

**OPTIMUM PWM MODULATION IMPLEMENTATION FOR A THREE PHASE  
INDUCTION MOTOR CONTROL USING THE - TMS320C6416 DSK - CARD****IMPLEMENTACION DE UNA MODULACION PWM OPTIMIZADA PARA EL  
CONTROL DE UN MOTOR TRIFASICO DE INDUCCION USANDO LA  
TARJETA DSP - TMS320C6416 DSK -**

**MSc. Jorge Luis Díaz Rodríguez, PhD. Aldo Pardo García**  
**Ing. Edwin Yesidh Ríos Contreras**

**Universidad de Pamplona**, Facultad de Ingenierías y Arquitectura  
Ciudadela Universitaria. Pamplona, Norte de Santander, Colombia.  
Tel.: 57-7-5685303 Ext. 164  
E-mail: {jdiazcu, apardo13, edwinye}@unipamplona.edu.co

**Abstract:** This paper deals with a procedure to implement an optimized PWM modulation by the direct method for controlling a three phase induction motor using a DSP TMS320C6416 DSK of the C6000 Texas Instruments family. The first part discusses the use of PWM modulation, exhibiting its principle and proposes a simulation model. The former model was compiled and downloaded into the card using the Matlab's Embedded *Target for TI 6000D DSP Toolbox* which describes the procedure for that purpose. Finally, an oscilloscope was used for acquiring the card resulting output signals .

**Resumen:** En este trabajo recoge un procedimiento para implementar una modulación PWM optimizada por el método directo para el control de un motor trifásico de inducción utilizando una tarjeta DSP TMS320C6416 DSK de la familia C6000 de la Texas Instruments. En la primera parte se comenta sobre la modulación PWM a utilizar, donde se exponen su principio y se obtiene un modelo de simulación. Este modelo se compila y descarga en la tarjeta utilizando la *Toolbox Embedded Target for TI 6000 DSP* del Matlab® donde se describe en detalle el procedimiento para tal fin. Finalmente, se adquiere en un osciloscopio las señales resultante en la salida da la tarjeta

**Keywords :** PWM, Simulation, Matlab®, Induction Motor, DSP.

## 1. INTRODUCCION

El auge de diferentes técnicas para optimizar los procesos industriales es elevado gracias a los avances tecnológicos que cada día se logran, con la motivación principal de ofrecer menos consumo energético, mayor eficiencia, mejor economía y cada vez mejor calidad, entre otros.

En la gran mayoría de los procesos industriales se ven involucrados motores eléctricos, principalmente los de corriente alterna (CA), que poco a poco se han venido tomando la industria moderna, desplazando por todas sus ventajas a los ya casi obsoletos motores de corriente directa. De lo anterior surge la necesidad de encontrar estrategias cada vez mejores para controlar este tipo de motores.

El avance de los dispositivos electrónicos hacen que el control de estos motores se realice de forma digital y óptima, aprovechando así todas las características que una máquina de este tipo puede ofrecer en la variación de sus parámetros mecánicos principales como la velocidad y el par de forma continua y precisa.

Más específicamente hablando Procesadores Digitales de Señales o simplemente DSP's que gracias a su alta capacidad de procesamiento fácilmente se puede implementar un control óptimo comparado con los sistemas de control que se encuentran en el mercado, pudiéndose adecuar a cualquier tipo de proceso sin limitación alguna como en ocasiones sucede con los convertidores de frecuencia comunes.

## 2. LA MODULACION PWM

### 2.1 Fundamentos de la Modulación PWM

En la Modulación por Ancho de Pulso, o simplemente PWM, la frecuencia se controla con pulsos positivos durante medio período y pulsos negativos durante el siguiente medio período. El circuito de control establece los tiempos de activación y desactivación de los dispositivos de potencia, que de acuerdo a la estrategia utilizada, puede ser en los puntos de intersección entre una tensión sinusoidal denominada onda moduladora o de referencia y una tensión triangular llamada onda portadora.

La frecuencia de la tensión sinusoidal debe ser igual a la frecuencia que se desea obtener del convertidor de frecuencia y por tanto, se controla por medio de la señal de entrada del circuito de control. La relación entre la amplitud de la tensión sinusoidal y la amplitud de la tensión triangular determina la anchura de los pulsos de tensión que se aplican al motor y por ende el valor de voltaje. En la figura 1 se muestra un esquema típico de modulación PWM basado en una portadora triangular. Los puntos de corte de la portadora con la señal de referencia se determinarán los instantes de conmutación de los transistores que configurarán cada rama.

En la figura 2 podemos ver las formas de onda de un esquema de este tipo para un inversor trifásico. En la parte superior se muestra la portadora y los voltajes de referencia de cada fase y en la parte inferior se puede observar los patrones de conmutación obtenidos para cada una de las fases.

