja wordt het bewaard. Dit alles gebeurt in “*getacceldata*”

Het hoofdprogramma bestaat vnl uit de verwerking van het signaal van de drukknop met de nodige acties:

Een korte druk op de knop zal de variable *displaynumber* veranderen en de overeenkomstige vaste informatie op het LCD plaatsen in “*displayfix*”. “*displayvar*” zal in functie van *displaynumber* de variabele data op het scherm vervolledigen.

In “*LCD\_bar*” wordt in functie van de actuele waarde en het actuele meetbereik een grafische voorstelling (bargraph) gemaakt dmv zwarte blokjes (*lcddata* (255)).

Houdt men de drukknop ingedrukt tot een korte beep dan wordt in “*resetmaxima*” de maxima op 0 gezet.

Houdt men de knop nog steeds ingedrukt, tot men een lange beep hoort, dan wordt in “calibrateoffset” de nieuwe offsets berekend zodat X en Y versnellingen op 0, de Z versnelling op 1 worden gezet, de maxima worden hier ook nog eens gereset.

In “send\_3D\_sample” wordt met de 3 actuele versnellingswaarden een string gemaakt die naar de seriele poort wordt gestuurd. De string ziet er als volgt uit: Table 2

+	X <sub>1</sub>	X <sub>2</sub>	X <sub>3</sub>	+	Y <sub>1</sub>	Y <sub>2</sub>	Y <sub>3</sub>	+	Z <sub>1</sub>	Z <sub>2</sub>	Z <sub>3</sub>	CR
---	----------------	----------------	----------------	---	----------------	----------------	----------------	---	----------------	----------------	----------------	----

Table 2

Elke gemeten waarde wordt met 3 significante digits en het teken doorgestuurd, om af te sluiten is een carriage return toegevoegd.

Als voorbeeld van de verdere verwerking van de seriele data is het programma Vbterm aangevuld volgens onderstaand screenshot (Figure 6). Vbterm is een terminalprogramma dat als active X voorbeeld is meegeleverd met Visual Basic 6.0 . De X en Y waarden worden op de “roos” geplaatst zodat de totale horizontale versnellingsvector zichtbaar wordt. De Z waarde is ernaast op een lineaire schaal gevisualiseerd. Het geeft een mooie grafische voorstelling van de gemeten data. Het programma laat ook toe de gemeten waarden te loggen zodat ze achteraf kunnen geanalyseerd worden.