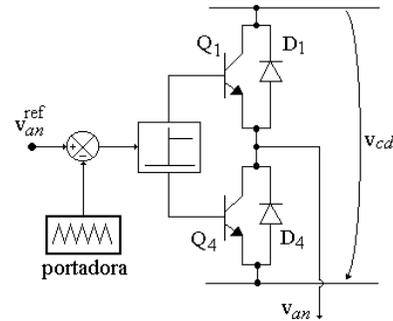


Fig. 1. Modulación por ancho de pulso con portadora triangular

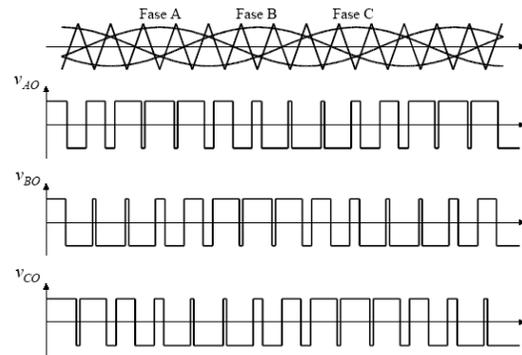


Fig. 2. Formas de onda de voltaje para un inversor trifásico PWM

Las formas de ondas no sinusoidales, obtenidas a la salida del inversor, se analizan aplicando el desarrollo en series de Fourier. Cuando la onda es periódica, se puede descomponer en la suma de una componente continua, una componente fundamental sinusoidal de la misma frecuencia y una serie de armónicos, también sinusoidales, cuya frecuencia es múltiplo de la frecuencia de la onda original.

Si existe simetría de semionda (semiciclos positivos y negativos iguales) desaparece la componente continua y todos los armónicos pares. Si en las tres salidas se generan las mismas formas de onda, desfasadas a  $120^\circ$ , desaparecen todos los armónicos múltiplos de 3. Para la forma de onda de la figura el primer armónico de importancia es el 5 (de secuencia inversa) seguido del 7, 11, etc.

Si todo esto se refiere a un motor de corriente alterna, es posible que algunos armónicos contribuyan a que el rotor gire en el sentido correcto, otros crean un campo en sentido contrario y la mayoría simplemente lo calientan disminuyendo la eficiencia del sistema. También, debido a la reactancia del motor, los armónicos de frecuencia muy elevada tienen poca influencia en

la intensidad y basta eliminar las frecuencias más bajas. A pesar del número reducido de frecuencias armónicas bajas, se producen pérdidas suplementarias en el motor y esto se ilustra más claramente por la recomendación de los fabricantes de los convertidores PWM de procurar que los motores que están funcionando de manera continua no estén sometidos a una carga superior al 85-95% del momento nominal. Por ello surge la necesidad de reducir los armónicos en la de salida de inversor, para mejorar sobre todo la eficiencia del sistema convertidor - motor, esto puede lograrse aplicando modulación por ancho de pulso sinusoidal, donde se actúa sobre los tiempos de conducción siguiendo una ley sinusoidal.

## 2.2 Modulación PWM Optimizada

Este tipo de modulación ha sido posible como resultado del creciente desarrollo en la tecnología basada microprocesadores, que ha facilitado la realización de estas estrategias. Ha sido publicado un gran número de investigaciones [2,3,4,5], las cuales confirman lo anterior y se han logrado algoritmos eficientes basados en microprocesadores, microcontroladores o DSP's para generar estas estrategias.

Debido a este continuo desarrollo, se considera importante incluir este tipo de modulación PWM con ayuda de un modelo computarizado, proponiendo un nuevo método basado en esta estrategia. Desde hace mucho tiempo se conocen las ventajas del uso del PWM optimizado, particularmente en bajas frecuencias.

Otros tipos de modulación PWM como la sinusoidal natural (SPWM) y sinusoidal regular (muestreada) son generadas utilizando circuitos analógicos y/o digitales simples [1,2,3]. En contraste, para generar PWM optimizada se necesita de un determinado conjunto de ángulos de conmutación y determinar estos ángulos utilizando métodos numéricos y computacionales implementados en microprocesadores.

Un ejemplo de forma de onda del PWM optimizada típica se ilustra en la figura. 3.

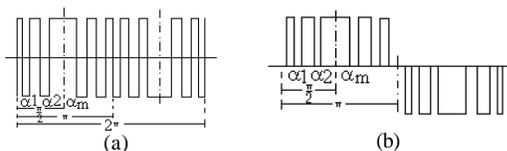


Fig. 3. PWM optimizada típica  
(a) PWM de dos niveles (b) PWM de tres niveles

Basado en los ángulos de conmutación, mostrados en la figura, es posible definir el espectro de armónicos de cada una de las formas de onda PWM. Por ejemplo, si se asume el cuarto de onda simétrica impar, entonces sólo existen armónicos impares y esos pueden ser definidos por las ecuaciones:

Para PWM de dos niveles.

$$V_k = \frac{4}{kp} \left\{ 1 + 2 \sum_{i=1}^m (-1)^i \cos(ka_i) \right\} \quad (1)$$

Para PWM de tres niveles.

$$V_k = \frac{4}{kp} \sum_{i=1}^m (-1)^{i+1} \cos(ka_i) \quad (2)$$

Donde:

- $V_k$  - Amplitud de los armónicos pares de la onda optimizada.
- $k$  - Número del armónico.
- $a_i$  - Angulo de conmutación.
- $m$  - Número de ángulos de conmutación por cuarto de ciclo de la onda PWM.

Las expresiones anteriores pueden ser utilizadas en varias formas para producir formas de onda PWM, las cuales han sido optimizadas con respecto a un criterio particular; por ejemplo, la eliminación de armónicos de baja frecuencia, o la minimización de la distorsión de armónicos o picos de corriente, utilizando criterios de este tipo se obtiene un conjunto de ecuaciones no lineales, en términos de los ángulos de conmutación desconocidos.

Estas ecuaciones pueden ser resueltas utilizando métodos computacionales y por técnicas de minimización numérica para determinar los ángulos de conmutación optimizados. Estos ángulos pueden ser consecuentemente ser almacenados dentro de la memoria de sólo lectura del sistema microprocesador y ser utilizados para generar la forma de onda de PWM en tiempo real. Basándose en los principios anteriores y partiendo del conocimiento de los valores deseados a la salida del inversor, puede reducirse considerablemente la realización del PWM optimizada, haciendo más factible su uso para sistemas a microprocesadores.

### 2.2.1 Método de Optimización Directo

A continuación se desarrollará un método que se clasifica dentro de la modulación PWM optimizada. Este método se denomina directo, ya que no utiliza la señal portadora para la obtención de la onda PWM de salida [4,5]. Tomando en

cuenta que el área bajo la curva de una señal equivale a la energía promedio de esta, se igualan el área bajo la curva de la señal modulada con la de la señal sinusoidal deseada, a fin de causar el mismo efecto.

La señal de referencia sinusoidal está dada por la expresión:

$$u = V_m \text{sen} \omega t \quad (3)$$

Donde:

- $u$  - Voltaje de salida deseado.
- $V_m$  - Valor máximo (Amplitud).
- $\omega$  - Frecuencia angular ( $\omega = 2\pi f$ ).
- $f$  - Frecuencia.

El área en un semiperíodo (de 0 a  $\pi/\omega$ ) sería:

$$A_{\text{seno}} = \int_0^{\pi/\omega} V_m \text{sen} \omega t \, dt = -\frac{V_m}{\omega} \cos \omega t \Big|_0^{\pi/\omega} \quad (4)$$

$$A_{\text{seno}} = \frac{2V_m}{\omega} \quad (5)$$

Esta se puede dividir en  $n_p$  intervalos de tiempo regulares como se muestra a continuación:

$$\left. \begin{aligned} t_0 &= 0 \\ t_1 &= \frac{P}{n_p \omega} \\ t_2 &= \frac{2P}{n_p \omega} \\ &\dots \\ t_n &= \frac{n_p P}{n_p \omega} = \frac{P}{\omega} \end{aligned} \right\} \Rightarrow t_i = \frac{iP}{n_p \omega} \quad i = 0, 1, 2, \dots, n_p \quad (6)$$

El área en cada uno de los intervalos viene dada por la siguiente expresión:

$$\begin{aligned} A_i &= \int_{t_{i-1}}^{t_i} V_m \text{sen} \omega t \, dt = -\frac{V_m}{\omega} \cos \omega t \Big|_{t_{i-1}}^{t_i} \\ &= \frac{V_m}{\omega} (\cos \omega t_{i-1} - \cos \omega t_i) \end{aligned} \quad (7)$$

Conociendo el área de cada uno de los intervalos y fijando el valor máximo de la onda PWM igual a la amplitud de la señal sinusoidal, se puede determinar el ancho de los pulsos.

$$t p_i = \frac{A_i}{V_m} \quad (8)$$

Finalmente, para la conformación de la señal PWM bajo el criterio de optimización antes expuesto, se ubican cada uno de los pulsos obtenidos en el centro de cada uno de los intervalos, expresando matemáticamente este conjunto de pulsos en un plano cartesiano  $xy$ , donde el eje  $x$  corresponde con

el tiempo y el eje  $y$  con la amplitud, como se muestra a continuación:

$$\text{Pulso}_i = \left\{ \begin{aligned} (x, y) / & \frac{x_{i-1} + x_i}{2} - \frac{t p_i}{2} \leq x \leq \frac{x_{i-1} + x_i}{2} + \frac{t p_i}{2}, \dots \\ \dots & 0 \leq y \leq V_m \end{aligned} \right\}$$

siendo  $i = 0, 1, 2, \dots, n_p$

(9)

### 2.2.2 Simulación

A continuación se muestra una simulación utilizando el Simulink® del Matlab® de esta estrategia PWM optimizada propuesta aplicada a un motor trifásico de inducción.