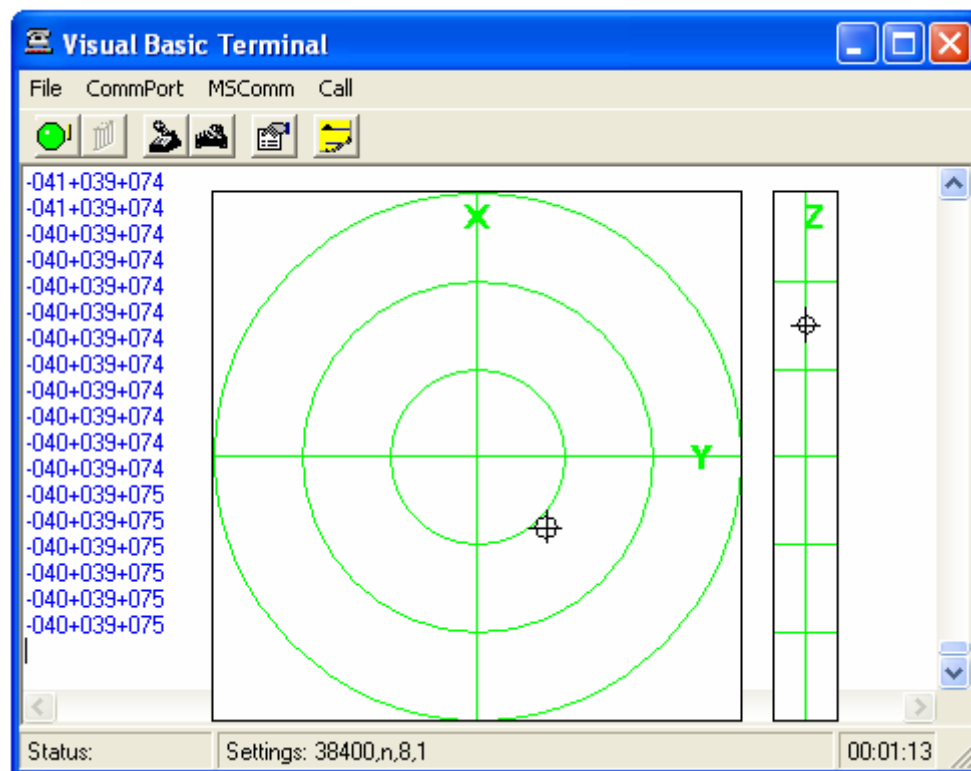


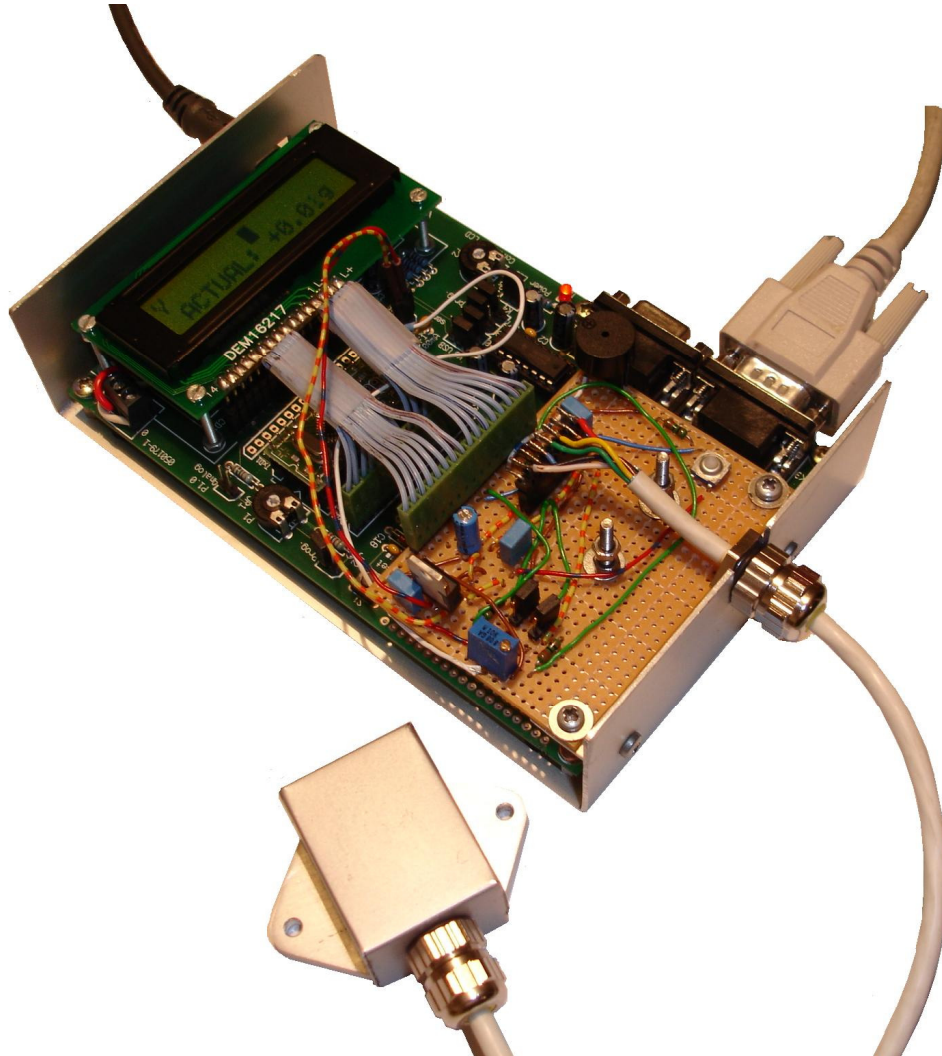
Figure 6

## Besluit

Het R8C elektuur project is voor mijzelf een echt success geworden en aan iedereen aan te raden die met microcontrollers wil starten. Ik heb er tot hertoe al uren plezier aan beleefd en het is een perfecte mix van electronica en software. Dit project zal



zeker niet het laatste zijn dat ik met deze controller bouw. Als vervolg wil ik dit projekt uitbreiden met een seriële draadloze communicatie zodat de sensor de gegevens naar een PC kan sturen zonder draadverbinding. Wordt vervolgd...



**Figure 7**