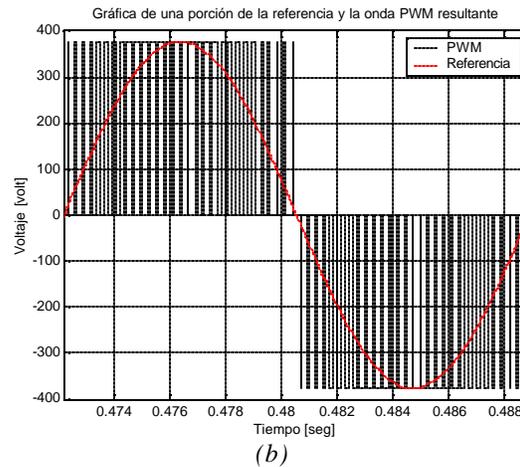
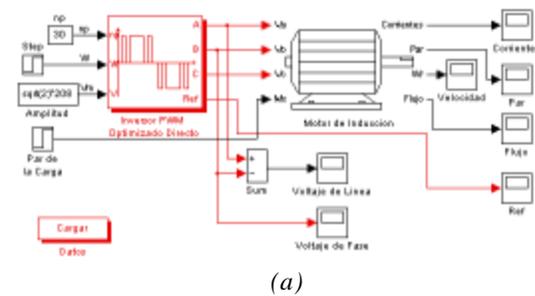


Fig. 4. Ejemplo de simulación  
(a) Esquema; (b) Modulación para  $n_p = 30$

Es notable destacar que a la hora de escoger el número de pulsos  $n_p$ , ha de tenerse en cuenta que este debe ser preferiblemente múltiplo de 3, para evitar los terceros armónicos y a su vez, si es posible, múltiplo de 4, para que la onda resultante mantenga la cuadratura, todo esto como se ha planteado anteriormente. Además, para la operación a bajas frecuencias se puede aumentar el número de pulsos para mejorar la eficiencia del sistema convertidor – motor.

### 3. LA TARJETA TMS320C6416 DSK

En la figura 5 muestra el diagrama de conexión de la tarjeta con la PC para efectuar la programación. Como se puede observar se conecta el cable de AC con la Fuente de 5 V y luego esta con el conector ubicado en la tarjeta, de esta forma la tarjeta ya queda energizada, luego se conecta el cable USB con la tarjeta y luego con el puerto de el PC. De esta forma ya queda lista la tarjeta para ser simulada y programada desde el Code Composer o desde el Matlab.

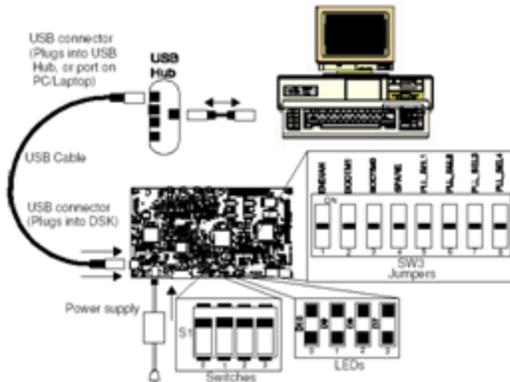


Fig. 5. Diagrama de conexión de la tarjeta

La figura 6 muestra el diagrama en bloques estructural de la tarjeta TMS320C6416 DSK donde se pueden observar todos sus componentes principales.

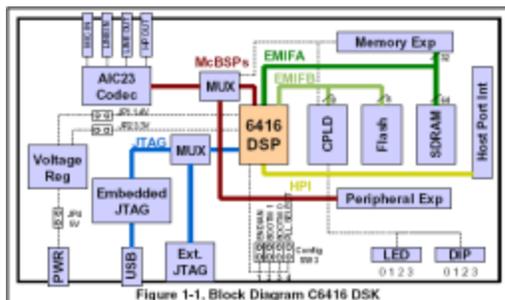


Fig. 6. Diagrama en bloques de la tarjeta

El AIC23 Codec es la parte que permite enlazar la tarjeta a aplicaciones de audio para procesar digitalmente este tipo de señales estos plug son stereo de 35 mm, posee dos líneas de entrada la Mic In y Line In al igual que dos de salida *Line Out* y *Hp Out* que son los conversores DAC y ADC de la tarjeta DSP.

El bloque JTAG es el que me hace la interfaz USB de la tarjeta y a través de este también me permite emular lo que esta sucediendo en la tarjeta y ver todos sus procesos desde la PC.

También se ve en el diagrama la parte principal de la tarjeta que es el DSP 6416 que es componente principal que me controla todos los demás bloques, junto a este se encuentran las expansiones; la PCI que se puede utilizar para diversas aplicaciones y enlazar otro tipo de hardware con la tarjeta tales como memorias, tarjetas de adquisición de datos... etc. La expansión para memorias externas, la expansión de periféricos donde de allí podemos obtener salidas y entradas digitales. La tarjeta también posee interfaz HPI, leds indicadores, interruptores programables. Estos son los bloques generales que posee la tarjeta en hardware.

### 4. ENTORNO DE DESARROLLO DEL CODE COMPOSER STUDIO®

El *Code Composer Studio®*, o simplemente CCS, es el entorno de desarrollo donde se programa la tarjeta, el lenguaje de programación es C, también fusionado en ocasiones en ensamblador. Desde este mismo software no sólo se digita el código sino también se programa la tarjeta. Para que el software detecte la tarjeta primero debemos de hacer la conexión USB y luego de que esté conectada en el menú *Debug* se oprime *Connect*.

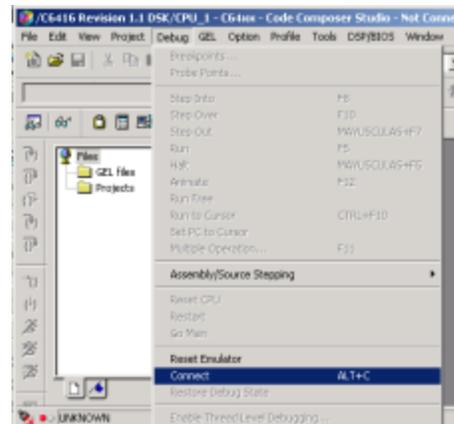


Fig. 7. Pantalla del CCS para hacer la conexión

Después de este paso el software detecta la tarjeta y ya queda lista para ser programada.

Ahora se puede crear el proyecto y el archivo en \*.C, de allí entonces se incluyen las librerías que se van a utilizar. Más tarde se compila el proyecto, y se programa la tarjeta con el archivo de extensión .out el cual se ubica en el menú *File /Load Program* de allí se abre la carpeta *Debug* de el proyecto y luego se abre el archivo específico, en ese instante es cuando se descarga el programa a la tarjeta.

Finalmente se selecciona la opción *Run* y la tarjeta empieza a funcionar de acuerdo a lo programado, esto lo sabremos cuando el LED *Busy* de la tarjeta se encuentra encendido, la ejecución del programa se puede detener desde el mismo CCS.

## 5. PROGRAMACION USANDO MATLAB®

Una de las grandes ventajas de usar esta tarjeta es la compatibilidad que tiene con el Simulink® del Matlab®, ya que este tiene una *Toolbox* diseñada específicamente para este tipo de tarjetas de la compañía *Texas Instruments*, la *Toolbox Embedded Target for TI C6000 DSP*, que hace que la programación sea mucho más sencilla. Pudiéndose hacer lazos de control y diagramas en bloques que luego el compilador que usa Matlab® transforma todo este tipo de diagrama en un lenguaje en C para que el compilador del CCS lo ajuste y lo descargue en la tarjeta DSP. En la figura 8 podemos observar la *Toolbox* específica para esta tarjeta para abrirla buscamos el menú *Embedded Target for TI 6000 DSP* y luego seleccionamos el menú C 6416 DSK que es la tarjeta de la familia de la que estamos utilizando. Al abrir esta librería vemos varios bloques correspondientes a bloques de la tarjeta como se puede observar en la figura 8.

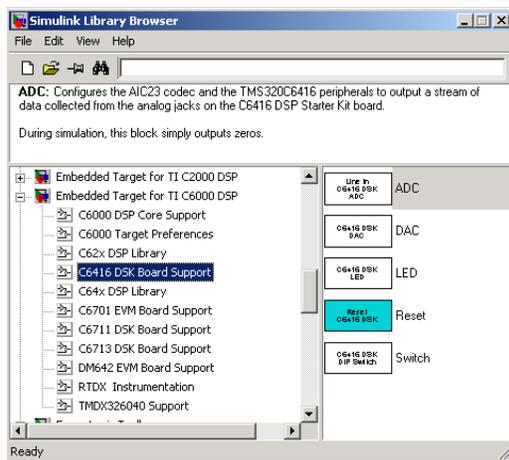


Fig. 8. *Toolbox Embedded Target para TI C6000*

A la derecha de la figura 8 se pueden observar 5 bloques. El ADC que es Conversor Análogo Digital que en hardware corresponde a las dos entradas análogas *stereo* abriendo este bloque y configurando sus propiedades podemos seleccionar el canal a utilizar ya sea la *Line In* o la *Mic In*, seleccionar también la frecuencia de muestreo, el tamaño en bits de la conversión, el tipo de dato a retornar y la cantidad de tramas por muestreo.

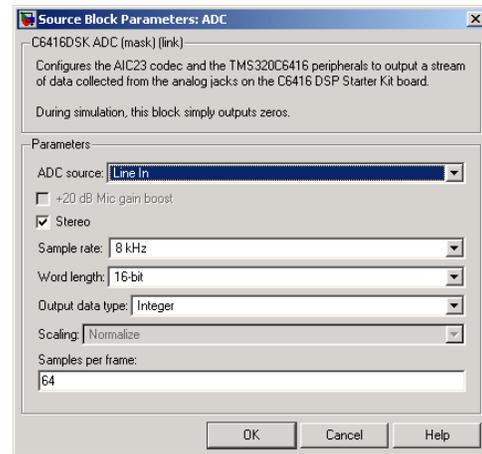


Fig. 8. *Selección de los parámetros del ADC*

Seguidamente se encuentra el DAC que es el Conversor Digital Análogo, el cual en hardware hace referencia a las dos salidas digitales que al igual que el ADC se puede configurar sus parámetros, como la entrada a utilizar, el tamaño en bits de los datos digitales a convertir el tipo de datos a recibir y la frecuencia de muestreo.

El tercero de los bloques es el de los LEDs que en la tarjeta hace referencia a 4 LEDs indicadores, en estos también se configura el tipo de dato a recibir, el bloque DIP SWITCH corresponde a los cuatro *Dip Switch* que hay en la tarjeta estos actúan como cuatro entradas digitales. También en este se configura el tipo de dato a retornar. Otro de los bloques es el de RESET, este simplemente lo que hace es reiniciar la tarjeta dando clic sobre este cuando el programa en la DSP se está ejecutando. Los anteriores bloques hacen referencia a las partes físicas del DSP.

Aparte de estas librerías también hay otras las cuales son de funciones internas de soporte de la tarjeta DSP, como la de guardar en memoria datos al igual que extraer datos de la memoria como se puede ver en la siguiente figura.

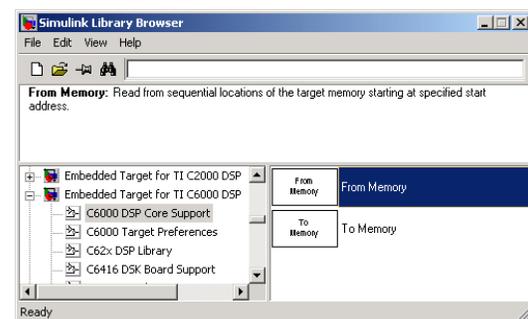


Fig. 9. *Funciones internas e soporte*

Hay otras librerías o bloques compatibles con varias tarjetas de la serie C6000 al cual pertenece la DSK6416, allí podemos encontrar filtros listos para su uso, tales como, FIR, FFT (Transformada Rápida de Fourier) como se pueden observar en la figura 10. Estos bloques los conseguimos en el menú C64X DSP Support.

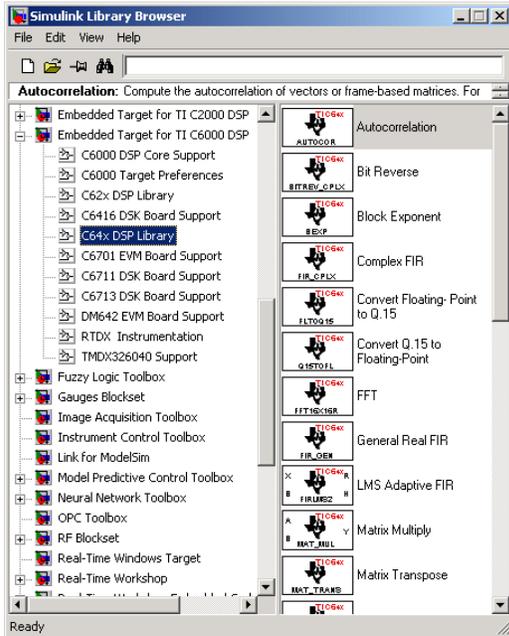


Fig. 10. Librería disponibles

Todos estos bloques anteriormente mencionados pueden interactuar con otras de las *Toolbox* del *Simulink*<sup>®</sup> haciendo de esta una herramienta aun más poderosa y eficiente.

En la próxima figura se muestra un ejemplo sencillo de una aplicación para la DSP, después de tener el modelo ya listo, se configura el *Matlab*<sup>®</sup> para habilitar que el *driver* reconozca la tarjeta y pueda hacer el enlace con el CCS, esto se hace en el menú *Simulation / Configuration Parameters*.

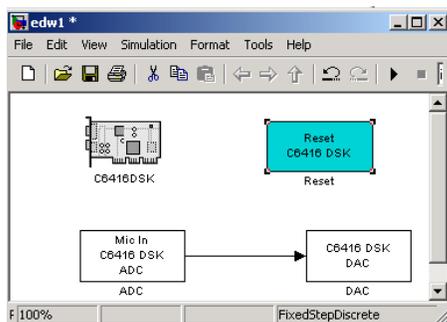


Fig. 11. Ejemplo de programación de una aplicación

En la siguiente ventana es donde se configura, en esta tenemos que seleccionar el modo discreto ya que la tarjeta trabaja de esta forma, también buscamos el *driver* para controlar la tarjeta con la cual estamos trabajando y se debe seleccionar el código al que debe transformar el modelo en este caso en código C.

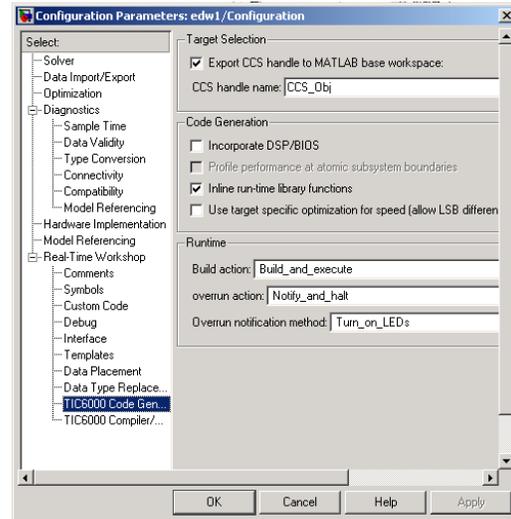


Fig. 12. Pantalla de configuración

En la casilla *Build Action* se selecciona lo que el *Matlab*<sup>®</sup> debe de hacer a la hora de compilar el modelo, aquí se encuentran varias opciones, entre las que están: sólo crear proyecto en CCS, crearlo y compilarlo, etc., hasta hacer todo el proceso completo y llegar a cargar el programa a la tarjeta sin necesidad de hacerlo desde el CCS.

Después se compila en el *Workspace* de *Matlab*<sup>®</sup> donde se ve todo el proceso completo de compilación como se muestra en la figura 13.



Fig. 13. Proceso de compilación

En el diagrama de la figura 14 podemos observar el diagrama en bloques del el sistema completo. La primera parte componen el PC que es donde se realiza el esquema de control a utilizar aprovechando que la tarjeta DSP se puede programar desde de *Matlab*<sup>®</sup>, esto es la gran ventaja del sistema que se puede acoplar a cualquier proceso donde se vea involucrado el

motor, estableciendo el lazo de control adecuado. Luego desde este continua el bloque de la tarjeta DSP donde se habilita una de sus expansiones para generar las señales digitales PWM que posteriormente llegan a la Etapa de Acople y Aislamiento Electrónico que adecua estas señales a los niveles necesarios para que la Etapa de Potencia las procese y el inversor trifásico logre mover el motor trifásico de inducción de acuerdo al sistema de control que se le implemente.

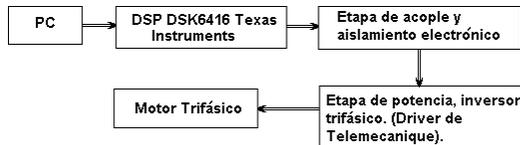


Fig. 14. Diagrama en bloques del sistema

## 6. FORMAS DE ONDAS OBTENIDAS

Una vez explicado el proceso de programación de la tarjeta pasamos a descargar el programa de la Modulación PWM Optimizada Directa y obtener los resultados de las ondas en las salidas de la tarjeta, donde conseguimos los siguientes resultados. Se selecciona una sólo fase de las tres fases de control, debido a que la deferenca entre ellas sólo radica en la fase, las cuales se desfasan  $120^\circ$  entre ellas. Adicionalmente se coloca la referencia sinusoidal para observar el cruce por cero de la onda. (Ver figura 15).

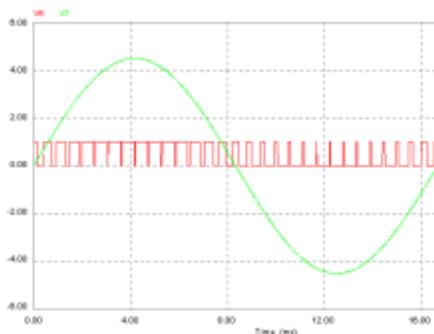


Fig. 15. Forma de onda PWM de la fase A

Luego se paso a variar frecuencia para observar los cambios en la forma de onda resultante, los resultados se muestran en la figura 16.

Por ultimo se realiza un análisis espectral de las ondas anteriores aplicando el método de FFT con un equipo de análisis espectral, obteniendo las siguientes gráficas que se muestran en la fig. 17.

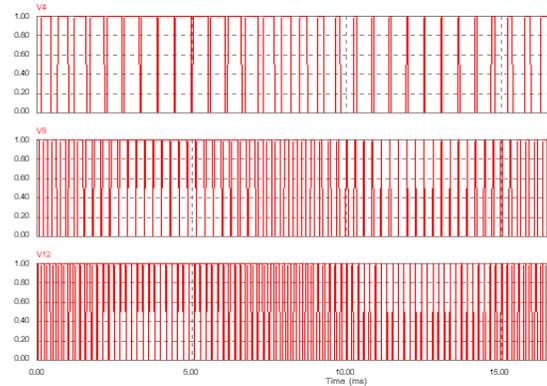


Fig. 16. Formas de onda para varias frecuencias

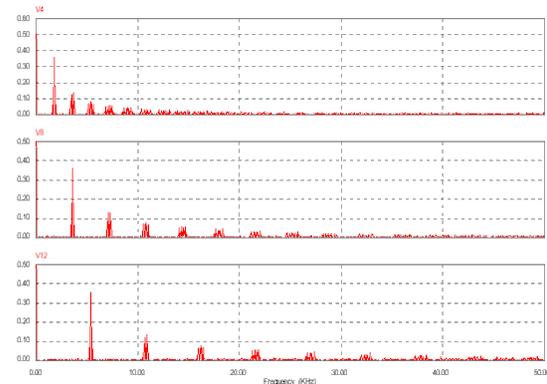


Fig. 17. Análisis espectral de las señales PWM

De la figura anterior se puede observar las componentes armónicas de la señal se van alejando del armónico fundamental, y también que los armónicos 5 y 7 van disminuyendo su valor. Lo contrario ocurre con los armónicos de orden superior que van aumentando, pero tienen menor influencia por que son divididos por el valor de la frecuencia, por lo no poseen gran importancia en el desempeño del motor.

A continuación se muestra una de las tablas obtenidas de tiempos de conmutación (on - off) para una modulación de  $n_p = 30$ .

Vector tiempos fase A ( $0^\circ$ ) = [137.2, 263.74, 319.47, 212.26, 366.22, 169.38, 413.81, 118.78, 458.83, 77.19, 492.71, 49, 512.44, 40.74, 521.42, 16.45, 16.45, 521.42, 40.74, 512.44, 49, 492.71, 77.19, 458.83, 118.78, 413.81, 169.38, 366.22, 212.26, 319.47, 263.74, 137.2, 137.2, 263.74, 319.47, 212.26, 366.22, 169.38, 413.81, 118.78, 458.83, 77.19, 492.71, 49, 512.44, 40.74, 521.42, 16.45, 16.45, 521.42, 40.74, 512.44, 49, 492.71, 77.19, 458.83, 118.78, 413.81, 169.38, 366.22, 212.26, 319.47, 263.74, 137.2]

Para obtener las dos fases restantes, las fases B y C, se utiliza el vector anterior desfasándolo  $-120^\circ$  para la fase B y  $+120^\circ$  para la fase C.

## 7. CONCLUSIONES

Las características operacionales de los convertidores PWM dependen intrínsecamente de procesos de modulación bastante complejos si se desea obtener un comportamiento óptimo. El modelado y la simulación en computador pueden proveer al diseñador del sistema de una herramienta clave para obtener conocimiento dentro de la estructura de armónicos de la forma de onda PWM. Una vez obtenido un modelo de un sistema PWM óptimo en el *Simulink*<sup>®</sup> es relativamente sencillo con el uso de la *Toolbox Embedded Target for TI 6000 DSP* descargarlo en la tarjeta DSP para luego ser implementado un inversor PWM. Finalmente, las respuestas obtenidas en cuanto al balance de armónicos muestran una reducción de armónicos, probando de esta forma que la onda PWM es óptima.

## REFERENCIAS

- [1]. Bose, B. K. *Power Electronics and AC drives*. Prentice-Hall, NY, USA, 2000.
- [2]. Bowes, S., "Computer aided design of PWM inverter system". IEEE Proceeding. 129(1) 1-10, 1991.
- [3]. Holtz, J., "Pulsewidth Modulation: a survey", IEEE Transactions on Industrial Electronics. 39(4) 410-420, 1992.
- [4]. Pardo G., A. y Díaz R., J. L. *Aplicación de los convertidores de frecuencia. Estrategias de control PWM*. Universidad de Pamplona, Colombia, 2004.
- [5]. Rojas, I. *Inversores de frecuencia con modulación del ancho de pulso empleado para el control del motor de inducción*. Tesis de Maestría. Universidad de Camagüey, Camagüey, Cuba, 2000.